

pgAdmin4

リリース *6.19.0*

pgAdmin Development Team

2023 年 01 月 11 日

目 次

第 1 章 Getting Started **3**
1.1 Deployment . . . 3
1.2 Login Dialog . . . 24
1.3 Enabling two-factor authentication (2FA) . . . 26
1.4 User Management Dialog . . . 27
1.5 Change Ownership Dialog . . . 29
1.6 Change User Password Dialog . . . 31
1.7 Lock/Restore Account . . . 32
1.8 Enabling LDAP Authentication . . . 33
1.9 Enabling Kerberos Authentication . . . 36
1.10 Enabling OAUTH2 Authentication . . . 38
1.11 Enabling Webserver Authentication . . . 40
1.12 User Interface . . . 42
1.13 Menu Bar . . . 44
1.14 Toolbar . . . 50
1.15 Tabbed Browser . . . 51
1.16 Tree Control . . . 57
1.17 Preferences Dialog . . . 61
1.18 Keyboard Shortcuts ： . . . 82
1.19 Search objects . . . 87

第 2 章 External database for pgAdmin user settings **89**
2.1 SQLite データベースを使用する . . . 89
2.2 外部データベースを使用 . . . 89
2.3 PostgreSQL データベースを使用する . . . 89

第 3 章 Connecting To A Server **91**
3.1 Server Group Dialog . . . 91
3.2 Server Dialog . . . 92
3.3 Cloud Deployment . . . 101
3.4 Master Password . . . 117
3.5 Connect to Server . . . 118
3.6 Connection Error . . . 120
3.7 Import/Export Servers . . . 122

第 4 章 Managing Cluster Objects **129**
4.1 Database Dialog . . . 129
4.2 Resource Group Dialog . . . 136
4.3 Login/Group Role Dialog . . . 139
4.4 Tablespace Dialog . . . 146

4.5 Role Reassign/Drop Own Dialog . . . 152

第 5 章 Managing Database Objects 155

5.1 Cast Dialog . . . 155
5.2 Collation Dialog . . . 159
5.3 Domain Dialog . . . 163
5.4 Domain Constraints Dialog . . . 169
5.5 Event Trigger Dialog . . . 172
5.6 Extension Dialog . . . 177
5.7 Foreign Data Wrapper Dialog . . . 181
5.8 Foreign Server Dialog . . . 186
5.9 Foreign Table Dialog . . . 192
5.10 FTS Configuration Dialog . . . 199
5.11 FTS Dictionary Dialog . . . 203
5.12 FTS Parser Dialog . . . 208
5.13 FTS Template Dialog . . . 211
5.14 Function Dialog . . . 214
5.15 Language Dialog . . . 221
5.16 Materialized View Dialog . . . 226
5.17 Package Dialog . . . 232
5.18 Procedure Dialog . . . 236
5.19 Publication Dialog . . . 244
5.20 Schema Dialog . . . 248
5.21 Sequence Dialog . . . 253
5.22 Subscription Dialog . . . 260
5.23 Synonym Dialog . . . 266
5.24 Trigger Function Dialog . . . 269
5.25 Type Dialog . . . 277
5.26 User Mapping Dialog . . . 287
5.27 View Dialog . . . 290

第 6 章 Creating or Modifying a Table 297

6.1 Check Dialog . . . 297
6.2 Column Dialog . . . 301
6.3 Compound Trigger Dialog . . . 310
6.4 Exclusion Constraint Dialog . . . 316
6.5 Foreign key Dialog . . . 319
6.6 Index Dialog . . . 326
6.7 Primary key Dialog . . . 330
6.8 RLS Policy Dialog . . . 334
6.9 Rule Dialog . . . 338
6.10 Table Dialog . . . 344
6.11 Trigger Dialog . . . 365
6.12 Unique Constraint Dialog . . . 372

第 7 章 Management Basics 377

7.1 Add Named Restore Point Dialog . . . 377
7.2 Change Password Dialog . . . 377

7.3 Grant Wizard . . . 379
7.4 Import/Export Data Dialog . . . 381
7.5 Maintenance Dialog . . . 386
7.6 Storage Manager . . . 387

第 8 章 Backup and Restore **391**
8.1 Backup Dialog . . . 391
8.2 Backup Globals Dialog . . . 397
8.3 Backup Server Dialog . . . 398
8.4 Restore Dialog . . . 402

第 9 章 Developer Tools **407**
9.1 Debugger . . . 407
9.2 Query Tool . . . 412
9.3 View/Edit Data . . . 438
9.4 Schema Diff . . . 444
9.5 ERD Tool . . . 448
9.6 PSQL Tool . . . 457

第 10 章 Processes **459**
10.1 プロセスウォッチャー . . . 460
10.2 デモを見る . . . 460

第 11 章 pgAgent **461**
11.1 Using pgAgent . . . 461
11.2 Installing pgAgent . . . 462
11.3 Creating a pgAgent Job . . . 464

第 12 章 pgAdmin Project Contributions **475**
12.1 Submitting Pull Requests . . . 475
12.2 Code Overview . . . 476
12.3 Coding Standards . . . 480
12.4 Code Review Notes . . . 483
12.5 Translations . . . 485

第 13 章 リリースノート **487**
13.1 バージョン 6.19 . . . 487
13.2 バージョン 6.18 . . . 488
13.3 バージョン 6.17 . . . 489
13.4 バージョン 6.16 . . . 490
13.5 バージョン 6.15 . . . 492
13.6 バージョン 6.14 . . . 494
13.7 バージョン 6.13 . . . 495
13.8 バージョン 6.12 . . . 497
13.9 バージョン 6.11 . . . 498
13.10 バージョン 6.10 . . . 499
13.11 バージョン 6.9 . . . 501
13.12 バージョン 6.8 . . . 502
13.13 バージョン 6.7 . . . 503

13.14 バージョン 6.6 . . . 504
13.15 バージョン 6.5 . . . 505
13.16 バージョン 6.4 . . . 506
13.17 バージョン 6.3 . . . 507
13.18 バージョン 6.2 . . . 508
13.19 バージョン 6.1 . . . 509
13.20 バージョン 6.0 . . . 511
13.21 バージョン 5.7 . . . 512
13.22 バージョン 5.6 . . . 513
13.23 バージョン 5.5 . . . 514
13.24 バージョン 5.4 . . . 515
13.25 バージョン 5.3 . . . 516
13.26 バージョン 5.2 . . . 517
13.27 バージョン 5.1 . . . 519
13.28 バージョン 5.0 . . . 521
13.29 バージョン 4.30 . . . 522
13.30 バージョン 4.29 . . . 524
13.31 バージョン 4.28 . . . 525
13.32 バージョン 4.27 . . . 526
13.33 バージョン 4.26 . . . 527
13.34 バージョン 4.25 . . . 528
13.35 バージョン 4.24 . . . 530
13.36 バージョン 4.23 . . . 531
13.37 バージョン 4.22 . . . 532
13.38 バージョン 4.21 . . . 534
13.39 バージョン 4.20 . . . 537
13.40 バージョン 4.19 . . . 538
13.41 バージョン 4.18 . . . 539
13.42 バージョン 4.17 . . . 540
13.43 バージョン 4.16 . . . 542
13.44 バージョン 4.15 . . . 544
13.45 バージョン 4.14 . . . 545
13.46 バージョン 4.13 . . . 547
13.47 バージョン 4.12 . . . 549
13.48 バージョン 4.11 . . . 550
13.49 バージョン 4.10 . . . 552
13.50 バージョン 4.9 . . . 552
13.51 バージョン 4.8 . . . 554
13.52 バージョン 4.7 . . . 554
13.53 バージョン 4.6 . . . 556
13.54 バージョン 4.5 . . . 557
13.55 バージョン 4.4 . . . 557
13.56 バージョン 4.3 . . . 559
13.57 バージョン 4.2 . . . 560
13.58 バージョン 4.1 . . . 562
13.59 バージョン 4.0 . . . 562
13.60 バージョン 3.6 . . . 563

13.61 バージョン 3.5 . . . 564
13.62 バージョン 3.4 . . . 565
13.63 バージョン 3.3 . . . 565
13.64 バージョン 3.2 . . . 566
13.65 バージョン 3.1 . . . 568
13.66 バージョン 3.0 . . . 569
13.67 バージョン 2.1 . . . 572
13.68 バージョン 2.0 . . . 575
13.69 バージョン 1.6 . . . 577
13.70 バージョン 1.5 . . . 580
13.71 バージョン 1.4 . . . 581
13.72 バージョン 1.3 . . . 582
13.73 バージョン 1.2 . . . 584
13.74 バージョン 1.1 . . . 586
13.75 バージョン 1.0 . . . 588

第 14 章 Licence **589**

索引 **591**



pgAdmin4 へようこそ。pgAdmin は、世界で最も先進的なオープンソースデータベースである Postgres の主要なオープンソース管理ツールです。pgAdmin4 は、初心者と経験豊富な Postgres ユーザーの両方のニーズを満たすように設計されており、データベースオブジェクトの作成、メンテナンス、使用を簡素化する強力なグラフィカルインターフェイスを提供します。

第1章　Getting Started

pgAdmin4 のプリコンパイルおよび構成済みインストールパッケージは、多くのデスクトップ環境で利用できます。可能な限りインストーラーを使用することをお勧めします。

サーバー展開 では、pgAdmin アプリケーションは Web サーバーの背後にまたは WSGI インターフェイスを使用して展開されます。サーバーモードで pgAdmin をインストールする場合、最初に pgAdmin に接続するときにロール名と pgAdmin パスワードを入力するように求められます。pgAdmin に登録された最初のロールは管理ユーザーになります。管理者ロールは pgAdmin *UserManagement* ダイアログを使用して、追加の pgAdmin ユーザーアカウントを作成および管理できます。ユーザーが pgAdmin で認証すると、pgAdmin ツリーコントロールにはそのログインロールに関連付けられたサーバー定義が表示されます。

デスクトップ展開 では、pgAdmin アプリケーションは、デスクトップランタイム環境を使用してサポートされているプラットフォームでプログラムをホストするように構成されています。通常、ユーザーは事前に構築されたパッケージをインストールして pgAdmin をデスクトップモードで実行しますが、手動デスクトップ展開をインストールすることもできます。

サーバーモードがセキュリティ用に事前構成された pgAdmin の **コンテナ展開** を使用することもできます。

1.1 Deployment

pgAdmin4 のプリコンパイルおよび構成済みインストールパッケージは、多くのデスクトップ環境で利用できます。可能な限りインストーラーを使用することをお勧めします。プロジェクトの詳細に興味がある場合、またはご使用の環境で pgAdmin インストーラーを利用できない場合、以下にリストされているページにカスタム展開の作成に関する詳細情報が記載されています。

1.1.1 The config.py File

pgAdmin によって起動時に読み取られる構成ファイルは複数あります。これらのファイルは、次の構成オプションに使用されます。

- アプリケーションの動作を制御するため、pgAdmin の起動前に設定する必要がある場合があります。
- システム管理者は、組織全体を制御してセキュリティポリシーを適用したい場合があります。
- はほとんどまたはほとんど変更されないため、ユーザーインターフェイスを介して変更できるようにしても意味がありません。

構成ファイルは次のとおりです。

- `config.py` ：これはメイン構成ファイルであり、変更しないでください。構成設定のリファレンスとして使用でき、次のファイルのいずれかでオーバーライドできます。

- config_distro.py ：このファイルは config.py の後に読み込まれ、パッケージャが pgAdmin ディストリビューションに必要な設定を変更することを目的としています。これには通常、特定のパスとファイルの場所が含まれます。このファイルはオプションであり、必要に応じてパッケージャーによって config.py と同じディレクトリに作成されます。
- config_local.py：このファイルは config_distro.py の後に読み込まれ、エンドユーザーがデフォルトまたはパッケージング固有の設定を変更して、ローカルの設定または標準に合わせて調整することを目的としています。このファイルはオプションであり、同じディレクトリにユーザーが作成できます必要に応じて config.py として。
- config_system.py：このファイルは config_local.py の後に読み取られ、システム管理者が、ユーザーが通常変更できない安全な場所からシステム全体に構成された設定を含めることを目的としています。このファイルの場所はプラットフォームによって異なり、必要な場合にのみ作成する必要があります。

Platform	File Location
Linux	/etc/pgadmin/config_system.py
macOS	/Library/Preferences/pgadmin/config_system.py
Windows	%CommonProgramFiles%//pgadmin//config_system.py

注釈: SERVER_MODE または DATA_DIR 設定が config_distro.py 、 config_local.py 、または config_system.py LOG_FILE、SQLITE_PATH、SESSION_DB_PATH、STORAGE_DIR、KERBEROS_CCACHE_DIR、および AZURE_CREDENTIAL_CArid_DIR の値に基づいて設定されていない限り、ファイルの値は明示的に変更されます。

参考までに、デフォルトの config.py ファイルを以下に示します。

1.1.2 Desktop Deployment

pgAdmin は、デスクトップモードで実行するようにアプリケーションを構成し、デスクトップランタイムを利用してサポートされている Windows、MacOSX または Linux インストールでプログラムをホストすることにより、デスクトップアプリケーションとして展開できます。

デスクトップランタイムは、起動すると pgAdmin サーバーを実行し、ウィンドウを開いてユーザーインターフェイスをレンダリングするスタンドアロンアプリケーションです。

注釈: プリコンパイルおよび構成済みのインストールパッケージは、多くのプラットフォームで利用できます。これらのパッケージは、可能な限りエンドユーザーが使用する必要があります。次の情報は、これらのパッケージのメンテナーおよび pgAdmin の動作を理解することに関心があるユーザーに役立ちます。

参考:

pgAdmin をゼロからビルドして構成する手順の詳細については、ソースコードの最上位ディレクトリにある README ファイルを参照してください。便宜上、ファイル here の最新バージョンを見つけることがで

きますが、これは特定のバージョンの pgAdmin のソースコードに含まれるバージョンとは異なる場合があることに注意してください。

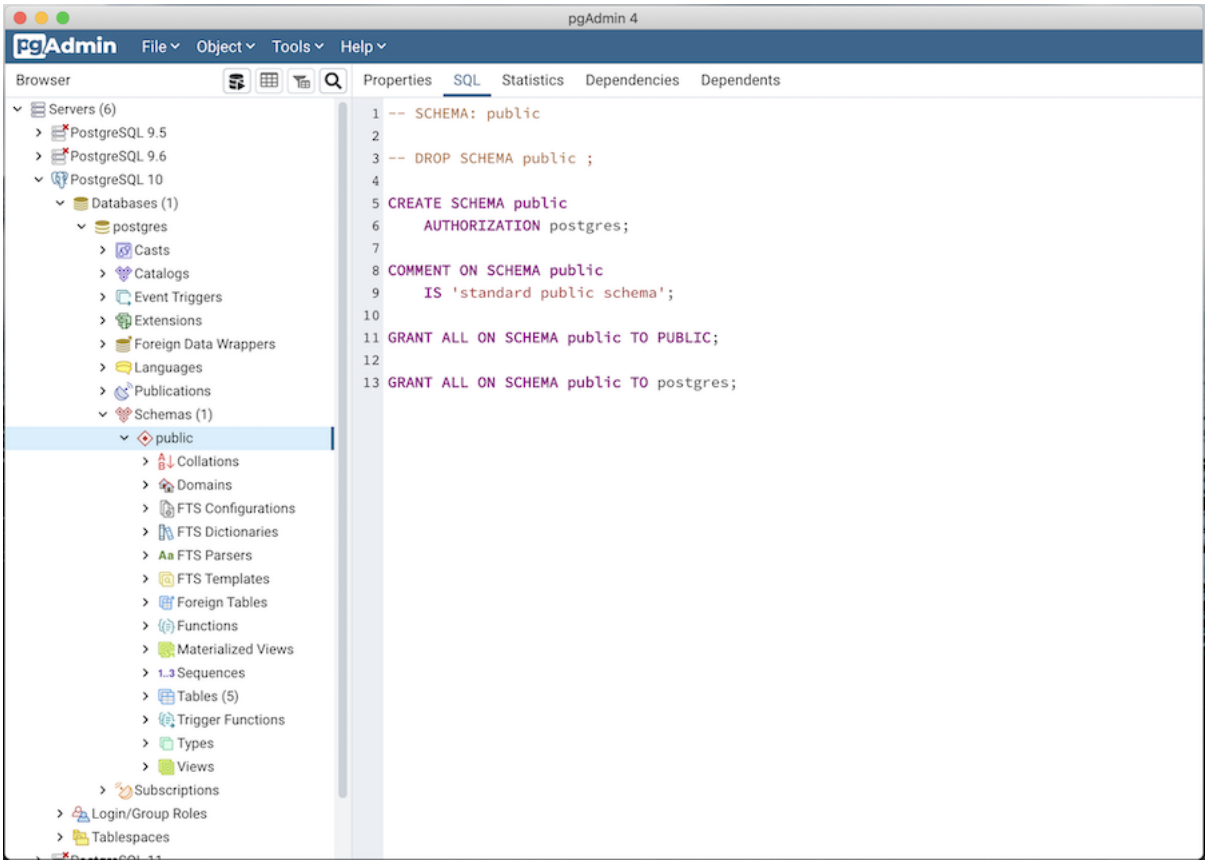
構成

pgAdmin4v2 以降、デフォルトの構成モードはサーバーですが、これは起動時にデスクトップランタイムによってオーバーライドされます。ほとんどの環境では、他のデフォルト設定をオーバーライドしない限り、Python 構成は必要ありません。

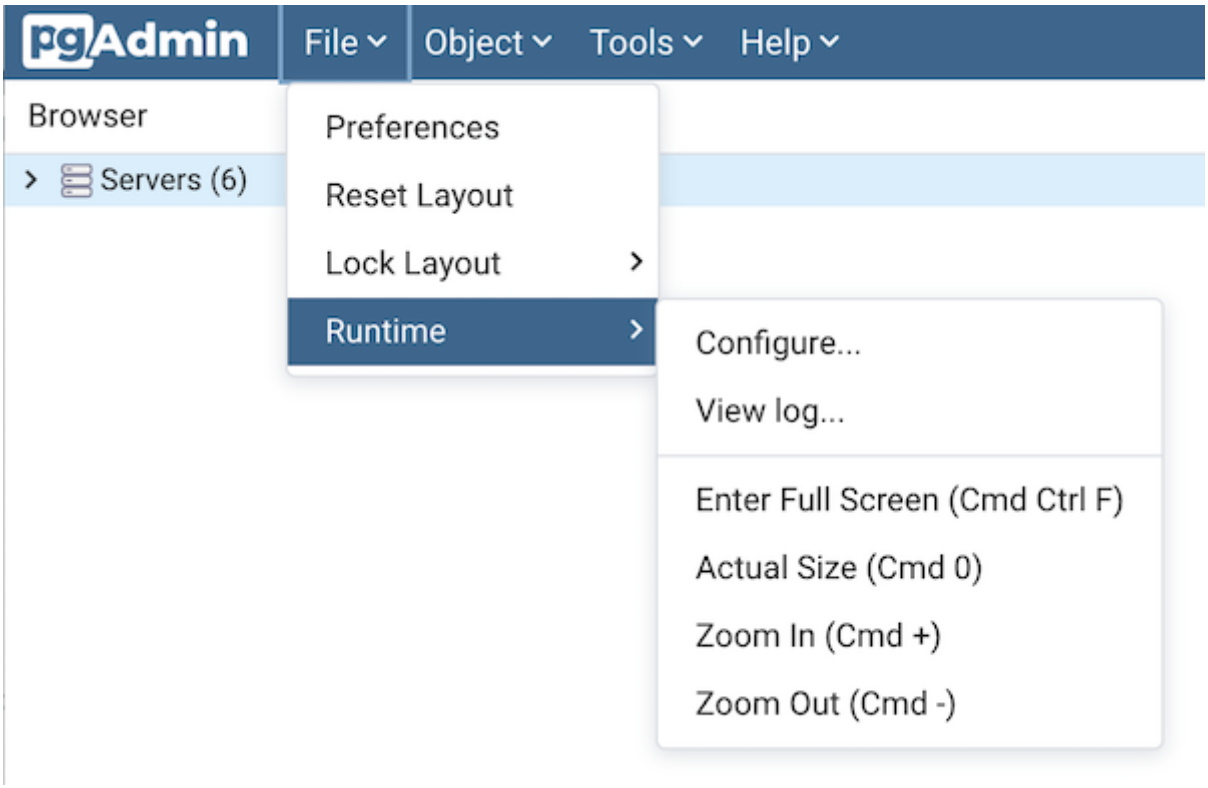
構成設定の詳細については、 *The config.py File* を参照してください。

デスクトップランタイムスタンドアロンアプリケーション

デスクトップランタイムは、ブラウザーとスタンドアロンアプリケーションを作成する Python サーバーを統合する NWjs に基づいています。



ランタイムメニュー

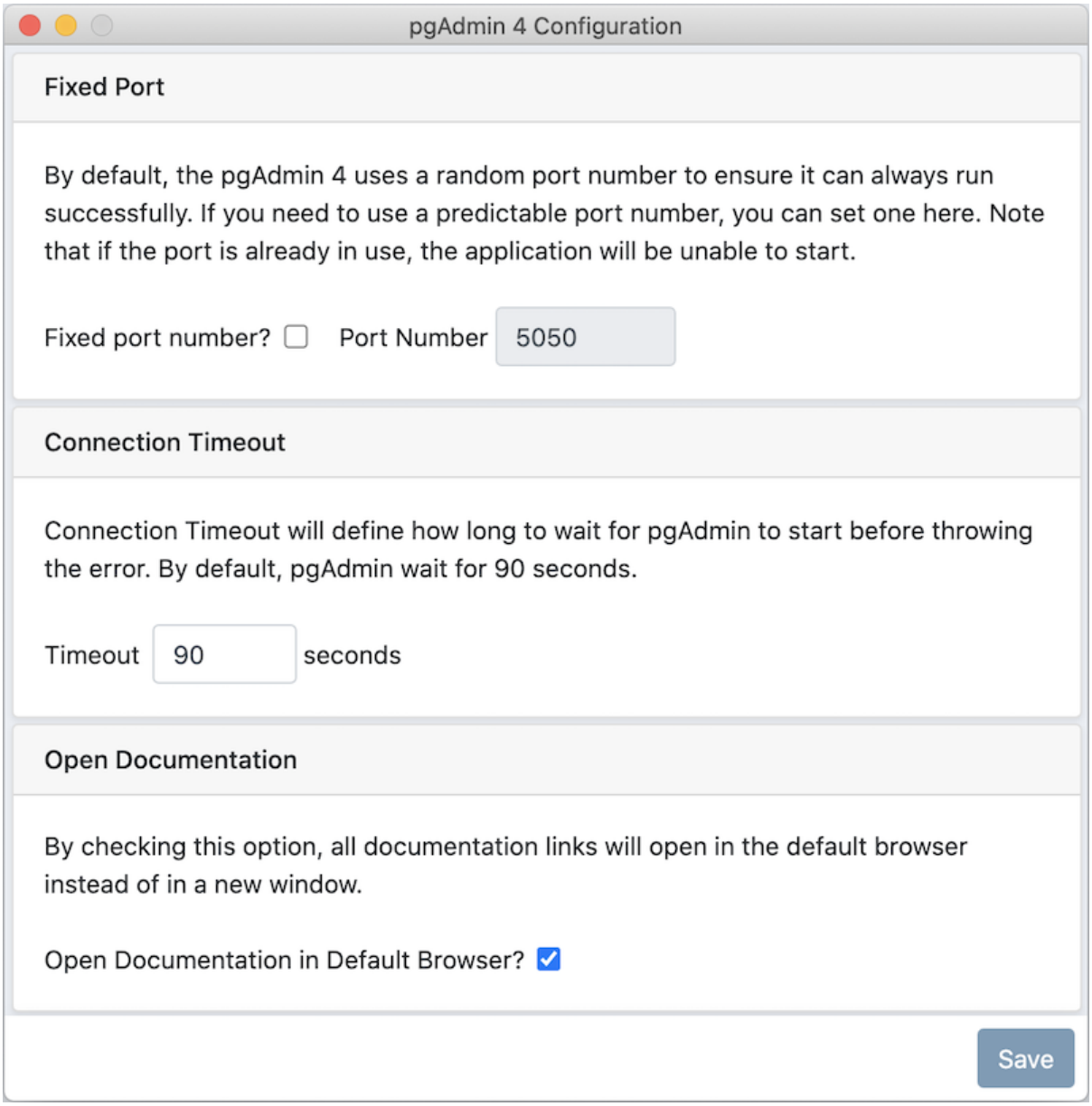


ファイルメニュー を使用して、 **ランタイムメニュー** にアクセスします。

オプション	アクション
設定...	クリックして構成ダイアログを開き、固定ポート、ポート番号、接続タイムアウトを構成します。
ログを表示...	クリックしてログの表示ダイアログを開き、pgAdmin4 ログを表示します。
フルスクリーンに入る	クリックして、フルスクリーンモードを開始/終了します。キーボードショートカット：OSX（Cmd+Ctrl+F）、その他の OS（F10）。
＊実寸＊	クリックしてウィンドウサイズを元のサイズに変更します。キーボードショートカット：OSX（Cmd+0）、その他の OS（Ctrl+0）。
ズームイン	クリックしてズームレベルを上げます。キーボードショートカット：OSX（Cmd++）、その他の OS（Ctrl++）。
ズームアウト	クリックしてズームレベルを下げます。キーボードショートカット：OSX（Cmd+-）、その他の OS（Ctrl+-）。

設定ダイアログ

ランタイムメニュー を使用して、 **構成** ダイアログにアクセスします。



以下は、**固定ポート番号**? 、**ポート番号** 、**接続タイムアウト** 、および「デフォルトのブラウザーでドキュメントを開く」の詳細です。構成パラメーター：

キー	タイプ	目的
固定ポート	ブール	ランダムではなく固定のネットワークポート番号を使用します。
ポート番号	整数	固定ポートを使用する場合、使用するポート番号。
接続タイムアウト	整数	アプリケーションサーバーの起動を待機する秒数。

次のページに続く

表 1 – 前のページからの続き

キー	タイプ	目的
デフォルトのブラウザーでドキュメントを開く	ブール	このオプションをチェックすると、すべてのドキュメントリンクが新しいウィンドウではなくデフォルトのブラウザーで開きます。

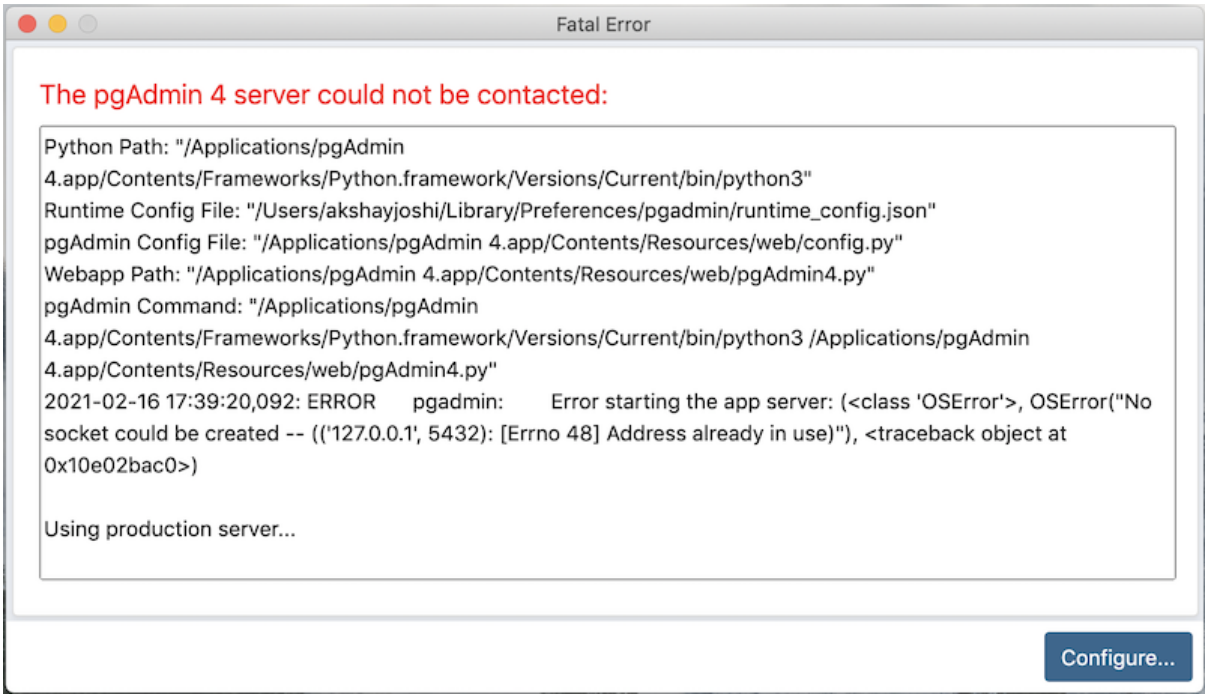
ログダイアログ

ランタイムメニュー を使用して、 **ログ** ダイアログにアクセスします。



下部の **リロード** ボタンをクリックして、pgAdmin4 サーバーの最新のログを表示します。

実行すると、ランタイムは自動的に pgAdminPython アプリケーションを実行しようとします。実行に失敗した場合、下部に **構成** ボタンを表示するエラーメッセージが表示されます。固定ポート番号を構成して、他のアプリケーションとのデフォルトのランダムポート番号の衝突を回避し、必要に応じて接続タイムアウトを構成できます。



エラーが PythonPath または pgAdminPython ファイルに関連している場合は、「dev_config.json」という名前のファイルを作成し、次のエントリを指定する必要があります。

```
{
    "pythonPath": "/path/to/python.exe",
    "pgadminFile": "/path/to/pgAdmin4.py"
}
```

dev_config.py ファイルは、標準インストールの外部で作業している開発者のみが必要であることに注意してください。

構成設定は *runtime_config.json* ファイルに保存されます。これは、Unix システム（~/.local/share/pgadmin/）、MacOSX（~/Library/Preferences/pgadmin）、および Windows（%APPDATA%/pgadmin）。

1.1.3 Server Deployment

pgAdmin は、サーバーモードで実行するようにアプリを構成し、リバースプロキシとして実行されている Web サーバーの背後に展開するか、WSGI インターフェイスを使用して展開できます。

サーバーモードで展開する場合、ユーザーには 2 つの重要な違いがあります。

- ユーザーは pgAdmin を使用する前にログインする必要があります。サーバーモードの初期化時に初期スーパーユーザーアカウントが作成され、このユーザーは必要に応じてスーパーユーザーと非スーパーユーザーを追加できます。
- ファイルストレージは、 STORAGE_DIR 構成パラメーターを使用して構成されたディレクトリの下の個々のユーザーの仮想ルートディレクトリに制限されています。ユーザーは、サーバーの完全なファイルシステムにアクセスできません。

次の手順は、 Apache HTTPD で mod_wsgi を使用して、 uWSGI または Gunicorn を使用してスタンドアロンで、 uWSGI または Gunicorn を使用して NGINX で pgAdmin を実行する方法を示しています。

参考:

pgAdmin をゼロからビルドして構成する手順の詳細については、ソースコードの最上位ディレクトリにある README ファイルを参照してください。便宜上、ファイル here の最新バージョンを見つけることができますが、これは特定のバージョンの pgAdmin のソースコードに含まれるバージョンとは異なる場合があることに注意してください。

要件

Important ：pgAdmin の一部のコンポーネントでは、クライアントセッションと特定のデータベース接続（たとえば、ユーザーが BEGIN コマンドに続けて多くの DMLSQL ステートメント、そして COMMIT を実行するクエリツール）。pgAdmin は、これを処理するためにビルトインの接続管理を使用して設計されていますが、クライアントセッションと複数の WSGI ワーカープロセスの 1 つとの間のアフィニティを維持することは簡単ではないため、単一の Python プロセスのみを使用する必要があります。

Windows システムでは、ApacheHTTP サーバーはシングルプロセス、マルチスレッドアーキテクチャを使用します。WSGI アプリケーションは embedded モードで実行されます。これは、すべての場合にこのプラットフォームに 1 つのプロセスのみが存在することを意味します。

Unix システムでは、ApacheHTTP サーバーは通常、マルチプロセス、シングルスレッドのアーキテクチャを使用します（これはコンパイル時に選択される MPM によって異なります）。WSGI アプリケーションに embedded モードが選択されている場合、Apache プロセスごとに 1 つの Python 環境があり、それぞれに独自の接続マネージャーがあり、接続アフィニティが失われます。したがって、単一のプロセスを使用するように構成された mod_wsgi の daemon モードを使用する必要があります。これにより、すべての Apache ワーカープロセスで使用される WSGI アプリケーションの単一のインスタンスが起動されます。

これが潜在的なパフォーマンスのボトルネックであることは事実ですが、実際には pgAdmin は、ビジーな Web サイトとは異なり、大量のトラフィックが発生する可能性が高い Web アプリケーションではないため、実際には問題になりません。

pgAdmin の将来のバージョンでは、この制限を克服するために共有接続マネージャープロセスが導入される可能性がありますが、これはかなりの量の作業であり、実際的なメリットはほとんどありません。

構成

サーバーモードで実行するように pgAdmin を構成するには、マルチユーザーモードで実行するように Python コードを構成し、コードを見つけて実行するように Web サーバーを構成する必要があります。

構成設定の詳細については、 *The config.py File* を参照してください。

パイソン

pgAdmin4v2 以降、サーバーモードがデフォルト構成です。デスクトップランタイムで実行している場合、これは自動的にオーバーライドされます。通常、サーバーモードを有効にするだけで構成を変更する必要はありませんが、使用されるパスの一部を調整することが望ましい場合があります。

Python コードを構成するには、次の手順に従います。

1. 既存の config.py ファイルと一緒に config_local.py ファイルを作成します。
2. config_local.py を編集し、以下の設定を追加します。ほとんどの場合、デフォルトのファイルの場所が適切です。

   *注：指定されたディレクトリが、Web サーバープロセスを実行するユーザーが書き込み可能であることを確認する必要があります。apache または www-data。DATA_DIR を指定して、DATA_DIR フォルダーの下に必要なすべてのディレクトリとファイルを作成できます。

```
LOG_FILE = '/var/log/pgadmin4/pgadmin4.log'
SQLITE_PATH = '/var/lib/pgadmin4/pgadmin4.db'
SESSION_DB_PATH = '/var/lib/pgadmin4/sessions'
STORAGE_DIR = '/var/lib/pgadmin4/storage'
AZURE_CREDENTIAL_CACHE_DIR = '/var/lib/pgadmin4/azurecredentialcache'
KERBEROS_CCACHE_DIR = '/var/lib/pgadmin4/kerberoscache'
```

4. 次のコマンドを実行して、構成データベースを作成します。

```
# python setup.py
```

5. たとえば、Web サーバーがグループ www-data のユーザー www-data として実行され、SQLite パスが /var/lib/pgadmin4/pgadmin4.db であると仮定します。

```
# chown www-data:www-data /var/lib/pgadmin4/pgadmin4.db
```

ホスティング

サーバーモードで pgAdmin をホストするには多くの方法があります。いくつかの例を以下に示します。

ApacheHTTPD 構成（Windows）

mod_wsgi をサポートするように ApacheHTTP を構成したら、pgAdmin アプリケーションを以下の例と同様に構成します。

```
<VirtualHost *>
    ServerName pgadmin.example.com
    WSGIScriptAlias / "C:\Program Files\pgAdmin4\web\pgAdmin4.wsgi"
    <Directory "C:\Program Files\pgAdmin4\web">
            Order deny,allow
```

(次のページに続く)

(前のページからの続き)

```
            Allow from all
    </Directory>
</VirtualHost>
```

お気に入りのエディターでファイル C:/Program Files/pgAdmin4/web/pgAdmin4.wsgi を開き、Apache サーバーの実行時に Python 仮想環境をアクティブにする以下のコードを追加します。

```
activate_this = 'C:\Program Files\pgAdmin4\venv\Scripts\activate_this.py'
exec(open(activate_this).read())
```

Note: pgAdmin4.wsgi ファイルに加えられた変更は、pgAdmin4 がアップグレードまたはダウングレードされると元に戻ります。

ApacheHTTPD 設定（Linux/Unix）

mod_wsgi をサポートするように ApacheHTTP を構成したら、pgAdmin アプリケーションを以下の例と同様に構成します。

```
<VirtualHost *>
    ServerName pgadmin.example.com

    WSGIDaemonProcess pgadmin processes=1 threads=25 python-home=/path/to/python/
↪virtualenv
    WSGIScriptAlias / /opt/pgAdmin4/web/pgAdmin4.wsgi

    <Directory /opt/pgAdmin4/web>
        WSGIProcessGroup pgadmin
        WSGIApplicationGroup %{GLOBAL}
        Order deny,allow
        Allow from all
    </Directory>
</VirtualHost>
```

Note: ApacheHTTPD2.4 以降を使用している場合は、次の行を置き換えます。

```
Order deny,allow
Allow from all
```

と：

```
Require all granted
```

アクセス制御の要件に合わせて、必要に応じて調整します。

スタンドアロンの Gunicorn 構成

pgAdmin は、以下に示すようなコマンドを実行するだけで、Gunicorn で直接ホストできます。この例では、PythonWheel を使用して pgAdmin がインストールされたことを前提としています（インストールに合わせてパスを調整する必要がある場合があります）。

```
gunicorn  --bind 0.0.0.0:80 \
          --workers=1 \
          --threads=25 \
          --chdir /usr/lib/python3.7/dist-packages/pgadmin4 \
          pgAdmin4:app
```

スタンドアロンの uWSGI 構成

pgAdmin は、以下に示すようなコマンドを実行するだけで uWSGI で直接ホストできます。この例では、PythonWheel を使用して pgAdmin がインストールされたことを前提としています（インストールに合わせてパスを調整する必要がある場合があります）。

```
uwsgi --http-socket 0.0.0.0:80 \
      --processes 1 \
      --threads 25 \
      --chdir /usr/lib/python3.7/dist-packages/pgadmin4/ \
      --mount /=pgAdmin4:app
```

Gunicorn を使用した NGINX 設定

pgAdmin は、NGINX を前にした Gunicorn でホストできます。これらの例では、PythonWheel を使用して pgAdmin がインストールされたことを前提としています（インストールに合わせてパスを調整する必要がある場合があります）。

サーバーのルートディレクトリで pgAdmin を実行するには、次のようなコマンドを使用して Gunicorn を起動します。

```
gunicorn --bind unix:/tmp/pgadmin4.sock \
         --workers=1 \
         --threads=25 \
         --chdir /usr/lib/python3.7/dist-packages/pgadmin4 \
         pgAdmin4:app
```

そして NGINX を構成します。

```
location / {
    include proxy_params;
    proxy_pass http://unix:/tmp/pgadmin4.sock;
}
```

または、サーバー上のサブディレクトリ（この場合は/pgadmin4）で pgAdmin をホストできます。ルートディレクトリを使用する場合と同じように Gunicorn を起動しますが、次のように NGINX を構成します。

```
location /pgadmin4/ {
    include proxy_params;
    proxy_pass http://unix:/tmp/pgadmin4.sock;
    proxy_set_header X-Script-Name /pgadmin4;
}
```

uWSGI を使用した NGINX 設定

pgAdmin は、NGINX を前に置いて uWSGI でホストできます。これらの例では、PythonWheel を使用して pgAdmin がインストールされたことを前提としています（インストールに合わせてパスを調整する必要がある場合があります）。

サーバーのルートディレクトリで pgAdmin を実行するには、次のようなコマンドを使用して uWSGI を起動します。

```
uwsgi --socket /tmp/pgadmin4.sock \
      --processes 1 \
      --threads 25 \
      --chdir /usr/lib/python3.7/dist-packages/pgadmin4/ \
      --manage-script-name \
      --mount /=pgAdmin4:app
```

そして NGINX を構成します。

```
location / { try_files $uri @pgadmin4; }
location @pgadmin4 {
    include uwsgi_params;
    uwsgi_pass unix:/tmp/pgadmin4.sock;
}
```

または、サーバー上のサブディレクトリ（この場合は/pgadmin4）で pgAdmin をホストできます。 mount パラメーターでディレクトリ名が指定されていることに注意して、uWSGI を起動します。

```
uwsgi --socket /tmp/pgadmin4.sock \
      --processes 1 \
      --threads 25 \
      --chdir /usr/lib/python3.7/dist-packages/pgadmin4/ \
      --manage-script-name \
      --mount /pgadmin4=pgAdmin4:app
```

次に、NGINX を構成します。

```
location = /pgadmin4 { rewrite ^ /pgadmin4/; }
location /pgadmin4 { try_files $uri @pgadmin4; }
```

(次のページに続く)

(前のページからの続き)

```
location @pgadmin4 {
  include uwsgi_params;
  uwsgi_pass unix:/tmp/pgadmin4.sock;
}
```

追加情報

注釈: pgAdmin は追加の Python プロセスを時々生成し、これを行うには Python の *sys.executable* 変数に依存しています。場合によっては、WSGI ホストプロセスの代わりに正しいインタープリターが使用されるように、その値をオーバーライドする必要があります。たとえば、uWSGI はこれを実現するための *--py-sys-executable* コマンドラインオプションを提供します。

1.1.4 Container Deployment

pgAdmin は、次のイメージを使用してコンテナにデプロイできます。

https://hub.docker.com/r/dpage/pgadmin4/

Docker を使用している場合は、次のようなコマンドを使用して、必要な pgAdmin のバージョンを取得するために選択できるさまざまなタグがあります。

```
docker pull dpage/pgadmin4:<tag name>
```

ここで、 *<tagname>* は次のいずれかです。

タグ名	説明
最新の	最新のリリース。
6.14	特定のバージョン（この場合は 6.14）。
6	特定のメジャーバージョン（この場合はメジャーバージョン 6）の最新リリース。
スナップショット	最新のナイトリーテストビルド。

PostgreSQL ユーティリティ

PostgreSQL ユーティリティ *pg_dump* 、 *pg_dumpall* 、 *pg_restore* および *psql* がコンテナに含まれており、バックアップの作成と復元、およびその他のメンテナンス機能を実行できます。さまざまなバージョンのデータベースサーバで使用できるように、複数のバージョンが次のディレクトリに含まれています。

- PostgreSQL10： */usr/local/pgsql-10*
- PostgreSQL11： */usr/local/pgsql-11*
- PostgreSQL12： */usr/local/pgsql-12*

- PostgreSQL13： */usr/local/pgsql-13*
- PostgreSQL14： */usr/local/pgsql-14*

コンテナに設定されているデフォルトのバイナリパスは次のとおりです。

```
DEFAULT_BINARY_PATHS = {
    'pg-15': '/usr/local/pgsql-15',
    'pg-14': '/usr/local/pgsql-14',
    'pg-13': '/usr/local/pgsql-13',
    'pg-12': '/usr/local/pgsql-12',
    'pg-11': '/usr/local/pgsql-11',
    'pg-10': '/usr/local/pgsql-10'
}
```

これは *Preferences Dialog* で変更される場合があります。

環境変数

コンテナは起動時に次の変数を受け入れます。

PGADMIN_DEFAULT_EMAIL

これは、pgAdmin にログインするための初期管理者アカウントを設定するときに使用されるメールアドレスです。この変数は必須であり、起動時に設定する必要があります。

PGADMIN_DEFAULT_PASSWORD

これは、pgAdmin にログインするための初期管理者アカウントを設定するときに使用されるパスワードです。この変数は必須であり、起動時に設定する必要があります。

PGADMIN_DEFAULT_PASSWORD_FILE

これは、pgAdmin にログインするための初期管理者アカウントを設定するときに使用されるパスワードです。パスワードを設定するには、この値を *dockersecret* に設定する必要があります。この変数は、dockerswarm 環境で、または dockercompose でコンテナを作成中にサポートされます。PGADMIN_DEFAULT_PASSWORD または PGADMIN_DEFAULT_PASSWORD_FILE 変数は必須であり、起動時に設定する必要があります。

PGADMIN_DISABLE_POSTFIX

デフォルト：*<null>*

設定しない場合、Postfix サーバーが起動してパスワードリセットメールを配信します。

任意の値に設定されている場合、Postfix サーバーは起動されず、以下の *PGADMIN_CONFIG_* オプションを使用して外部メールサーバーを使用するように pgAdmin を構成する必要があります。

このオプションは、sudo を使用して Postfix を起動できない環境で実行している場合、または外部のメールサーバーを使用したい場合に役立ちます。

PGADMIN_ENABLE_TLS

デフォルト：*<null>*

設定しない場合、コンテナはポート 80 でプレーンテキストの接続を監視します。任意の値に設定すると、コンテナはポート 443 で TLS 接続を監視します。

TLS が有効になっている場合、証明書とキーを提供する必要があります。通常、これらはホストファイルシステムに保存し、コンテナからマウントする必要があります。予想されるパスは/certs/server.cert および/certs/server.key

PGADMIN_LISTEN_ADDRESS

デフォルト：*[::]*

サーバーがリッスンするローカルアドレスを指定します。デフォルトはほとんどのユーザーにとって機能するはずです。IPv4 のみの環境では、これを 0.0.0.0 に設定する必要がある場合があります。

PGADMIN_LISTEN_PORT

デフォルト：*80* または *443*（*TLS* が有効な場合）

サーバーがリッスンするポートをデフォルトを使用するのではなく特定の値に設定できるようにします。

PGADMIN_SERVER_JSON_FILE

デフォルト：*/pgadmin4/servers.json*

サーバー定義リストのデフォルトのファイルパスをオーバーライドします。詳細については、以下の/pgadmin4/servers.json マッピングされたファイルを参照してください。 JSON file のフォーマットを参照してください。

GUNICORN_ACCESS_LOGFILE

デフォルト：-（stdout）

Gunicorn アクセスログを stdout に送信する代わりに、出力ファイルを指定します。

GUNICORN_LIMIT_REQUEST_LINE

デフォルト：*8190*

HTTP リクエスト行の最大サイズをバイト単位で設定します。デフォルトでは、一部の要求は非常に大きくなる可能性があるため、pgAdmin コンテナは Gunicorn が提供する最大制限サイズを使用します。例外的に、この値を 0（ゼロ）に設定して「無制限」を指定できますが、これによりサービス拒否の危険が発生します。

GUNICORN_THREADS

デフォルト：*25*

Gunicorn サーバーが着信要求を処理するために使用するスレッドの数を調整します。これは、増加する可能性のある高負荷のシステムを除き、通常はそのままにしておく必要があります。

PGADMIN_CONFIG_ *

これは、pgAdmin の *config.py* ファイルの構成オプションをオーバーライドするために使用できる変数プレフィックスです。 *config.py* の変数名に *PGADMIN_CONFIG_* プレフィックスを追加し、文字列の場合は「文字列値」、ブール値の場合は True/False、数値の場合は 123 の形式で値を指定します。例については、以下を参照してください。

設定は、コンテナ内の */pgadmin4/config_distro.py* に書き込まれ、 */pgadmin4/config.py* の後、 */pgadmin4/config_local.py* の前に読み取られます。したがって、指定された設定は config.py のすべてをオーバーライドしますが、config_local.py の設定でオーバーライドできます。

設定は、通常、コンテナの最初の起動時に 1 回のみ */pgadmin4/config_distro.py* に書き込まれます。 */pgadmin4/config_distro.py* に 1 つ以上の行が含まれている場合、変更は行われません。たとえば、コンテナインスタンスが再起動された場合、または */pgadmin4/config_distro.py* が永続ストレージ上のファイルにマップされている場合（非推奨-代わりに */pgadmin4/config_local.py* を使用してください）！

使用可能な構成設定の詳細については、 *The config.py File* を参照してください。

マップされたファイルとディレクトリ

次のファイルまたはディレクトリをコンテナからホストマシンにマッピングして、構成をカスタマイズしてインスタンス間で共有できます。

警告: 警告：pgAdmin は、コンテナの *pgadmin* グループ（GID：5050）の *pgadmin* ユーザー（UID：5050）として実行されます。すべてのファイルがホストマシンでこのユーザーに対して読み取り可能であり、必要な場合（作業/セッションディレクトリなど）に書き込み可能であることを確認する必要があります。例：

```
sudo chown -R 5050:5050 <host_directory>
```

拡張属性をサポートしていない一部のファイルシステムでは、 *PGADMIN_LISTEN_PORT* に 1024 より大きい値を指定せずに pgAdmin を実行できない場合があります。そのような場合、環境変数を追加してコンテナを起動するときに代替ポートを指定し、例：

```
-e 'PGADMIN_LISTEN_PORT=5050'
```

それに応じてホストコンテナポートマッピングを調整することを忘れないでください。

/var/lib/pgadmin

これは、pgAdmin がセッションデータ、ユーザファイル、構成ファイル、および構成データベースを保存する作業ディレクトリです。このディレクトリをホストマシンにマッピングすると、コンテナの呼び出し間で構成を簡単に維持できます。

/pgadmin4/config_local.py

このファイルを使用して、pgAdmin の構成設定をオーバーライドできます。config.py にある設定は、必要に応じて展開固有の値でオーバーライドできます。config_local.py の設定は、 *PGADMIN_CONFIG_* プレフィックス付き変数を介してコンテナ環境で指定されたものもオーバーライドします。

/pgadmin4/servers.json

このファイルがマップされている場合、その中にあるサーバー定義が起動時にロードされます。これにより、接続情報をコンテナ内の pgAdmin のインスタンスにプリロードできます。サーバー定義は、最初の起動、つまり構成データベースの作成時にのみロードされ、同じ構成データベースを使用した後続の起動ではロードされないことに注意してください。

/certs/server.cert

TLS が有効になっている場合、このファイルはサーバーの TLS 証明書として使用されます。

/certs/server.key

TLS が有効になっている場合、このファイルはサーバーの TLS 証明書のキーファイルとして使用されます。

例

ポート 80 で単純なコンテナーを実行します。

```
docker pull dpage/pgadmin4
docker run -p 80:80 \
    -e 'PGADMIN_DEFAULT_EMAIL=user@domain.com' \
    -e 'PGADMIN_DEFAULT_PASSWORD=SuperSecret' \
    -d dpage/pgadmin4
```

いくつかの構成オプションを設定して、ポート 80 で単純なコンテナーを実行します。

```
docker pull dpage/pgadmin4
docker run -p 80:80 \
    -e 'PGADMIN_DEFAULT_EMAIL=user@domain.com' \
    -e 'PGADMIN_DEFAULT_PASSWORD=SuperSecret' \
    -e 'PGADMIN_CONFIG_ENHANCED_COOKIE_PROTECTION=True' \
    -e 'PGADMIN_CONFIG_LOGIN_BANNER="Authorised users only!"' \
    -e 'PGADMIN_CONFIG_CONSOLE_LOG_LEVEL=10' \
    -d dpage/pgadmin4
```

ホスト上の/private/var/lib/pgadmin、およびホスト上の/tmp/servers.json からプリロードされたサーバーを使用して、TLS で保護されたコンテナを実行します。

```
docker pull dpage/pgadmin4
docker run -p 443:443 \
    -v /private/var/lib/pgadmin:/var/lib/pgadmin \
    -v /path/to/certificate.cert:/certs/server.cert \
    -v /path/to/certificate.key:/certs/server.key \
    -v /tmp/servers.json:/pgadmin4/servers.json \
    -e 'PGADMIN_DEFAULT_EMAIL=user@domain.com' \
    -e 'PGADMIN_DEFAULT_PASSWORD=SuperSecret' \
    -e 'PGADMIN_ENABLE_TLS=True' \
    -d dpage/pgadmin4
```

リバースプロキシ

場合によっては、ユーザーが実行されているコンテナに直接ではなく、リバースプロキシを介して pgAdmin に接続することが望ましいです。次の例は、これを実現する方法を示しています。 Nginx などの従来のリバースプロキシサーバーでは、pgAdmin は同じホスト上のコンテナで実行されており、ホストのポート 5050 はコンテナのポート 80 にマッピングされています。

```
docker pull dpage/pgadmin4
docker run -p 5050:80 \
    -e "PGADMIN_DEFAULT_EMAIL=user@domain.com" \
    -e "PGADMIN_DEFAULT_PASSWORD=SuperSecret" \
    -d dpage/pgadmin4
```

pgAdminX-Forwarded-*構成

pgAdmin は、どの値を信頼するかを知るために、各ヘッダーを設定するプロキシの数を理解する必要があります。この目的に使用される X-Forwarded-*オプションの構成パラメーターをデフォルト値とともに示します。

pgAdmin は、非標準ポートでもリバースプロキシの背後で実行できるように構成されており、通常、これらの構成オプションを変更する必要はありません。通常とは異なる構成（複数のリバースプロキシなど）を実行している場合は、適合するように構成を調整できます。

```
# Number of values to trust for X-Forwarded-For
PROXY_X_FOR_COUNT = 1

# Number of values to trust for X-Forwarded-Proto.
PROXY_X_PROTO_COUNT = 0

# Number of values to trust for X-Forwarded-Host.
PROXY_X_HOST_COUNT = 0

# Number of values to trust for X-Forwarded-Port.
PROXY_X_PORT_COUNT = 1

# Number of values to trust for X-Forwarded-Prefix.
PROXY_X_PREFIX_COUNT = 0
```

Nginx を介した HTTP

次のような構成を使用して、 Nginx ですべてのホスト名をリッスンする単純な HTTP リバースプロキシを作成できます。

```
server {
    listen 80;
    server_name _;

    location / {
        proxy_set_header Host $host;
        proxy_pass http://localhost:5050/;
        proxy_redirect off;
    }
}
```

サーバーのルートではなくサブディレクトリの下で pgAdmin をホストする場合は、場所を指定し、パスを書き換える方法を pgAdmin コンテナに指示する *X-Script-Name* ヘッダーを設定する必要があります。

```
server {
    listen 80;
    server_name _;

    location /pgadmin4/ {
        proxy_set_header X-Script-Name /pgadmin4;
        proxy_set_header Host $host;
        proxy_pass http://localhost:5050/;
        proxy_redirect off;
    }
}
```

Nginx もコンテナで実行されている場合、2 つのコンテナが同じ Docker ネットワークで実行されている限り、pgAdmin ポートをホストにマッピングする必要はありません。このような構成では、 *proxy_pass* オプションは、Docker ネットワーク内の pgAdmin コンテナを指すように変更されます。

Nginx を介した HTTPS

次の構成を使用して、バックエンドコンテナがプロキシサーバーにプレーン HTTP を提供しているときに、HTTPS を介して pgAdmin をユーザーに提供できます。この構成では、 *X-Script-Name* を設定するだけでなく、 *X-Scheme* も設定して、正しいスキームを使用して URL を生成するように pgAdmin サーバーに指示します。HTTP から HTTPS へのリダイレクトも含まれています。証明書とキーのパスは、特定の展開に合わせて調整する必要がある場合があります。

```
server {
    listen 80;
    return 301 https://$host$request_uri;
```

(次のページに続く)

(前のページからの続き)

```
}

server {
    listen 443;
    server_name _;

    ssl_certificate /etc/nginx/server.cert;
    ssl_certificate_key /etc/nginx/server.key;

    ssl on;
    ssl_session_cache builtin:1000 shared:SSL:10m;
    ssl_protocols TLSv1 TLSv1.1 TLSv1.2;
    ssl_ciphers HIGH:!aNULL:!eNULL:!EXPORT:!CAMELLIA:!DES:!MD5:!PSK:!RC4;
    ssl_prefer_server_ciphers on;

    location /pgadmin4/ {
        proxy_set_header X-Script-Name /pgadmin4;
        proxy_set_header X-Scheme $scheme;
        proxy_set_header Host $host;
        proxy_pass http://localhost:5050/;
        proxy_redirect off;
    }
}
```

トラフィク

Traefik の構成は、ローカルマシンで実行されているコンテナからコンテンツを提供するように自動的に構成され、 *<container_name>.<domain_name>* でそれらをホストするため、HTTP または HTTPS のいずれかで簡単ドメイン名は、Traefik 構成で指定されたものです。コンテナは通常、以下の例に従って起動されます。

```
docker pull dpage/pgadmin4
docker run --name "pgadmin4" \
    -e "PGADMIN_DEFAULT_EMAIL=user@domain.com" \
    -e "PGADMIN_DEFAULT_PASSWORD=SuperSecret" \
    -d dpage/pgadmin4
```

Nginx の例のように TCP/IP ポートはホストにマップされておらず、コンテナ名はホスト名として使用され、DNS に追加する必要があるため、既知の値に設定されていることに注意してくださいゾーンファイル。

次の構成は、Traefik に同梱されているデフォルトの証明書を使用して、ポート 80 と 443 でリッスンし、80 を 443 にリダイレクトします。LetsEncrypt または他の発行者からの証明書を使用するオプションについては、Traefik のドキュメントを参照してください。

```
defaultEntryPoints = ["http", "https"]

[entryPoints]
  [entryPoints.http]
    address = ":80"
      [entryPoints.http.redirect]
        entryPoint = "https"
  [entryPoints.https]
    address = ":443"
      [entryPoints.https.tls]

[docker]
domain = "domain_name"
watch = true
```

Traefik を使用してサブディレクトリで pgAdmin をホストする場合、構成の変更は通常、Traefik 自分自身ではなくコンテナの起動方法に対して行われます。たとえば、ルートディレクトリではなく */pgadmin4/* で pgAdmin をホストするには、Traefik のバージョン v1 の使用中にコンテナが次のように起動された場合、上記の Traefik 構成を使用できます。

```
docker pull dpage/pgadmin4
docker run --name "pgadmin4" \
    -e "PGADMIN_DEFAULT_EMAIL=user@domain.com" \
    -e "PGADMIN_DEFAULT_PASSWORD=SuperSecret" \
    -e "SCRIPT_NAME=/pgadmin4" \
    -l "traefik.frontend.rule=PathPrefix:/pgadmin4" \
    -d dpage/pgadmin4
```

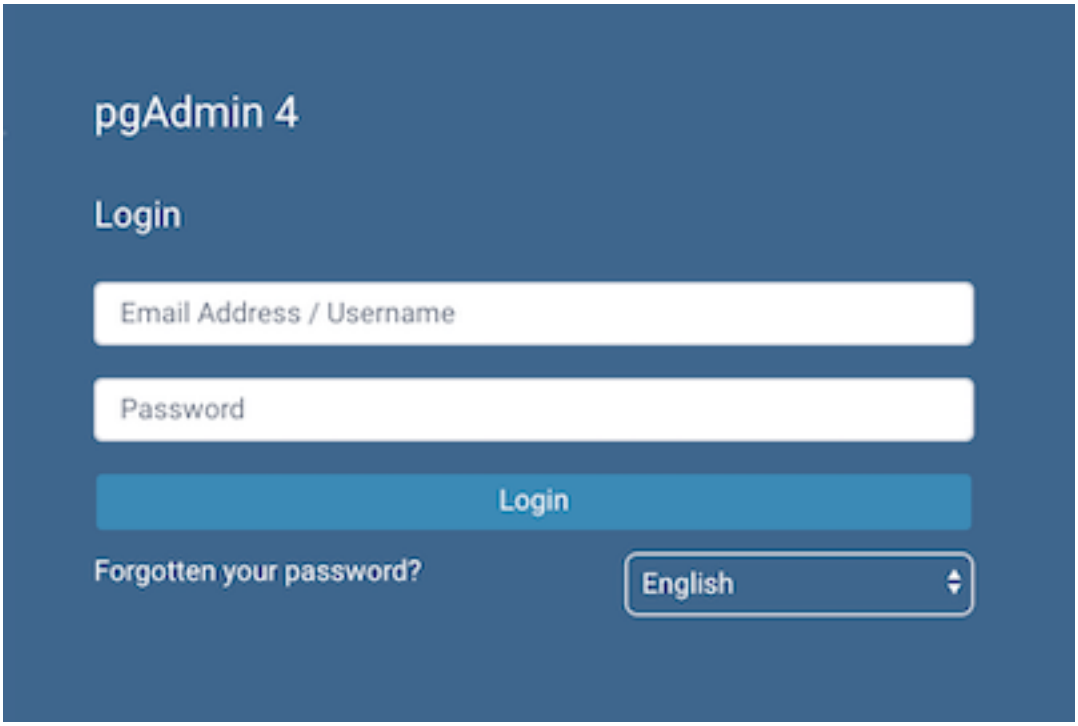
SCRIPT_NAME 環境変数は、サブディレクトリでホストされていることをコンテナに通知するように設定されており（*X-Script-Name* ヘッダーが Nginx で使用されるのと同じ方法で）、Traefik に伝えるラベルが追加されましたサブディレクトリの下のリクエストをこのコンテナにルーティングします。

サブディレクトリの下で pgAdmin をホストするためにバージョン v2 の Traefik 構成を使用している場合、コンテナは通常、以下の例に従って起動されます。

```
docker pull dpage/pgadmin4
docker run --name "pgadmin4" \
    -e "PGADMIN_DEFAULT_EMAIL=user@domain.com" \
    -e "PGADMIN_DEFAULT_PASSWORD=SuperSecret" \
    -e "SCRIPT_NAME=/pgadmin4" \
    -l "traefik.frontend.pgadmin4.rule=Host(`host.example.com`) && PathPrefix(`/
↪pgadmin4`)" \
    -d dpage/pgadmin4
```

1.2 Login Dialog

Login ダイアログを使用して、pgAdmin にログインします。



[*Login*] ダイアログのフィールドを使用して、接続を認証します。接続を認証するには 2 つの方法があります。

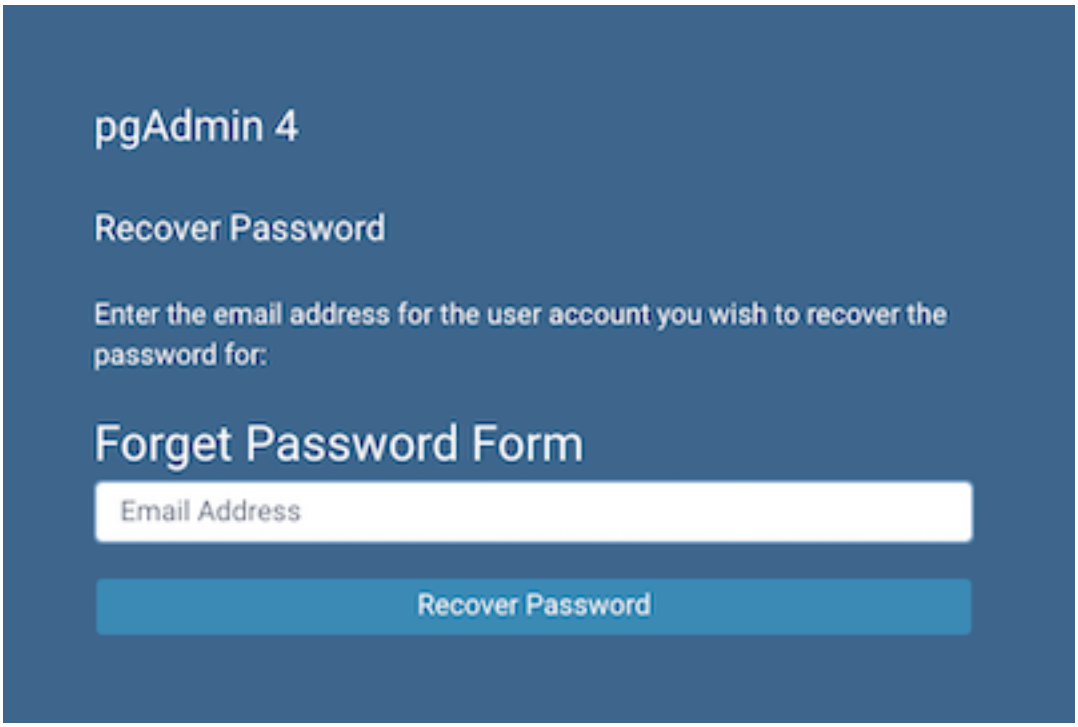
- pgAdmin バージョン 4.21 以降、LDAP 認証のサポートが追加されました。pgAdmin アプリケーションで LDAP 認証が有効になっている場合、LDAP 資格情報を使用して pgAdmin にログインできます。
  - **電子メールアドレス/ユーザー名** フィールドに LDAP ユーザー名を入力します。
  - [Password] フィールドに LDAP パスワードを入力します。
- または、次の情報を使用して pgAdmin にログインできます。
  - アカウントに関連付けられたメールアドレスを **メールアドレス/ユーザー名** フィールドに入力します。
  - *Password* フィールドにパスワードを入力します。

ログイン ボタンをクリックして、pgAdmin に安全にログインします。

pgAdmin サーバーを再起動すると、ログアウトされることに注意してください。続行するには再ログインする必要があります。

1.2.1 紛失したパスワードの回復

パスワードを入力できない場合は、 **パスワードをお忘れですか**? ボタンをクリックして、パスワード回復ユーティリティを起動します。



- アカウントに関連付けられたメールアドレスを **メールアドレス** フィールドに入力します。
- [パスワードの回復] ボタンをクリックして、回復を開始します。 **メールアドレス** フィールドに入力したアドレスに、パスワードのリセット方法が記載されたメールが送信されます。

アカウントに関連付けられたメールアドレスを忘れた場合は、管理者に連絡してください。

このダイアログでは LDAP パスワードを復元できないことに注意してください。[**電子メールアドレス/ユーザー名**] フィールドに LDAP ユーザー名を入力し、電子メールを入力してパスワードを回復すると、LDAP 管理者に連絡して LDAP パスワードを回復するように求めるエラーメッセージが表示されます。

1.2.2 ブルートフォース攻撃の回避

ログイン試行の最大数に達したら、 `MAX_LOGIN_ATTEMPTS` を設定することにより、アカウントをロックする可能性があります。値をゼロに設定することで、この機能を無効にできます。

1.3 Enabling two-factor authentication (2FA)

1.3.1 二要素認証について

二要素認証（2FA）は、Web サイトまたはアプリにログインするときに使用される追加のセキュリティレイヤーです。2FA では、ユーザー名とパスワードでログインし、自分だけが知っている、またはアクセスできる別の形式の認証を提供する必要があります。

1.3.2 二要素認証を設定する

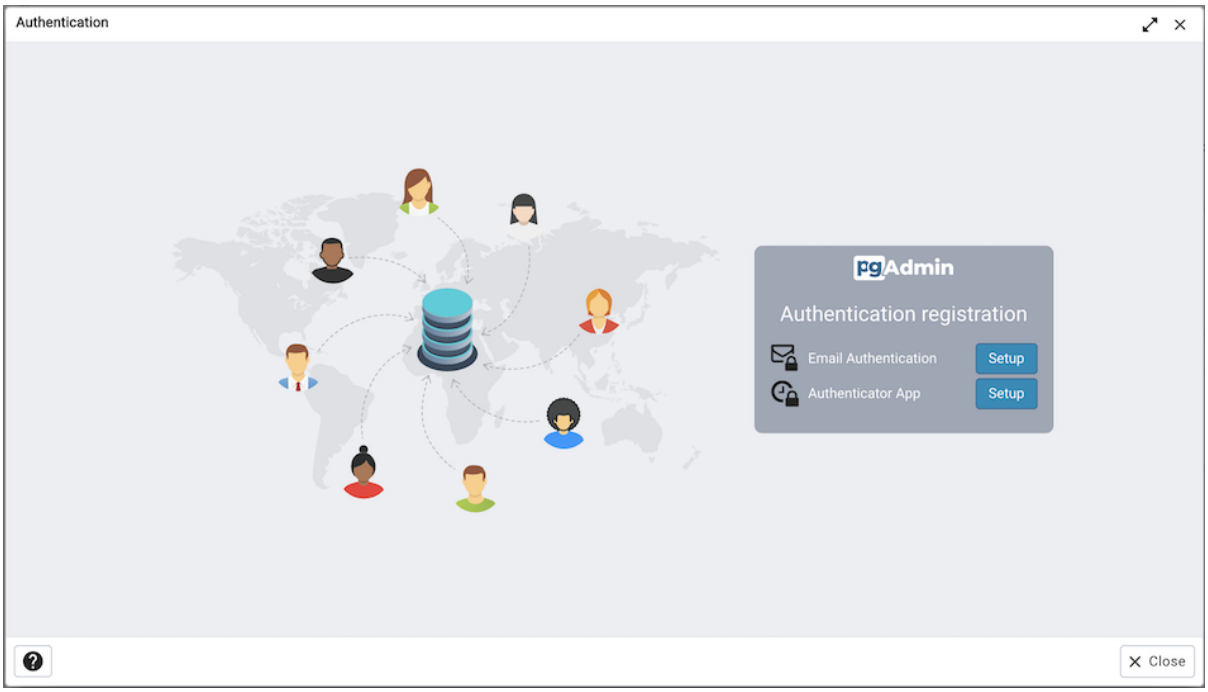
pgAdmin4 の 2FA をセットアップするには、pgAdmin がサーバーモードでインストールされているシステムの *config_local.py* または *config_system.py*（*config.py* ドキュメントを参照）で 2 要素認証設定を構成する必要があります。これらの設定を *config.py* ファイルからコピーし、次のパラメーターの値を変更できます。

Parameter	Description
MFA_ENABLED	このパラメーターのデフォルト値は True です。2FA を無効にするには、値を *False* に設定します
SUPPORTED_MFA_LIST	サポートする認証方法を設定する
MFA_EMAIL_SUBJECT	<APP_NAME>-確認コード pgAdmin4-検証コード
MFA_FORCE_REGISTRATION	ログイン時に認証方法を構成するようにユーザーに強制します（認証がまだ構成されていない場合）。

注：*2* 要素認証方法として「電子メール」を使用するには、*config_local.py* または *config_system.py* で「メールサーバー設定」を設定する必要があります（*config.py* *ドキュメントを参照）。

1.3.3 二要素認証を構成する

ユーザーの 2FA を構成するには、右上隅の `User` メニューで [二要素認証] をクリックする必要があります。サポートされているすべての多要素認証方法がリストされます。これらの方法のいずれかの [セットアップ] をクリックし、各認証方法の手順に従います。既に設定されている認証方法の `Delete` ボタンが表示されます。`Delete` ボタンをクリックすると、現在のユーザーの認証方法の登録が解除されます。



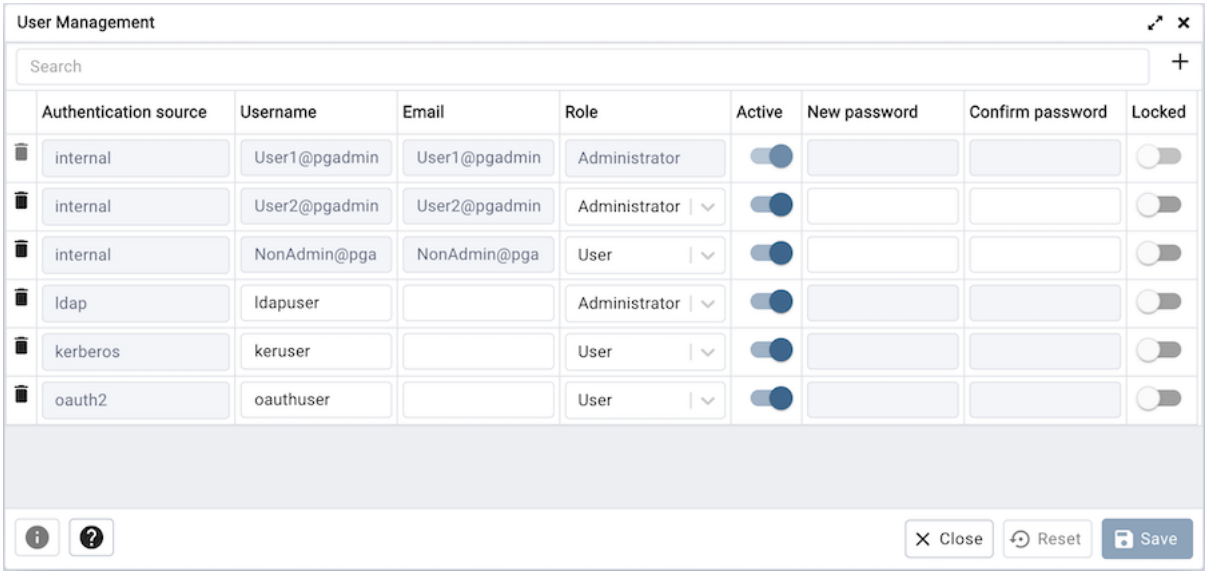
MFA_FORCE_REGISTRATION パラメーターを *True* に設定して、ユーザーにログイン時に 2 要素認証方法を構成するように強制することもできます。

1.4 User Management Dialog

デスクトップモードで pgAdmin を呼び出すと、パスワードがランダムに生成され、無視されます。サーバーモードで pgAdmin をインストールすると、pgAdmin クライアントの管理者のメールアドレスとパスワードの入力が求められます。

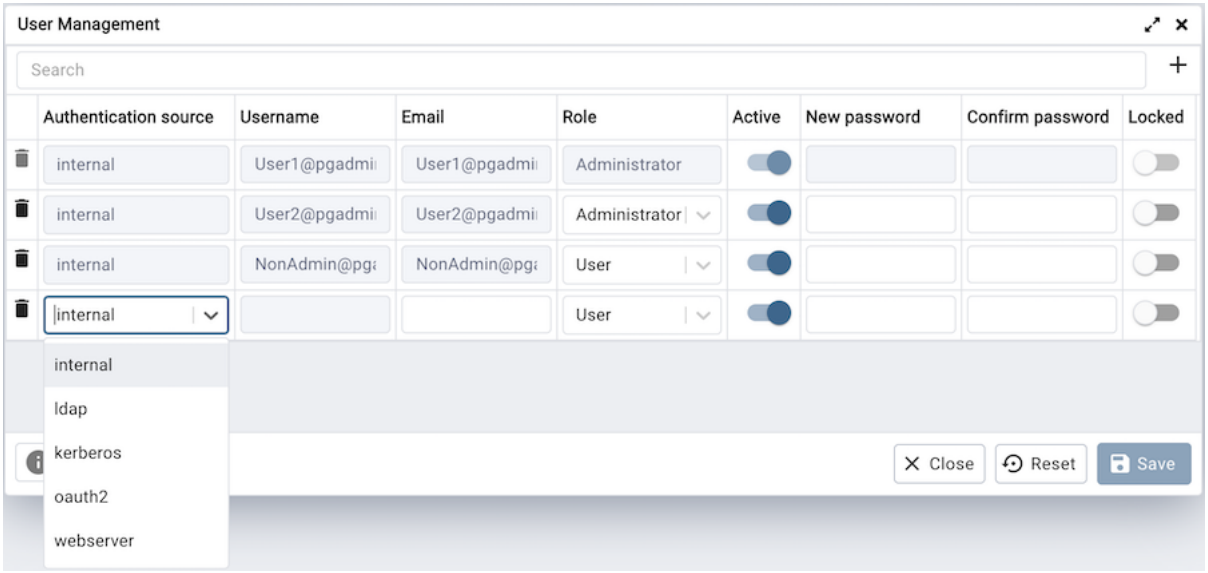
pgAdmin で認証すると、そのログインロールに関連付けられたサーバー定義がツリーコントロールで利用できるようになります。管理ユーザーは *UserManagement* ダイアログを使用して次のことができます。

- pgAdmin ロールを追加または削除します
- 特権を割り当てる
- ロールに関連付けられたパスワードを管理する



検索 フィールドを使用して、基準を指定し、指定した基準に一致するユーザーのリストを確認します。次の条件タイプに一致する値を入力できます。 **認証ソース** 、 **ユーザー名** 、または **電子メール** 。たとえば、検索ボックスに *ldap* と入力すると、認証元として *ldap* を持つレコードのみが *UserManagement* テーブルに表示されます。

ユーザーを追加するには、右上隅の追加（+）ボタンをクリックします。



行に新しい pgAdmin ロールに関する情報を入力します。

- **認証ソース** フィールドの横にあるドロップダウンリストボックスを使用して、ユーザーに使用する認証の種類を選択します。認証ソースが「internal」のみの場合、*Authenticationsource* フィールドは無効になります。サポートされている **認証ソース** は、internal、ldap、kerberos、oauth2、および webserver です。
- *Username* フィールドをクリックし、ユーザーのユーザー名を入力します。このフィールドは、*internal* 以外の認証ソースを選択した場合にのみ有効になります。認証元として *internal* を選択した場合、ユーザー名フィールドにメールアドレスが表示されます。
- **電子メール** フィールドをクリックし、ユーザーの電子メールアドレスを入力します。

- **ロール** の横にあるドロップダウンリストボックスを使用して、ユーザーが **管理者** か **ユーザー** かを選択します。
  - ユーザーが pgAdmin クライアント内で管理者権限を持つ場合は、 *Administrator* を選択します。
  - *User* を選択して、非管理者ユーザーアカウントを作成します。
- アカウントが現在アクティブでない場合は、 **アクティブ** スイッチを **いいえ** の位置に移動します。デフォルトは **はい** です。このスイッチを使用して、アカウントを削除せずにアカウントアクティビティを無効にします。
- **新しいパスワード** フィールドを使用して、 **電子メール** フィールドで指定されたユーザーに関連付けられたパスワードを入力します。 *internal* 以外の認証ソースを選択した場合、このフィールドは無効になります。
- **パスワードの確認** フィールドにパスワードを再入力します。認証元として *ldap* を選択した場合、このフィールドは無効になります。
- *Locked* スイッチは、 *False* に設定されている場合、デフォルトで無効になっています。これは、ユーザーがログインに失敗してロックされた場合にのみ有効になります。アカウントのロックを解除する場合は、スイッチを *False* の位置に移動します。

ユーザーを破棄し、pgAdmin へのアクセスを取り消すには、行の左側にあるごみ箱アイコンをクリックし、*Deleteuser?* ダイアログで削除を確認します。ユーザーが共有サーバーを作成している場合、共有サーバーの所有権を変更するための *Change Ownership* ダイアログが表示されます。

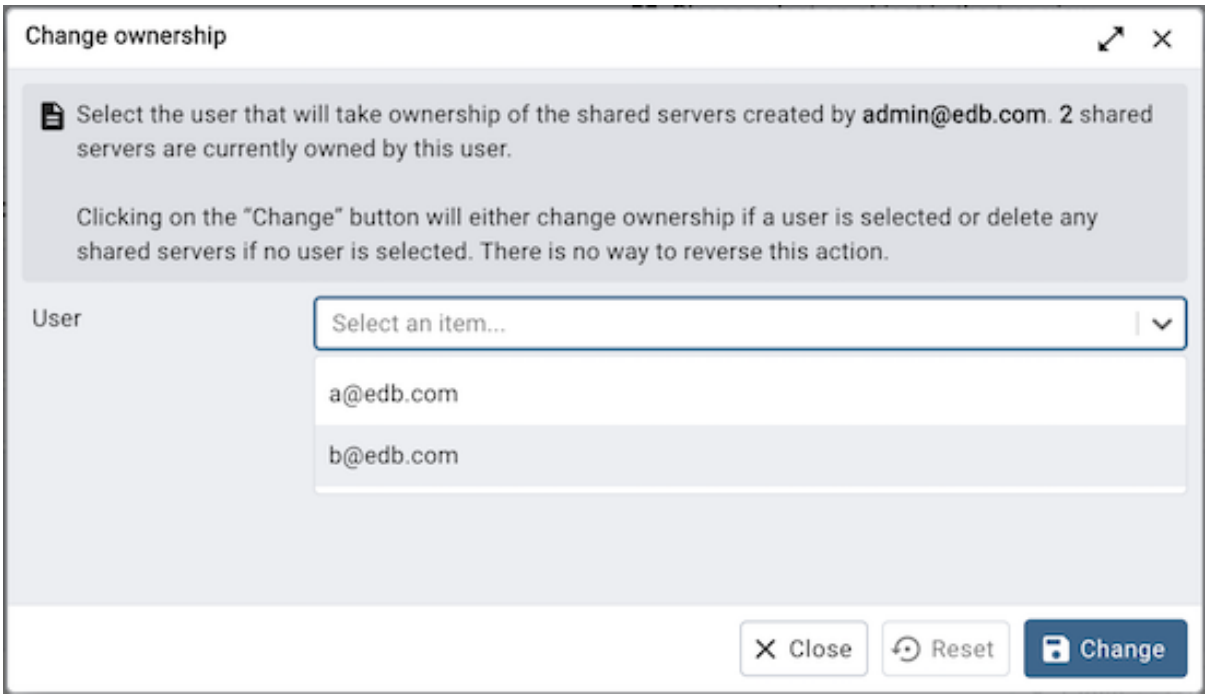
Administrator ロールを持つユーザーは、pgAdmin ユーザーを追加、編集、削除できますが、それ以外は *User* ロールを持つユーザーと同じです。

- オンラインヘルプにアクセスするには、 **ヘルプ** ボタン（?）をクリックします。
- **閉じる** ボタンをクリックして作業を保存します。選択を保存できない場合は、ダイアログに戻るように求められます。

1.5 Change Ownership Dialog

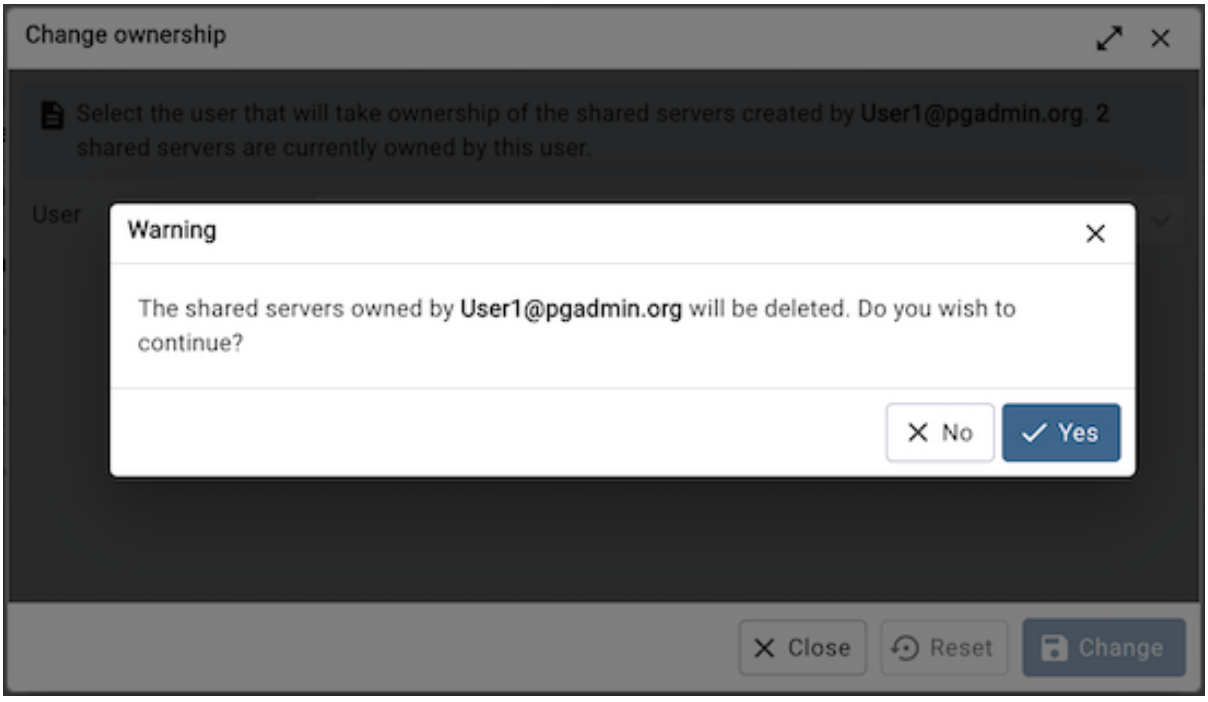
所有権の変更 ダイアログを使用して、共有サーバーの所有権を変更します。このダイアログは、ユーザーが *User Management* から削除され、いくつかの共有サーバーを所有している場合に表示されます。

ドロップダウンから共有サーバーを所有するユーザーを選択します。



所有権を変更するには、 *Change* ボタンをクリックします。

ドロップダウンからユーザーを選択しないと、ユーザーが所有する共有サーバーが削除されます。



変更 ボタンをクリックして所有権を変更します。 *Close* をクリックしてダイアログを終了します。

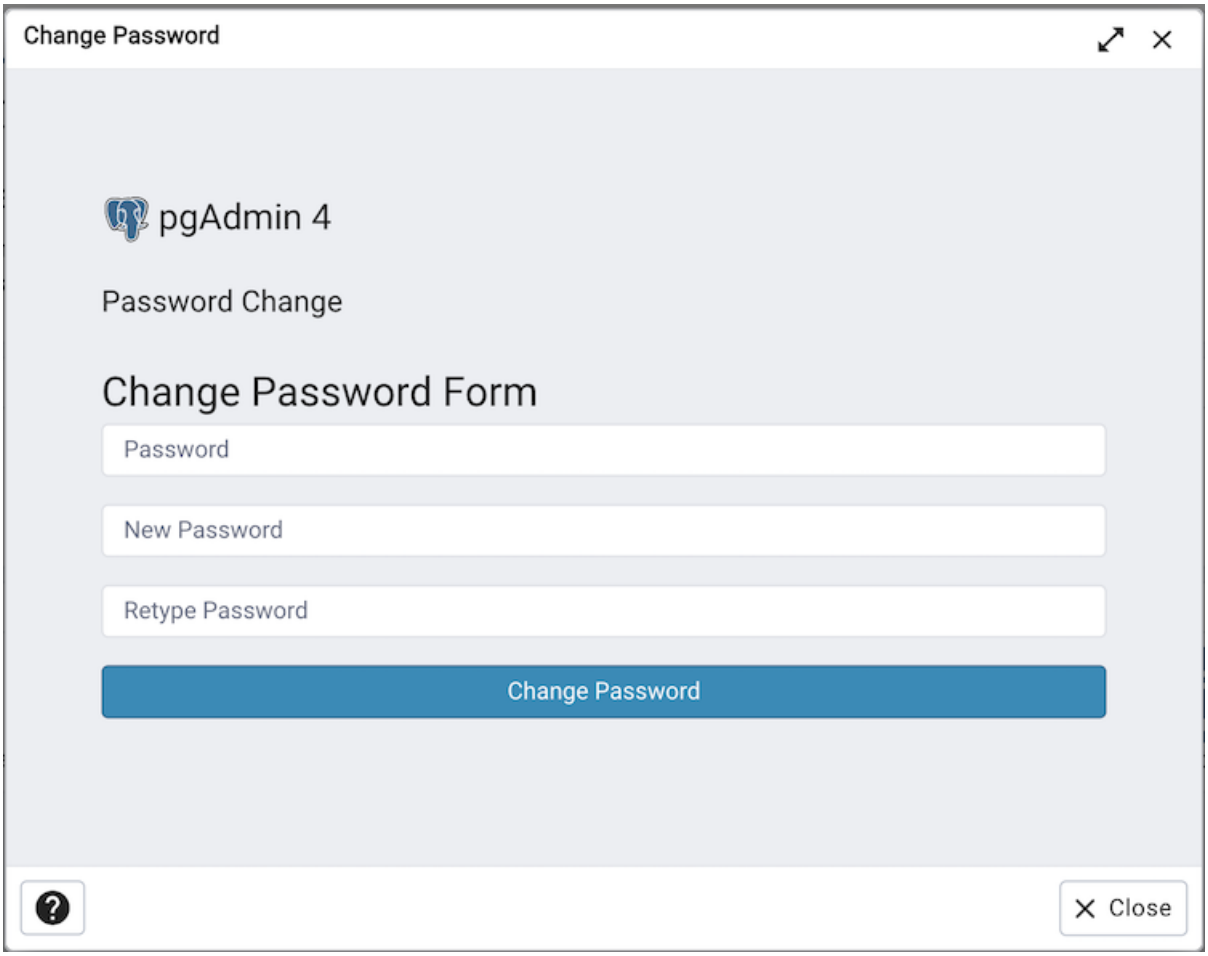
1.6 Change User Password Dialog

「安全」な環境でも、データを保護するためにパスワードを定期的に変更することをお勧めします。職場では、適切なパスワードポリシーを適用しないと、データ保護法に違反する可能性があります。

パスワードを選択するときは、次のガイドラインを考慮してください。

- パスワードが十分な長さであることを確認してください。6 文字は、パスワードの絶対最小文字数である必要があります。
- パスワードが辞書攻撃に対して開かれていないことを確認します。大文字、小文字、数字を組み合わせて使用し、単語や名前は避けてください。覚えやすいが聞き慣れない頭字語のフレーズを使って、各単語の最初の文字を使用することを検討してください。
- パスワードは定期的に変更してください。少なくとも 90 日ごとに交換してください。

上記のガイドラインは出発点と考えてください。それらは包括的なリストではなく、 **will not guarantee security** 。



[パスワードの変更] ダイアログを使用して、パスワードを変更します。

- **現在のパスワード** フィールドに既存のパスワードを入力します。
- **新しいパスワード** フィールドに目的のパスワードを入力します。
- *ConfirmPassword* フィールドに新しいパスワードを再入力します。

ChangePassword ボタンをクリックしてパスワードを変更します。 *Close* をクリックしてダイアログを終了します。

1.7 Lock/Restore Account

ユーザー管理ダイアログによるアカウントのロック/ロック解除：

管理者は、ユーザー管理ダイアログからユーザーをロックおよびロック解除できます。これにより、管理者はログインに数回失敗した場合にユーザーをロックまたは復元できます。このページでは、ユーザーごとのロック/復元の構成について説明します。詳細については、 *User management* をご覧ください。

Parameter	Description
MAX_LOGIN_ATTEMPTS	許可されるログインの失敗の数を設定します。この値を超えると、アカウントがロックされ、管理者がリセットできます。変数を値ゼロに設定すると、この機能は無効になります。

Administrator 自分自身がロックされた場合、復元するには次の手順が考えられます。

- MAX_LOGIN_ATTEMPTS を増やし、正常にログインしてアカウントを復元してみてください。

または

構成データベースを更新することにより

SQLiteDB（pgAdmin4.db）：

- pgAdmin4.db ファイルを見つけて、任意の DBBrowser（または DBBrowserforSQLite）を使用して開きます
- DB ファイルを開いたら、「ExecuteSQL」セクションに進みます。
- 以下のクエリを実行します-

UPDATEUSERSETLOCKED=false、LOGIN_ATTEMPTS=0WHEREUSERNAME=<YOUR_EMAIL_ID>

外部データベース：

- データベースに接続します。
- 以下のクエリを実行します-

UPDATEUSERSETLOCKED=false、LOGIN_ATTEMPTS=0WHEREUSERNAME=<YOUR_EMAIL_ID>

- クエリの変更がコミットされていることを確認します。

ログイン失敗によるアカウントのロック：



pgAdmin アプリケーションは、ログインに連続して失敗した回数を超えた場合にユーザーアカウントをロックするように構成されています。

MAX_LOGIN_ATTEMPTS のデフォルトは 3 回のログイン試行で、その後はアカウントがロックされます。

ユーザーアカウントを復元する唯一の方法は、管理者に連絡してロックを解除するように依頼することです。

1.8 Enabling LDAP Authentication

pgAdmin の LDAP 認証を有効にするには、pgAdmin がサーバーモードでインストールされているシステムで *config_local.py* または *config_system.py* ファイル（ *config.py* ドキュメントを参照）で LDAP 設定を構成する必要があります。これらの設定を *config.py* ファイルからコピーし、次のパラメーターの値を変更できます。

LDAP を構成するには 3 つの方法があります。

- pgAdmin ユーザーとしてバインド
- 匿名バインド
- 専用ユーザーバインド

Parameter	Description
AUTHENTICATION_SOURCES	このパラメータのデフォルト値は *internal* です。LDAP 認証を有効にするには、このパラメーターの値のリストに *ldap* を含める必要があります。次のように値を変更できます。 • ['ldap']：pgAdmin は LDAP 認証のみを使用します。 • ['ldap','internal']：pgAdmin は最初に LDAP を介してユーザーを認証しようとします。その認証が失敗した場合、pgAdmin の内部ユーザーエントリが認証に使用されます。 • ['internal','ldap']：pgAdmin は最初に内部ユーザーエントリを介してユーザーを認証しようとします。その認証が失敗すると、LDAP 認証が使用されます。
LDAP_AUTO_CREATE_USER	LDAP ユーザー資格情報に対応する pgAdmin ユーザーを自動的に作成するかどうかを指定します。LDAP パスワードは pgAdmin データベースに保存されないことに注意してください。
LDAP_CONNECTION_TIMEOUT	LDAP 認証の接続タイムアウト（秒）を指定します。
LDAP_SERVER_URI	LDAPURI は、接続プロトコル（ldap または ldaps）、接続するディレクトリサーバーの IP アドレス/ホスト名、およびポートの組み合わせです。たとえば、「ldap://172.16.209.35:389」は有効な LDAP_SERVER_URI です。ldap は接続プロトコル、172.16.209.35 は IP アドレス、389 はポートです。ポート 636 は ldaps 通信プロトコルに使用されます。
LDAP_USERNAME_ATTRIBUTE	ユーザー名を含む LDAP 属性を指定します。LDAP 認証の場合、その特定の属性の値をユーザー名として入力する必要があります。たとえば、LDAP_USERNAME_ATTRIBUTE の値を「cn」に設定し、LDAP サーバーのエントリで「cn=admin」を定義している場合、 *EmailAddress/Username* フィールドに「admin」と入力して認証できます。 *Password* フィールドに対応するパスワード。
LDAP_SEARCH_BASE_DN	検索する最上位のユーザディレクトリの識別名（DN）を指定します。このパラメーターを使用して、検索要求を特定のユーザーグループに制限できます。たとえば、sales という名前の組織単位内のみを検索する場合、LDAP_SEARCH_BASE_DN パラメーターの値を次のように定義できます。 これは、pgAdmin ユーザーとしてバインドしている場合のみのオプションパラメーターです。LDAP_SEARCH_BASE_DN の値を指定しない場合、LDAP_BASE_DN の値は同じと見なされます。
LDAP_SEARCH_FILTER	LDAP 検索要求で一致するエントリを取得するための基準を定義します。たとえば、LDAP_SEARCH_FILTER='(objectclass=HR)' 設定は、objectClass 属性として HR を持つユーザーのみを検索します。

次のページに続く

表 5 – 前のページからの続き

Parameter	Description
LDAP_SEARCH_SCOPE	検索要求の潜在的な一致と見なされるベース DN 以下のエントリのセットを示します。検索の範囲は、 **ベース** 、 **レベル** 、または **サブツリー** 検索として指定できます。 **ベース** 検索は、検索をベースオブジェクトに制限します。 **レベル** の検索は、ベースオブジェクトの直接の子に制限されますが、ベースオブジェクト自分自身は除外されます。 **サブツリー** 検索には、ベースオブジェクトだけでなくすべての子オブジェクトが含まれます。
LDAP_DN_CASE_SENSITIVE	DN（識別名）で大文字と小文字を区別するかどうかを示します。可能な値は True または False です。デフォルトでは False に設定されています。
LDAP_USE_STARTTLS	LDAP クライアントと LDAP サーバー間の安全な通信に TransportLayerSecurity(TLS) を使用するかどうかを指定します。*LDAP_SERVER_URI* で接続プロトコルを *ldaps* として指定した場合、このパラメーターは無視されます。
LDAP_CA_CERT_FILE	信頼できる CA 証明書ファイルへのパスを指定します。このパラメーターは、接続プロトコルとして *ldaps* を使用している場合、または *LDAP_USE_STARTTLS* パラメーターを *True* に設定している場合にのみ適用されます。
LDAP_CERT_FILE	サーバー証明書ファイルへのパスを指定します。このパラメーターは、接続プロトコルとして *ldaps* を使用している場合、または *LDAP_USE_STARTTLS* パラメーターを *True* に設定している場合にのみ適用されます。
LDAP_KEY_FILE	サーバー秘密鍵ファイルへのパスを指定します。このパラメーターは、接続プロトコルとして *ldaps* を使用している場合、または *LDAP_USE_STARTTLS* パラメーターを *True* に設定している場合にのみ適用されます。
Bind as pgAdmin user	
LDAP_BASE_DN	サーバーがユーザーの検索を開始するベース DN を指定します。たとえば、ユーザーの LDAP 検索は、サーバーによってベース DN（dc=example,dc=com）から実行されます。ベース DN が一致する場合、完全な DN（cn=admin,dc=example,dc=com）を使用して、指定されたパスワードとバインドします。
Anonymous bind	
LDAP_ANONYMOUS_BIND	匿名バインドの場合、このパラメーターを *True* に設定します。接続が確立されると、pgadmin ログインユーザーは、ログイン画面で指定されたユーザー名とパスワードによってさらに認証されます。
Dedicated user bind	

次のページに続く

表 5 – 前のページからの続き

Parameter	Description
LDAP_BIND_USER	シンプルバインドにログインするユーザーのアカウント。専用ユーザーを使用して接続をバインドできるようにするには、このパラメーターを設定します。接続が確立されると、pgadmin ログインユーザーは、ログイン画面で指定されたユーザー名とパスワードによってさらに認証されます。ログイン画面で。
LDAP_BIND_PASSWORD	シンプルバインドのパスワード。LDAP_BIND_USER パラメーターを設定している場合は値を指定します。

1.9 Enabling Kerberos Authentication

Prerequisite: Kerberos の理解とセットアップ

参照：https://web.mit.edu/kerberos/

Kerberos 認証を構成するには、Kerberos サーバーをセットアップし、 *kinit* を使用してクライアントでチケットを取得する必要があります。

注:Kerberos をサポートする ActiveDirectory ユーザーは、kinit を必要としません。

pgAdmin の Kerberos 認証を有効にするには、pgAdmin がサーバーモードでインストールされているシステムで *config_local.py* または *config_system.py* （ *config.py* ドキュメントを参照）で Kerberos 設定を構成する必要があります。これらの設定を *config.py* ファイルからコピーし、次のパラメーターの値を変更できます。

Parameter	Description
AUTHENTICATION_SOURCES	このパラメータのデフォルト値は *internal* です。Kerberos 認証を有効にするには、このパラメーターの値のリストに *kerberos* を含める必要があります。次のように値を変更できます。 • ['kerberos']：pgAdmin は Kerberos 認証のみを使用します。 • ['kerberos','internal']：pgAdmin は最初に kerberos を介してユーザーを認証しようとします。その認証が失敗すると、ログインダイアログに戻り、認証用に内部 pgAdmin ユーザ資格情報を提供する必要があります。
KERBEROS_AUTO_CREATE_USER	正常に認証された Kerberos ユーザーに対応する pgAdmin ユーザーを自動的に作成する場合は、値を *True* に設定します。パスワードは pgAdmin データベースに保存されないことに注意してください。
KRB_APP_HOST_NAME	*pgAdminWeb* **サーバーのホスト名** の名前を指定します。設定されていない場合、 *default_server* パラメーターの値が使用されることに注意してください。

1.9.1 HTTP サービスの keytab ファイル

- HTTP サービスプリンシパル HTTP/<host-name>@realm の *Keytab* ファイルを生成し、*pgAdmin* ウェブサーバーマシンにコピーします。*pgAdmin* Web サーバーを所有するオペレーティングシステムユーザーがこのファイルの所有者であり、そのユーザーがアクセスできることを確認してください。
- *krb5.conf* ファイルで *default_keytab_name* パラメーターを設定するか、環境変数 *KRB5_KTNAME* を設定する必要があることに注意してください。設定されていない場合、*KRB_KTNAME* を *config_local.py* または *config_system.py* ファイルの *Keytab* ファイルの場所に設定します。

1.9.2 ApacheHTTPD 構成

pgAdmin サーバーが Apache サーバーの下にある場合、*Apache HTTPD Configuration* の *Directory* 指示子に次のパラメーターを追加する必要があります。

- WSGIScript リロードオン
- WSGIPass 承認オン

1.9.3 Kerberos 認証を構成するためのブラウザー設定

SPNEGO を介して Kerberos 認証を使用するには、クライアントマシンでブラウザー設定を構成する必要があります。

- MozillaFirefox の場合
  - アドレスバーに *about:config* と入力して、低レベルの Firefox 構成ページを開きます。
  - [検索] テキストボックスに、次のように入力します。
  - *network.generate-auth.trusted-uris* 設定をダブルクリックし、KerberosHTTPSPNEGO で保護されている Web サーバーのホスト名またはドメインを入力します。複数のドメインとホスト名はコンマで区切ります。
  - [OK] をクリックします。
- GoogleChrome の場合
  - Windows：
    - コントロールパネルを開き、[インターネットオプション] ダイアログにアクセスします。
    - [セキュリティ] タブを選択します。
    - ローカルイントラネットゾーンを選択し、[サイト] ボタンをクリックします。

* 最初の 2 つのオプション、 **他のゾーンにリストされていないすべてのローカル（イントラネット）サイトを含める** と **プロキシサーバーをバイパスするすべてのサイトを含める** がチェックされていることを確認します。
* 「詳細設定」をクリックし、KerberosHTTPSPNEGO で保護されているドメインの名前を一度に 1 つずつ Web サイトのリストに追加します。例、myhost.example.com。[閉じる] をクリックします。
* OK をクリックして、構成の変更を保存します。

– Linux または macOS の場合：

* *--auth-server-whitelist* パラメーターを google-chrome コマンドに追加します。たとえば、Linux プロンプトから Chrome を実行するには、次のように google-chrome コマンドを実行します。

```
google-chrome --auth-server-whitelist = "hostname/domain"
```

1.9.4 Kerberos 認証を構成するための PostgreSQL サーバー設定

- Kerberos 認証で PostgreSQL サーバーに接続するには、PostgreSQL のビルド時に GSSAPI サポートを有効にし、必要な configuration を配置する必要があります。
- pgAdmin では、サーバーダイアログで/"Kerberosauthentication/" フラグを True に設定して、PostgreSQL サーバーの Kerberos 認証を有効にする必要があります。有効にすると、pgAdmin はパスワードの入力を求めず、Kerberos を使用して PostgreSQL サーバーに接続しようとします。
- Kerberos を使用して PostgreSQL に接続するには、Kerberos 認証を使用して pgAdmin にログインする必要があることに注意してください。

1.9.5 マスターパスワード

マルチユーザーモードでは、pgAdmin はユーザーのログインパスワードを使用して、PostgreSQL サーバーのパスワードを暗号化/復号化します。Kerberos 認証では、pgAdmin ユーザーはパスワードを持っていないため、Kerberos 認証を使用するように構成されていないサーバーの PostgreSQL サーバーのパスワードを保存するための暗号化キーが必要です。これを行うには、構成パラメーター MASTER_PASSWORD を *True* に設定します。これにより、マスターパスワードを設定すると、パスワードの保存中に暗号化キーとして使用されます。False の場合、サーバーのパスワードを保存できません。

1.10 Enabling OAUTH2 Authentication

pgAdmin の OAUTH2 認証を有効にするには、pgAdmin がサーバーモードでインストールされているシステムの *config_local.py* または *config_system.py* ファイル（ *config.py* ドキュメントを参照）で OAUTH2 設定を構成する必要があります。これらの設定を *config.py* ファイルからコピーし、次のパラメーターの値を変更できます。

Parameter	Description
AUTHENTICATION_SOURCES	このパラメータのデフォルト値は *internal* です。OAUTH2 認証を有効にするには、このパラメーターの値のリストに *oauth2* を含める必要があります。次のように値を変更できます。 • ['oauth2'、'internal']：pgAdmin は oauth2 で認証するための追加のボタンを表示します
OAUTH2_NAME	Oauth2 プロバイダーの名前、例：Google、Github
OAUTH2_DISPLAY_NAME	pgAdmin での Oauth2 表示名
OAUTH2_CLIENT_ID	Oauth2 クライアント ID
OAUTH2_CLIENT_SECRET	Oauth2 クライアントシークレット
OAUTH2_TOKEN_URL	Oauth2 アクセストークンエンドポイント
OAUTH2_AUTHORIZATION_URL	ユーザー承認のエンドポイント
OAUTH2_SERVER_METADATA_URL	OAuth2 プロバイダーのサーバーメタデータ URL
OAUTH2_API_BASE_URL	リクエストを簡単にする Oauth2 ベース URL エンドポイント、例： *https://api.github.com/*
OAUTH2_USERINFO_ENDPOINT	ユーザーエンドポイント、例： *user* （github の場合）および *useinfo* （google の場合）
OAUTH2_SCOPE	Oauth スコープ、例：「openidemailprofile」。結果のプロファイルには「email」クレームが必要であることに注意してください。
OAUTH2_ICON	oauth2 ボタンに配置される Font-awesome アイコン、例：fa-github
OAUTH2_BUTTON_COLOR	Oauth2 ボタンの色
OAUTH2_USERNAME_CLAIM	**ユーザー名に使用されるクレーム。値が空の場合** 電子メールはユーザー名として使用されますが、値が指定されている場合、クレームが存在する必要があります。例： *oid* （AzureAD の場合）
OAUTH2_AUTO_CREATE_USER	**自動的に設定したい場合は値を *True* に設定します** 正常に認証された Oauth2 ユーザーに対応する pgAdmin ユーザーを作成します。パスワードは pgAdmin データベースに保存されないことに注意してください。

1.10.1 リダイレクト URL

Oauth2 サーバーを構成するためのリダイレクト URL は *http://<pgAdminServerURL>/oauth2/authorize*

1.10.2 マスターパスワード

マルチユーザーモードでは、pgAdmin はユーザーのログインパスワードを使用して、PostgreSQL サーバーのパスワードを暗号化/復号化します。Oauth2 認証では、pgAdmin はユーザーのパスワードを保存しないため、PostgreSQL サーバーのパスワードを保存するための暗号化キーが必要です。これを行うには、構成パラメーター MASTER_PASSWORD を *True* に設定します。これにより、マスターパスワードを設定すると、パスワードの保存中に暗号化キーとして使用されます。False の場合、サーバーのパスワードを保存できません。

1.10.3 ログインページ

構成後、再起動すると、Oauth2 ログインボタンがあるログインページが表示されます。



1.11 Enabling Webserver Authentication

Web サーバー認証を構成するには、REMOTE_USER 環境変数を設定する限り、認証プラグイン（Shibboleth、HTTPBASICauth など）で Web サーバーをセットアップする必要があります。pgAdmin のウェブサーバー認証を有効にするには、pgAdmin がサーバーモードでインストールされているシステムの *config_local.py* または *config_system.py* ファイル（*config.py* ドキュメントを参照）でウェブサーバー設定を構成する必要があります。これらの設定を *config.py* ファイルからコピーし、次のパラメーターの値を変更できます。

Parameter	Description
AUTHENTICATION_SOURCES	このパラメータのデフォルト値は *internal* です。OAUTH2 認証を有効にするには、このパラメーターの値のリストに *webserver* を含める必要があります。次のように値を変更できます。 • ['webserver']：pgAdmin は Web サーバー認証のみを使用します。 • ['webserver','internal']：pgAdmin は最初に Web サーバーを介してユーザーを認証しようとします。その認証が失敗すると、ログインダイアログに戻り、認証用に内部 pgAdmin ユーザ資格情報を提供する必要があります。
WEBSERVER_AUTO_CREATE_USER	**自動的に設定したい場合は値を *True* に設定します** 正常に認証された Web サーバーユーザーに対応する pgAdmin ユーザーを作成します。パスワードは pgAdmin データベースに保存されないことに注意してください。
WEBSERVER_REMOTE_USER	**Web サーバーのリモートユーザーの詳細を取得するには、この変数をヘッダーに設定するか、** Web サーバー認証後に Web サーバーから取得される環境変数名。デフォルト値は REMOTE_USER で、指定できる値は REMOTE_USER、HTTP_X_FORWARDED_USER、X-Forwarded-User です。

1.11.1 マスターパスワード

マルチユーザーモードでは、pgAdmin はユーザーのログインパスワードを使用して、PostgreSQL サーバーのパスワードを暗号化/復号化します。Web サーバー認証では、pgAdmin はユーザーのパスワードを保存しないため、PostgreSQL サーバーのパスワードを保存するための暗号化キーが必要です。これを行うには、構成パラメーター MASTER_PASSWORD を *True* に設定します。これにより、マスターパスワードを設定すると、パスワードの保存中に暗号化キーとして使用されます。False の場合、サーバーのパスワードを保存できません。

注釈: プリコンパイルおよび構成済みのインストールパッケージは、多くのプラットフォームで利用できます。これらのパッケージは、可能な限りエンドユーザーが使用する必要があります。次の情報は、これらのパッケージのメンテナーおよび pgAdmin の動作を理解することに関心があるユーザーに役立ちます。

---

pgAdmin4 クライアントは、デスクトップ環境を最大限に活用するように配置できるドラッグアンドドロップパネルを備えた高度にカスタマイズ可能な表示を備えています。

ツリーコントロールは、管理対象サーバーと各サーバーにあるオブジェクトの概要を提供します。ツリーコントロール内のノードを右クリックして、選択したオブジェクトの管理タスクにすばやくアクセスできるコンテキスト依存メニューにアクセスします。

タブ付きブラウザーを使用すると、ツリーコントロール内の各オブジェクトに関する統計情報、および pgAdmin ツールとユーティリティ（クエリツールやデバッガーなど）にすばやくアクセスできます。pgAdmin は、pgAdmin ツールが提供する拡張機能にアクセスするたびに追加の機能タブを開きます。必要に応じて、機能タブを開いたり、閉じたり、再配置したりできます。

設定 ダイアログを使用して、pgAdmin 表示の内容と動作をカスタマイズします。 **設定** ダイアログを開くには、 **ファイル** メニューから **設定** を選択します。

各ダイアログの左下隅にある **ヘルプ** ボタンは、ダイアログのオンラインヘルプを開きます。 **ヘルプ** メニューをナビゲートし、開くリソースの名前を選択することにより、追加の Postgres ヘルプにアクセスできます。

Search objects を使用してデータベース内のオブジェクトを検索できます

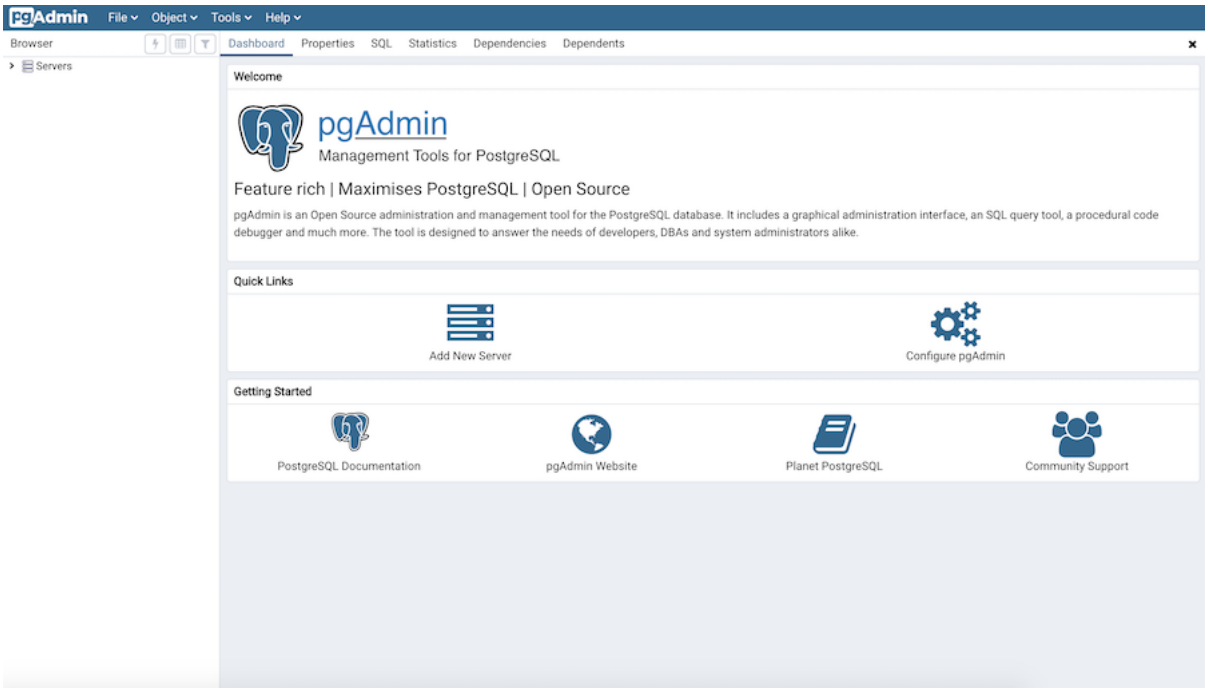
1.12 User Interface

pgAdmin4 は、単純な SQL クエリの作成から複雑なデータベースの開発まで、PostgreSQL のすべての機能をサポートしています。アクティブなデータベースを（リアルタイムで）照会するように設計されているため、変更や実装を最新の状態に保つことができます。

pgAdmin4 の機能は次のとおりです。

- 実行時に検出されるオブジェクトの自動検出とサポート
- データを直接編集できるライブ SQL クエリツール
- 管理照会のサポート
- 構文を強調表示する SQL エディター
- 再設計されたグラフィカルインターフェイス
- 一般的なタスクのための強力な管理ダイアログとツール
- 応答性のある状況依存の動作
- サポートエラーメッセージ
- 役立つヒント
- オンラインヘルプと、pgAdmin ダイアログとツールの使用に関する情報。

pgAdmin が開くと、インターフェイスにはメニューバーと 2 つのペインに分割されたウィンドウがあります。左側のペインの **ブラウザー** ツリーコントロールと右側のペインのタブ付きブラウザー。



ダッシュボード タブの **クイックリンク** パネルからアイコンを選択します。

- **新しいサーバーの追加** ボタンをクリックして *Create - Server dialog* を開き、新しいサーバー定義を追加します。
- *ConfigurepgAdmin* ボタンをクリックして *Preferences dialog* を開き、pgAdmin クライアントをカスタマイズします。

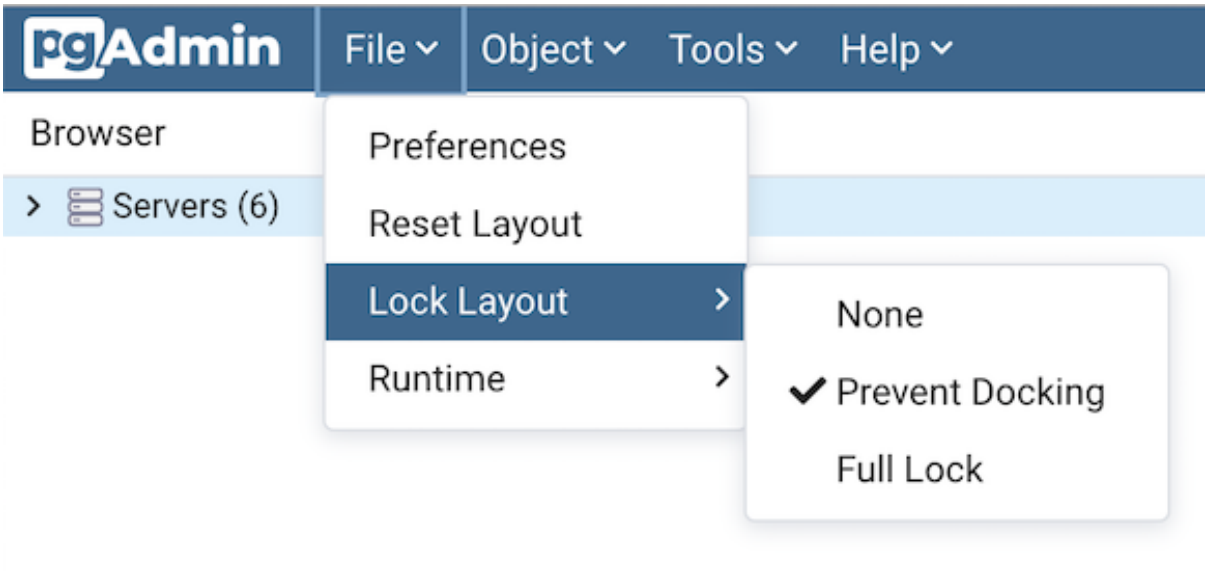
GettingStarted パネルのリンクは、Postgres ユーザーに役立つ情報を提供する新しいブラウザータブを開きます。

- *PostgreSQLDocumentation* リンクをクリックして、PostgreSQL オープンソースプロジェクトの *Documentation* ページに移動します。プロジェクトサイトで、現在サポートされているバージョンの PostgreSQL サーバーのマニュアルを確認できます。
- *pgAdminWeb* **サイト** リンクをクリックして、pgAdmin プロジェクトの Web サイトに移動します。pgAdmin サイトには、最近の pgAdmin リリースに関するニュースやその他のプロジェクト情報があります。
- *PlanetPostgreSQL* リンクをクリックして、Postgres 関連のブログのブログアグリゲーターに移動します。
- **コミュニティサポート** リンクをクリックして、PostgreSQL オープンソースプロジェクトサイトの **コミュニティ** ページに移動します。このページでは、PostgreSQL 機能のサポートの取得に関する情報を提供します。

1.13 Menu Bar

pgAdmin メニューバーには、オプション、コマンド、およびユーティリティにアクセスするためのドロップダウンメニューがあります。メニューバーには、 **ファイル** 、 **オブジェクト** 、ツール 、**および** ヘルプ **の選択が表示されます。選択はグレー表示され、** pgAdmin*ツリーコントロールで現在選択されているオブジェクトで無効になっていることを示します。

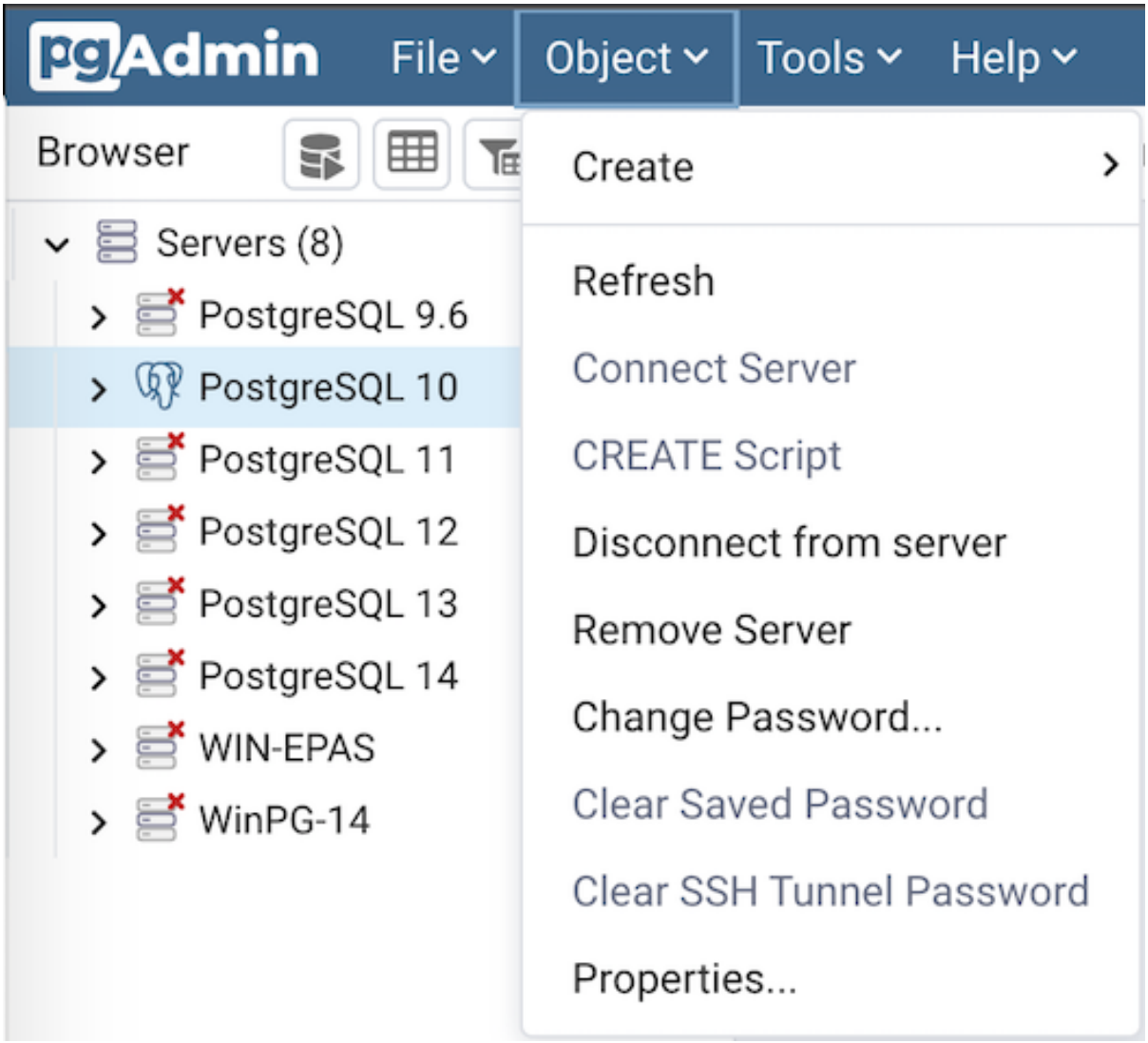
1.13.1 ファイルメニュー



ファイル メニューを使用して、次のオプションにアクセスします。

オプション	アクション
設定	クリックして *Preferences* ダイアログを開き、pgAdmin 設定をカスタマイズします。
レイアウトをリセット	ワークスペースを変更した場合は、をクリックしてデフォルトのレイアウトに戻します。
ロックレイアウト	クリックしてサブメニューを開き UI レイアウトをロックするレベルを選択こちらもブラウザ表示から変更可能 *preferences*
ランタイム	クリックして、構成、ログ、およびズーム設定を行うサブメニューを開きます。pgAdmin4 がデスクトップモードで実行されている場合にのみ表示されます。ランタイムメニューについてもっと知るには *click here*

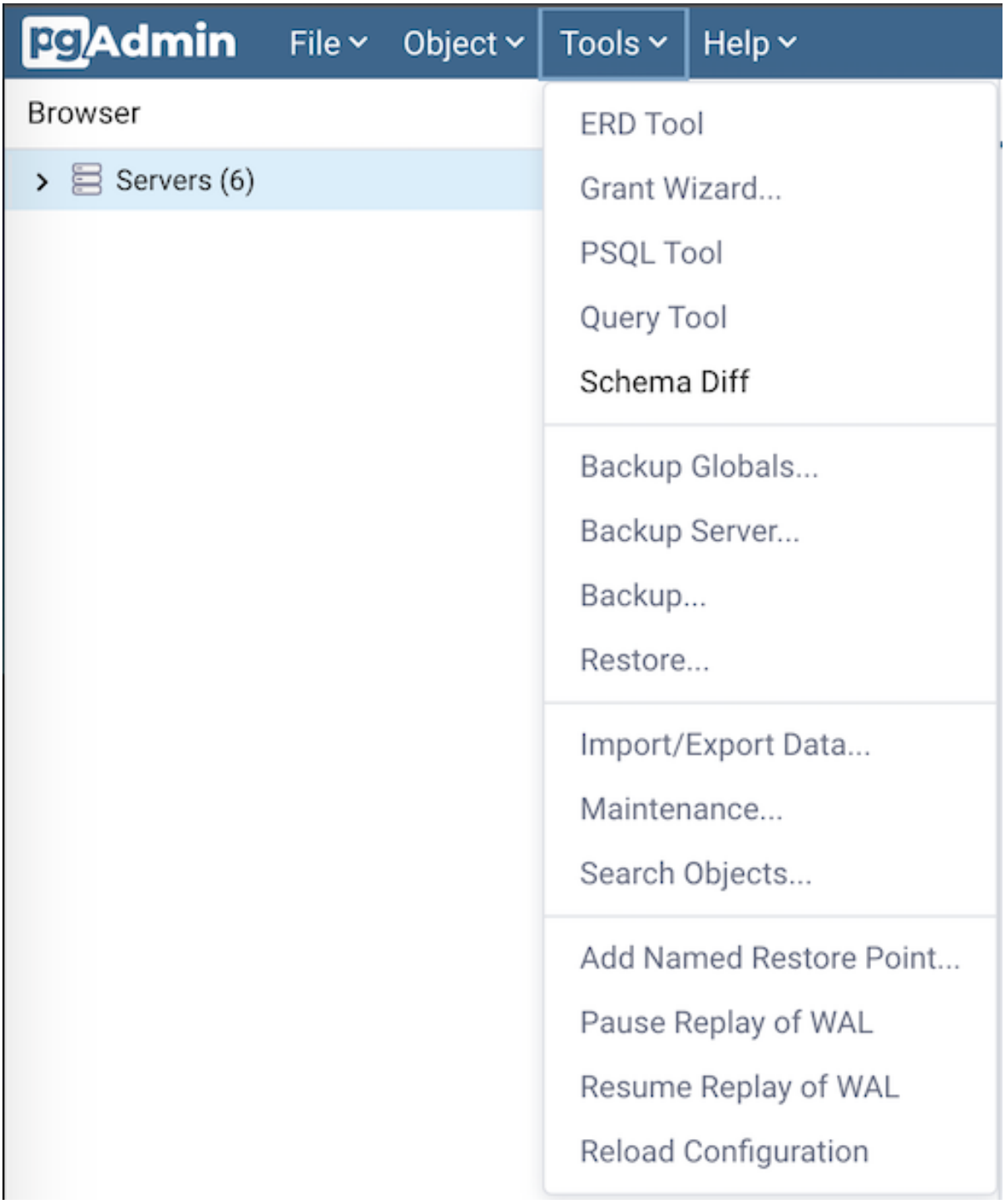
1.13.2 オブジェクトメニュー



Object メニューは状況依存です。 *Object* メニューを使用して、次のオプションにアクセスします（アルファベット順）。

オプション	アクション
パスワードの変更...	クリックして *Change Password...* ダイアログを開き、パスワードを変更します。
保存されたパスワードをクリアする	データベースサーバーのパスワードを保存している場合は、クリックして保存したパスワードをクリアします。パスワードが既に保存されている場合にのみ有効にします。
***SSH* トンネルのパスワードをクリアする**	ssh トンネルのパスワードを保存している場合は、クリックして保存したパスワードをクリアします。パスワードが既に保存されている場合にのみ有効にします。
サーバーに接続	クリックして *Connect to Server* ダイアログを開き、サーバーとの接続を確立します。
作成	*Create* をクリックして、状況依存の選択を提供するコンテキストメニューにアクセスします。選択すると、新しいオブジェクトを作成するための *Create* ダイアログが開きます。
削除/ドロップ	クリックして、現在選択されているオブジェクトをサーバーから削除します。
サーバーから切断する	クリックして、現在選択されているサーバーから切断します。
ドロップカスケード	クリックして、現在選択されているオブジェクトとすべての依存オブジェクトをサーバーから削除します。
プロパティ...	クリックして、現在選択されているオブジェクトのプロパティを確認または変更します。
リフレッシュ	クリックして、現在選択されているオブジェクトを更新します。
サーバーを削除	クリックして、現在選択されているサーバーを削除します。
スクリプト	クリックして *Query tool* を開き、フライアウトメニューから選択したスクリプトを編集します。
トリガー	クリックして、現在選択されているテーブルのトリガーを **無効** または **有効** にします。オプションはフライアウトメニューに表示されます。
切り捨て	クリックして、テーブルからすべての行を削除する（ *Truncate* ）、テーブルとその子テーブルからすべての行を削除する（ *TruncateCascade* ）、またはテーブルからすべての行を削除して列が所有するシーケンスを自動的にリスタートします（ **アイデンティティ** ）。オプションはフライアウトメニューに表示されます。
データを見る	クリックして、データを表示するためのいくつかのオプションを提供するコンテキストメニューにアクセスします（以下を参照）。
データベースの *ERD*	クリックして、選択したデータベースの図が自動生成された ERD ツールを開きます。このオプションは、データベースが選択されている場合にのみ使用できます。オプションはフライアウトメニューに表示されます。
テーブルの *ERD*	クリックして、選択したテーブルの図が自動生成された ERD ツールを開きます。このオプションは、テーブルが選択されている場合にのみ使用できます。オプションはフライアウトメニューに表示されます。

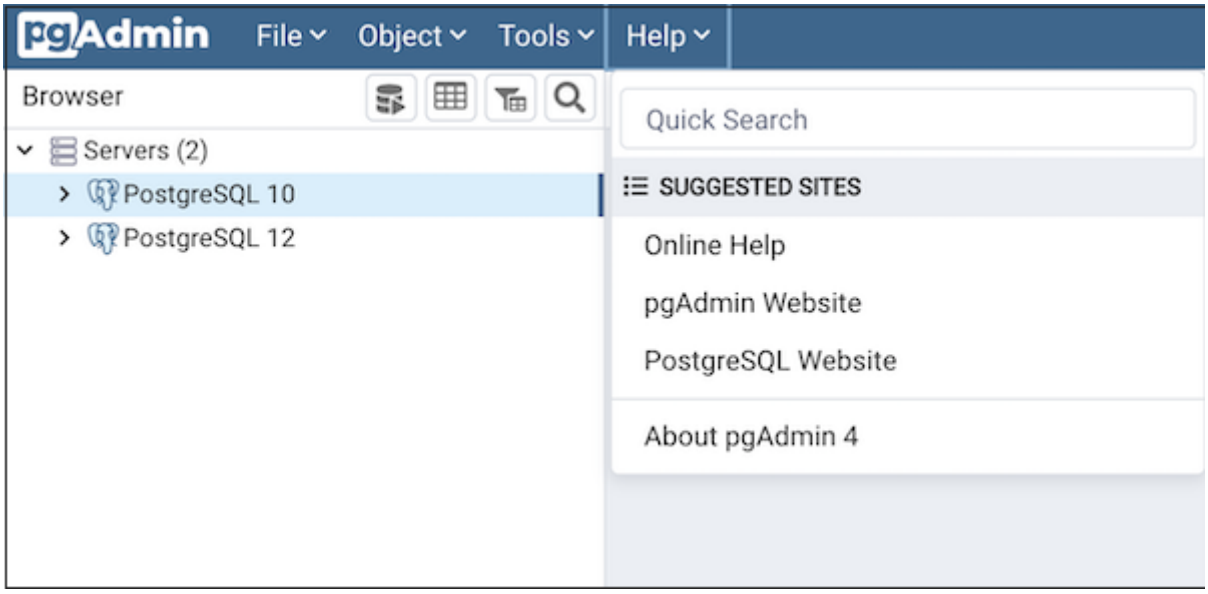
1.13.3 ツールメニュー



[ツール] メニューを使用して、次のオプションにアクセスします（アルファベット順）。

オプション	アクション
ERD ツール	クリックして *ERD Tool* を開き、データベースの設計を開始します。
Grant ウィザード...	クリックして、 *Grant Wizard* ツールにアクセスします。
PSQL ツール	クリックして *PSQL Tool* を開き、現在のデータベースコンテキストで PSQL を起動します。
クエリツール	クリックして、現在選択されているオブジェクトの *Query tool* を開きます。
スキーマの相違	クリックして *Schema Diff* を開き、2 つのデータベースまたは 2 つのスキーマの比較を開始します。
バックアップグローバル...	クリックして *Backup Globals...* ダイアログを開き、クラスターオブジェクトをバックアップします。
バックアップサーバー...	クリックして *Backup Server...* ダイアログを開き、サーバーをバックアップします。
バックアップ...	クリックして *Backup...* ダイアログを開き、データベースオブジェクトをバックアップします。
復元...	クリックして *Restore* ダイアログにアクセスし、バックアップからデータベースファイルを復元します。
データのインポート/エクスポート...	クリックして *Import/Export data...* ダイアログを開き、テーブルからデータをインポートまたはエクスポートします。
メンテナンス...	クリックして、VACUUM、ANALYZE、REINDEX、または CLUSTER の *Maintenance...* ダイアログを開きます。
オブジェクトを検索...	クリックして *Search Objects...* を開き、データベース内の任意の種類のオブジェクトの検索を開始します。
名前付けられた復元ポイントを追加	クリックして *Add named restore point...* ダイアログを開き、現在のサーバー状態の特定の時点のスナップショットを撮ります。
WAL のリプレイを一時停止	クリックして、WAL ログの再生を一時停止します。
WAL のリプレイを再開	クリックして、WAL ログの再生を再開します。
設定をリロード...	クリックして、サーバーを再起動せずに構成ファイルを更新します。
ストレージマネージャー	クリックして *Storage Manager* を開き、バックアップファイルをアップロード、削除、またはダウンロードします。

1.13.4 ヘルプメニュー

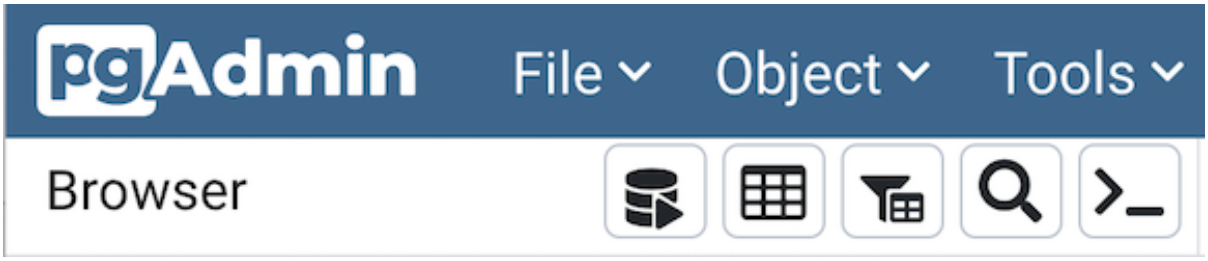


[**ヘルプ**] メニューのオプションを使用して、オンラインヘルプドキュメントにアクセスするか、pgAdmin インストールに関する情報を確認します（アルファベット順）。

オプション	アクション
クイック検索	クイック検索フィールドにキーワードを入力します。3 文字以上入力すると、一致するすべてのものがメニュー項目に表示され、関連ドキュメントがヘルプ記事に表示されます。メニュー項目の下のオプションをクリックして、特定の機能またはオブジェクトのアクションを実行します。ヘルプ記事をクリックすると、そのトピックのヘルプが別のウィンドウでテキストが強調表示されます。 **Note** :-メニュー項目の下のオプションのいずれかが無効になっている場合、情報アイコンを介して情報が提供されます。
pgAdmin4 **について**	クリックしてウィンドウを開き、pgAdmin に関する情報を確認します。これには、現在のバージョンと現在のユーザーが含まれます。
オンラインヘルプ	クリックして、pgAdmin ユーティリティ、ツール、ダイアログの使用に関するドキュメントサポートを開きます。左側のブラウザーペインで（新しく開いたタブ内で）ヘルプドキュメントを移動するか、検索バーを使用してトピックを指定します。
pgAdmin **ウェブサイト**	クリックして、ブラウザーウィンドウで *pgAdmin.org* Web サイトを開きます。
PostgreSQL **ウェブサイト**	クリックして、PostgreSQL サイトでホストされている PostgreSQL のコアドキュメントにアクセスします。このサイトでは、ガイド、チュートリアル、リソースも提供しています。

1.14 Toolbar

pgAdmin ツールバーには、データの表示や pgAdmin で最も頻繁に使用されるクエリツールなど、頻繁に使用される機能のショートカットボタンがあります。このツールバーは、[ブラウザー] パネルに表示されます。ボタンは、選択したブラウザノードに基づいて有効/無効になります。

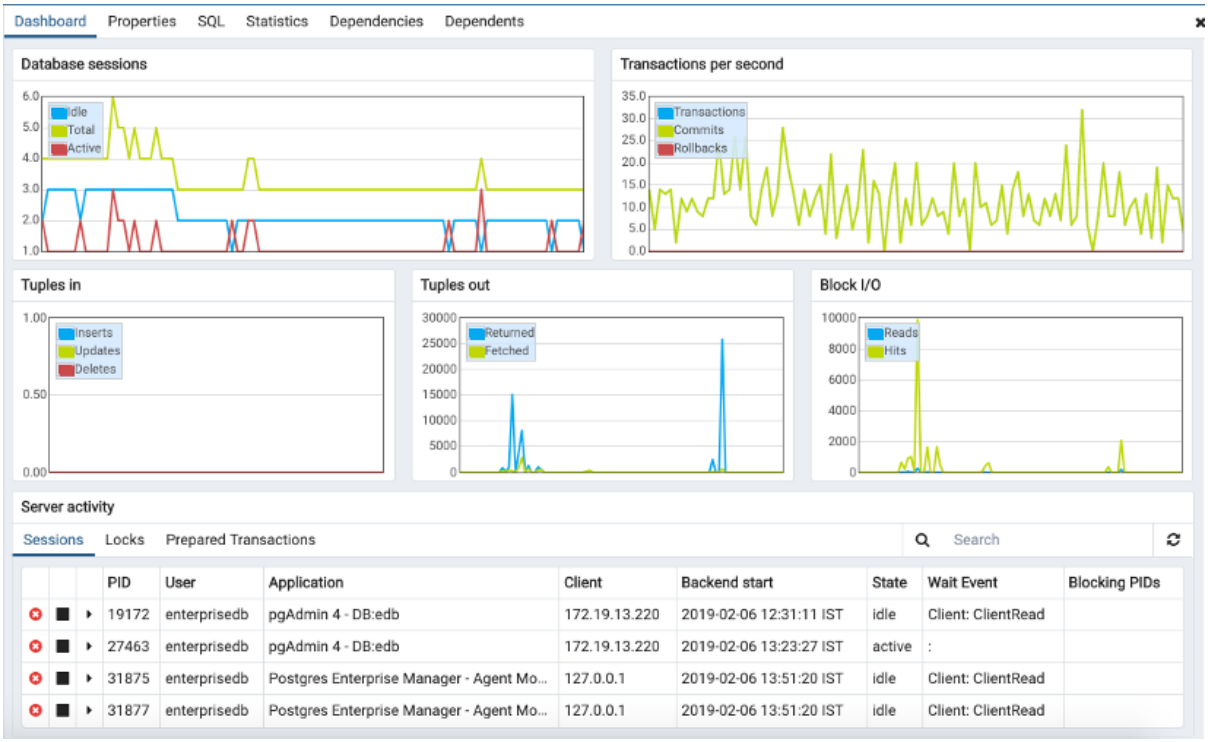


- *Query Tool* ボタンを使用して、現在のデータベースコンテキストでクエリツールを開きます。
- *View Data* ボタンを使用して、選択したテーブルに格納されているデータを表示/編集します。

- *Filtered Rows* ボタンを使用して DataFilter ポップアップにアクセスし、表示/編集するデータのセットにフィルターを適用します。
- *Search objects* ボタンを使用して、オブジェクトの検索ダイアログにアクセスします。データベースオブジェクトを検索するのに役立ちます。
- *PSQL Tool* ボタンを使用して、現在のデータベースコンテキストで PSQL を開きます。

1.15 Tabbed Browser

pgAdmin ウィンドウの右側のペインには、左側のウィンドウの *pgAdmin* ツリーコントロールで現在選択されているオブジェクトに関する情報を表示するタブのコレクションがあります。タブを選択して、ツリーコントロールで強調表示されたオブジェクトに関する情報にアクセスします。



ダッシュボード タブのグラフは、選択したサーバーまたはデータベースの使用統計のアクティブな分析を提供します。

- **サーバーセッション** または **データベースセッション** グラフには、サーバーまたはデータベースとの対話が表示されます。
- *1* **秒あたりのトランザクション数** グラフには、サーバーまたはデータベースで実行されている 1 秒あたりのコミット、ロールバック、合計トランザクションが表示されます。
- **タプルイン** グラフには、サーバーまたはデータベースで挿入、更新、および削除されたタプルの数が表示されます。
- **タプルアウト** グラフには、サーバーまたはデータベースからフェッチされて返されたタプルの数が表示されます。

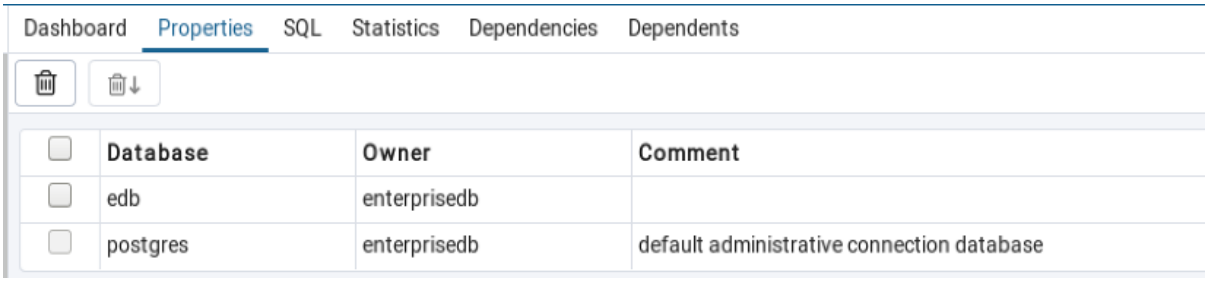
- **ブロック** *I/O* グラフには、サーバーまたはデータベースのファイルシステムから読み取られたブロックまたはバッファーキャッシュ（オペレーティングシステムのファイルシステムキャッシュではない）からフェッチされたブロックの数が表示されます。

サーバーアクティビティ パネルには、セッション、ロック、準備されたトランザクション、およびサーバー構成（該当する場合）が表示されます。情報は、状況依存のテーブルに表示されます。表の上にあるコントロールを使用して、次のことを行います。

- *Refresh* ボタンをクリックして、各テーブルに表示される情報を更新します。
- **検索** ボックスに値を入力して、テーブルのコンテンツを検索条件を満たす 1 つ以上のセッションに制限します。たとえば、プロセス ID を入力して特定のセッションを見つけたり、セッション状態（ *idle* など）を入力してアイドル状態のすべてのセッションを見つけたりできます。

セッション テーブルのアイコンを使用して、セッションの状態を確認または制御できます。

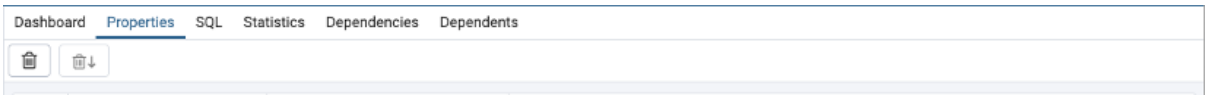
- （最初の列にある） **終了** アイコンを使用して、セッションを停止し、テーブルからセッションを削除します。サーバーがセッションを終了する前に、選択を確認するように求められます。
- セッションを閉じずにアクティブなクエリを終了するには、 *Cancel* アイコン（2 列目にあります）を使用します。クエリをキャンセルする前に、サーバーは選択を確認するように求めます。クエリをキャンセルすると、テーブルの **状態** 列に表示される値が **アクティブ** から **アイドル** に更新されます。セッションは、セッションが終了するまでテーブルに残ります。
- **詳細** アイコン（3 列目にあります）を使用して、 **詳細** タブを開きます。タブには、選択したセッションに関する情報が表示されます。



[**プロパティ**] タブには、選択したオブジェクトに関する情報が表示されます。

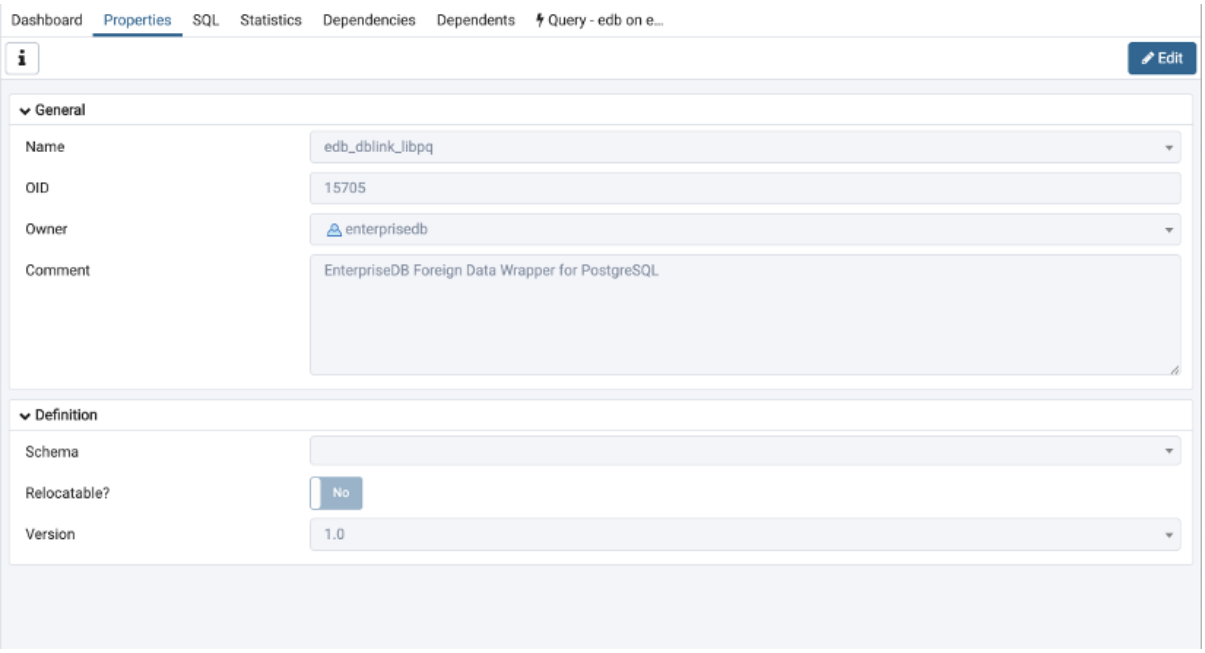
ブラウザータブの下のツールバーの **削除** アイコンをクリックして、[プロパティ] パネルで選択したオブジェクトを削除します。

ブラウザータブの下のツールバーの **ドロップカスケード** アイコンをクリックして、[プロパティ] パネルで選択したオブジェクトとすべての依存オブジェクトを削除します。

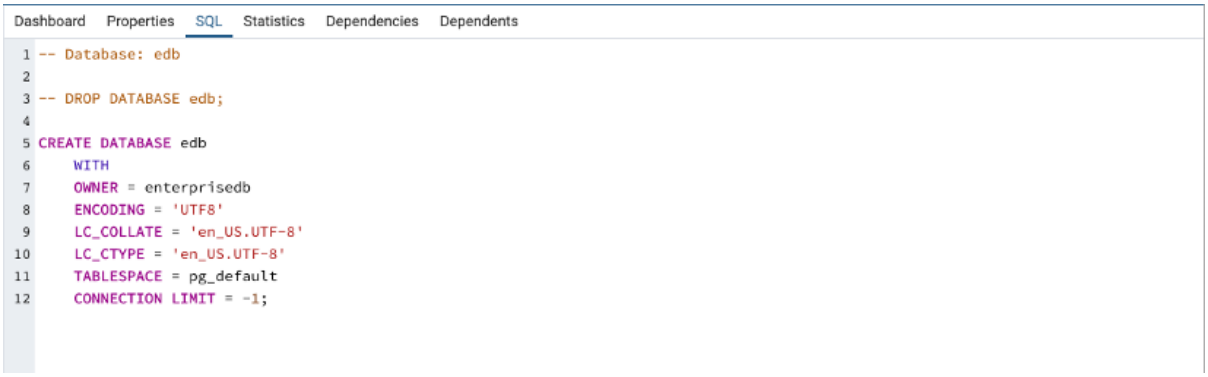


ブラウザータブの下のツールバーの **編集** アイコンをクリックして、選択したオブジェクトの **プロパティ** ダイアログを起動します。

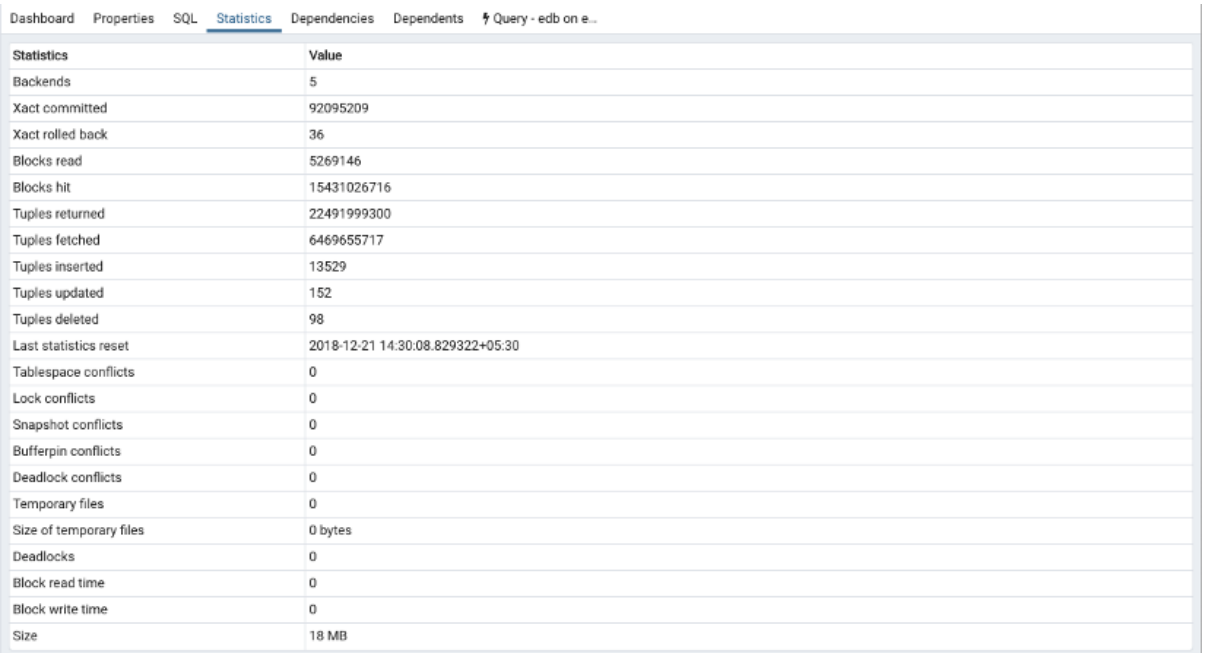
[**プロパティ**] ダイアログへの変更を保持するには、 **保存** アイコンをクリックします。変更は更新された **プロパティ** タブに表示されます。



ツリーコントロールで強調表示されたオブジェクトに関する詳細は、1 つ以上の折りたたみ可能なパネルに表示されます。各パネルラベルの左側にある矢印を使用して、パネルを開閉できます。



SQL タブには、強調表示されたオブジェクトを作成した SQL スクリプトと、該当する場合、選択したオブジェクトを *DROP* する（コメントアウトされた）SQL ステートメントが表示されます。カットアンドペーストのショートカットを使用して、SQL ステートメントを選択したエディターにコピーできます。



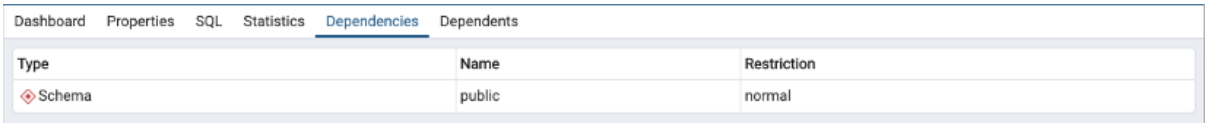
[**統計**] タブには、ツリーコントロールの各オブジェクトについて収集された統計が表示されます。テーブルに表示される統計は、選択したオブジェクトのタイプによって異なります。列見出しをクリックして、列に表示されているデータでテーブルを並べ替えます。再度クリックすると、ソート順が逆になります。次の表に、使用可能な統計の一部を示します。

パネル	説明
PID	行に関連付けられたプロセス ID。
ユーザー	オブジェクトを所有するユーザーの名前。
データベース	データベース名を表示します。
バックエンド	データベースへの現在の接続数を表示します。
バックエンドスタート	バックエンドプロセスの開始時刻。
正確なコミット	過去 1 週間にデータベースにコミットされたトランザクションの数を表示します。
正確にロールバック	過去 1 週間にロールバックされたトランザクションの数を表示します。
読み取りをブロック	過去 1 週間にメモリから読み取られたブロック数（メガバイト）を表示します。
ブロックヒット	過去 1 週間にキャッシュでヒットしたブロック数（メガバイト）を表示します。
返されるタプル	過去 1 週間に返されたタプルの数を表示します。
取得されたタプル	先週取得したタプルの数を表示します。
タプルが挿入されました	過去 1 週間にデータベースに挿入されたタプルの数を表示します。
タプルが更新されました	過去 1 週間にデータベースで更新されたタプルの数を表示します。
削除されたタプル	過去 1 週間にデータベースから削除されたタプルの数を表示します。
最後の統計リセット	データベースの統計が最後にリセットされた時刻を表示します。

次のページに続く

表 9 – 前のページからの続き

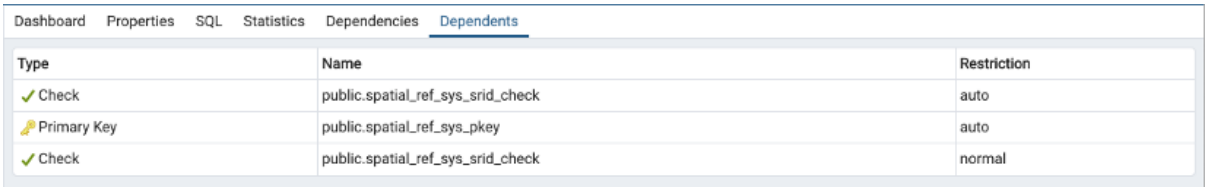
パネル	説明
テーブルスペースの競合	データベース内の削除されたテーブルスペースとのリカバリの競合が原因でキャンセルされたクエリの数を表示します。
ロックの競合	データベースのロックとのリカバリの競合が原因でキャンセルされたクエリの数を表示します。
スナップショットの競合	データベース内の古いスナップショットとのリカバリの競合が原因でキャンセルされたクエリの数を表示します。
バッファーピンの競合	データベース内の固定バッファとのリカバリの競合が原因でキャンセルされたクエリの数を表示します。
一時ファイル	統計コレクターによって使用されるものを含む、一時ファイルの合計数を表示します。
一時ファイルのサイズ	一時ファイルのサイズを表示します。
デッドロック	データベースのデッドロックとのリカバリの競合が原因でキャンセルされたクエリの数を表示します。
ブロック読み取り時間	読み取られたブロックの読み取りに必要なミリ秒数を表示します。
ブロック書き込み時間	読み取ったブロックの書き込みに必要なミリ秒数を表示します。
サイズ	選択したデータベースのサイズ（メガバイト）を表示します。



[**依存関係**] タブには、現在選択されているオブジェクトが依存するオブジェクトが表示されます。依存関係が削除されると、pgAdmin ツリーコントロールで現在選択されているオブジェクトが影響を受けます。データベース構造全体の整合性を保証するために、データベースサーバは、他のオブジェクトが依存しているオブジェクトを誤って削除しないようにします。依存関係を持つオブジェクトを削除するには、DROPCASCADE コマンドを使用する必要があります。

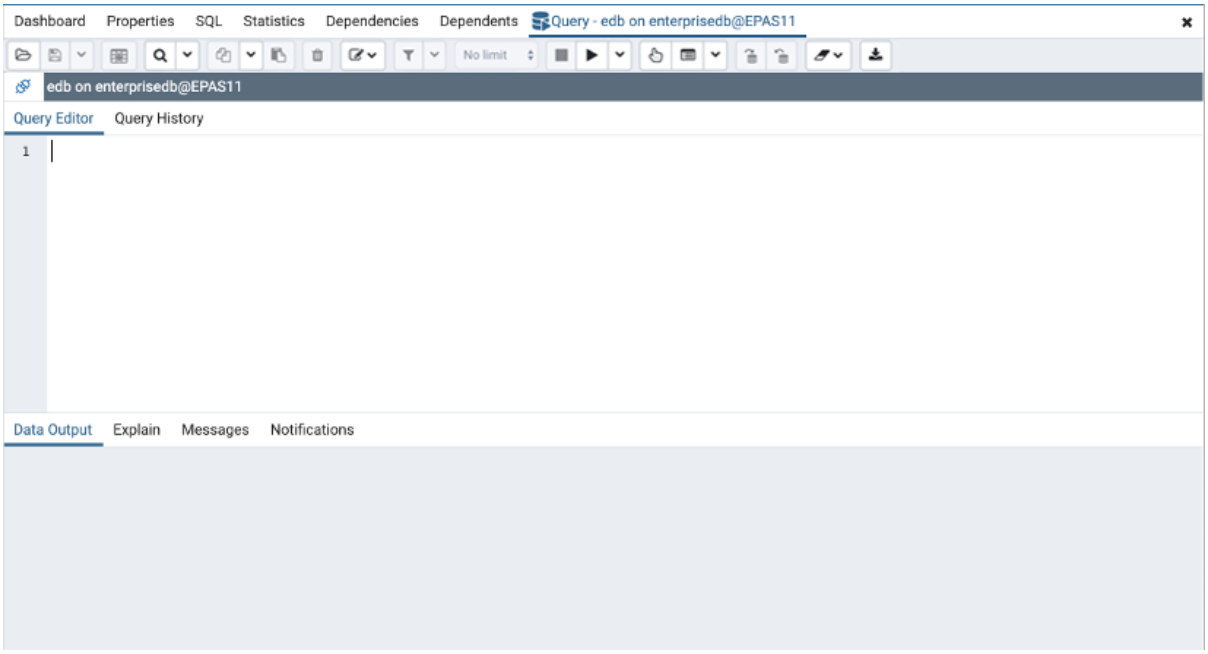
依存関係 テーブルには、次の情報が表示されます。

- *Type* フィールドは、親オブジェクトのタイプを指定します。
- *Name* フィールドは、親オブジェクトの識別名を指定します。
- **制限** フィールドは、現在選択されているオブジェクトと親間の依存関係を説明します。
  - フィールドが *auto* の場合、選択したオブジェクトは親オブジェクトとは別にドロップでき、親オブジェクトがドロップされるとドロップされます。
  - フィールドが *internal* の場合、選択したオブジェクトは親オブジェクトの作成中に作成され、親オブジェクトが削除されると削除されます。
  - フィールドが **通常** の場合、親オブジェクトをドロップせずに選択したオブジェクトをドロップできます。
  - フィールドが **空白** の場合、選択したオブジェクトはシステムに必要であり、削除できません。



Dependents タブには、*pgAdmin* ブラウザーで現在選択されているオブジェクトに依存するオブジェクトのテーブルが表示されます。依存オブジェクトは、*pgAdmin* ツリーコントロールで現在選択されているオブジェクトに影響を与えずに削除できます。

- *Type* フィールドは、依存オブジェクトのタイプを指定します。
- *Name* フィールドは、依存オブジェクトの識別名を指定します。
- *Database* フィールドは、オブジェクトが存在するデータベースを指定します。

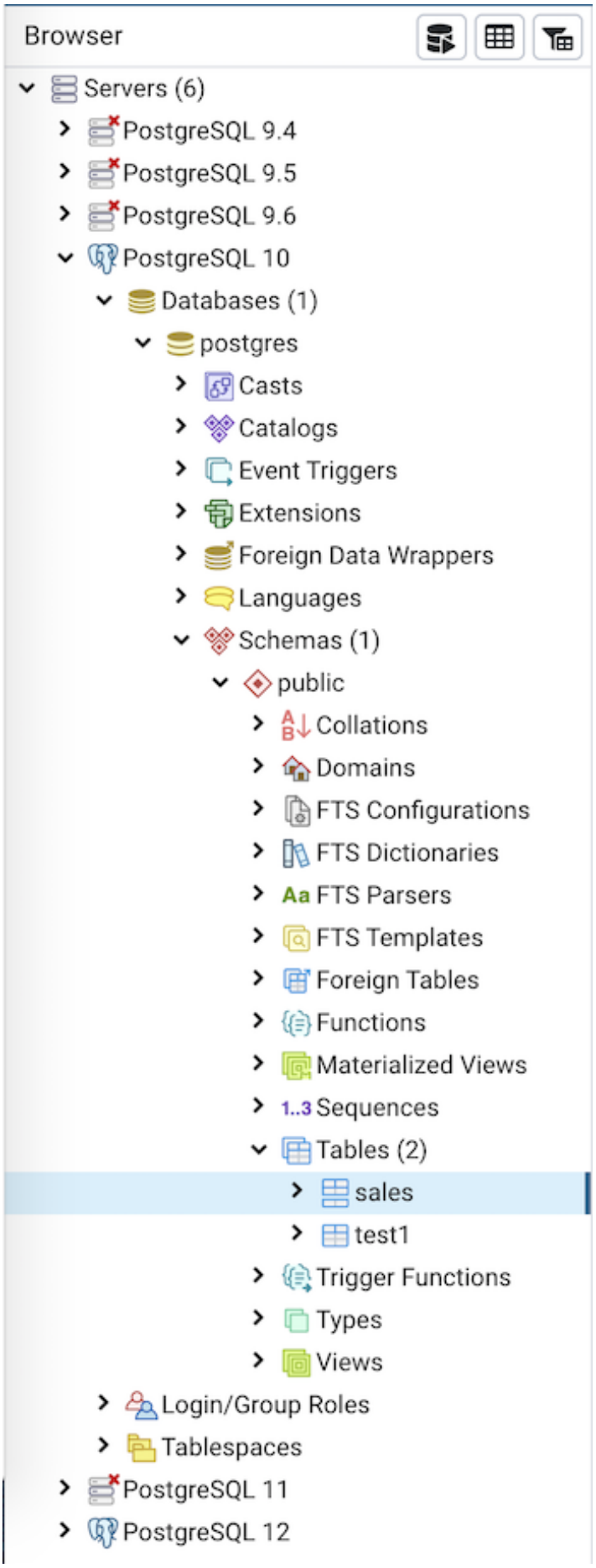


pgAdmin ツール（クエリツール、デバッガ、SQL エディタなど）が提供する拡張機能にアクセスすると、追加のタブが開きます。ツールの使用が終了したら、各タブの右上隅にある閉じるアイコン（X）を使用してタブを閉じます。永続的なタブと同様に、これらのタブは pgAdmin クライアントウィンドウで再配置できます。

デフォルトでは、ツールを開くたびに、pgAdmin は新しいブラウザータブを開きます。各ツールの **設定** ダイアログの *Display* ノードを変更することにより、この動作を制御できます。 **設定** ダイアログを開くには、**ファイル** メニューから **設定** を選択します。

1.16 Tree Control

メインウィンドウの左側のペインには、サーバー上にあるオブジェクトへのアクセスを提供するツリーコントロール（*pgAdmin* ツリーコントロール）が表示されます。


Browser
Servers (6)
PostgreSQL 9.4
PostgreSQL 9.5
PostgreSQL 9.6
PostgreSQL 10
Databases (1)
postgres
Casts
Catalogs
Event Triggers
Extensions
Foreign Data Wrappers
Languages
Schemas (1)
public
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Sequences
Tables (2)
sales
test1
Trigger Functions
Types
Views
Login/Group Roles
Tablespaces
PostgreSQL 11
PostgreSQL 12

ツリーコントロールのノードを展開して、選択したサーバーにあるデータベースオブジェクトを表示できます。ツリーコントロールが展開され、階層ビューが表示されます。

- ノードの左側にあるプラス記号（+）を使用して、ツリーコントロールのセグメントを展開します。
- ノードの左側にあるマイナス記号（-）をクリックして、そのノードを閉じます。

特定のオブジェクトをクエリツールに **drag and drop** すると、長いオブジェクト名を入力する時間を節約できます。オブジェクト名を含むテキストは、スキーマで完全修飾されます。必要に応じてダブルクォーテーションが追加されます。関数とプロシージャの場合、関数名とパラメーター名がクエリツールに貼り付けられます。

ツリーコントロールのノードを右クリックしてコンテキスト依存メニューにアクセスして、一般的なタスクを実行します。メニューには、次の選択項目を 1 つ以上含むオプションが表示されます（オプションはアルファベット順に表示されます）。

オプション	アクション
名前付けられた復元ポイントを追加	クリックして、復元ポイントの名前を作成し、入力します。
バックアップ...	クリックして *Backup...* ダイアログを開き、データベースオブジェクトをバックアップします。
バックアップグローバル...	クリックして *Backup Globals...* ダイアログを開き、クラスターオブジェクトをバックアップします。
バックアップサーバー...	クリックして *Backup Server...* ダイアログを開き、サーバーをバックアップします。
サーバーに接続...	クリックして *Connect to Server* ダイアログを開き、サーバーとの接続を確立します。
作成	クリックして、状況依存の選択を提供するコンテキストメニューにアクセスします。選択すると、新しいオブジェクトを作成するための *Create* ダイアログが開きます。
CREATE **スクリプト**	クリックして *Query tool* を開き、CREATE スクリプトを編集または表示します。
デバッグ	クリックして *Debug* ツールを開くか、[**ブレークポイントの設定**] を選択してスクリプトの実行を停止または一時停止します。
削除/ドロップ	クリックして、現在選択されているオブジェクトをサーバーから削除します。
データベースを切断...	クリックして、データベース接続を終了します。
サーバーから切断する	クリックして、現在選択されているサーバーから切断します。
ドロップカスケード	クリックして、現在選択されているオブジェクトとすべての依存オブジェクトをサーバーから削除します。
デバッグ	クリックして、 *Debugger* ツールにアクセスします。
Grant **ウィザード**	クリックして、 *Grant Wizard* ツールにアクセスします。
メンテナンス...	クリックして、VACUUM、ANALYZE、REINDEX、または CLUSTER の *Maintenance...* ダイアログを開きます。
プロパティ...	クリックして、現在選択されているオブジェクトのプロパティを確認または変更します。
更新...	クリックして、現在選択されているオブジェクトを更新します。
設定をリロード...	クリックして、サーバーを再起動せずに構成ファイルを更新します。
復元...	クリックして *Restore* ダイアログにアクセスし、バックアップからデータベースファイルを復元します。
データを見る	**クエリツール** の **データ出力** タブで選択したテーブルに保存されているデータにアクセスするには、 **データの表示** オプションを使用します。

テーブル とネストされた **テーブル** ノードに関連付けられた状況依存メニューには、追加の表示オプションがあります（オプションはアルファベット順に表示されます）。

オプション	アクション
データのインポート/エクスポート...	クリックして *Import/Export...* ダイアログを開き、選択したテーブルとデータをインポートまたはエクスポートします。
統計をリセット	クリックして、選択したテーブルの統計をリセットします。
スクリプト	クリックして *Query tool* を開き、フライアウトメニューから選択したスクリプトを編集します。
切り捨て	クリックして、テーブルからすべての行を削除します。
カスケードを切り捨てる	クリックして、テーブルとその子テーブルからすべての行を削除します。
最初の *100* 行を表示	クリックして、選択したテーブルの最初の 100 行を表示するデータグリッドにアクセスします。
最後の *100* 行を表示	クリックして、選択したテーブルの最後の 100 行を表示するデータグリッドにアクセスします。
すべての行を表示	クリックして、選択したテーブルのすべての行を表示するデータグリッドにアクセスします。
フィルタされた行を表示...	クリックして *DataFilter* ポップアップにアクセスし、データのセットにフィルターを適用します。

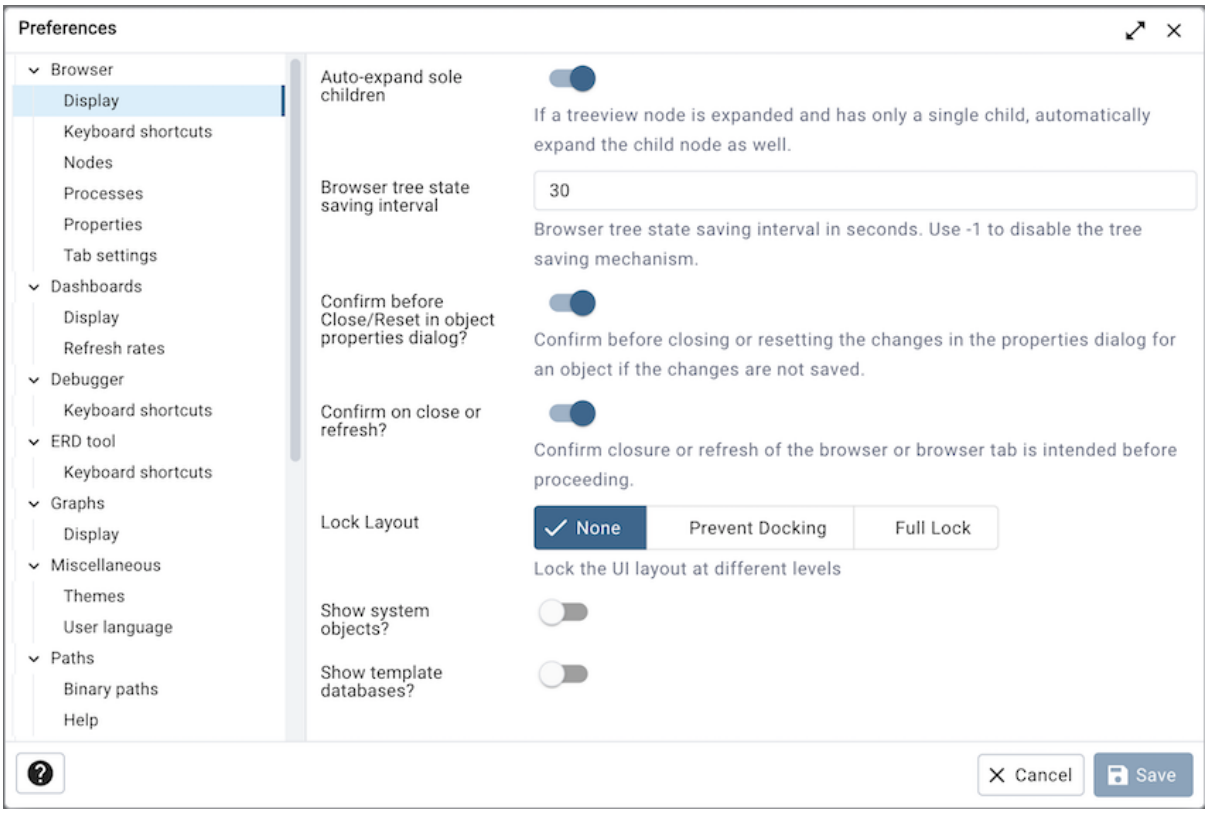
1.17 Preferences Dialog

設定 ダイアログのオプションを使用して、クライアントの動作をカスタマイズします。 **設定** ダイアログを開くには、 **ファイル** メニューから **設定** を選択します。 **設定** ダイアログの左ペインにはツリーコントロールが表示されます。ツリーコントロールの各ノードは、それらが表示されるノードに関連するオプションへのアクセスを提供します。

- ノード名の左側にあるプラス記号（+）を使用して、ツリーコントロールのセグメントを展開します。
- ノード名の左側にあるマイナス記号（-）を使用して、そのノードを閉じます。

1.17.1 ブラウザノード

ツリーコントロールの **ブラウザー** ノードにある設定を使用して、ワークスペースをパーソナライズします。



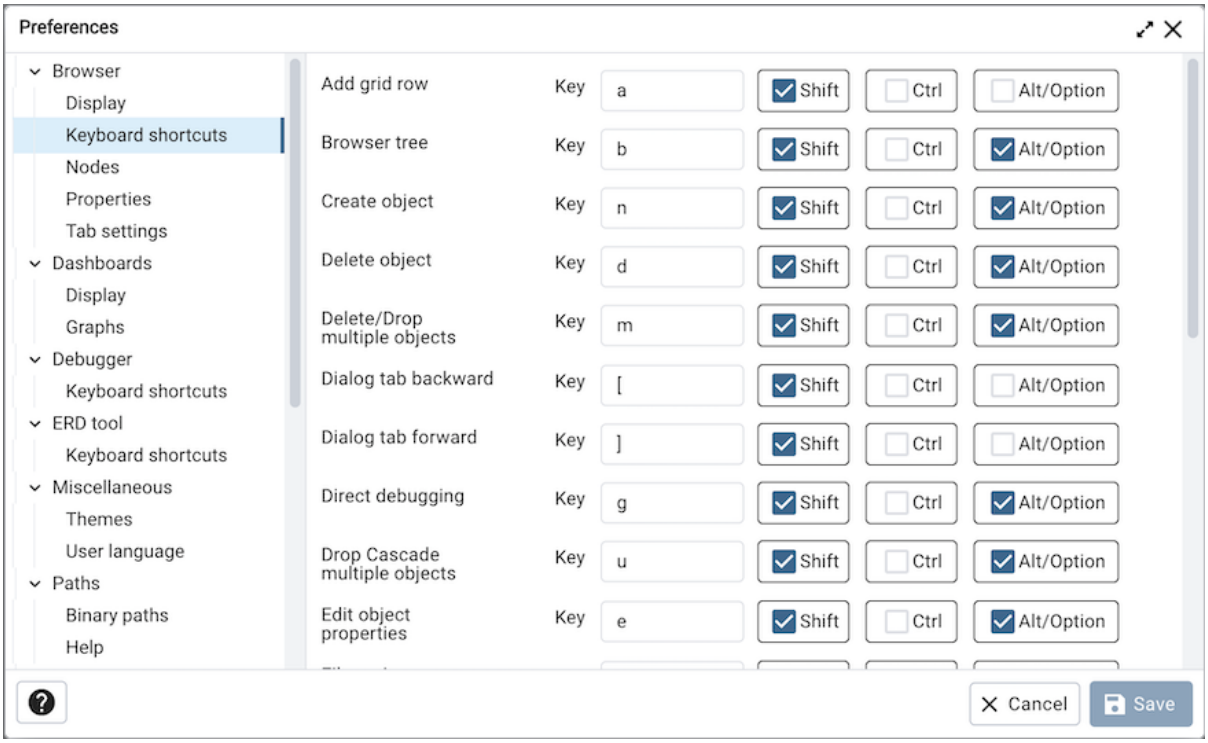
Display パネルのフィールドを使用して、一般的な表示設定を指定します。

- *Auto-expandSolechildren* スイッチが *True* に設定されている場合、ツリービューノードが展開され、子が 1 つだけの場合、子ノードは自動的に展開されます。
- **ブラウザツリーの状態の保存間隔** フィールドを使用して、ツリービューの状態の保存間隔を設定します。 *-1* の値は、ツリービューの状態保存機能を無効にします。
- **未保存の変更でプロパティを閉じる前に確認** スイッチが *True* に設定されている場合、未保存の変更がある場合、pgAdmin はオブジェクトのプロパティダイアログを閉じる前に警告します。ユーザーの確認で、プロパティダイアログが閉じます。
- **閉じるまたは更新時に確認** スイッチが *True* に設定されている場合、pgAdmin はブラウザーを閉じるまたは更新するイベントをキャッチしようとし、続行する前にプロンプトを表示します。
- **共有サーバーを非表示にしますか?** スイッチが *True* に設定されている場合、クライアントはブラウザーツリーからすべての共有サーバーを非表示にします。
- **レイアウトのロック** フィールドを使用して、さまざまなレベルで UI レイアウトをロックします。これは、 *menu bar* の [ファイル] メニューから変更することもできます

オプション	アクション
なし	ロックなし。すべてのパネルはサイズ変更とドッキングが可能です。
ドッキングを防ぎます	これにより、パネルのドッキング/ドッキング解除が無効になります
フル	これにより、パネルのサイズ変更、ドッキング/ドッキング解除が無効になります

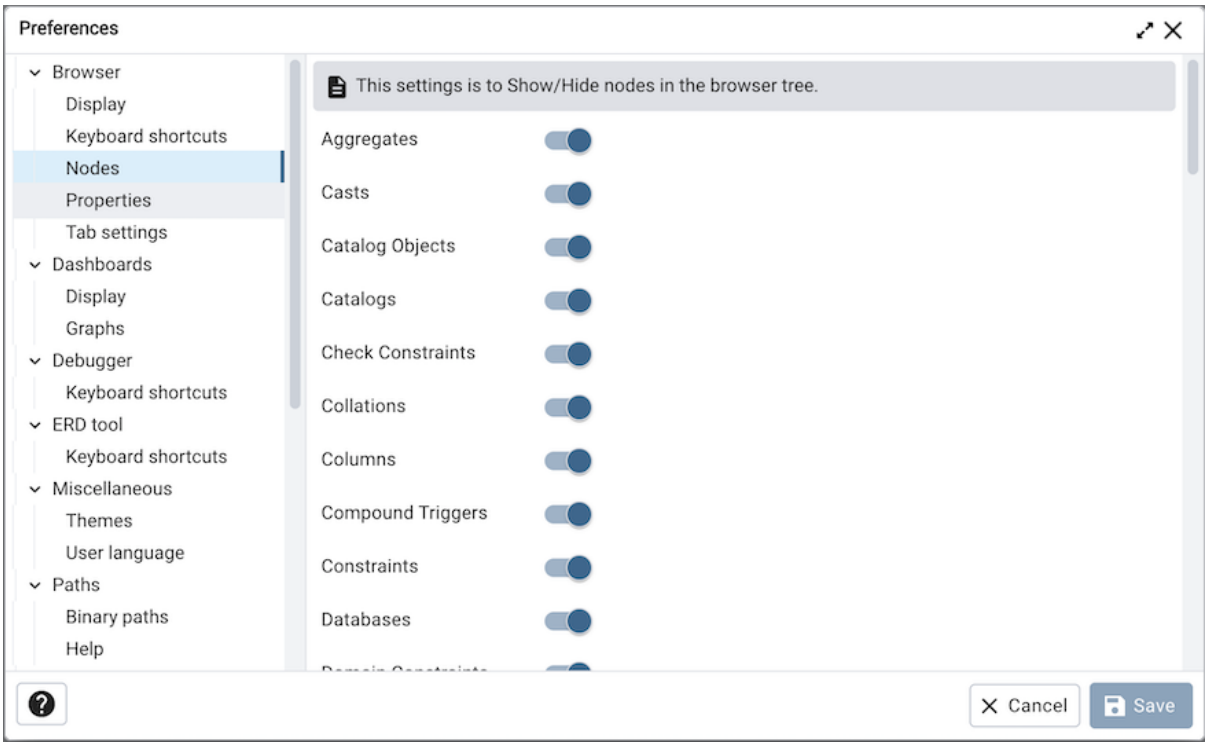
- **システムオブジェクトを表示しますか**? スイッチが *True* に設定されている場合、クライアントはシステムスキーマ（*pg_temp* など）やシステム列（*xmin* や *ctid* など）などのシステムオブジェクトを表示しますツリーコントロール。
- **テンプレートデータベースを表示しますか**? スイッチが *True* に設定されている場合、クライアントはテンプレートデータベースを表示します。

[**キーボードショートカット**] パネルのフィールドを使用して、メインウィンドウのナビゲーションのショートカットを構成します。



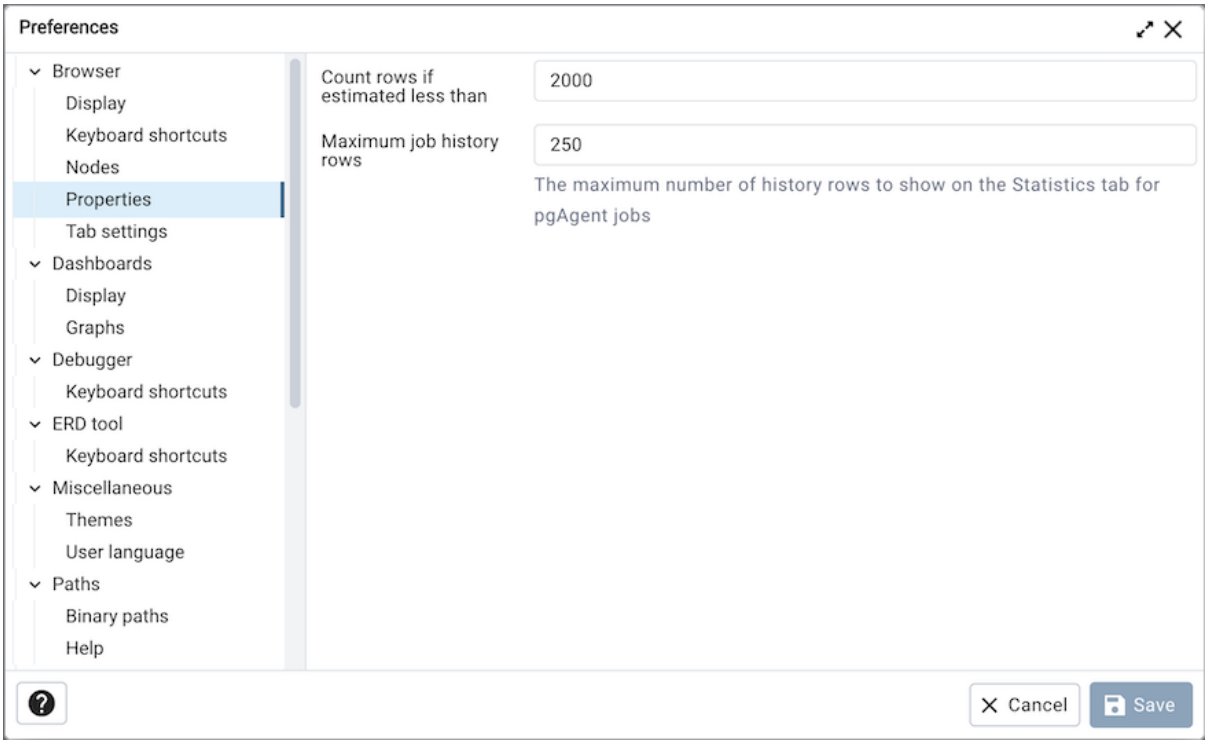
- パネルには、メインウィンドウで使用可能なキーボードショートカットのリストが表示されます。修飾キーとキーの組み合わせを選択して、各ショートカットを構成します。

ノード パネルのフィールドを使用して、 **ブラウザー** ツリーコントロールに表示されるオブジェクトタイプを選択します。



- パネルにはデータベースオブジェクトのリストが表示されます。各オブジェクトの横にあるスイッチをスライドして、データベースオブジェクトを **表示** または **非表示** にします。システムカタログを照会するときに、表示されるオブジェクトタイプの数を減らして速度を上げることができます。

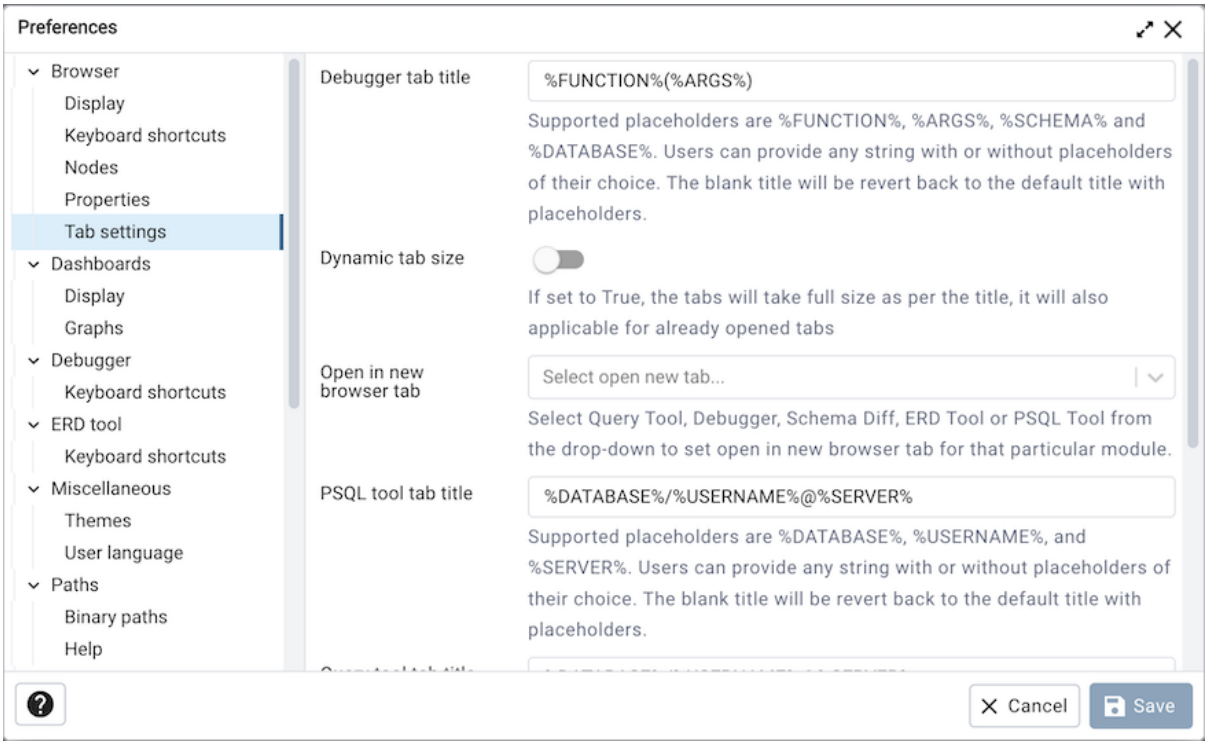
[**プロパティ**] パネルのフィールドを使用して、ブラウザーのプロパティを指定します。



- テーブル内の推定行数（テーブル統計から読み取った）が指定された制限を下回る場合、 **推定された場合は行をカウント** フィールドに値を含めて SELECTcount( ) **を実行します。***SELECTcount( )* を実行した後、pgAdmin は行数を表示します。デフォルトは 2000 です。

- **最大ジョブ履歴行** フィールドに値を入力して、pgAgent ジョブの統計タブに表示する行数を制限します。デフォルトは 250 です。

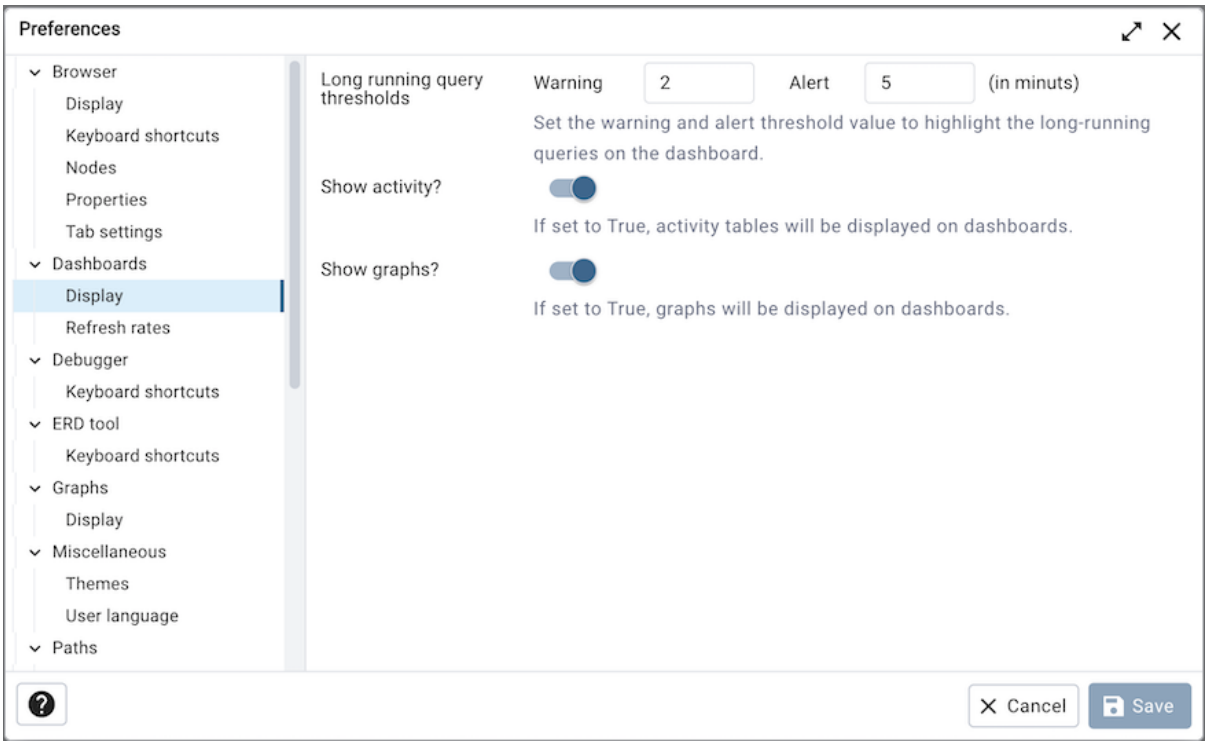
タブ設定 パネルのフィールドを使用して、タブ関連のプロパティを指定します。



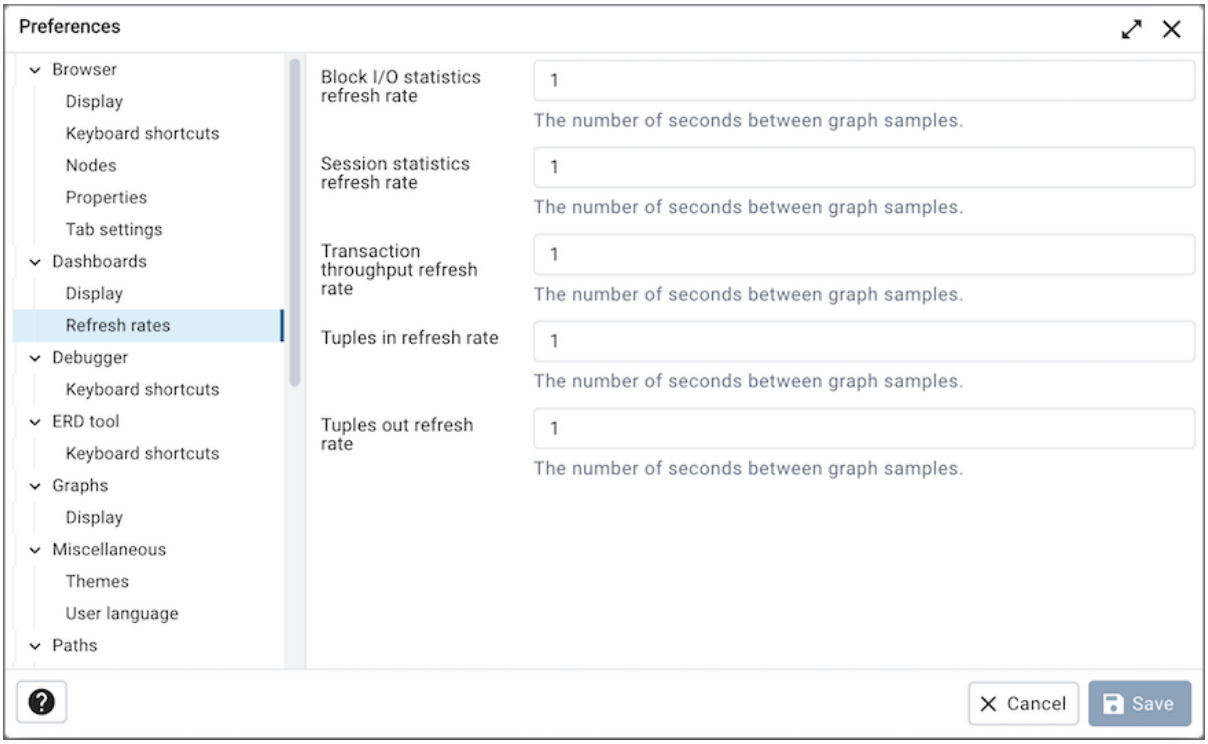
- *Debuggertabtitle* **プレースホルダー** フィールドを使用して、[デバッガー] タブのタイトルをカスタマイズします。
- **動的タブサイズ** True に設定されている場合、タブはタイトルごとにフルサイズになり、既に開いているタブにも適用されます
- クエリツール、スキーマ差分またはデバッガーで **新しいブラウザータブで開く** フィールドが選択されている場合、呼び出されたときに新しいブラウザータブで開きます。
- **クエリツールタブのタイトルプレースホルダー** フィールドを使用して、クエリツールのタブのタイトルをカスタマイズします。
- *View/Edittabtitle* **プレースホルダー** フィールドを使用して、[View/EditData] タブのタイトルをカスタマイズします。

1.17.2 ダッシュボードノード

ダッシュボード ノードを展開して、ダッシュボードの表示設定を指定します。



- 警告とアラートのしきい値を設定して、ダッシュボードで実行時間の長いクエリを強調表示します。
- **アクティビティを表示しますか**? スイッチが *True* に設定されている場合、アクティビティテーブルはダッシュボードに表示されます。
- **グラフを表示しますか**? スイッチが *True* に設定されている場合、グラフはダッシュボードに表示されます。



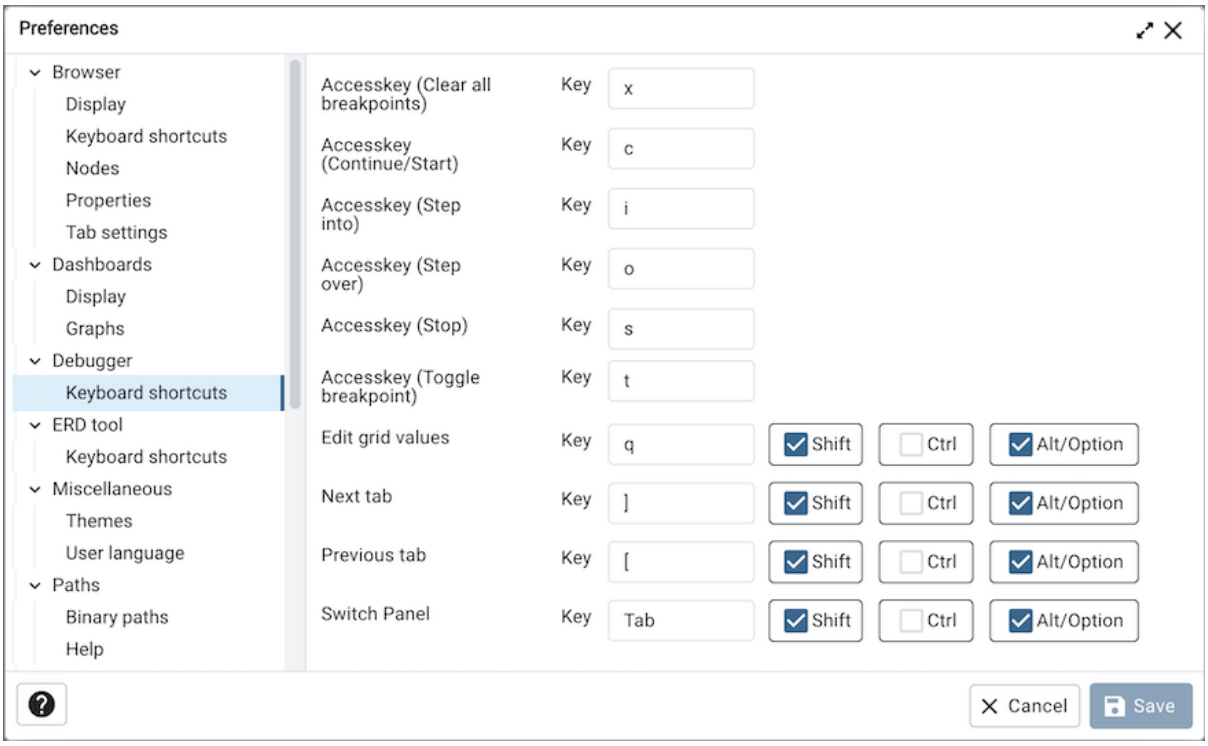
リフレッシュレート パネルのフィールドを使用して、 **ダッシュボード** タブのグラフの参照レート設定を指定します。

- **ブロック *I/O* 統計のリフレッシュレート** フィールドを使用して、グラフに表示されるブロック I/O 統計のサンプル間の秒数を指定します。
- **セッション統計のリフレッシュレート** フィールドを使用して、グラフに表示されるセッション統計サンプル間の秒数を指定します。
- **トランザクションスループットのリフレッシュレート** フィールドを使用して、グラフに表示されるトランザクションスループットのサンプル間の秒数を指定します。
- *Tuplesinrefreshrate* フィールドを使用して、グラフに表示されるタプルインのサンプル間の秒数を指定します。
- **タプルアウトリフレッシュレート** フィールドを使用して、グラフに表示されるタプルアウトサンプル間の秒数を指定します。

1.17.3 デバッガーノード

Debugger ノードを展開して、デバッガーの表示設定を指定します。

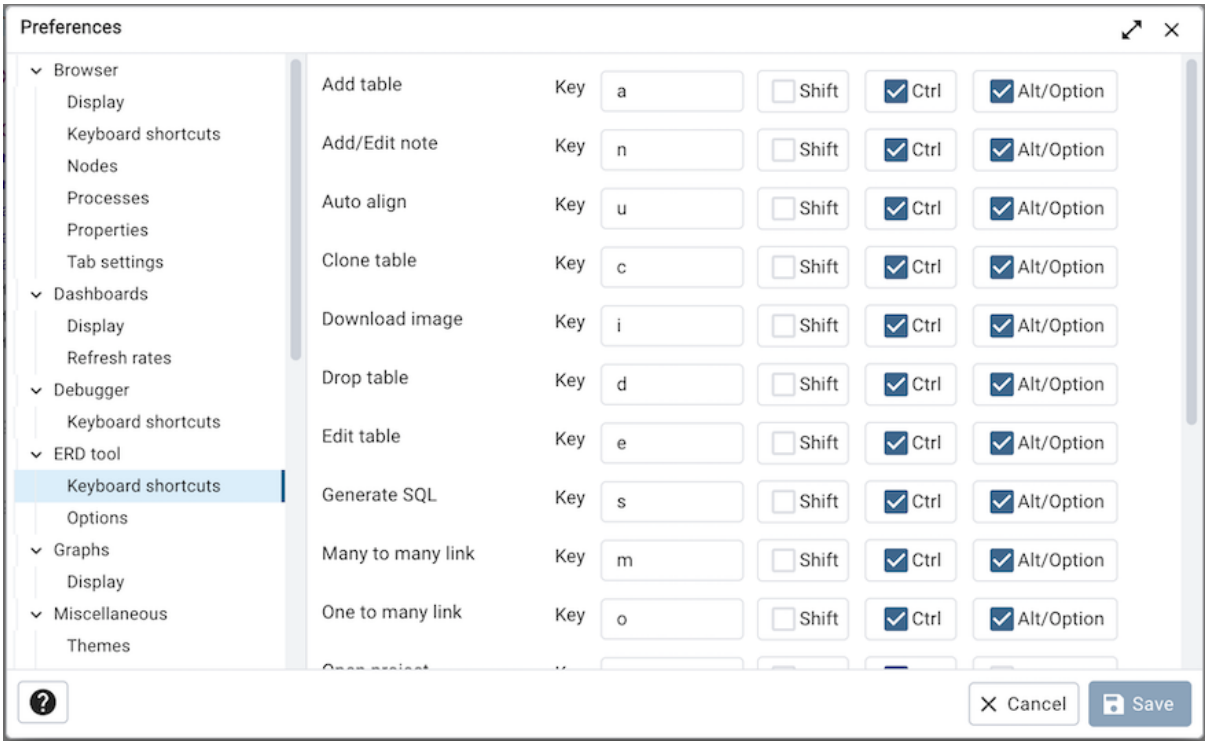
[**キーボードショートカット**] パネルのフィールドを使用して、デバッガーウィンドウのナビゲーションのショートカットを構成します。



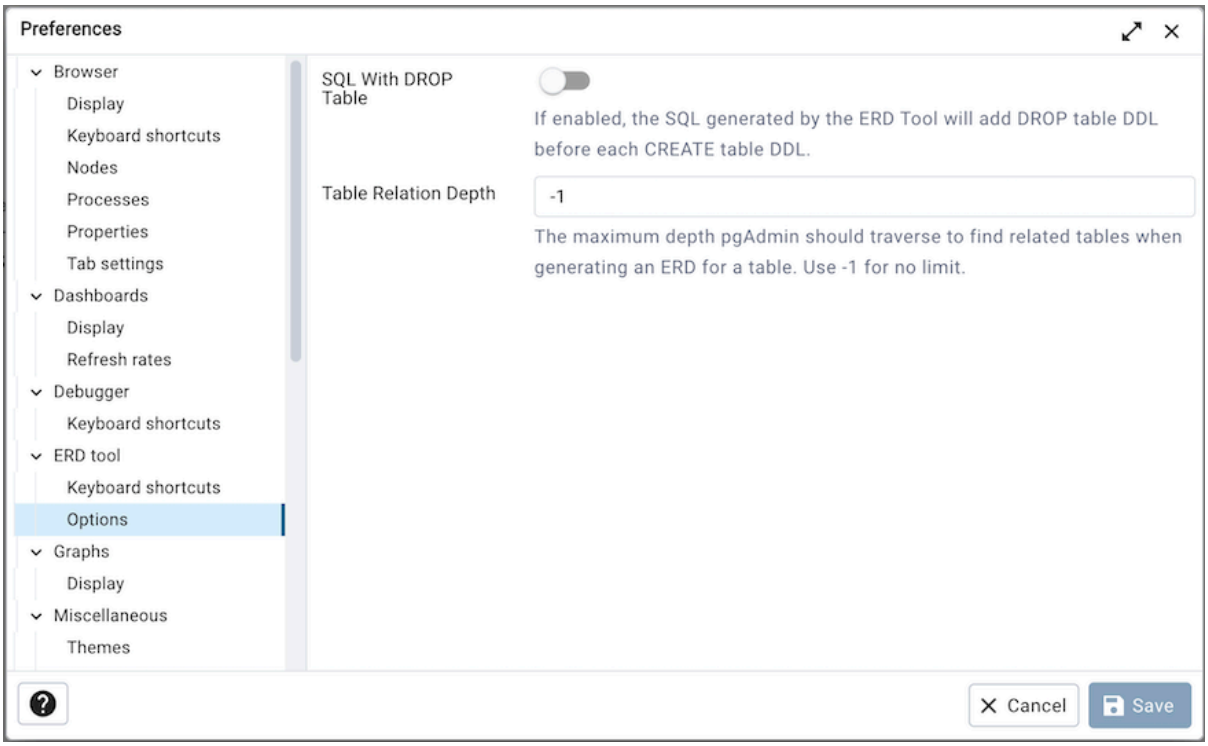
1.17.4 ERD ツールノード

ERDTool ノードを展開して、ERDTool の表示設定を指定します。

[**キーボードショートカット**] パネルのフィールドを使用して、ERD ツールウィンドウのナビゲーションのショートカットを構成します。



Options パネルのフィールドを使用して、ERD 設定を管理します。



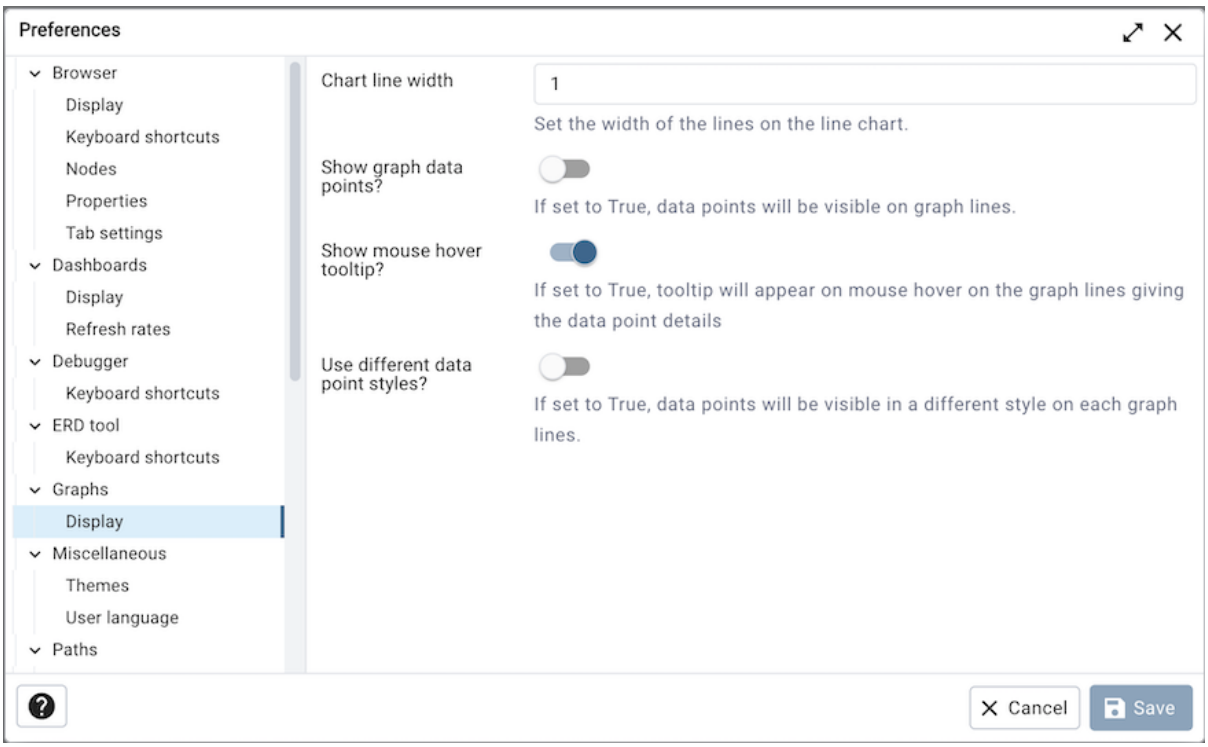
- *SQLWithDROPTable* スイッチが *True* に設定されている場合、ERD ツールによって生成された SQL

は各 CREATEtableDDL の前に DROPtableDDL を追加します。

- **テーブルリレーションの深さ** は、テーブルの ERD を生成するときに役立ちます。pgAdmin がリレーションを見つけるためにトラバースする必要がある深さレベルに制限を設定できます。制限を設定しない場合は-1 を使用します。

1.17.5 グラフノード

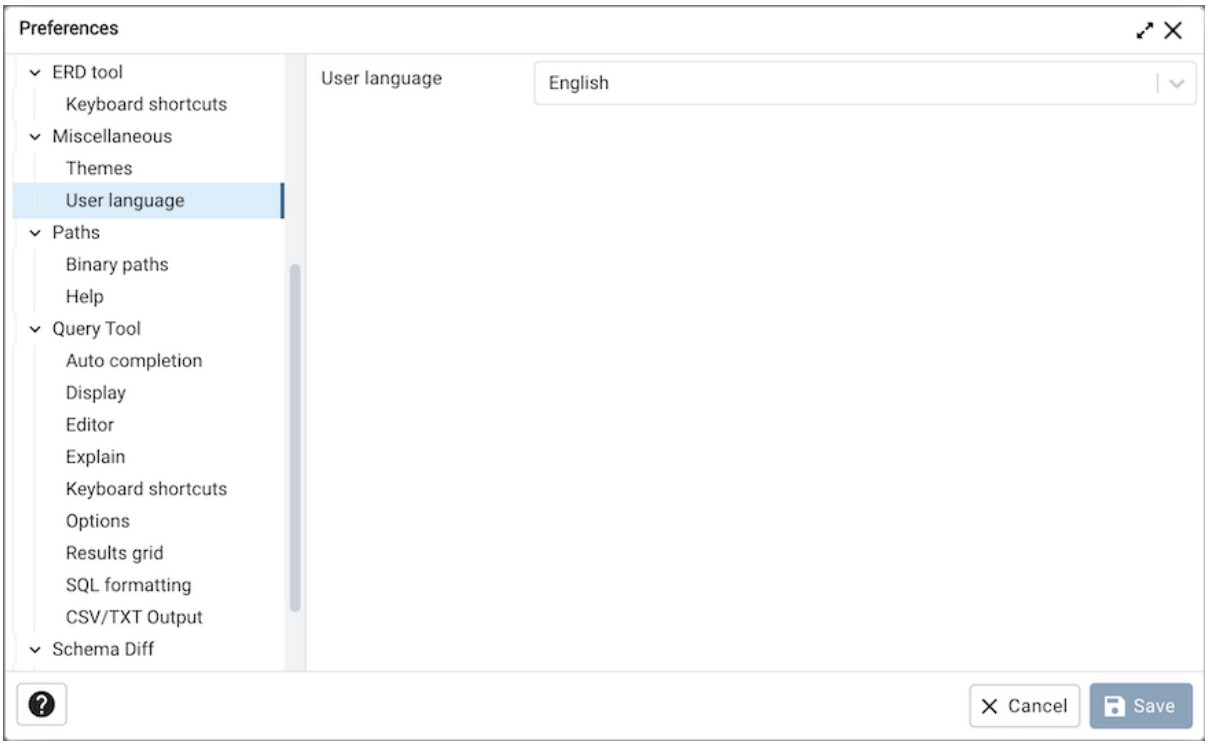
Graphs ノードを展開して、グラフ表示設定を指定します。



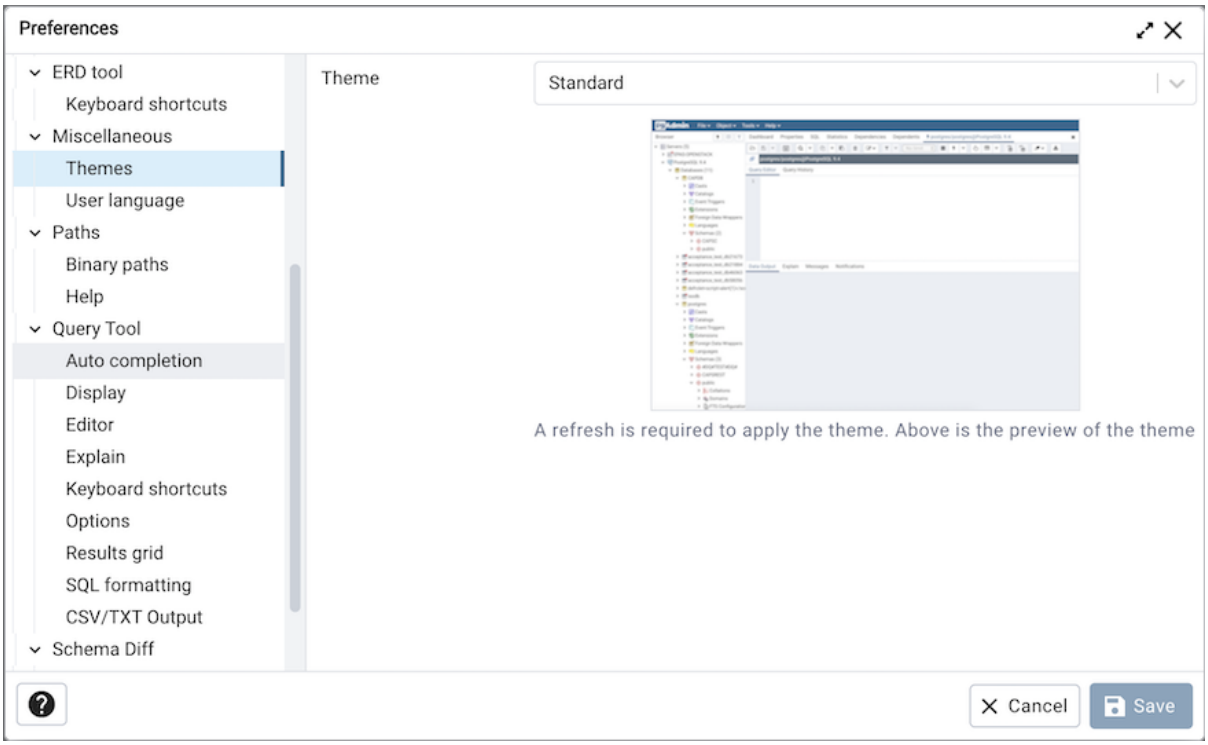
- *Chartlinewidth* フィールドを使用して、折れ線グラフの線の幅を指定します。
- **グラフデータポイントを表示しますか***?* スイッチが *True* に設定されている場合、データポイントはグラフの線に表示されます。
- **マウスホバーツールチップを表示しますか***?* スイッチが *True* に設定されている場合、グラフの線にマウスを合わせるとツールチップが表示され、データポイントの詳細が示されます。
- **異なるデータポイントスタイルを使用しますか***?* スイッチが *True* に設定されている場合、データポイントは各グラフの線に異なるスタイルで表示されます。

1.17.6 その他のノード

Miscellaneous ノードを展開して、その他の表示設定を指定します。



- **ユーザー言語** ドロップダウンリストボックスを使用して、クライアントの表示言語を選択します。

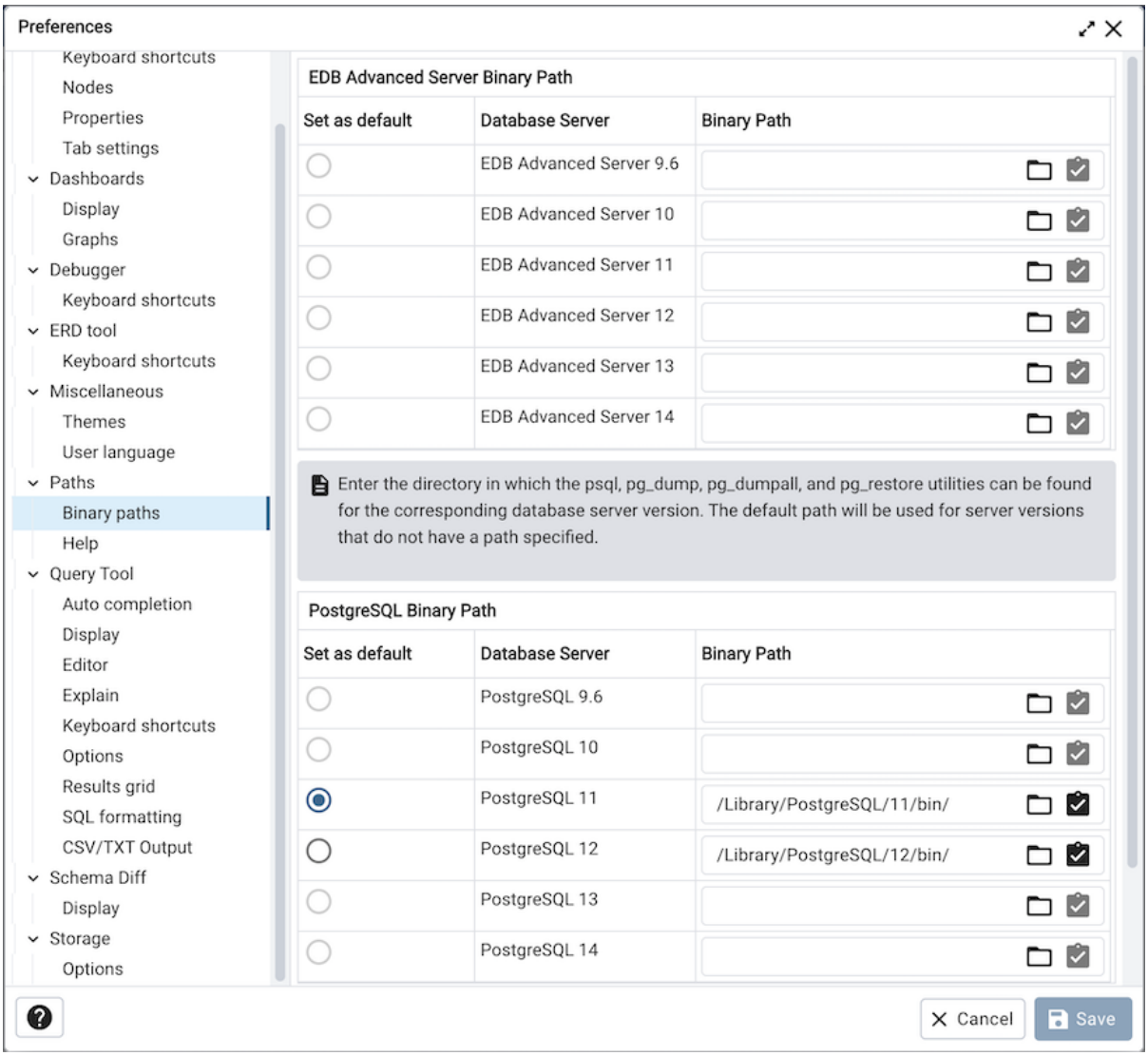


- *Themes* ドロップダウンリストボックスを使用して、pgAdmin のテーマを選択します。ドロップダウンのすぐ下にプレビューも表示されます。テーマを適用するには、pgAdmin ページを更新する必要があることに注意してください。独自のテーマを送信することもできます。 here を確認してくださ

い。現在、スタンダード、ダーク、ハイコントラストのテーマをサポートしています。

1.17.7 パスノード

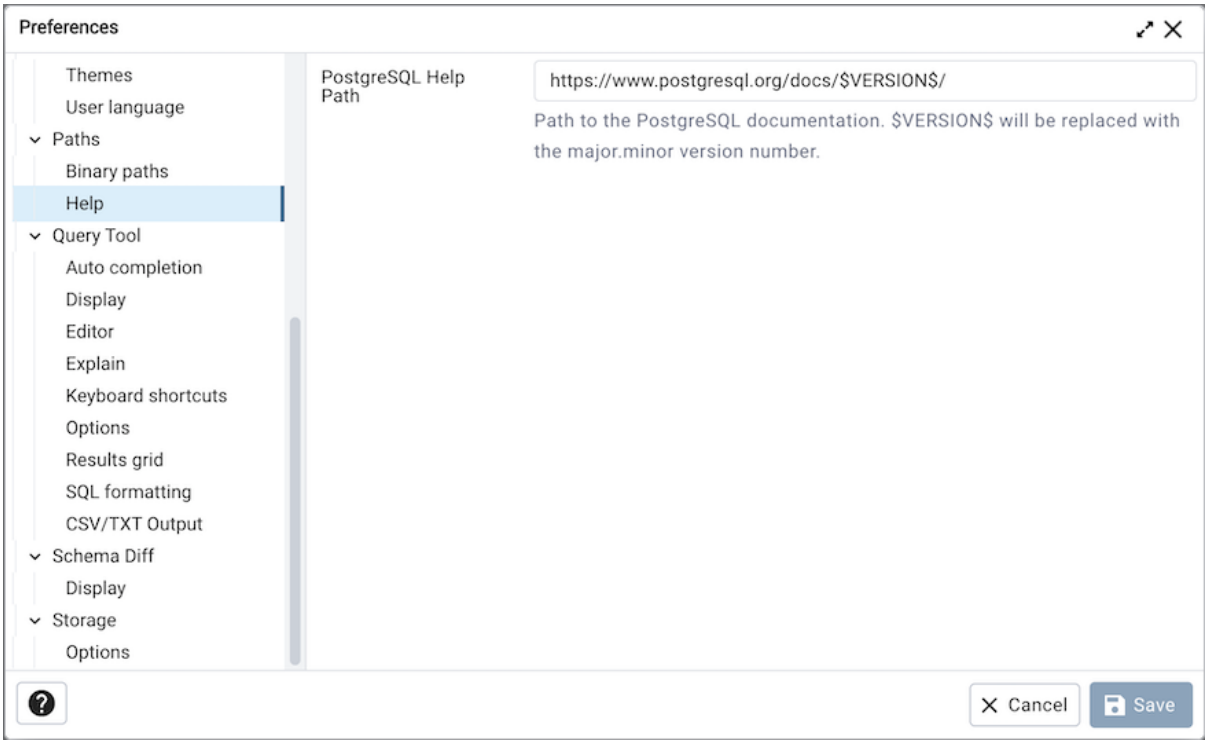
パス ノードを展開して、サポートするユーティリティとヘルプファイルの場所を指定します。



[**バイナリパス**] パネルのフィールドを使用して、監視対象データベースのユーティリティプログラム（pg_dump、pg_dumpall、pg_restore、および psql）を含むディレクトリへのパスを指定します。

- *EDBAdvancedServer* **バイナリパス** グリッドを使用して、サーバーバージョンに基づいて EDBPostgresAdvancedServer ユーティリティプログラムの場所を指定します。それぞれのパスが設定されていない場合、pgAdmin は「デフォルトとして設定」がチェックされているパスを選択します。そうでない場合、EnterpriseDB が使用する標準の場所でユーティリティを見つけようとします。
- *PostgreSQL* **バイナリパス** グリッドを使用して、サーバーバージョンに基づいて PostgreSQL ユーティリティプログラムの場所を指定します。それぞれのパスが設定されていない場合、pgAdmin は「デフォルトとして設定」がチェックされているパスを選択します。そうでない場合、pgAdmin は PostgreSQL で使用される標準の場所でユーティリティを見つけようとします。

Note: [パスの検証] ボタンを使用して、ユーティリティプログラム（pg_dump、pg_dumpall、pg_restore、および psql）とそれぞれのバージョンの存在を確認します。



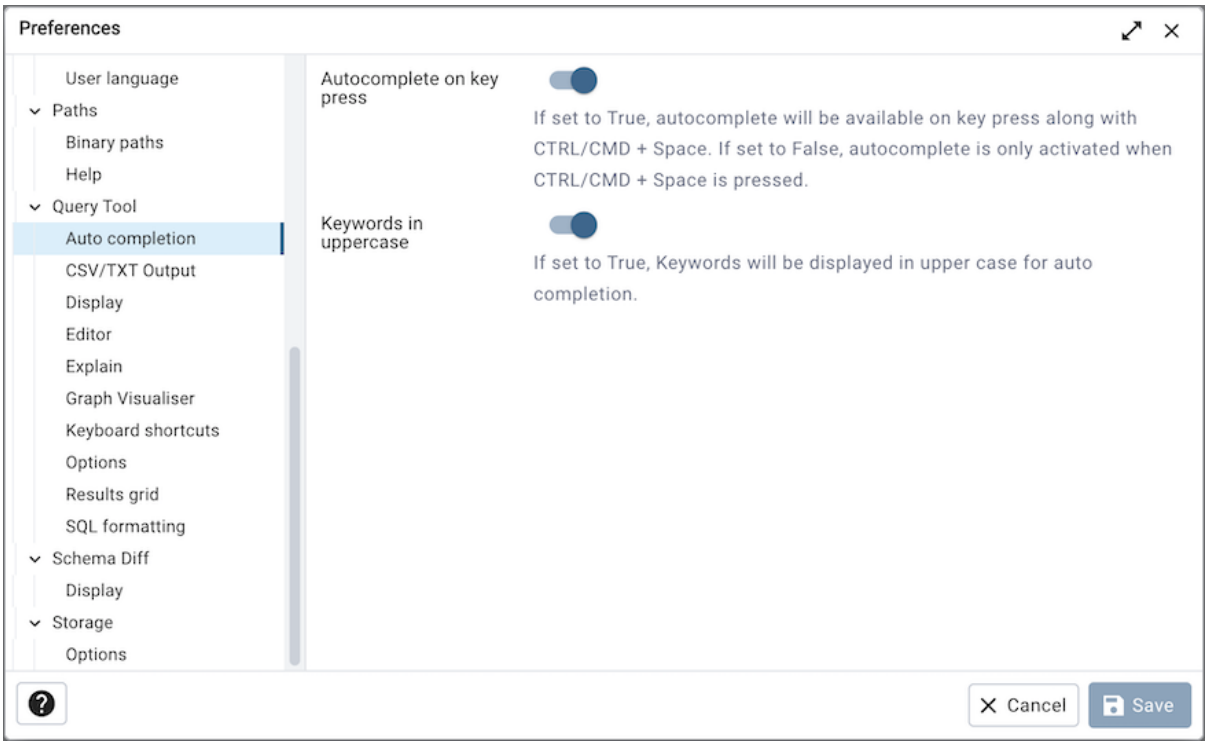
[**ヘルプ**] パネルのフィールドを使用して、ヘルプファイルの場所を指定します。

- *EDBAdvancedServer* **ヘルプパス** を使用して、EDBPostgresAdvancedServer ドキュメントへのパスを指定します。
- *PostgreSQLHelpPath* を使用して、PostgreSQL ドキュメントへのパスを指定します。

注：デフォルトのヘルプパスには *VERSION* プレースホルダーが含まれています。$VERSION$プレースホルダーは現在のデータベースバージョンに置き換えられます。

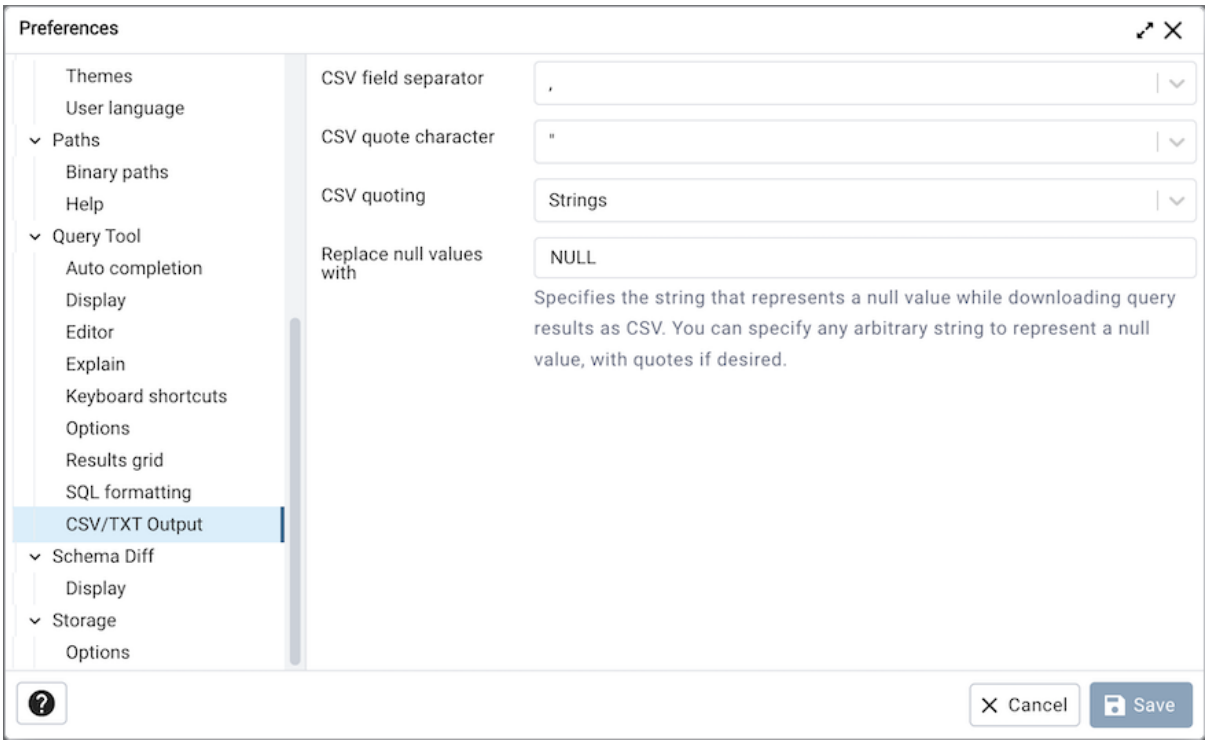
1.17.8 クエリツールノード

クエリツール ノードを展開して、クエリエディターツールの設定を指定できるパネルにアクセスします。



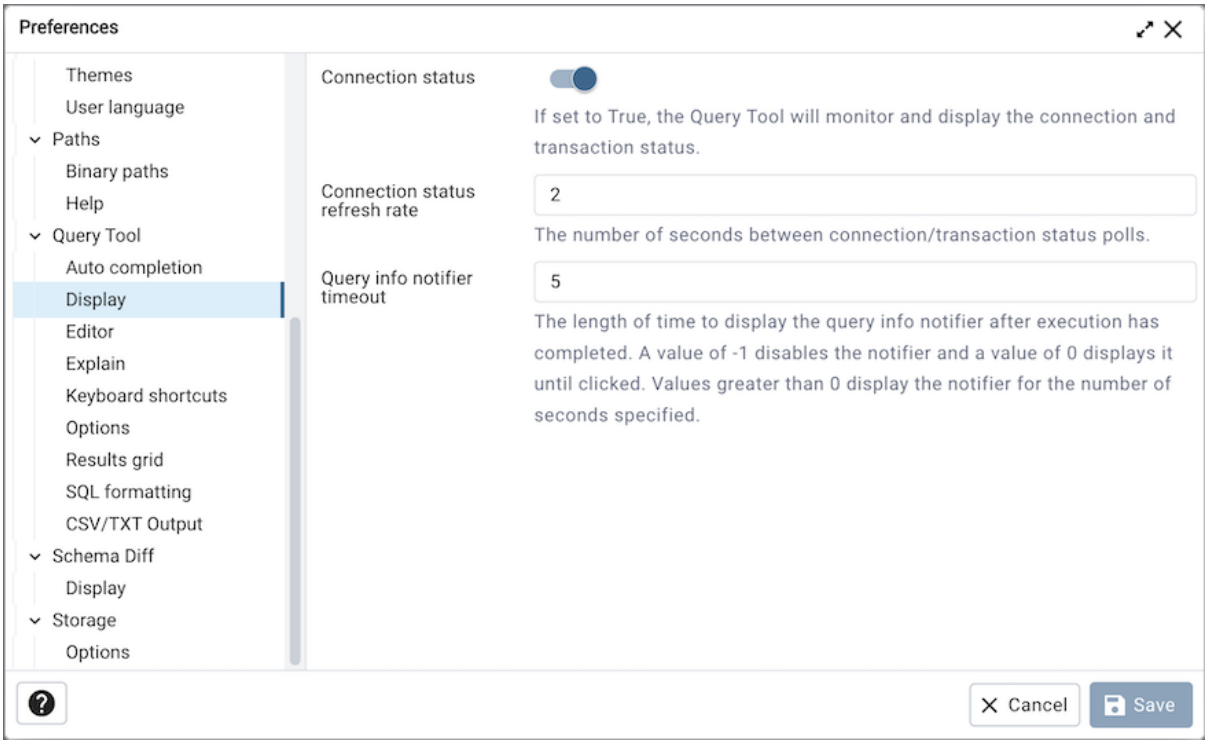
オートコンプリート パネルのフィールドを使用して、オートコンプリートオプションを設定します。

- **キーを押すとオートコンプリート** スイッチが *True* に設定されている場合、CTRL/CMD+Space とともにキーを押すとオートコンプリートが利用できます。 *False* に設定されている場合、オートコンプリートは CTRL/CMD+Space が押されたときにのみアクティブになります。
- *Keywordsinuppercase* スイッチが *True* に設定されている場合、キーワードは大文字で表示されます。



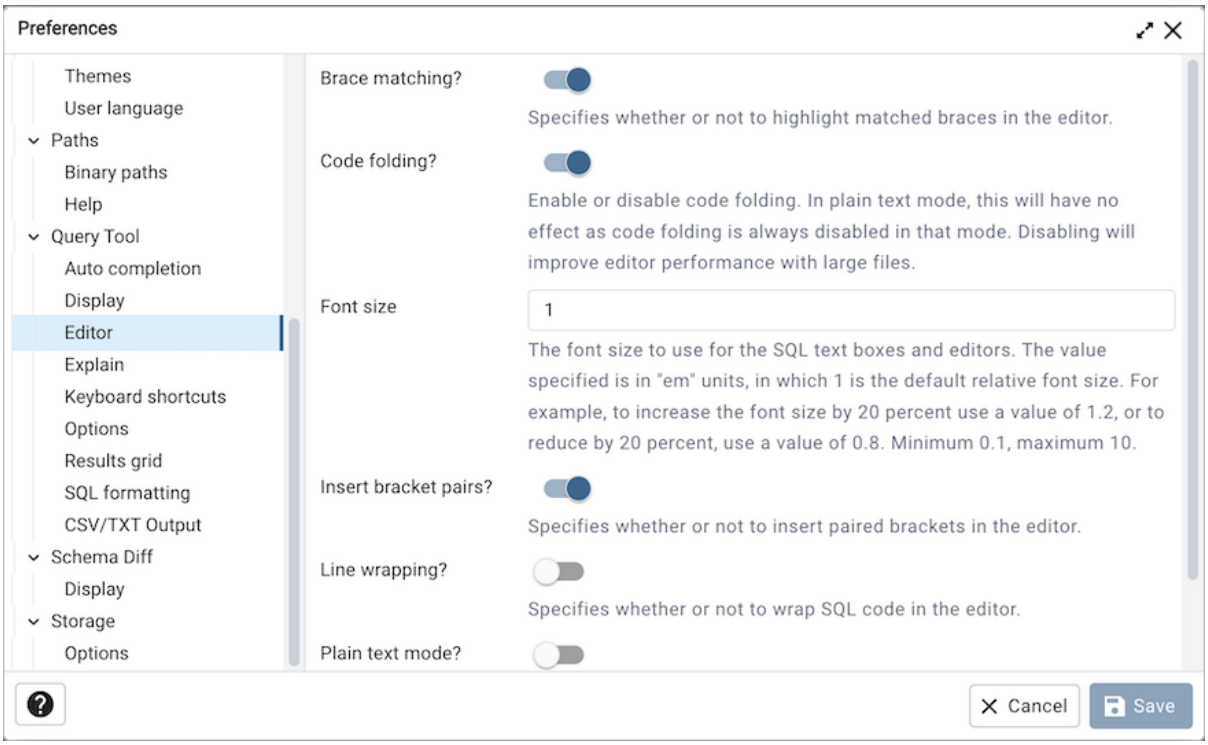
CSV/TXTOutput パネルのフィールドを使用して、CSV/TXT 出力を制御します。

- *CSV* **フィールドセパレーター** ドロップダウンリストボックスを使用して、CSV/TXT 出力で使用される区切り文字を指定します。
- *CSVquotecharacter* ドロップダウンリストボックスを使用して、CSV/TXT 出力で使用される引用符を指定します。
- *CSV* **引用** ドロップダウンリストボックスを使用して、CSV/TTX 出力で引用されるフィールドを選択します。 **文字列** 、 **すべて** 、または **なし** を選択します。
- **ヌル値を置換** オプションを使用して、出力ファイル内のヌル値を指定した文字列に置き換えます。デフォルトは「NULL」に設定されています。



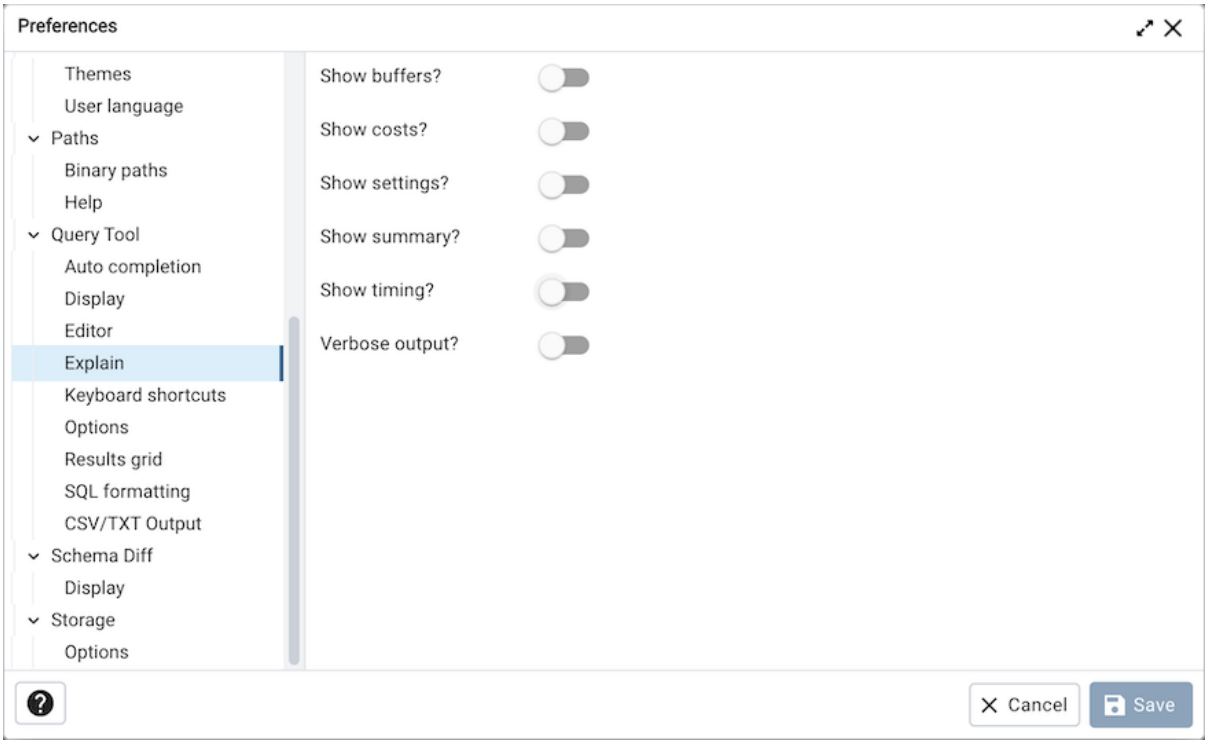
[*Display*] パネルのフィールドを使用して、クエリツールの表示設定を指定します。

- **接続ステータス** スイッチが *True* に設定されている場合、クエリツールの新しいインスタンスごとに接続とトランザクションのステータスが表示されます。
- **接続ステータスのリフレッシュレート** フィールドを使用して、接続/トランザクションステータスの更新間の秒数を指定します。
- **クエリ情報通知のタイムアウト** フィールドを使用して、クエリの実行が完了したときに表示される通知の動作を制御します。値が *-1* で通知が無効になり、値が 0 でクリックされるまで表示されます。ゼロより大きい正の値を指定すると、指定した秒数だけ通知が表示されます。デフォルトは *5* です。



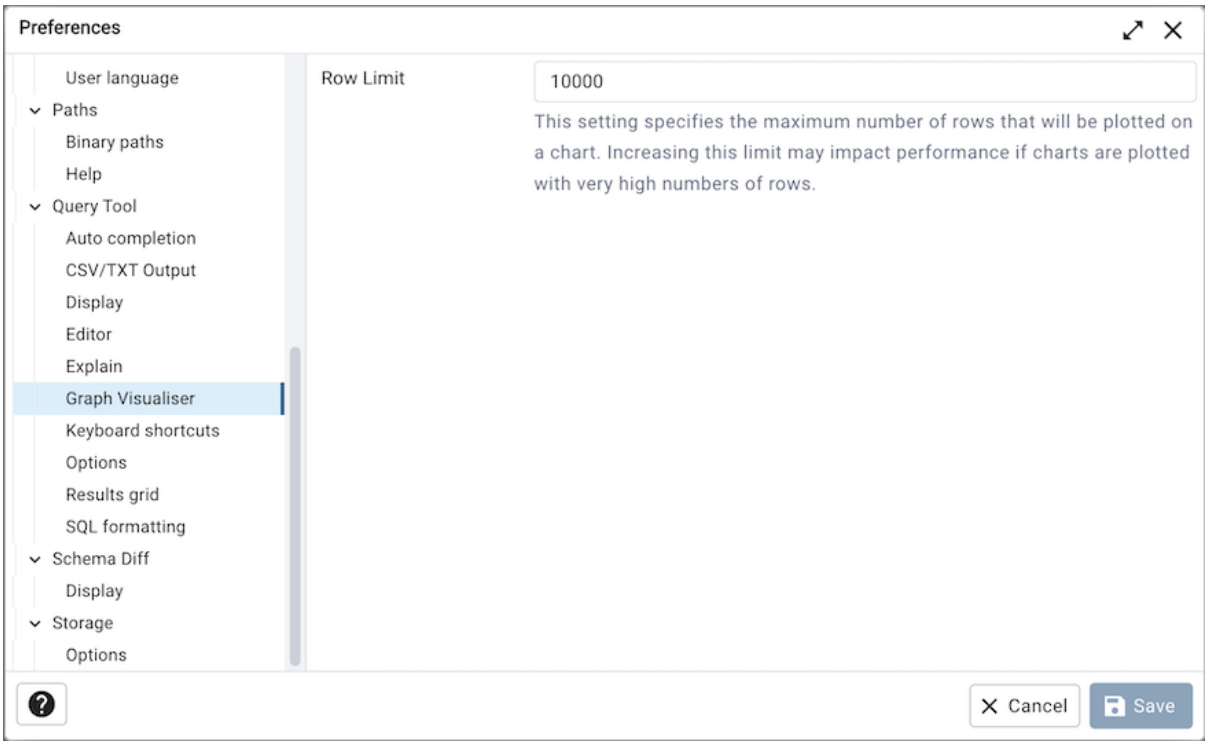
[エディター] パネルのフィールドを使用して、クエリエディターの設定を変更します。

- **中かっこが一致しますか**? スイッチが *True* に設定されている場合、エディターは一致した中かっこのペアを強調表示します。
- **コードの折りたたみ**? スイッチが *False* に設定されている場合、エディターはコードの折りたたみを無効にします。無効にすると、大きなファイルでのエディターのパフォーマンスが向上します。
- **フォントサイズ** フィールドを使用して、テキストボックスとエディターで使用されるフォントサイズを指定します。
- **ブラケットのペアを挿入しますか**? スイッチが *True* に設定されている場合、エディターは自動的にペアのブラケットを挿入します。
- **行の折り返し** スイッチが *True* に設定されている場合、エディターは行の折り返し動作を実装します。
- *Plaintextmode?* スイッチが *True* に設定されている場合、エディターモードは text/plain に変更されます。キーワードの強調表示とコードの折りたたみは無効になります。これにより、大きなファイルでのエディターのパフォーマンスが向上します。



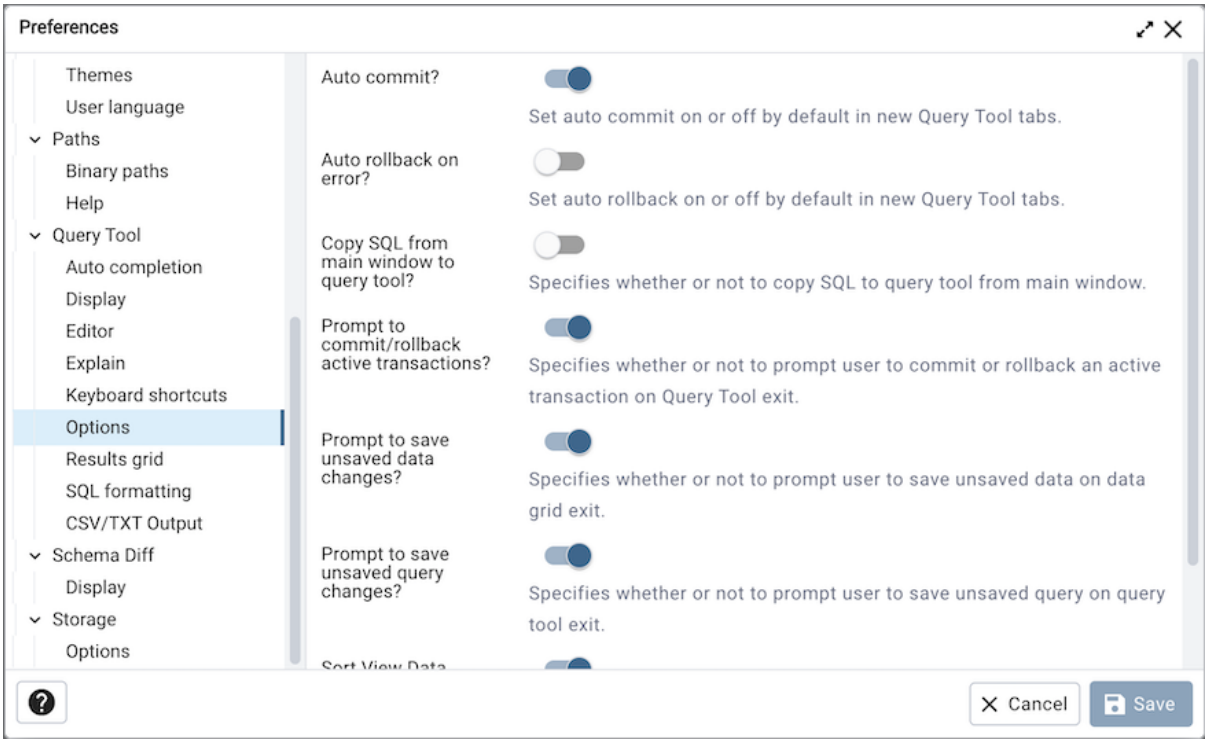
Explain パネルのフィールドを使用して、グラフィカル EXPLAIN に含まれる詳細レベルを指定します。

- *ShowBuffers?* スイッチが *True* に設定されている場合、グラフィカルな説明の詳細にバッファーの使用状況に関する情報が含まれます。
- **コストを表示しますか***?* スイッチが *True* に設定されている場合、グラフィカルな説明の詳細には、各プランの推定スタートアップと合計コスト、および推定行数と各行の推定幅が含まれます。
- *ShowTiming?* スイッチが *True* に設定されている場合、グラフィカルな説明の詳細には、出力の各ノードで費やされた時間が含まれます。
- *Verboseoutput?* スイッチが *True* に設定されている場合、グラフィカルな説明の詳細には、クエリ実行プランに関する拡張情報が含まれます。



GraphVisualiser パネルのフィールドを使用して、グラフに関連する設定を指定します。

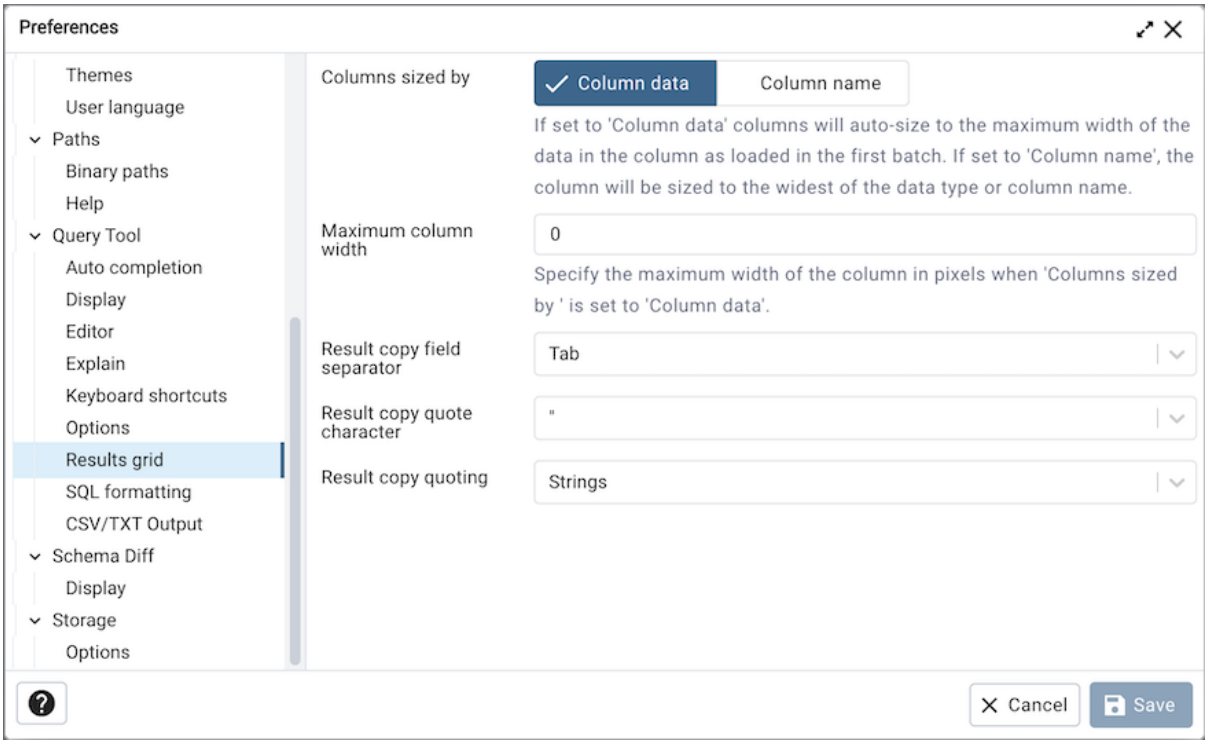
- **行制限** フィールドを使用して、チャートにプロットされる行の最大数を指定します。



オプション パネルのフィールドを使用して、エディターの設定を管理します。

- *Autocommit?* スイッチが *True* に設定されている場合、成功した各クエリは実行後にコミットされます。
- *Autorollbackonerror?* スイッチが *True* に設定されている場合、失敗したクエリはロールバックされます。

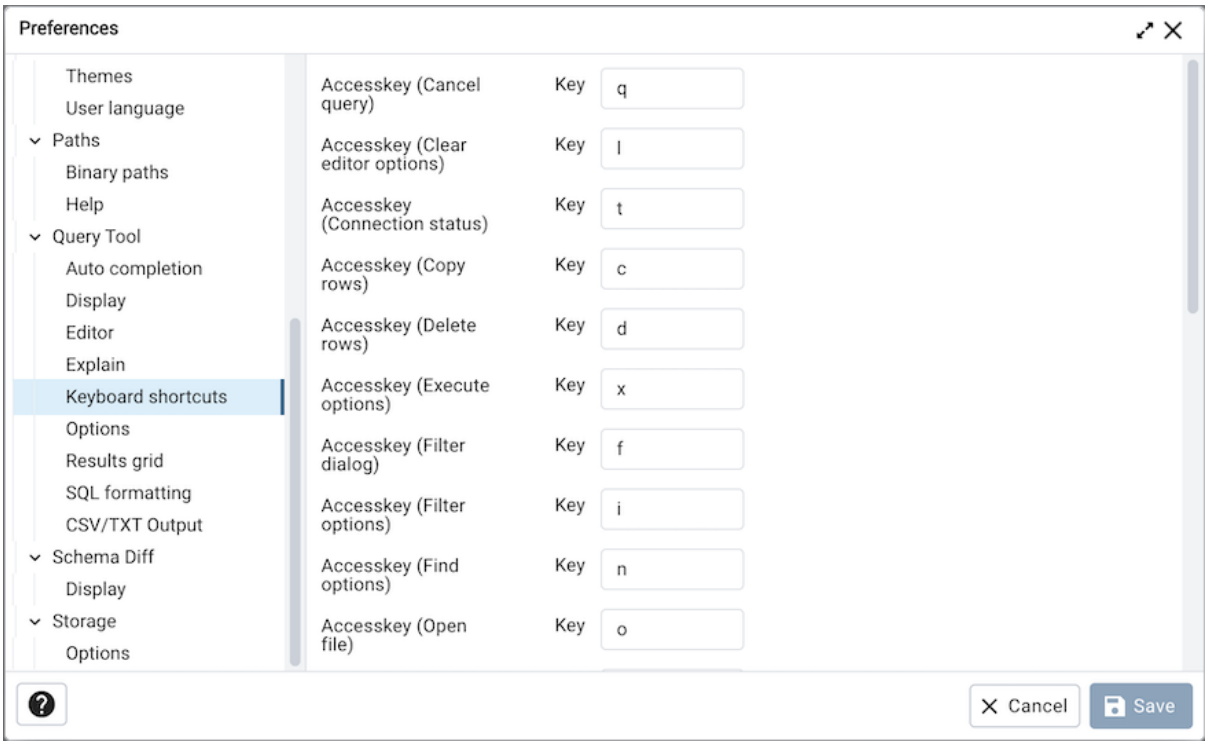
- **メインウィンドウからクエリツールに *SQL* をコピーしますか***?*** スイッチが *True* に設定されている場合、クエリツールが開いたときに選択したオブジェクトの CREATEsql がクエリツールにコピーされます。
- *Prompttosaveunsaveddatachanges?* スイッチが *True* に設定されている場合、エディターはデータエディターを終了するときにユーザーに未保存のデータを保存するように求めます。
- *Prompttosaveunsavedquerychanges?* スイッチが *True* に設定されている場合、エディターはクエリツールを終了するときにユーザーに未保存のクエリの変更を保存するように求めます。
- *Prompttocommit/rollbackactivetransactions?* スイッチが *True* に設定されている場合、エディターは、現在のトランザクションがコミットされていないときにクエリツールを終了すると、ユーザーに変更をコミットまたはロールバックするように求めます。
- **主キー列でデータの結果を並べ替えますか***?* *True* に設定されている場合、データの表示/編集-すべての行オプションを使用したときに返されるデータは、デフォルトで主キー列で並べ替えられます。最初/最後の 100 行オプションを使用すると、データは常に並べ替えられます。



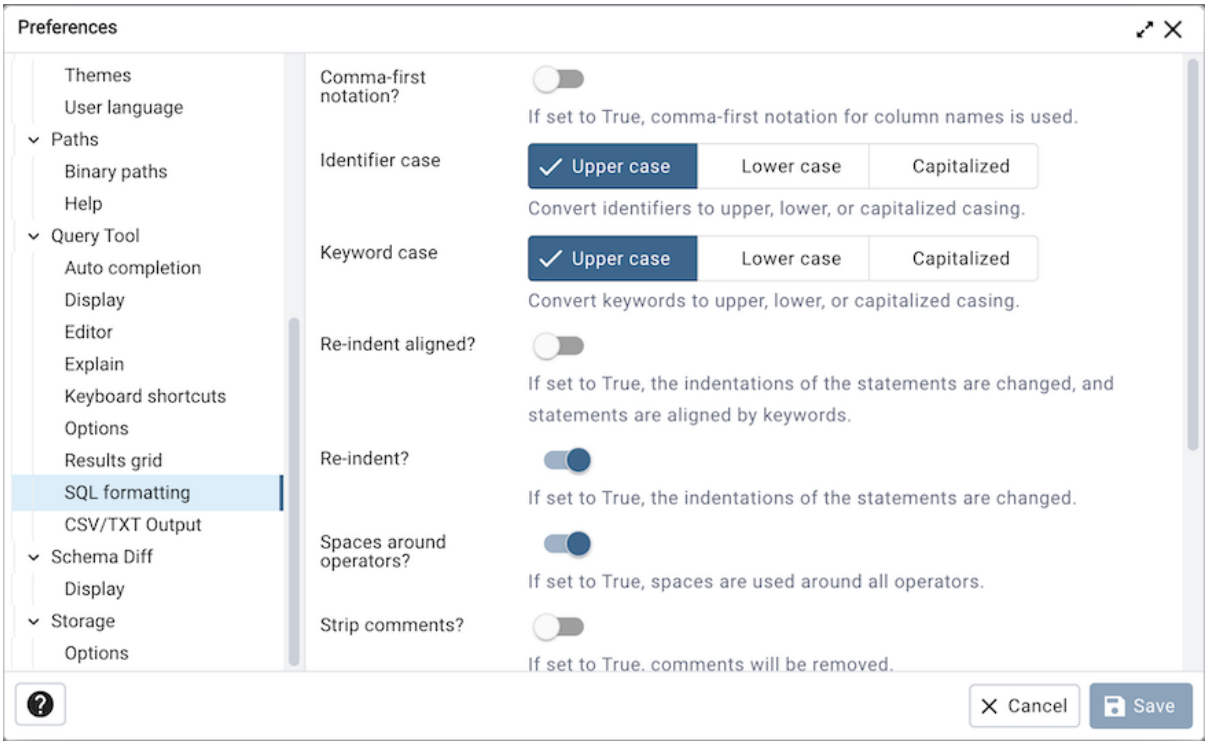
結果グリッド パネルのフィールドを使用して、コピーしたデータのフォーマット設定を指定します。

- **列のサイズ設定** が **列データ** に設定されている場合、データ列は最初のバッチでロードされた列のデータの最大幅に自動サイズ設定されます。 **列名** に設定すると、列のサイズはデータ型または列名の最大幅になります。
- [列のサイズ] が **列データ** に設定されている場合、列の最大幅をピクセル単位で指定します。[列のサイズ] が **列名** に設定されている場合、この設定は効果がありません。
- **結果のコピーフィールド区切り文字** ドロップダウンリストボックスを使用して、コピーされたデータのフィールド区切り文字を選択します。
- **結果コピー引用文字** ドロップダウンリストボックスを使用して、コピーされたデータの引用文字を選択します。

- **結果コピーの引用** ドロップダウンリストボックスを使用して、引用が必要なフィールドの種類を選択します。 **すべて** 、 **なし** 、または **文字列** を選択します。



[**キーボードショートカット**] パネルのフィールドを使用して、クエリツールウィンドウのナビゲーションのショートカットを構成します。



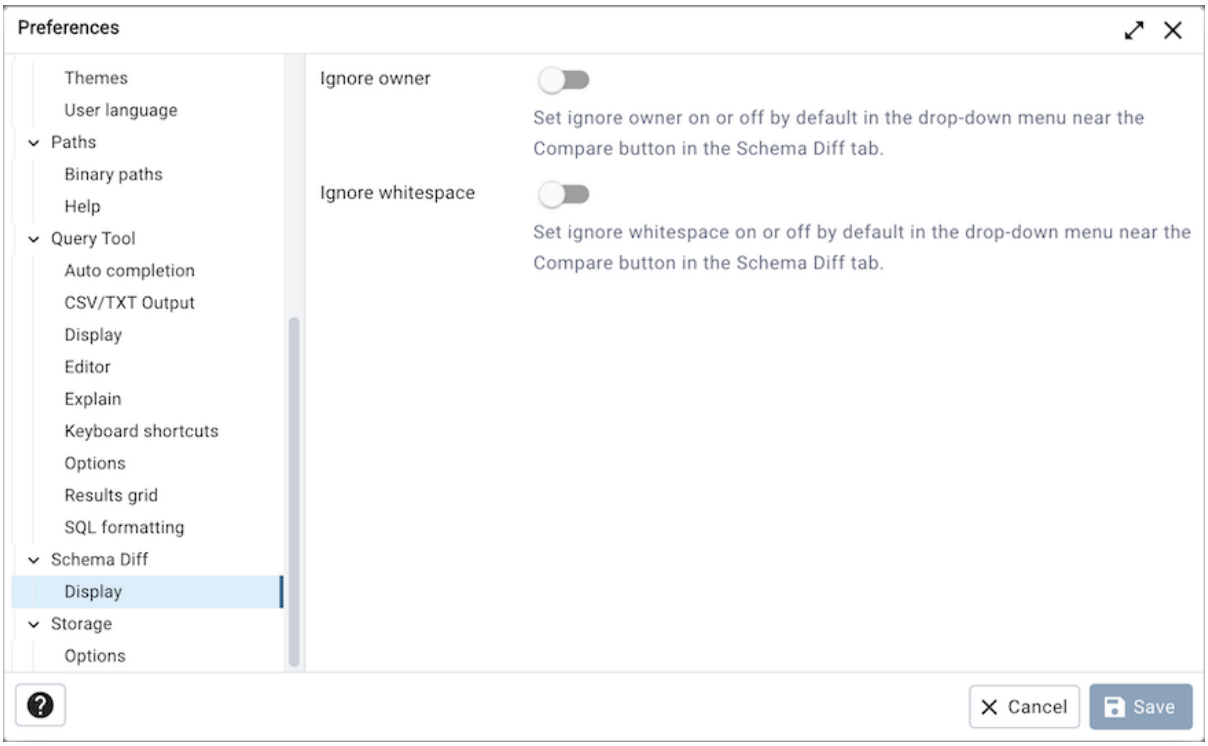
SQL **フォーマット** パネルのフィールドを使用して、SQL の再フォーマット設定を指定します。

- **コンマファースト表記** オプションを使用して、列名の前または後にコンマを配置するかを指定します。

- **識別子のケース** オプションを使用して、識別子（オブジェクト名）を大文字、小文字、または大文字に変更するかどうかを指定します。
- *Keywordcase* オプションを使用して、キーワードを大文字、小文字、または大文字に変更するかどうかを指定します。
- *Re-indentAligned?* オプションを使用して、ステートメントのインデントをキーワードで揃えて変更するように指定します。
- *Re-indent?* オプションを使用して、ステートメントのインデントを変更するように指定します。
- **演算子の周りにスペース***?* オプションを使用して、演算子のいずれかの側にスペースを含めるかどうかを指定します。
- *Stripcomment?* オプションを使用して、コメントを削除するかどうかを指定します。
- **タブサイズ** オプションを使用して、タブまたはインデントごとのスペースの数を指定します。
- **スペースを使用しますか***?* オプションを使用して、インデントするときにスペースを使用するかタブを使用するかを選択します。
- *N* **文字の後にラップ** オプションを使用して、列で区切られたリスト（テーブル内の列名など）の列制限を指定します。0（ゼロ）に設定した場合、各項目は自分の行に表示されます。

1.17.9 スキーマ差分ノード

SchemaDiff ノードを展開して、表示設定を指定します。

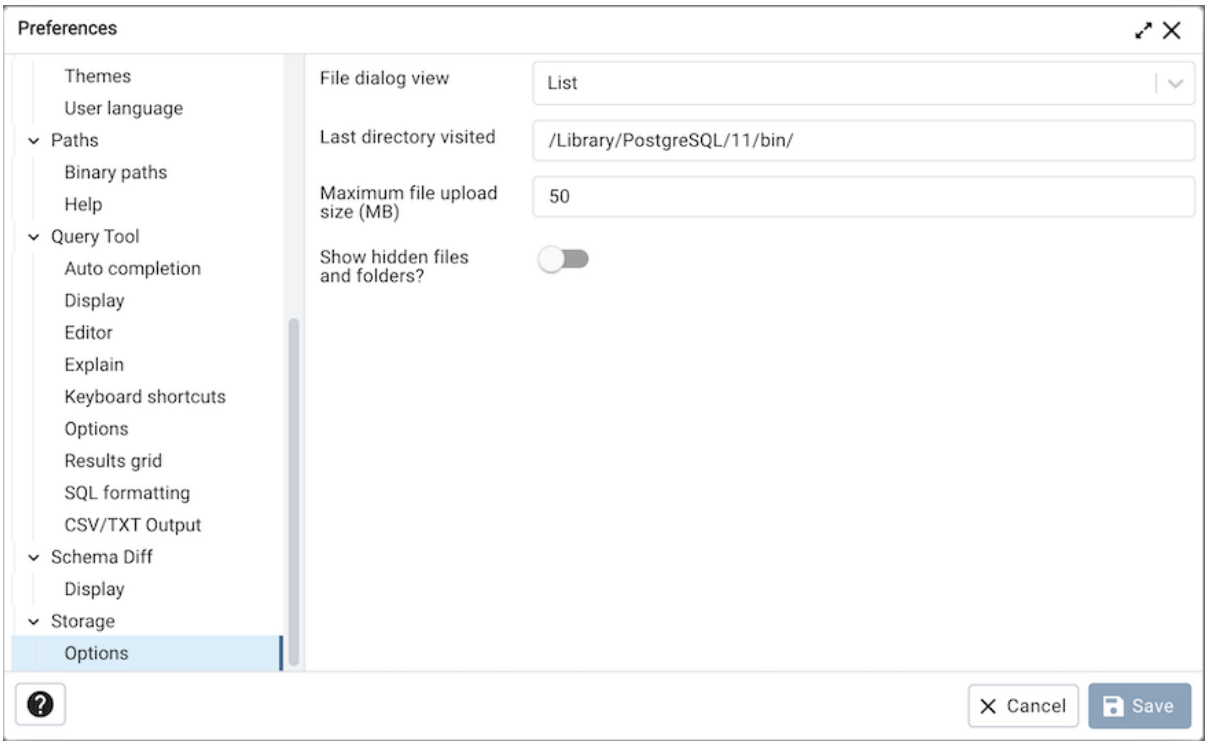


オブジェクトの比較中に所有者を無視するには、 **所有者を無視** スイッチを使用します。

Ignorewhitespace スイッチを使用して、文字列オブジェクトの比較中に空白を無視します。ホワイトスペースには、スペース、タブ、CRLF が含まれます。

1.17.10 ストレージノード

Storage ノードを展開して、ストレージ設定を指定します。



[**オプション**] パネルのフィールドを使用して、ストレージ設定を指定します。

- **ファイルダイアログビュー** ドロップダウンリストボックスを使用して、ファイルマネージャーを開いたときに表示されるアイコンのスタイルと表示形式を選択します。 **リスト** を選択してリストビューを表示するか、 **グリッド** を選択してフォルダーアイコンを表示します。
- **最後にアクセスしたディレクトリ** フィールドを使用して、ファイルマネージャーが開くフォルダーの名前を指定します。
- **Storage** ノードの **オプション** パネルの **最大ファイルアップロードサイズ** *(MB)* フィールドを使用して、アップロードの最大ファイルサイズを指定します。
- **隠しファイルとフォルダーを表示しますか**? スイッチが *True* に設定されている場合、ファイルマネージャーは隠しファイルとフォルダーを表示します。

1.18 Keyboard Shortcuts ：

pgAdmin にはキーボードショートカットが提供されているため、特定の機能に簡単にアクセスできます。代替のショートカットは、必要に応じて [ファイル]>[設定] で構成できます。

1.18.1 メインブラウザウィンドウ

メインブラウザーウィンドウを使用している場合、次のキーボードショートカットを使用できます。

すべてのプラットフォームのショートカット	ファンクション
Shift+Ctrl+a	グリッド行を追加
Shift+Alt+b	ブラウザツリー
Shift+Alt+n	オブジェクトを作成
Shift+Alt+d	オブジェクトを削除
Shift+Alt+m	複数のオブジェクトを削除/ドロップ
Shift+Ctrl+[	ダイアログタブを後ろに
Shift+Ctrl+]	ダイアログタブを進む
Shift+Alt+g	ダイレクトデバッグ
Shift+Alt+u	複数のオブジェクトをカスケードする
Shift+Alt+e	オブジェクトのプロパティを編集する
Shift+Alt+f	ファイルメインメニュー
Shift+Alt+h	ヘルプメインメニュー
Shift+Alt+o	オブジェクトメインメニュー
Shift+Alt+c	コンテキストメニューを開く
Shift+Alt+q	クエリツールを開く
Shift+Ctrl+f	クイック検索
F5	ブラウザツリーを更新
Shift+Alt+s	オブジェクトを検索
Shift+Alt+[	タブ付きパネルの後方
Shift+Alt+]	タブ付きパネルを前方に
Shift+Alt+l	ツールメインメニュー
Shift+Alt+v	データを見る

1.18.2 ダイアログタブ

以下のショートカットを使用して、ダイアログのタブセットを移動します。

すべてのプラットフォームのショートカット	ファンクション
Control+Shift+[	ダイアログタブを後ろに

次のページに続く

表 11 – 前のページからの続き

すべてのプラットフォームのショートカット	ファンクション
Control+Shift+]	ダイアログタブを進む

1.18.3 プロパティグリッドコントロール

プロパティグリッドコントロールを使用する場合は、以下のショートカットを使用します。

すべてのプラットフォームのショートカット	ファンクション
Control+Shift+A	グリッドに行を追加
タブ	次のコントロールにフォーカスを移動する
Shift+Tab	前のコントロールにフォーカスを移動する
戻る	コンボボックスで選択された項目を選ぶ
Control+Shift+A	グリッドに行を追加

1.18.4 SQL エディター

構文強調表示 SQL エディターを使用する場合、次のショートカットを使用できます。

ショートカット（Windows/Linux）	ショートカット（Mac）	ファンクション
Alt+左	Option+左	行の先頭に移動
Alt+右	Option+右	行の最後に移動
Ctrl+Alt+左	Cmd+Option+左	1 単語左に移動
Ctrl+Alt+右	Cmd+Option+右	1 単語右に移動
Ctrl+/	Cmd+/	選択したコードをコメントする（インライン）
Ctrl+.	Cmd+。	選択したコードのコメントを外します（インライン）
Ctrl+Shift+/	Cmd+Shift+/	コードのコメント/アンコメント（ブロック）
Ctrl+A	Cmd+	すべて選択
Ctrl+c	Cmd+c	選択したテキストをクリップボードにコピー
Ctrl+r	Cmd+r	最後の編集をやり直す
Ctrl+v	Cmd+v	クリップボードからテキストを貼り付ける
Ctrl+z	Cmd+z	最後の編集を元に戻す
タブ	タブ	選択したテキストをインデント
Shift+Tab	Shift+Tab	選択したテキストのインデントを解除
Alt+g	Option+g	ジャンプ（行：列へ）

次のページに続く

表 13 – 前のページからの続き

ショートカット（Windows/Linux）	ショートカット（Mac）	ファンクション
Ctrl+スペース	Ctrl+スペース	オートコンプリート
Ctrl+f	Cmd+f	見つける
Ctrl+g	Cmd+g	次を探す
Ctrl+Shift+g	Cmd+Shift+g	前を検索
Ctrl+Shift+f	Cmd+Shift+f	置き換える

1.18.5 クエリツール

クエリツールを使用する場合、次のショートカットを使用できます。

ショートカット（Windows/Linux）	ショートカット（Mac）	ファンクション
<アクセスキー>+q	<アクセスキー>+q	クエリをキャンセル
<アクセスキー>+t	<アクセスキー>+t	接続状態
<アクセスキー>+d	<アクセスキー>+d	行を削除する
<アクセスキー>+×	<アクセスキー>+×	実行オプション
<アクセスキー>+f	<アクセスキー>+f	フィルタダイアログ
<アクセスキー>+i	<アクセスキー>+i	フィルタオプション
<アクセスキー>+n	<アクセスキー>+n	オプションを探す
<アクセスキー>+o	<アクセスキー>+o	ファイルを開く
<アクセスキー>+p	<アクセスキー>+p	行を貼り付ける
<アクセスキー>+r	<アクセスキー>+r	行制限
<アクセスキー>+s	<アクセスキー>+s	ファイルを保存
Ctrl+Alt+L	Ctrl+option+L	クエリをクリア
Shift+Ctrl+m	Shift+Ctrl+m	コミットする
F8	F8	結果をダウンロード
Shift+F7	Shift+F7	EXPLAINANALYZE クエリ
F7	F7	EXPLAIN クエリ
F5	F5	クエリを実行
Shift+Alt+]	Shift+option+]	次のタブ
Shift+Alt+[	Shift+option+[	前のタブ
Shift+Ctrl+r	Shift+Ctrl+r	ロールバック
F6	F6	データの変更を保存する
Shift+Alt+Tab	Shift+オプション+Tab	スイッチパネル
Shift+Ctrl+u	Shift+Ctrl+u	選択したテキストのトグル

1.18.6 デバッガー

デバッガーを使用する場合、次のショートカットを使用できます。

ショートカット（Windows/Linux）	ショートカット（Mac）	ファンクション
<アクセスキー>+×	<アクセスキー>+×	すべてのブレークポイントをクリア
<アクセスキー>+c	<アクセスキー>+c	続行/開始
<アクセスキー>+i	<アクセスキー>+i	踏み込む
<アクセスキー>+o	<アクセスキー>+o	ステップオーバー
<アクセスキー>+s	<アクセスキー>+s	停止
<アクセスキー>+t	<アクセスキー>+t	ブレークポイントをトグル
Shift+Alt+q	Shift+option+q	グリッド値を編集する
Shift+Alt+]	Shift+option+]	次のタブ
Shift+Alt+[	Shift+option+]	前のタブ
Shift+Alt+Tab	Shift+オプション+Tab	スイッチパネル

1.18.7 ERD ツール

ERD ツールを使用する場合、次のショートカットを使用できます。

ショートカット（Windows/Linux）	ショートカット（Mac）	ファンクション
Ctrl+Alt+a	Ctrl+option+a	テーブルを追加
Ctrl+Alt+n	Ctrl+option+n	ノートを追加/編集
Ctrl+Alt+l	Ctrl+option+l	自動整列
Ctrl+Alt+c	Ctrl+option+c	クローンテーブル
Ctrl+Alt+i	Ctrl+option+i	画像をダウンロード
Ctrl+Alt+d	Ctrl+option+d	ドロップテーブル
Ctrl+Alt+e	Ctrl+option+e	テーブルを編集
Ctrl+Alt+s	Ctrl+option+s	SQL を生成
Ctrl+Alt+m	Ctrl+option+m	多対多リンク
Ctrl+Alt+o	Ctrl+option+o	1 対多のリンク
Ctrl+o	Ctrl+o	プロジェクトを開く
Ctrl+s	Ctrl+s	プロジェクトを保存
Shift+Ctrl+s	Shift+Ctrl+s	プロジェクトを名前を付けて保存
Ctrl+Alt+t	Ctrl+option+t	詳細を表示する/少なくする
Shift+Alt++	Shift+オプション++	ズームイン
Shift+Alt+-	Shift+option+-	ズームアウト
Shift+Alt+f	Shift+option+f	ズーム

1.18.8 内部のタブとパネルのナビゲーション

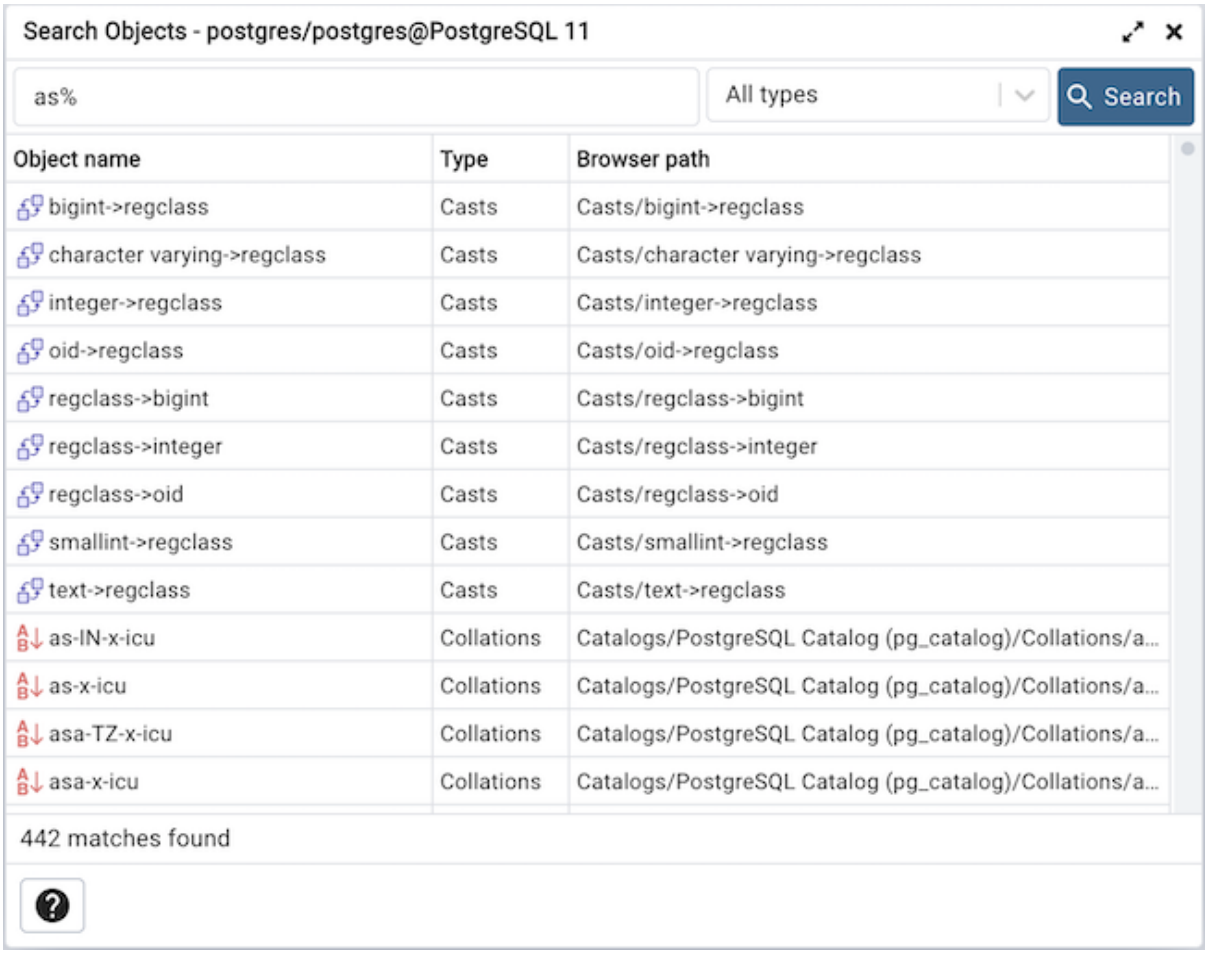
クエリツールとデバッガーを使用している場合、内部パネルのナビゲーションに次のショートカットを使用できます。

1.18.9 アクセスキー

<accesskey>はブラウザーとプラットフォームに依存します。次の表に、サポートされているブラウザーのデフォルトのアクセスキーを示します。

	Windows	Linux	マック
エッジ	代替		
Chrome	代替	代替	Ctrl+Option
Firefox	Alt+Shift	Alt+Shift	Ctrl+Option
サファリ	代替		Ctrl+Option

1.19 Search objects



このダイアログでは、データベース内のほぼすべての種類のオブジェクトを検索できます。

データベースまたはその子ノードを右クリックし、[オブジェクトの検索] を選択するとアクセスできます。ショートカット（デフォルトは ALT+SHIFT+S）を押してアクセスすることもできます。

パターンの最小長は 3 文字です。実行される検索は大文字と小文字を区別せず、名前にパターンが含まれるすべてのオブジェクトを見つけます。現在、オブジェクト名のみを検索できます。例：abc、%ab%、ab%c、%%%など

結果は、オブジェクト名、オブジェクトタイプ、および *browser* のオブジェクトツリーパスとともにグリッドに表示されます。結果の行をダブルクリックして、*browser* のオブジェクトを選択できます。オブジェクトがグレー表示されている場合、これは *browser* でこれらのオブジェクトタイプを有効にしていないことを意味するため、ダブルクリックできません。関数とプロシージャの名前に追加された省略記号をクリックして、引数を表示できます。

オブジェクトタイプのドロップダウンからオブジェクトタイプを選択して、特定のオブジェクトタイプに基づいてフィルタリングできます。オブジェクトタイプのいずれかが選択されているときに検索ボタンが押されると、それらのタイプのみがデータベースからフェッチされます。データベースサーバがオブジェクトタイプをサポートしていない場合、または *preferences* から有効になっていない場合、オブジェクトタイプはドロップダウンに表示されません。

pgAdmin を使用してサーバーにあるオブジェクトを管理する前に、サーバーへの接続を定義する必要があ

ります。詳細については、次のセクションの **サーバーへの接続** を参照してください。

第2章　External database for pgAdmin user settings

pgAdmin が使用するユーザー設定は SQLite データベースに保存されます。このデータベースには、設定、ユーザーアカウント、自動検出サーバーなど、多くの設定が保存されます。

SQLite はファイルベースのデータベースであり、ファイルシステムのどこにでも配置できるため、障害を処理するように設計されていません（HA サポートなし）。さらに、複数の接続を同時に読み取り/書き込みするように設計されていません。例：Kubernetes などの環境では、代替バックエンドを使用して、非エフェメラルストレージでの SQLite の使用を回避し、設定データベースの HA を許可すると役立つ場合があります。

これを防ぐために、pgAdmin は、*config.py* ファイルの新しい「CONFIG_DATABASE_URI」パラメータを使用して、外部データベースにユーザ設定を保存することをサポートするようになりました。

2.1 SQLite データベースを使用する

SQLite データベースを使用するには、CONFIG_DATABASE_URI パラメーターが'' のような空の文字列に設定されていることを確認してください。デフォルトでは config.py の空の文字列に設定されているため、SQLite データベースを使用する場合は何も変更する必要はありません。

2.2 外部データベースを使用

外部データベースを使用するには、CONFIG_DATABASE_URI パラメーターが'/"dialect+driver://username:password@host:port/database/"'のように設定されていることを確認してください。

Note データベースは事前に作成しておくことをおすすめします。

2.3 PostgreSQL データベースを使用する

以下は、PostgreSQL を外部データベースとして使用するためのフォーマットです。

基本的な構文：

```
postgresql://username:password@host:port/database
```

特定のスキーマを使用する（事前にスキーマを作成することをお勧めします）：

```
postgresql://username:password@host:port/database?options=-csearch_path=<schema name>
```

サービスアカウントのデフォルトの pgpass パスを使用します。

```
postgresql://username@host:port?options=-csearch_path=<schema name>
```

pgpass ファイルパスの指定：

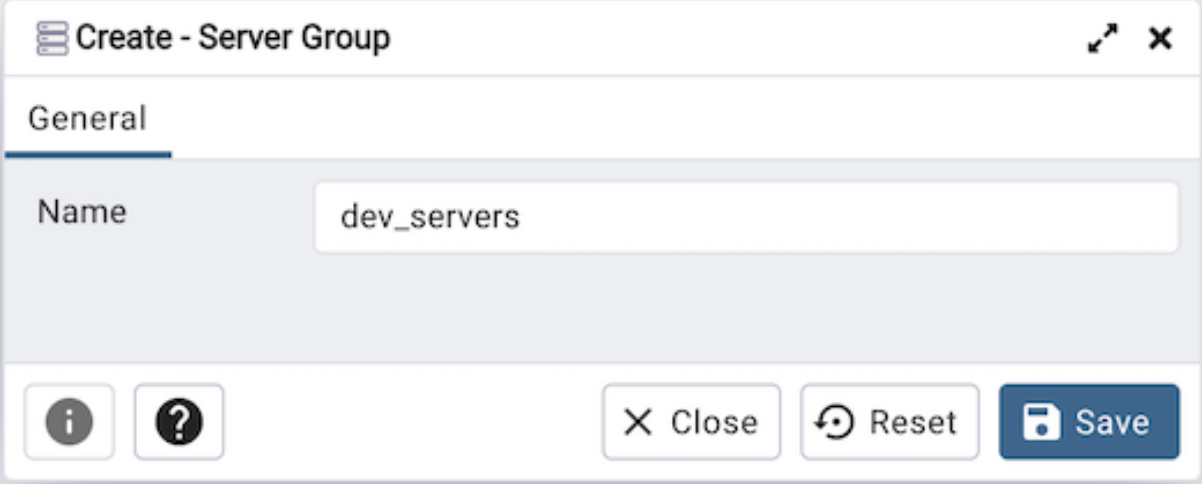
```
postgresql://username@host:port?passfile=<path of the pgpass file>&options=-csearch_
↪path=<schema name>
```

第3章　Connecting To A Server

pgAdmin クライアントを使用して Postgres サーバーにあるオブジェクトを管理する前に、サーバーへの接続を定義する必要があります。（オプションで）**サーバーグループ** ダイアログを使用してサーバーグループを作成し、管理を簡単にするためにツリーコントロール内のサーバー接続を整理できます。 *ServerGroup* ダイアログを開くには、ツリーコントロールの *Servers* ノードを右クリックし、*Create* メニューから *ServerGroup* を選択します。

3.1 Server Group Dialog

ServerGroup ダイアログを使用して、新しいサーバーグループを追加します。サーバーをサーバーグループに割り当てて、複数のサーバーの管理を簡素化します。サーバーグループは *pgAdmin* ツリーコントロールの一部として表示されます。



ServerGroup ダイアログの *Name* フィールドを使用して、 *pgAdmin* ツリーコントロールでサーバーグループを識別する名前を指定します。

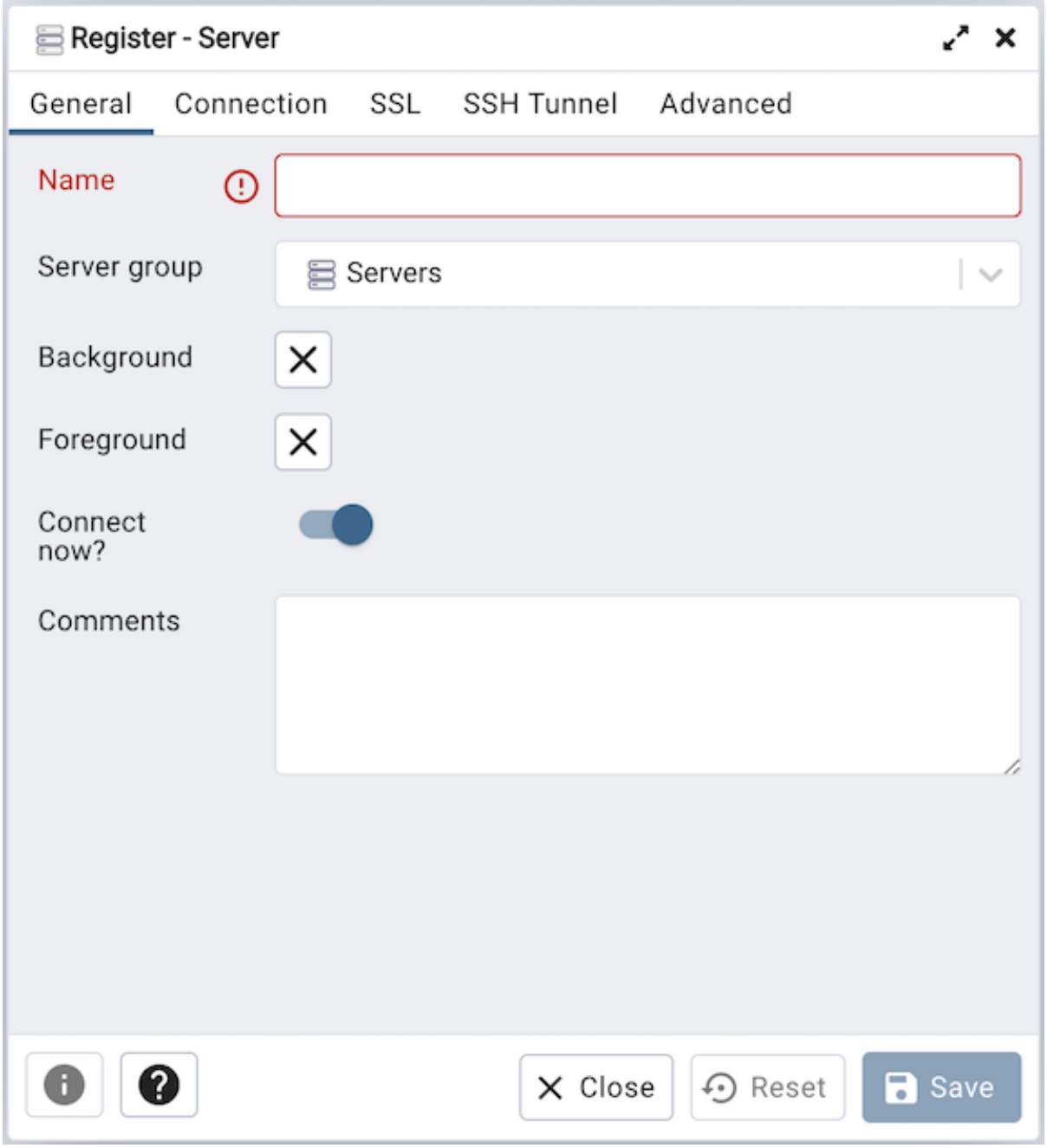
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

サーバーグループにサーバー接続を作成するには、名前付けたサーバーグループを右クリックし、 *Create* オプションを選択して *Create-Server* ダイアログを開きます。

Server ダイアログのフィールドを使用して、pgAdmin で管理する新しいサーバーごとに接続プロパティを定義します。 *Server* ダイアログを開くには、ツリーコントロールの *Servers* ノードを右クリックし、 *Register* メニューから *Server* を選択します。

3.2 Server Dialog

Server ダイアログを使用して、サーバーへの接続を説明します。注：接続しているサーバーの pg_hba.conf ファイルでクライアントのホストからの接続が許可されていることを確認する必要があります。



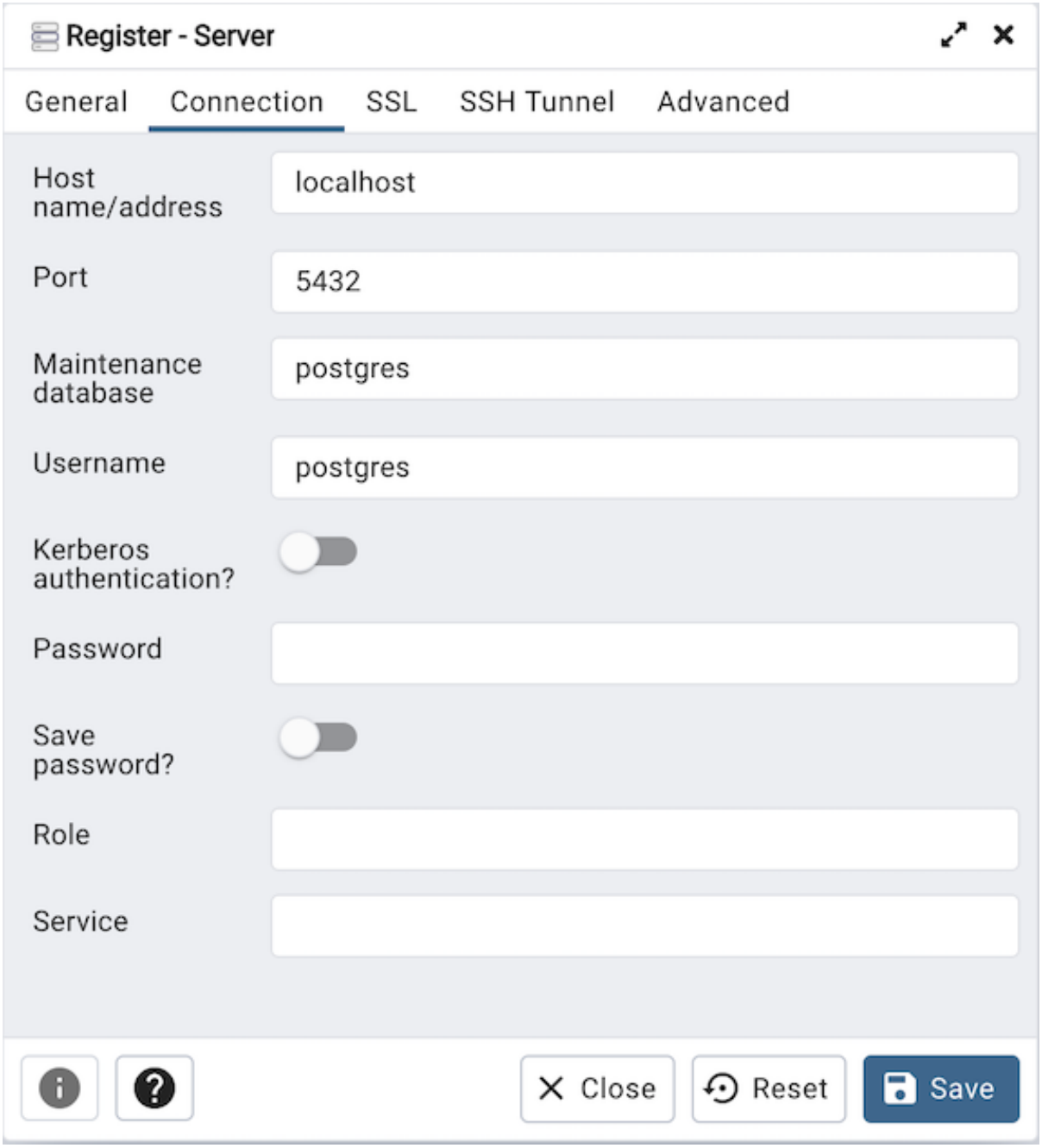
[**全般**] タブのフィールドを使用して、サーバーを識別します。

- *Name* フィールドを使用して、サーバーの説明的な名前を追加します。指定された名前は **ブラウザー** ツリーコントロールに表示されます。
- **サーバーグループ** フィールドのドロップダウンリストボックスを使用して、サーバーの親ノードを選択します。サーバーは指定されたグループ内の **ブラウザー** ツリーコントロールに表示されます。
- *Background* フィールドの color-picker を使用して、サーバーの背景色を指定します。

- *Foreground* フィールドの color-picker を使用して、サーバーの前景色を指定します。
- [今すぐ接続しますか?] チェックボックスがオンになっている場合、クライアントはダイアログの完了時にサーバーへの接続を試行します。これがデフォルトです
- *Shared?* スイッチを *Yes* にすると、そのサーバーを他のすべてのユーザーと共有できます。このオプションは、管理ユーザのみが使用できます。ユーザーの詳細については、 *User Management Dialog* を参照してください。ユーザーはいくつかの制限付きで共有サーバーにアクセスできます。共有サーバーでの次の操作は許可されません。
  - サーバーを削除する
  - サーバーの名前を変更する
  - グループサーバーの名前を変更する
  - ホスト、ポート、およびメンテナンスデータベースの変更

  サーバーが共有されると、ブラウザツリーのアイコンが変更されることに注意してください。
- サーバーに関するコメントを *Comments* フィールドに入力します。

[**接続**] タブをクリックして続行します。



[**接続**] タブのフィールドを使用して、接続を構成します。

- サーバーホストの IP アドレス、または完全修飾ドメイン名を **ホスト名/アドレス** フィールドに指定します。UNIX ドメインソケットを提供する場合、ディレクトリ名は/"//"で始まる必要があります。
- *Port* フィールドにサーバーホストのリスナーポート番号を入力します。デフォルトは *5432* です。
- **メンテナンスデータベース** フィールドを使用して、クライアントが接続する初期データベースの名前を指定します。pgAgent または adminpack オブジェクトを使用する場合は、そのデータベースに pgAgent スキーマと adminpack オブジェクトをインストールする必要があります。
- *Username* フィールドを使用して、サーバーでの認証時に使用されるロールの名前を指定します。
- *Kerberosauthentication?* が *True* に設定されている場合、pgAdmin は Kerberos 認証を使用して PostgreSQL サーバーに接続しようとします。

- *Password* フィールドを使用して、サーバーで認証するときに指定されるパスワードを入力します。
- *Savepassword?* の横にあるチェックボックスをオンにして、将来の使用のためにパスワードを保存するように pgAdmin に指示します。 *Clear Saved Password* を使用して、保存したパスワードを削除します。
- *Role* フィールドを使用して、サーバーでの認証後にクライアントに伝達される特権を持つロールの名前を指定します。この選択により、1 つのロールとして接続し、接続が確立された後にこの指定されたロールの権限を引き受けることができます。接続するロールは、指定されたロールのメンバーである必要があることに注意してください。
- *Service* フィールドを使用して、サービス名を指定します。詳細については、Section 33.16 of the Postgres documentation を参照してください。

[*SSL*] タブをクリックして続行します。



[*SSL*] タブのフィールドを使用して、SSL を構成します。

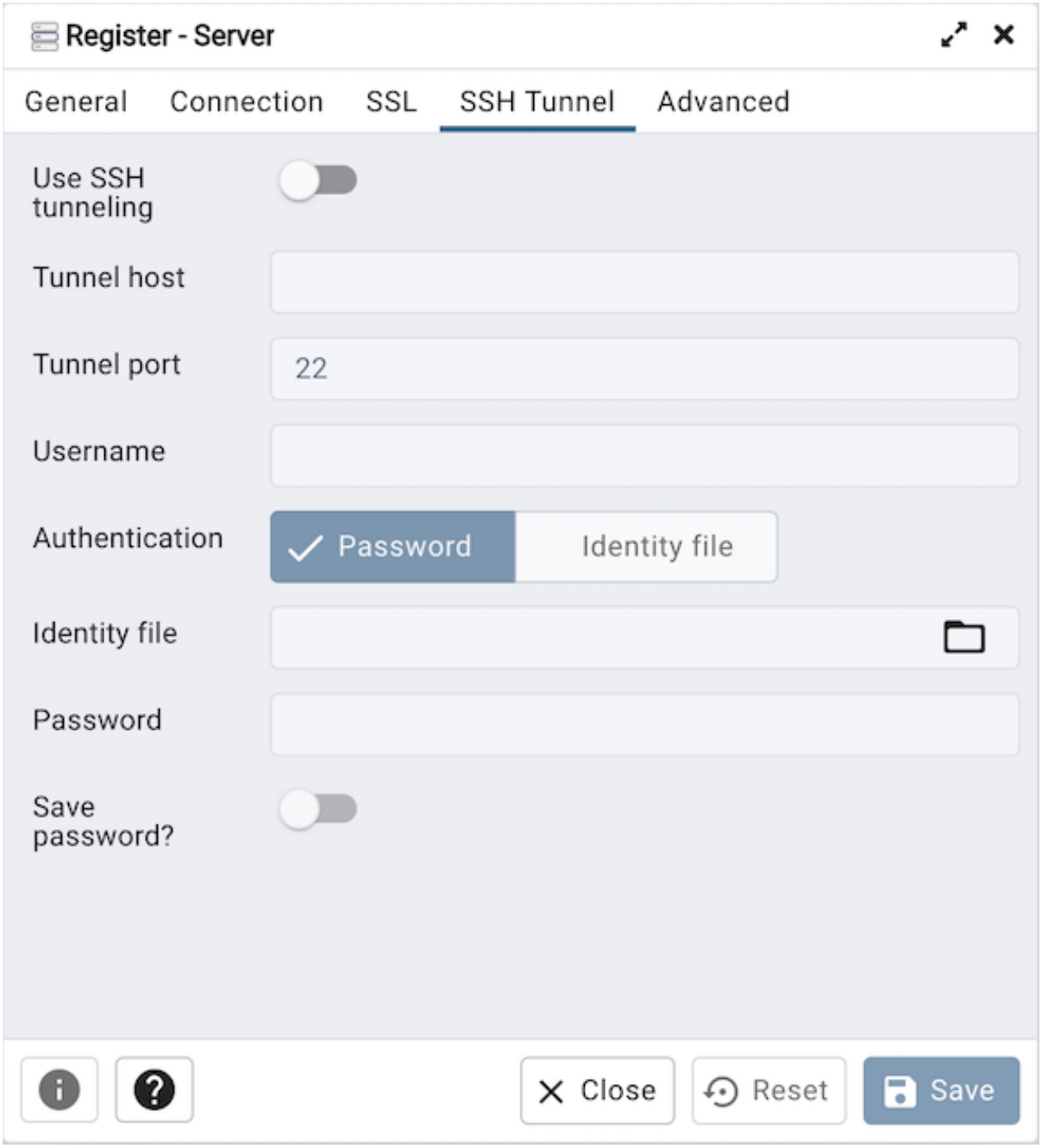
- *SSL* フィールドのドロップダウンリストボックスを使用して、サーバーが使用する SSL 接続のタイプを選択します。SSL 暗号化の使用の詳細については、 Section 33.18 of the Postgres documentation を参照してください。

pgAdmin がサーバーモード（デフォルトモード）でインストールされている場合、プラットフォーム固有のファイルマネージャーダイアログを使用して、SSL 暗号化をサポートするファイルをサーバーにアップロードできます。[ファイルマネージャー] ダイアログにアクセスするには、次の各フィールドの右側にあるアイコンをクリックします。

- **クライアント証明書** フィールドを使用して、クライアント SSL 証明書を含むファイルを指定します。このファイルは、pgAdmin がデスクトップモードでインストールされている場合はデフォルトの *~/.postgresql/postgresql.crt* 、pgAdmin が Web モードでインストールされている場合は *<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.crt* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **クライアント証明書キー** フィールドを使用して、クライアント証明書に使用される秘密鍵を含むファイルを指定します。このファイルは、pgAdmin がデスクトップモードでインストールされている場合はデフォルトの *~/.postgresql/postgresql.key* 、pgAdmin が Web モードでインストールされている場合は *<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.key* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **ルート証明書** フィールドを使用して、SSL 認証局を含むファイルを指定します。このファイルは、デフォルトの *~/.postgresql/root.crt* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **証明書失効リスト** フィールドを使用して、SSL 証明書失効リストを含むファイルを指定します。このリストは、 *~/.postgresql/root.crl* にあるデフォルトのリストを置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- *SSL* **圧縮***?* が *True* に設定されている場合、SSL 接続で送信されるデータは圧縮されます。デフォルト値は *False* です（圧縮は無効です）。SSL 接続が確立されていない場合、このパラメーターは無視されます。

警告: サーバーモードでは、証明書、プライベートキー、および失効リストは、pgAdmin サーバープロセスが実行されるユーザーアカウントが所有するサーバー上のユーザーごとのファイル格納領域に格納されます。これは、サーバーの管理者がこれらのファイルにアクセスできることを意味します。この機能の使用を選択する前に、適切な注意を払う必要があります。

[*SSH* **トンネル**] タブをクリックして続行します。



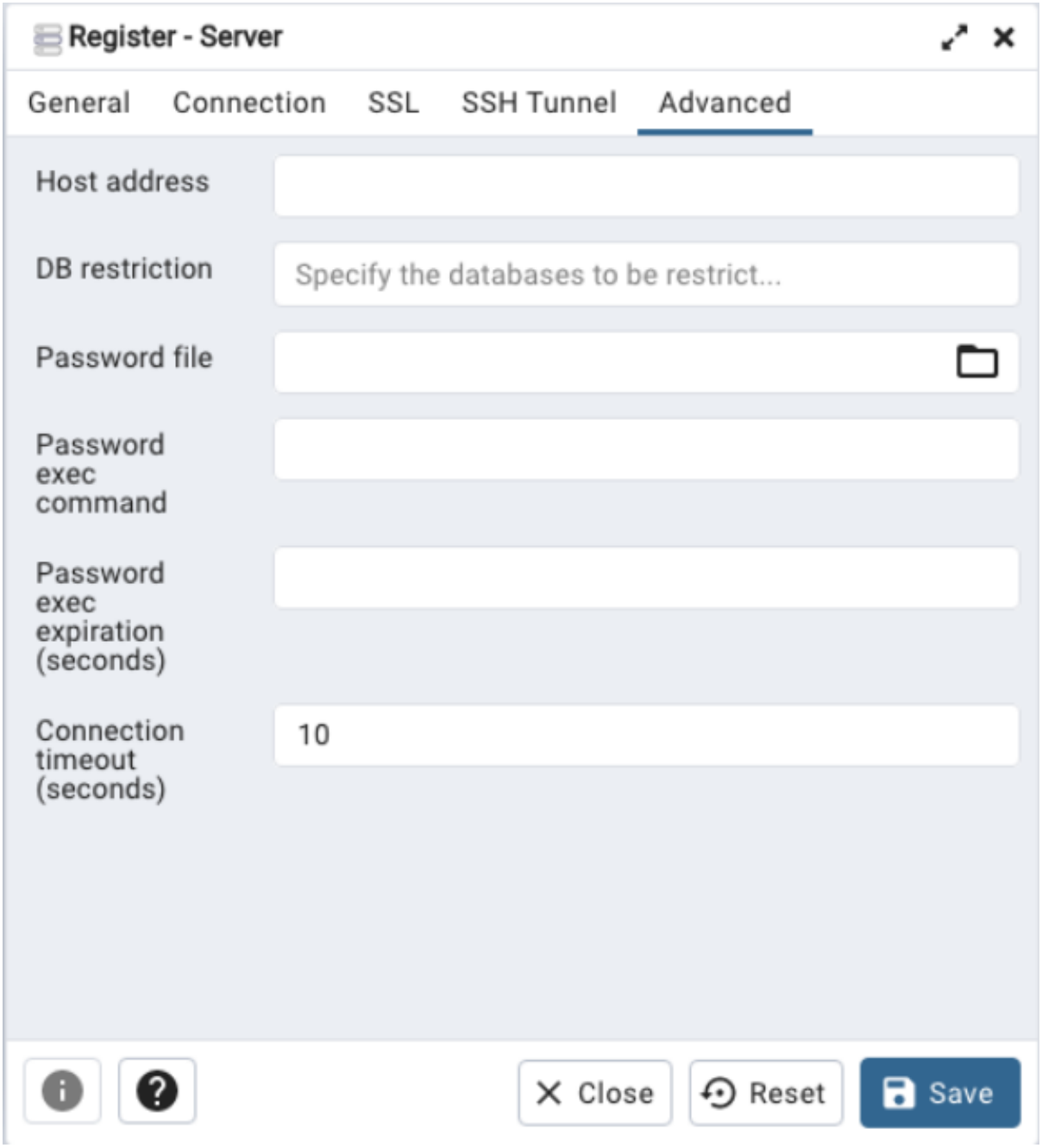
[*SSHTunnel*] タブのフィールドを使用して、SSH トンネリングを構成します。

/"SSH トンネル/"タブを使用して、クライアントが直接接続できないネットワーク上にあるサーバーに pgAdmin を（中間プロキシホストを介して）接続できます。

- /"SSH トンネリングを使用する/"を *Yes* に設定して、指定されたサーバーへの接続時に pgAdmin が SSH トンネルを使用するように指定します。
- **トンネルホスト** フィールドに SSH ホスト（クライアント接続が転送される）の名前または IP アドレスを指定します。
- **トンネルポート** フィールドに SSH ホスト（クライアント接続が転送される）のポートを指定します。
- *Username* フィールドに SSH ホストのログイン権限を持つユーザーの名前を指定します。

- *Authentication* フィールドで、SSH ホストに接続するときに使用される認証の種類を指定します。
  - *Password* オプションを選択して、pgAdmin が SSH ホストへの認証にパスワードを使用することを指定します。これがデフォルトです。
  - *ID* **ファイル** を選択して、接続時に pgAdmin が秘密キーファイルを使用することを指定します。
- SSH ホストが認証用のプライベートキーファイルを想定している場合は、 *Identityfile* フィールドを使用してキーファイルの場所を指定します。
- SSH ホストがユーザー名のパスワード、または使用されている場合は ID ファイルを想定している場合は、 *Password* フィールドを使用してパスワードを指定します。
- *Savepassword?* の横にあるチェックボックスをオンにして、将来の使用のためにパスワードを保存するように pgAdmin に指示します。 *Clear SSH Tunnel Password* を使用して、保存したパスワードを削除します。

[**詳細**] タブをクリックして続行します。



[**詳細**] タブのフィールドを使用して、接続を構成します。

- **ホストアドレス** フィールドにサーバーホストの IP アドレスを指定します。このフィールドを使用してホスト IP アドレスを指定すると、接続時の DNS ルックアップを回避して時間を節約できますが、Kerberos、GSSAPI、または SSPI 認証方法を使用する場合、および完全な SSL 証明書の検証。
- *DBRestriction* フィールドを使用して、表示するデータベースを制限するために pg_database テーブルに対して使用される SQL 制限を提供します。たとえば、次のように入力します。 *live_dbtest_db* live_db と test_db のみが pgAdmin ブラウザーに表示されます。入力するときは、コンマまたはタブでエントリを区切ります。
- *PasswordFile* フィールドを使用して、パスワードファイル（.pgpass）の場所を指定します。.pgpass ファイルを使用すると、ユーザーは接続時にパスワードを入力せずにログインできます。詳細については、 Section 33.15 of the Postgres documentation を参照してください。

- *Passwordexeccommand* フィールドを使用して、SQL 認証に使用されるパスワードを取得するために実行するシェルコマンドを指定します。コマンドの `stdout` は SQL パスワードとして使用されます。これは、 `stdout` シナリオで接続するときにパスワードを提供する代わりに、パスワードを一時的な承認トークンとして生成する必要がある場合に役立ちます。
- *Passwordexec* **有効期限** フィールドを使用して、 *Passwordexec* **コマンド** で生成されたパスワードの最大有効期間を秒単位で指定します。指定しない場合、パスワードは pgAdmin セッションが期限切れになるまで期限切れになりません。ゼロは、新しい接続または再接続が行われるたびにコマンドが実行されることを意味します。生成されたパスワードが無期限に有効でない場合は、この値を期限切れになる少し前に設定します。
- **接続タイムアウト** フィールドを使用して、接続の最大待機時間を秒単位で指定します。ゼロまたは指定されていない場合、無期限に待機します。2 秒未満のタイムアウトを使用することはお勧めしません。デフォルトでは 10 秒に設定されています。

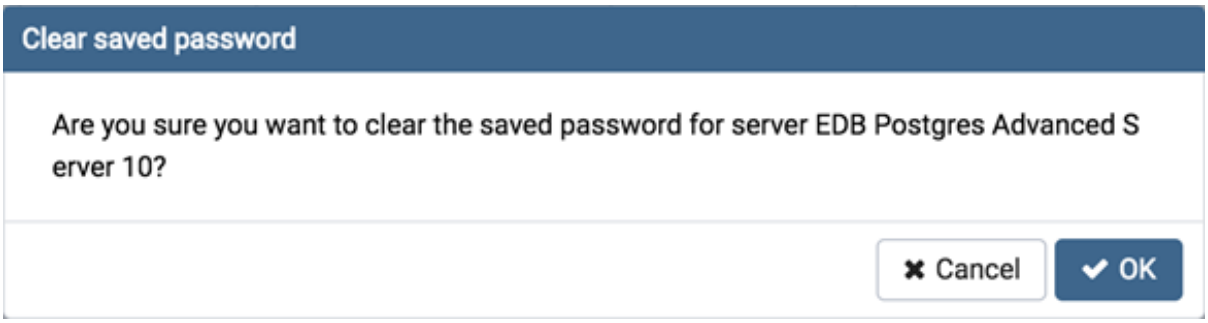
注釈: パスワードファイルオプションは、pgAdmin が libpqv10.0 以降を使用してサーバーに接続している場合にのみサポートされます。

注釈: Passwordexec オプションは、pgAdmin がデスクトップモードで実行されている場合にのみサポートされます。

- [**保存**] ボタンをクリックして、作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、Server ダイアログで指定した値を元の状態に戻します。

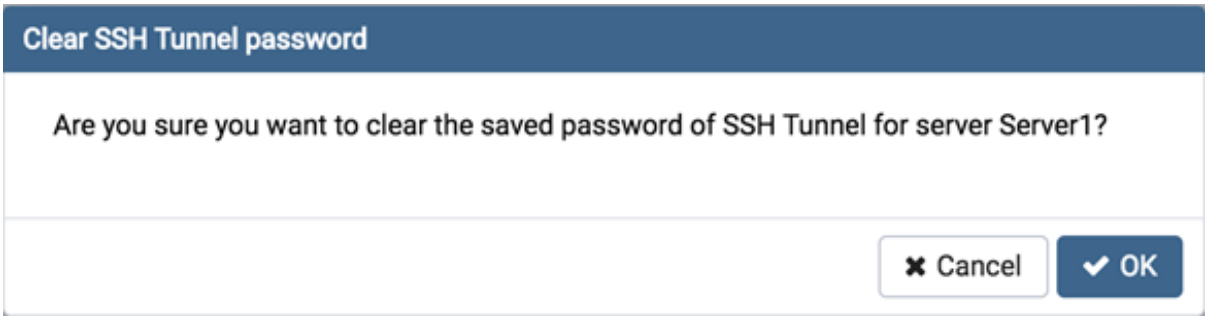
3.2.1 Clear Saved Passwords

保存されたパスワードのクリア 機能を使用して、データベースサーバーの保存されたパスワードをクリアします。



保存されたパスワードのクリア は、選択したサーバーのコンテキストメニューと、トップメニューバーの **オブジェクト** メニューの下に表示されます。

ClearSSHTunnelPassword 機能を使用して、データベースサーバーに接続するための SSHTunnel の保存されたパスワードをクリアします。



クリア *SSH* トンネルパスワード は、選択したサーバーのコンテキストメニューと、トップメニューバーの **オブジェクト** メニューの下に表示されます。

Note: 選択したデータベースサーバのパスワードが既に保存されている場合に有効/表示されます。

または、 **クラウド展開** ウィザードを使用して、クラウドに新しい PostgreSQL インスタンスを展開し、接続することもできます。

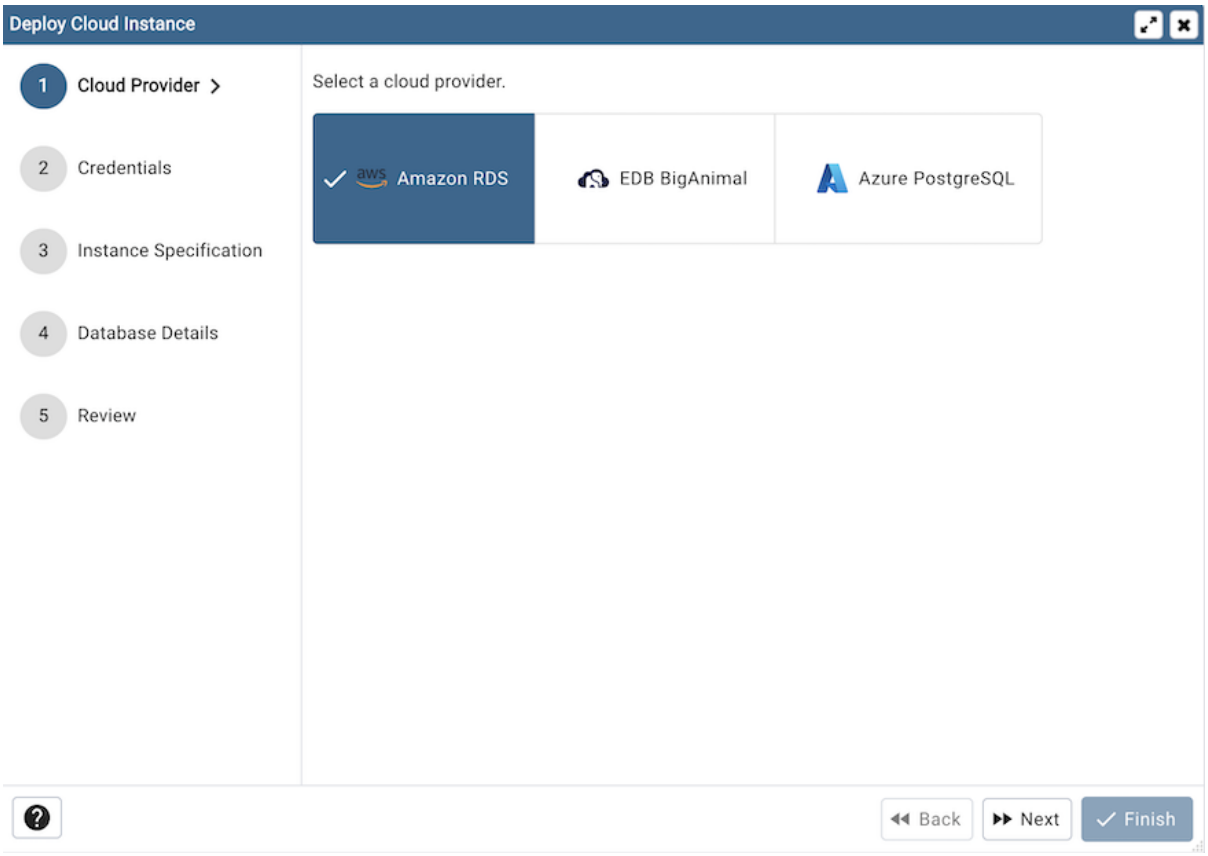
3.3 Cloud Deployment

このモジュールを使用して、PostgreSQL サーバーを AmazonAWS および EDBBigAnimal クラウドにデプロイできます。将来的には、より多くのクラウドオプションが利用できるようになります。

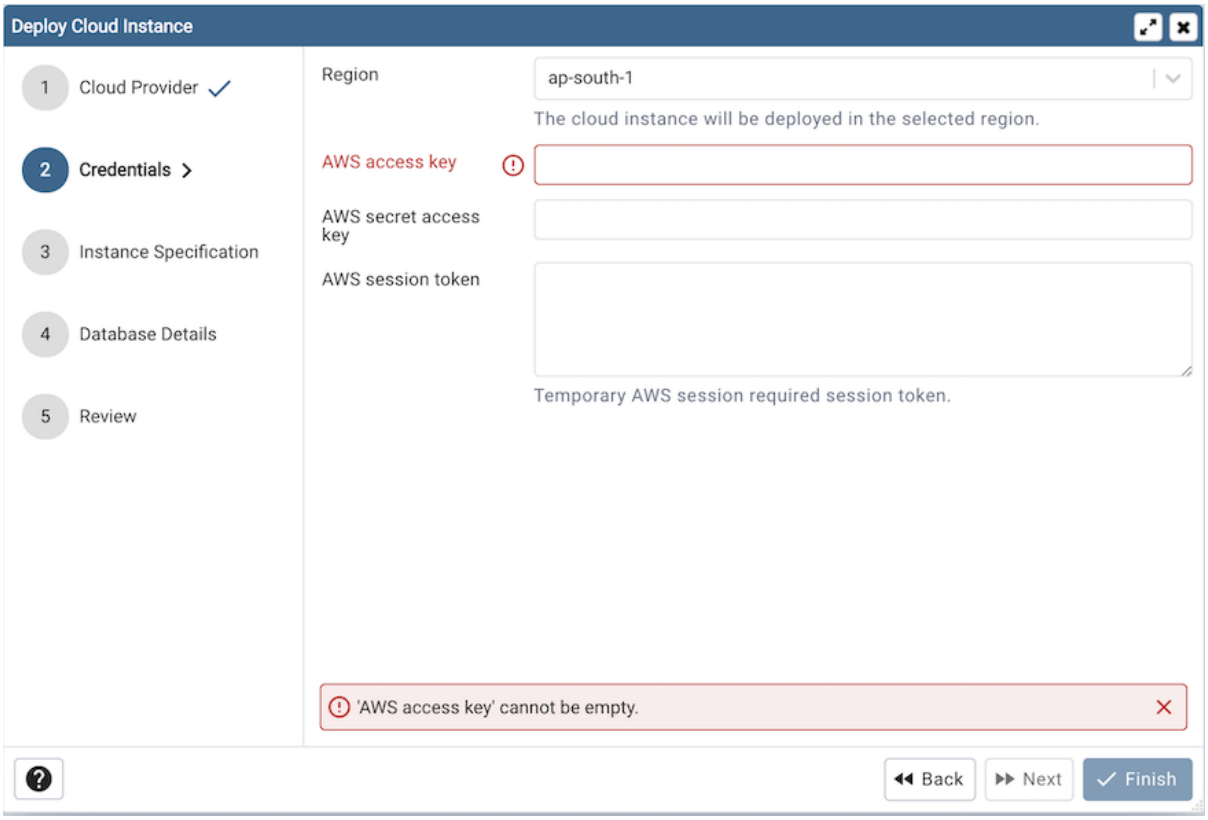
クラウド展開... ツールを起動するには、ツリーコントロールの **サーバーグループ** または **サーバー** を右クリックし、 **登録** メニューから **クラウドインスタンスの展開** を選択します。

3.3.1 Amazon AWS RDS Cloud Deployment

AmazonAWS クラウドに PostgreSQL サーバーをデプロイするには、以下の手順に従います。



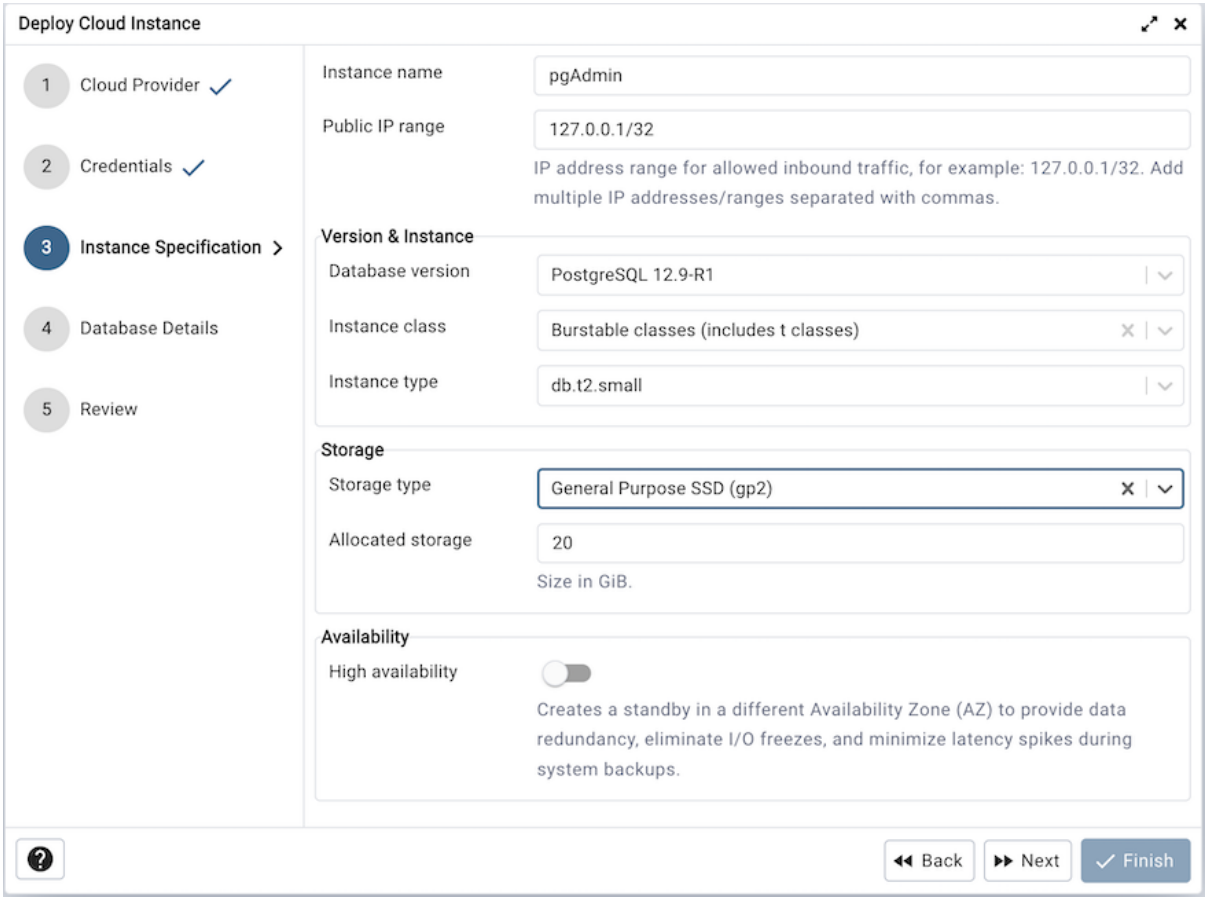
ツールを起動したら、[AmazonRDS] オプションを選択します。[**次へ**] ボタンをクリックして続行します。



[資格情報] ダイアログで、インスタンスを展開するリージョンと、 *AWS* **アクセスキー** と *AWS* **シークレットアクセスキー** を指定します。AWS セッションが一時的な場合にのみ *AWS* **セッショントークン** を提供し

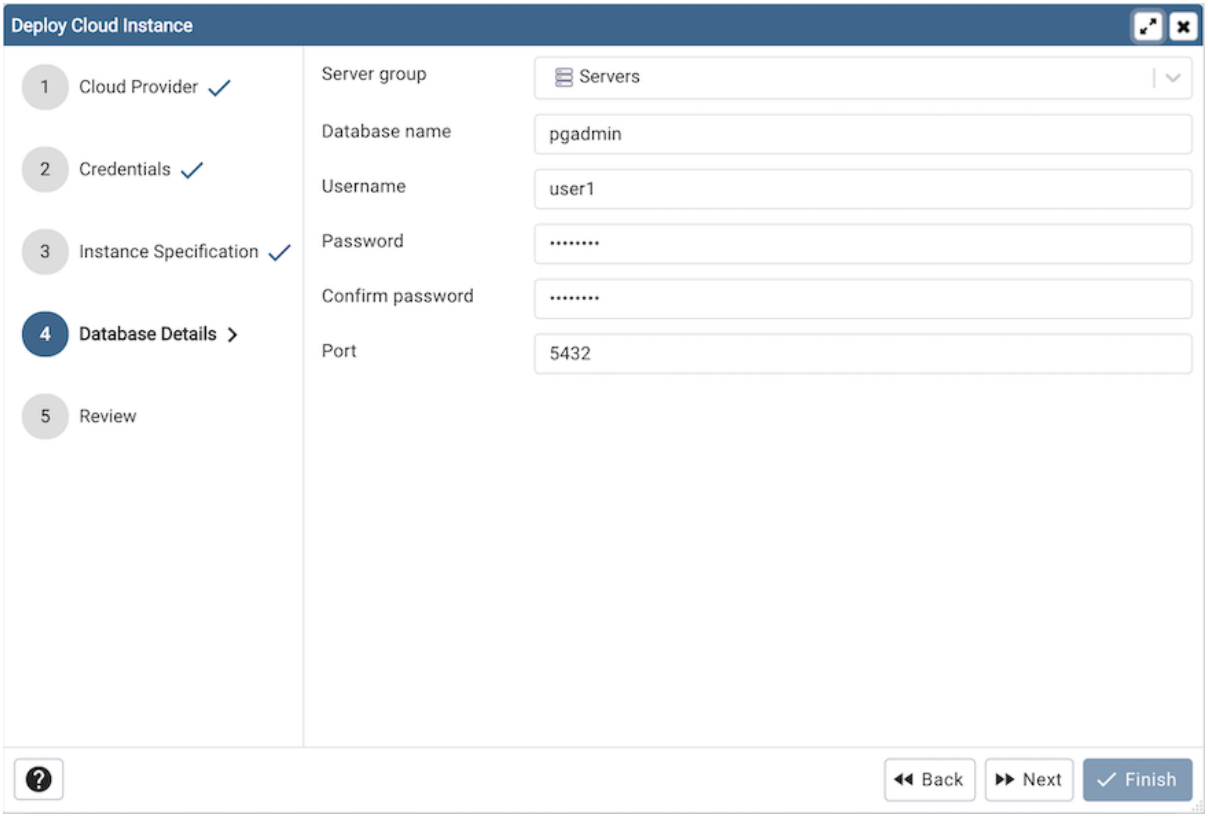
ます。

さらに進むには、[次へ] ボタンをクリックします。先に進む前に、pgAdmin は資格情報を検証します。



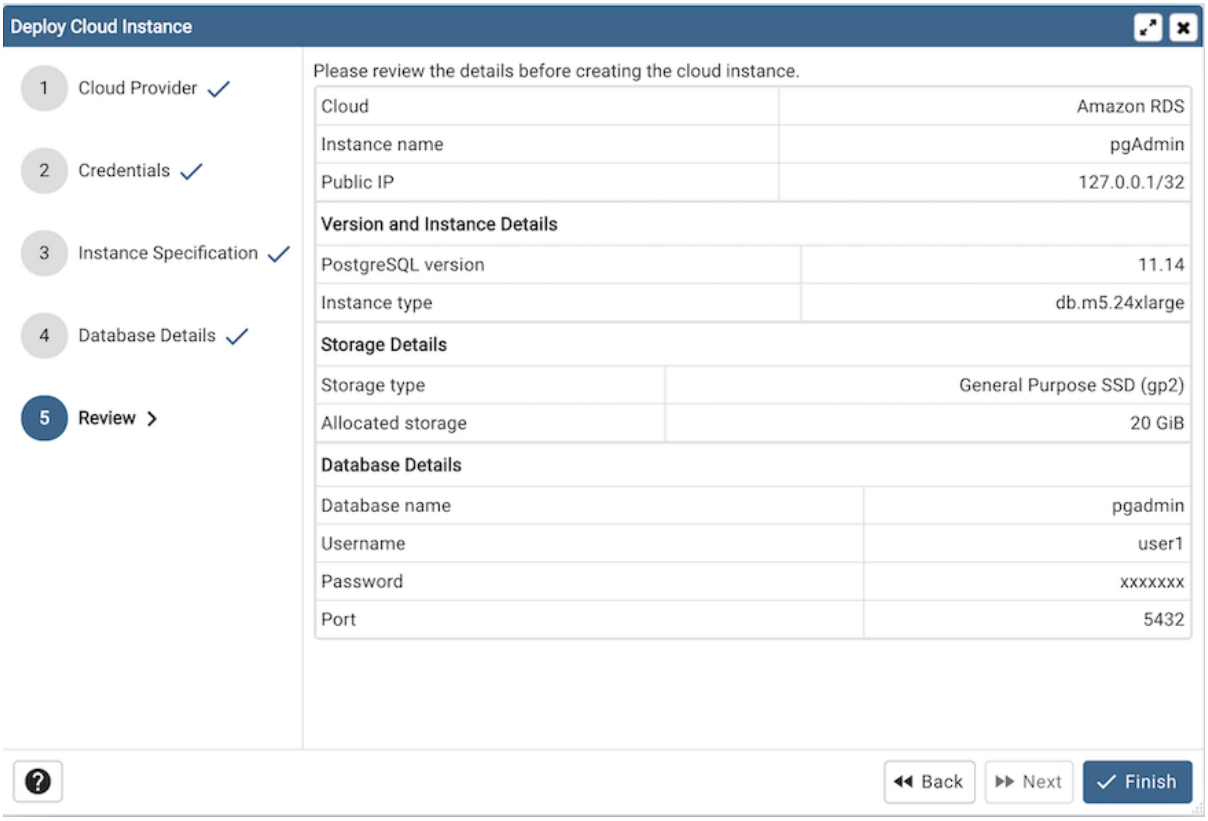
[インスタンスの仕様] タブのフィールドを使用して、インスタンスの詳細を指定します。

- **インスタンス名** フィールドを使用して、PostgreSQL サーバーのインスタンス名を追加します。指定された名前は **ブラウザー** ツリーコントロールにも表示されます。
- **パブリック** *IP* フィールドを使用して、インバウンドトラフィックを許可する IP アドレス範囲を指定します。
- **データベースバージョン** フィールドを使用して、展開する PostgreSQL のバージョンを指定します。
- **インスタンスクラス** フィールドを使用して、この DB インスタンスの計画されたワークロードに必要なコンピューティング、ネットワーク、およびメモリ容量を割り当てます。
- **インスタンスタイプ** フィールドを使用して、インスタンスタイプを選択します。
- **ストレージタイプ** フィールドを使用して、インスタンスのストレージタイプを選択します。3 つのオプションがあります。汎用（SSD）ストレージ、プロビジョンド IOPS（SSD）、およびマグネティックストレージ。
- **割り当てられたストレージ** フィールドを使用して、ストレージ容量を GiB で指定します。
- プロビジョンド IOPS(SSD) ストレージタイプの場合は、 **プロビジョンド** *IOPS* を使用します。
- **高可用性** オプションを使用して、高可用性オプションを指定します。このオプションは、別のアベイラビリティーゾーン (AZ) にスタンバイを作成します。

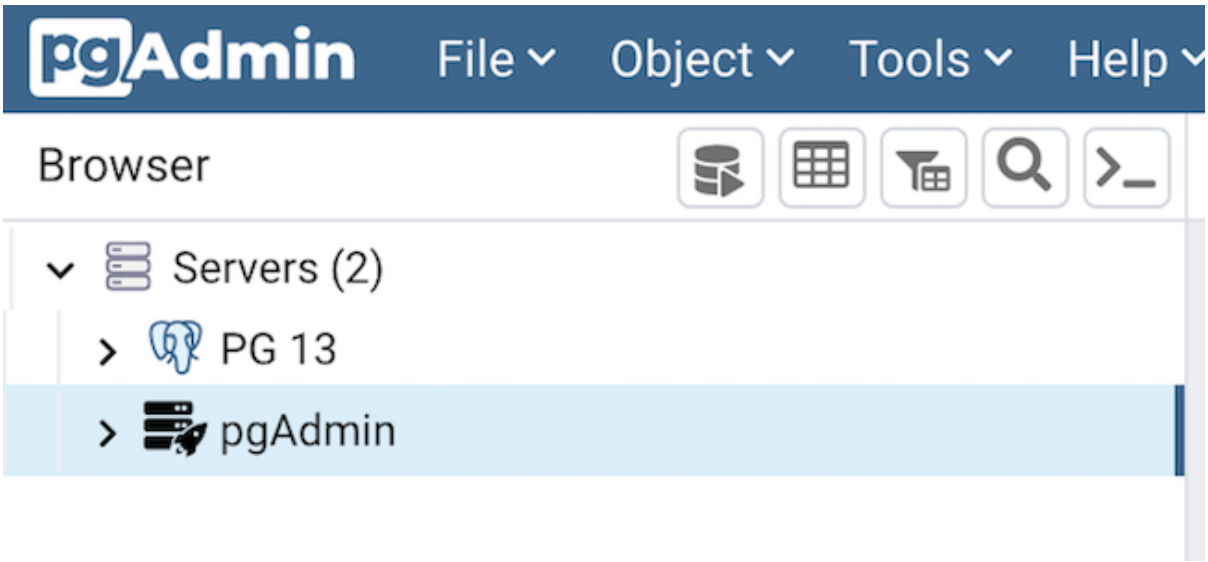


[データベースの詳細] タブのフィールドを使用して、インスタンスの詳細を指定します。

- **サーバーグループ** フィールドのドロップダウンリストボックスを使用して、サーバーの親ノードを選択します。サーバーは指定されたグループ内の **ブラウザー** ツリーコントロールに表示されます。
- **データベース名** フィールドを使用して、PostgreSQL サーバーのデータベース名を追加します。
- *Username* フィールドを使用して、サーバーでの認証時に使用されるロールの名前を指定します。
- *Password* フィールドを使用して、サーバーで認証するときに指定されるパスワードを入力します。
- **パスワードの確認** フィールドを使用して、パスワードを繰り返します。
- *Port* フィールドにサーバーホストのリスナーポート番号を入力します。



最後に、指定したインスタンスの詳細を確認します。Finish ボタンをクリックして、AmazonRDS にインスタンスをデプロイします。

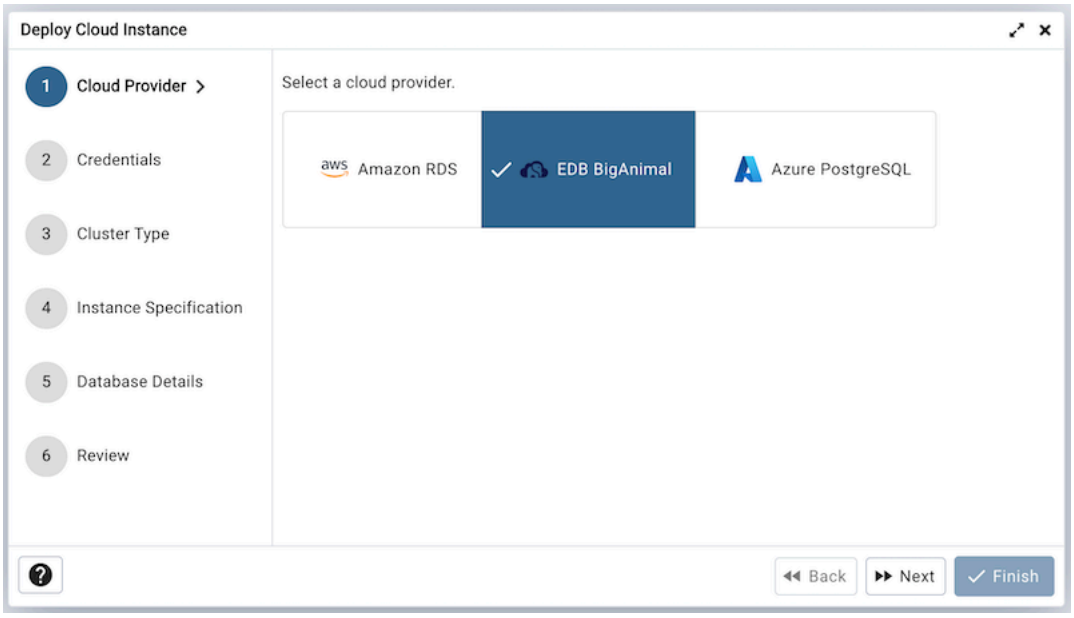


Finish をクリックすると、1 つのバックグラウンドプロセスが開始され、クラウドにインスタンスがデプロイされ、デプロイの進行状況が監視されます。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

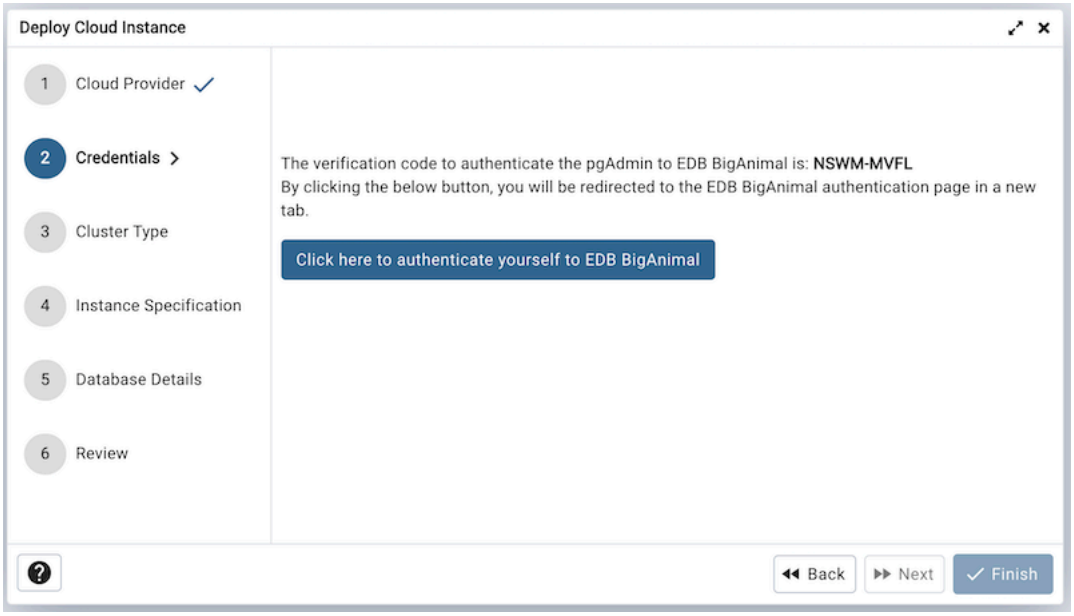
サーバーは、クラウド展開アイコンでツリーに追加されます。展開が完了すると、サーバーの詳細が更新されます。

3.3.2 EDB BigAnimal Cloud Deployment

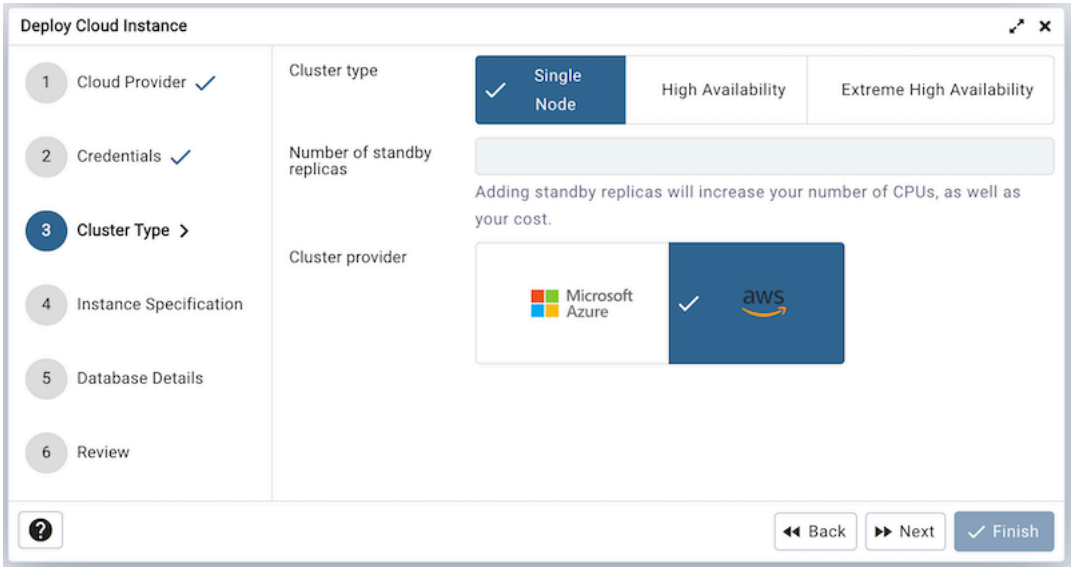
EDBBigAnimal クラウドに PostgreSQL サーバーをデプロイするには、以下の手順に従います。



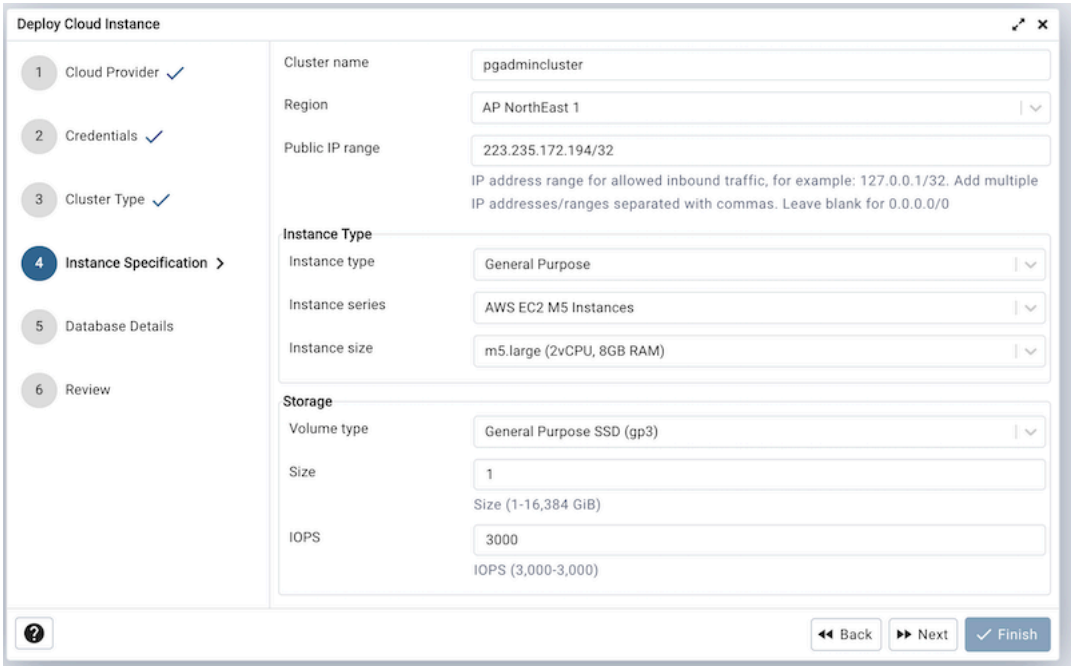
ツールを起動したら、EDBBigAnimal オプションを選択します。[**次へ**] ボタンをクリックして続行します。



次のステップでは、EDBBigAninal に対してユーザーを認証します。指定されたボタンをクリックして認証します。ボタンをクリックすると、ユーザーは検証のための新しいタブにリダイレクトされます。ワンタイムコードを確認すると、pgAdmin が自動的に検出し、次へボタンが有効になります。さらに進むには、[次へ] ボタンをクリックします。



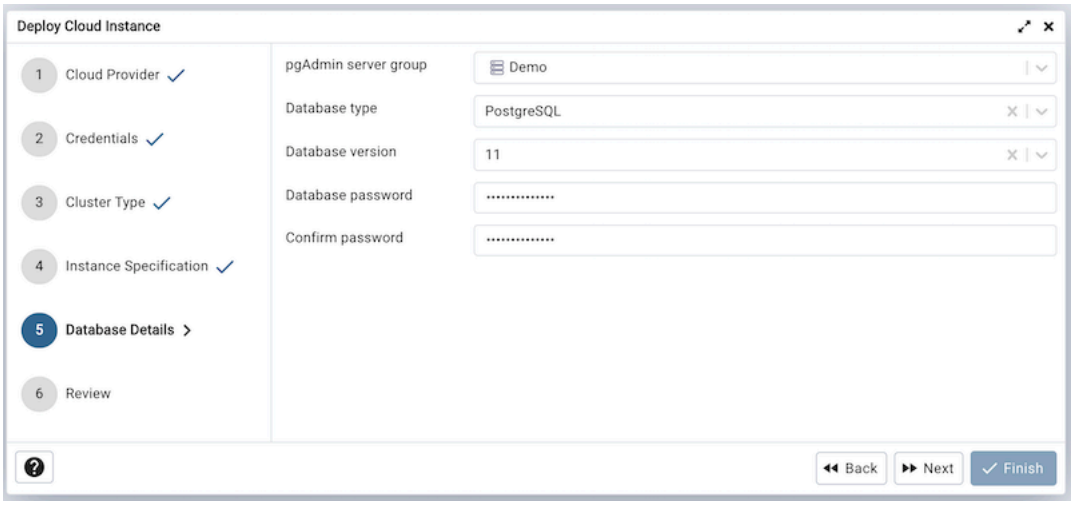
- **クラスタータイプ** フィールドを使用して、クラスタータイプを選択します。
- **番号を使用してください。***ofStandbyReplicas* フィールドを使用して、高可用性クラスターを選択した場合はレプリカを指定します。
- プロバイダーを選択するには、 **クラスタープロバイダー** フィールドを使用します。



[インスタンスの仕様] タブのフィールドを使用して、インスタンスの詳細を指定します。

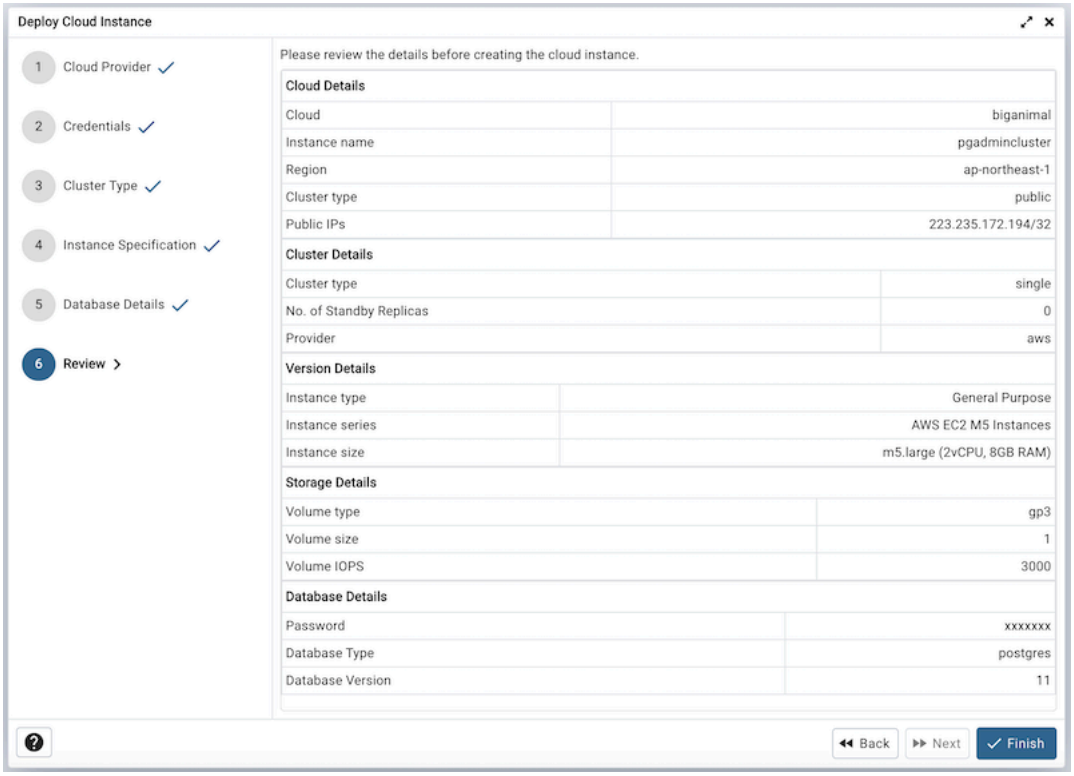
- **クラスター名** フィールドを使用して、PostgreSQL サーバーのクラスター名を追加します。指定された名前は **ブラウザー** ツリーコントロールにも表示されます。
- *Region* フィールドを使用して地域を選択します。
- **パブリック** *IP* **範囲** フィールドを使用して、インバウンドトラフィックを許可する IP アドレス範囲を指定します。0.0.0.0/0 の場合は空白のままにします
- **インスタンスタイプ** フィールドを使用して、インスタンスタイプを選択します。

- **インスタンスシリーズ** フィールドを使用して、インスタンスシリーズを選択します。
- **インスタンスサイズ** フィールドを使用して、この DB インスタンスの計画されたワークロードに必要なコンピューティング、ネットワーク、およびメモリの容量を割り当てます。
- **ボリュームタイプ** フィールドを使用して、インスタンスのストレージタイプを選択します。
- **ボリュームのプロパティ** フィールドを使用して、ストレージ容量を指定します。このフィールドは Azure に固有です。
- **ボリュームサイズ** フィールドを使用して、ストレージサイズを指定します。このフィールドは AWS に固有です。
- *VolumeIOPS* フィールドを使用して、ストレージ IOPS を指定します。このフィールドは AWS に固有です。

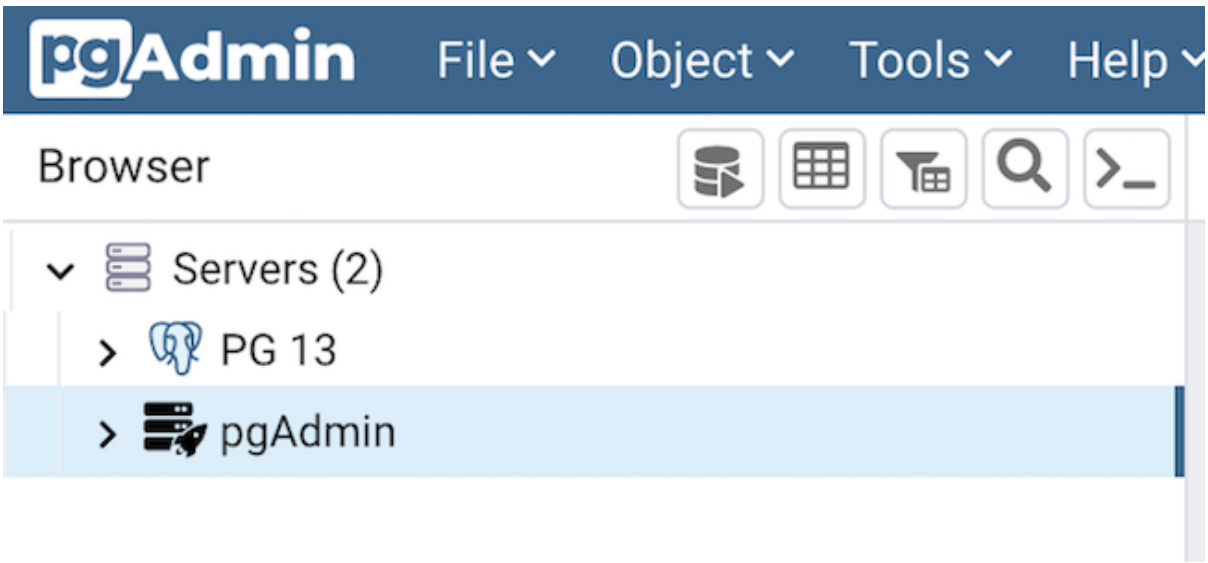


[データベースの詳細] タブのフィールドを使用して、インスタンスの詳細を指定します。

- **サーバーグループ** フィールドのドロップダウンリストボックスを使用して、サーバーの親ノードを選択します。サーバーは指定されたグループ内の **ブラウザー** ツリーコントロールに表示されます。
- **データベースの種類** フィールドを使用して、PostgreSQL の種類、EnterpriseDBPostgreSQLAdvanced-Server または PostgreSQL を指定します。
- *PostgreSQLversion* フィールドを使用して、データベースのバージョンを選択します。
- **データベースパスワード** フィールドを使用して、サーバーでの認証時に指定されるパスワードを入力します。
- **パスワードの確認** フィールドを使用して、パスワードを繰り返します。



最後に、指定したクラスターの詳細を確認します。Finish ボタンをクリックして、インスタンスを EDBBigAnimal にデプロイします。

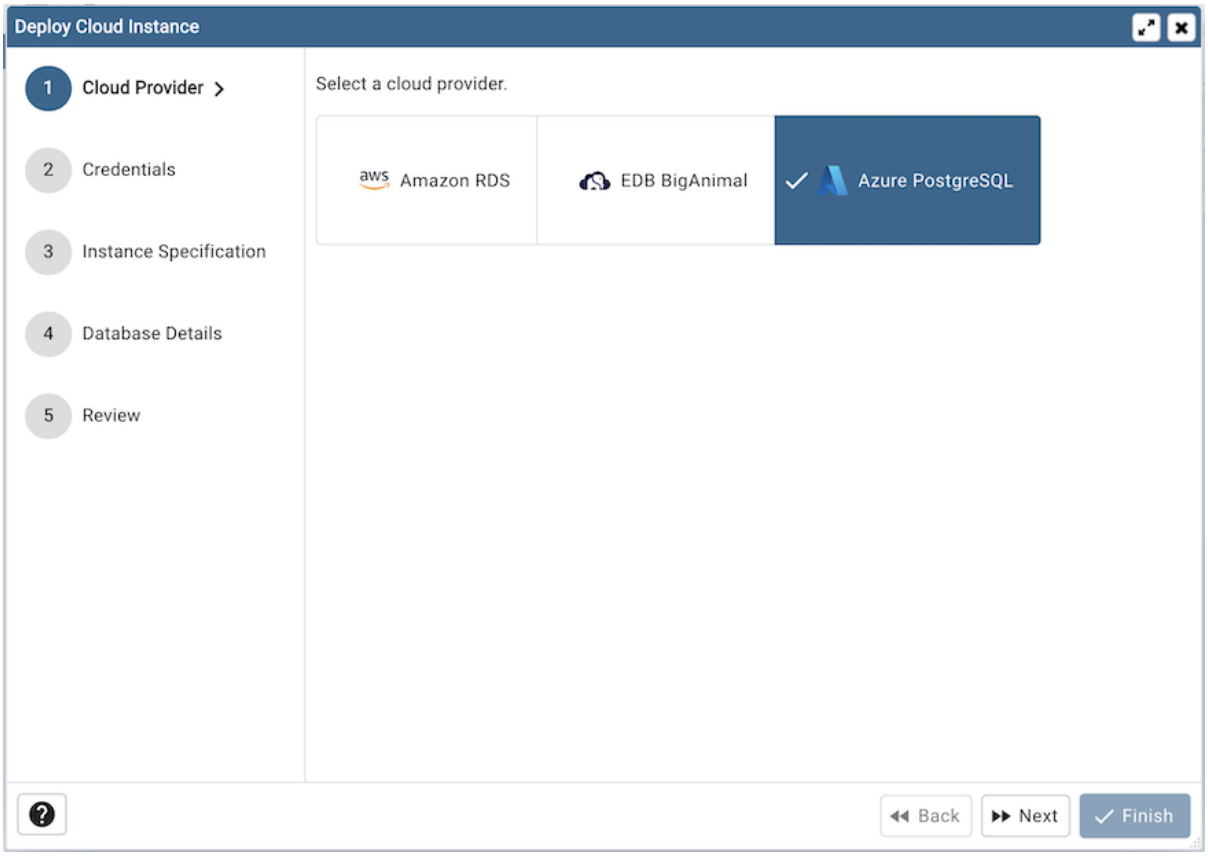


Finish をクリックすると、1 つのバックグラウンドプロセスが開始され、クラウドにインスタンスがデプロイされ、デプロイの進行状況が監視されます。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

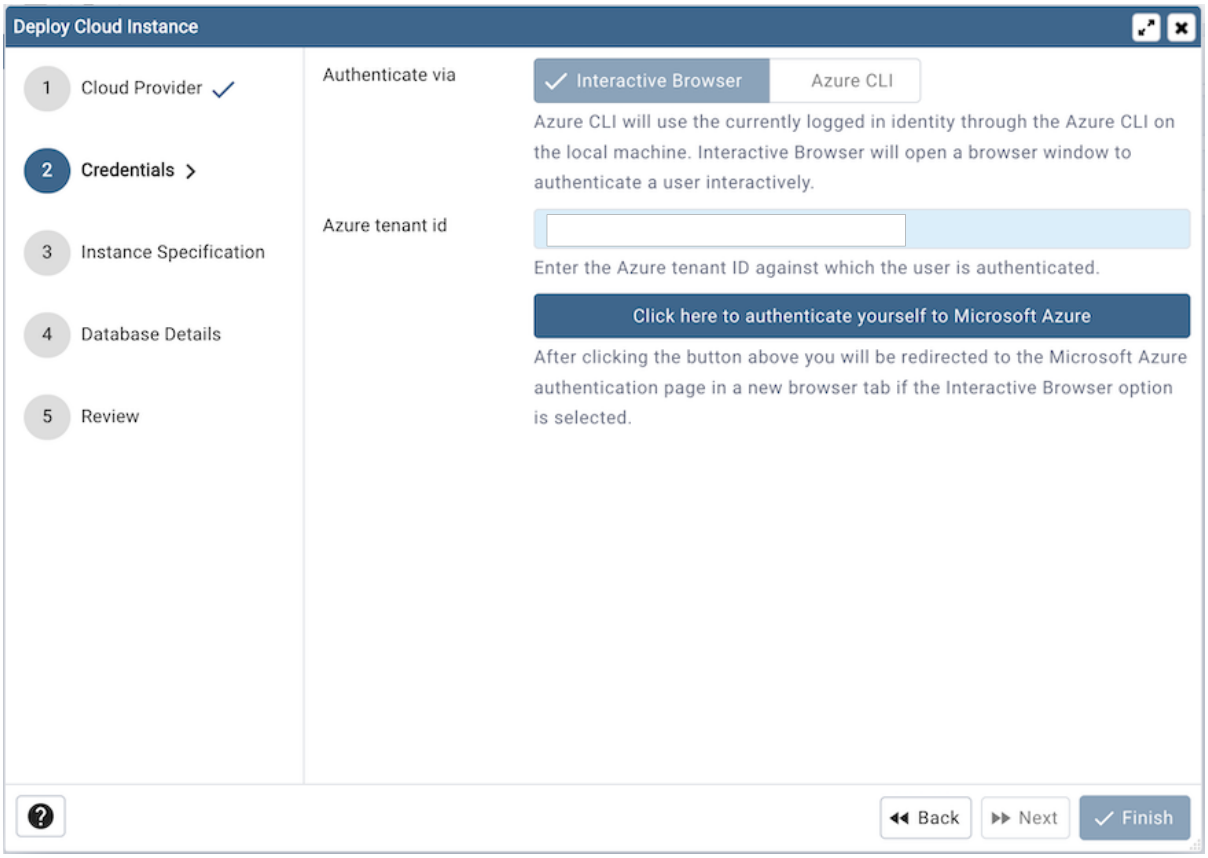
サーバーは、クラウド展開アイコンでツリーに追加されます。展開が完了すると、サーバーの詳細が更新されます。

3.3.3 Azure PostgreSQL Cloud Deployment

Azure クラウドに PostgreSQL サーバーを展開するには、次の手順に従います。



ツールを起動したら、[AzurePostgreSQL] オプションを選択します。[**次へ**] ボタンをクリックして続行します。

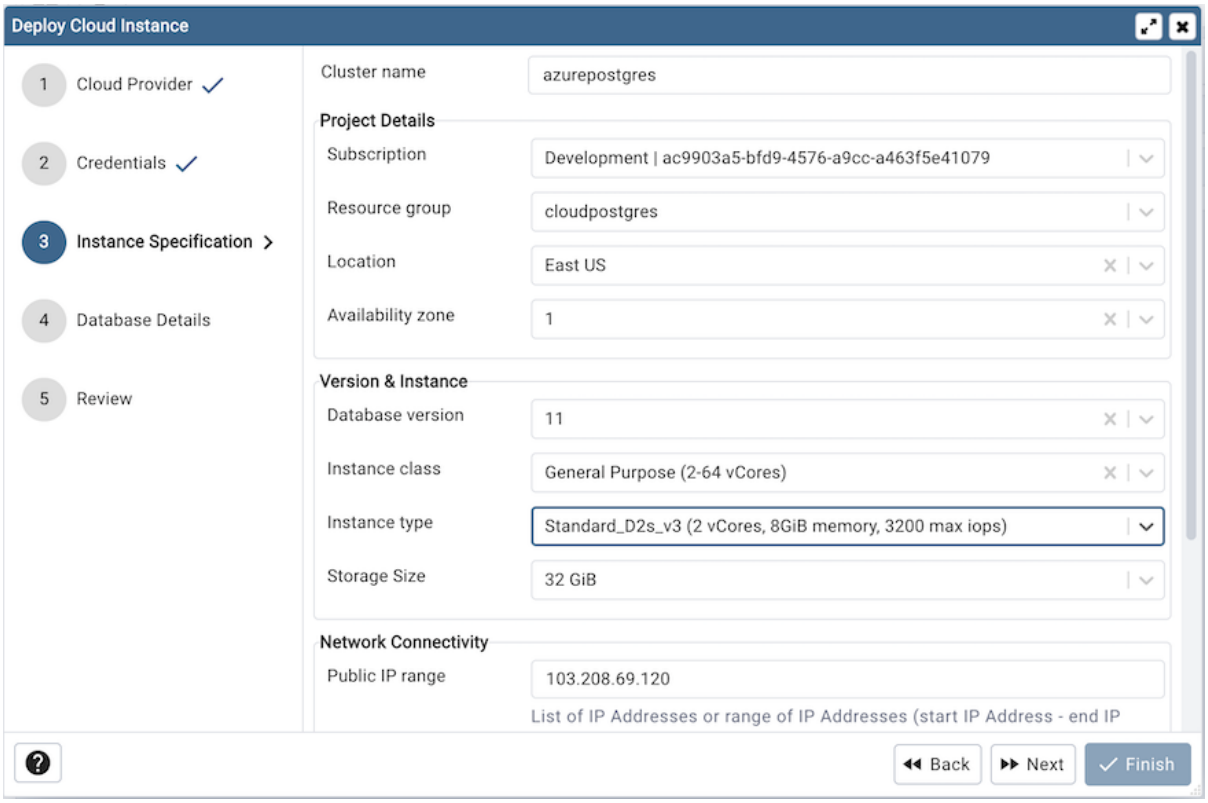


[資格情報] ダイアログで、対話型ブラウザーまたは AzureCLI のいずれかの認証方法を選択します。AzureCLI は、ローカルマシン上の AzureCLI を介して現在ログインしている ID を使用します。InteractiveBrowser はブラウザーウィンドウを開き、ユーザーを対話的に認証します。

Azuretenant id を使用して、ユーザーを認証する Azure テナント ID を指定します。

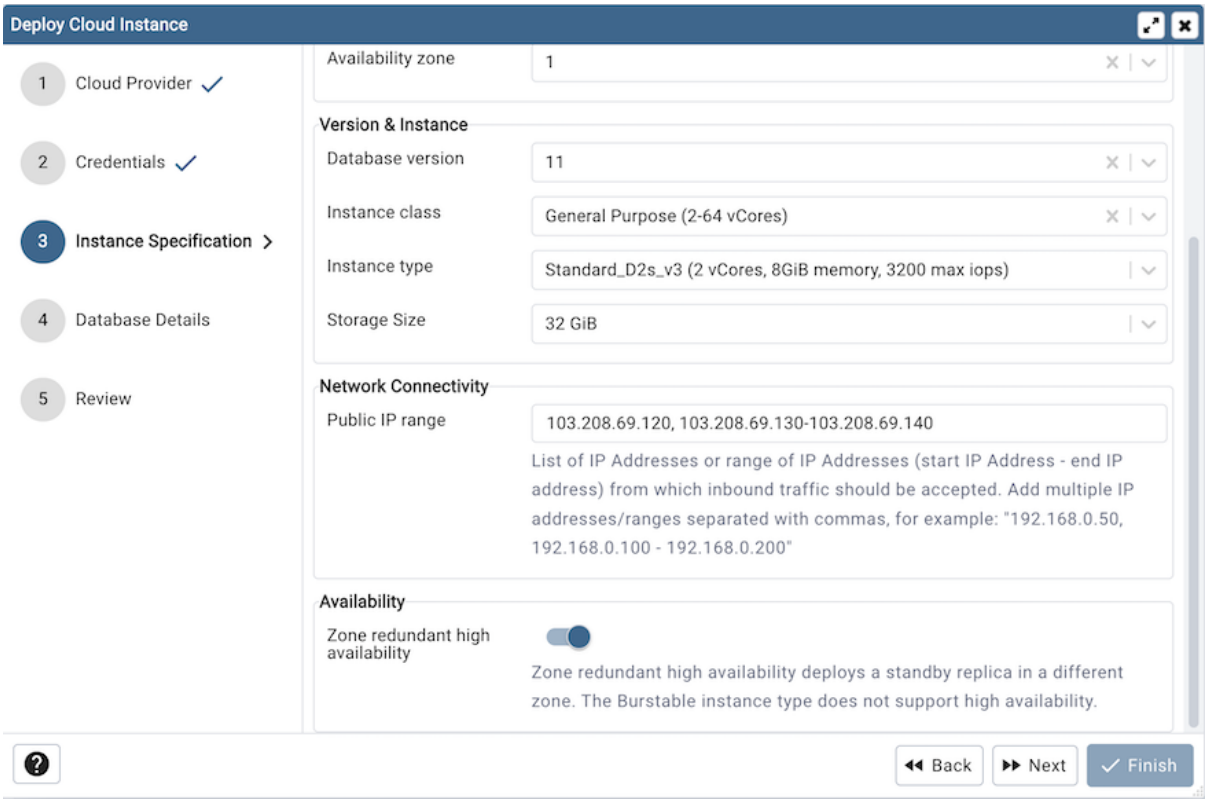
[対話型ブラウザー] オプションが選択されている場合、 **ここをクリックして自分自身を** *MicrosoftAzure* で認証すると、ユーザーは新しいブラウザータブの MicrosoftAzure 認証ページにリダイレクトされます。AzureCLI 認証は、デスクトップモードでのみ使用できます。

認証が完了したら、[次へ] ボタンをクリックして続行します。

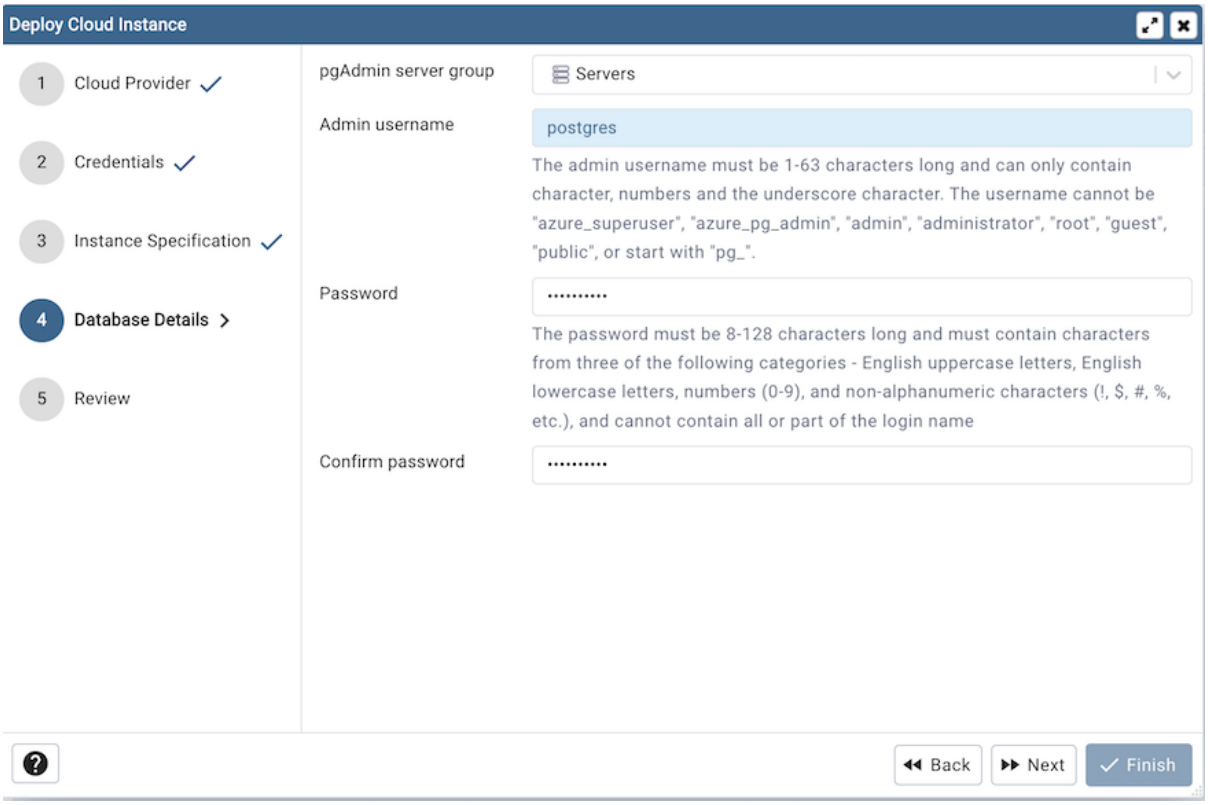


[インスタンスの仕様] タブのフィールドを使用して、インスタンスの詳細を指定します。

- **クラスター名** フィールドを使用して、PostgreSQL サーバーの名前を追加します。指定された名前は **ブラウザー** ツリーコントロールにも表示されます。
- AzurePortal のユーザーアクセスレベルに基づいて入力される **サブスクリプション** オプションからサブスクリプションを選択します。
- PostgreSQL インスタンスを作成するリソースグループを **リソースグループ** ドロップダウンから選択します。
- *Location* オプションから PostgreSQL インスタンスを展開する場所を選択します。
- **アベイラビリティーゾーン** のオプションから、指定したリージョンのアベイラビリティーゾーンを選択して、PostgreSQL インスタンスをデプロイします。
- **データベースバージョン** オプションを使用して、PostgreSQL データベースのバージョンを指定します。
- **インスタンスクラス** フィールドを使用して、この DB インスタンスの計画されたワークロードに必要なコンピューティング、ネットワーク、およびメモリ容量を割り当てます。
- **インスタンスタイプ** フィールドを使用して、インスタンスタイプを選択します。
- **ストレージサイズ** オプションを使用して、ストレージ容量を指定します。



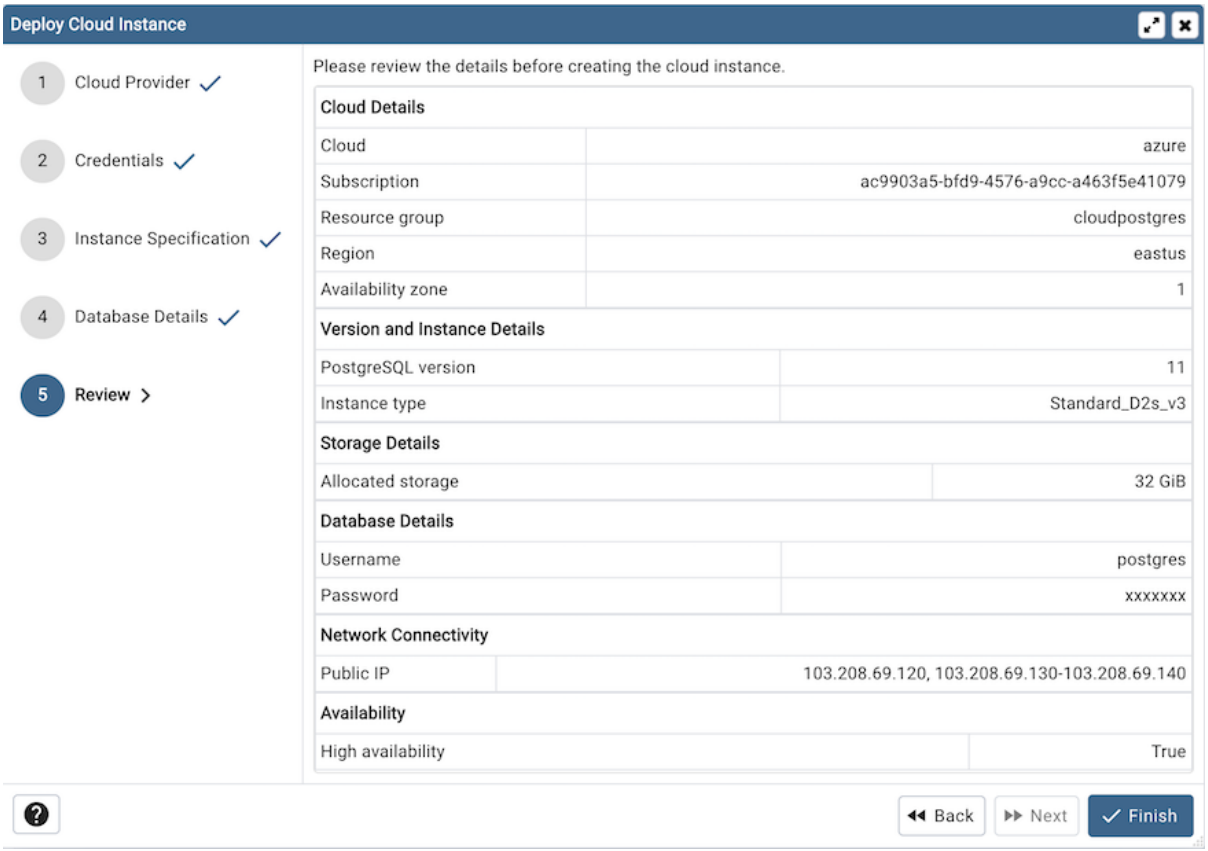
- **パブリック** *IP* フィールドを使用して、受信トラフィックを受け入れる IP アドレスまたは IP アドレスの範囲（開始 IP アドレス-終了 IP アドレス）を指定します。複数の IP アドレス/範囲をコンマで区切って追加します。例：/"192.168.0.100-192.168.0.200/"
- **ゾーン冗長高可用性** オプションを使用して、高可用性オプションを指定します。ゾーン冗長高可用性は、別のゾーンにスタンバイレプリカをデプロイします。Burstable インスタンスタイプは高可用性をサポートしていません。



[データベースの詳細] タブのフィールドを使用して、PostgreSQL データベースの詳細を指定します。

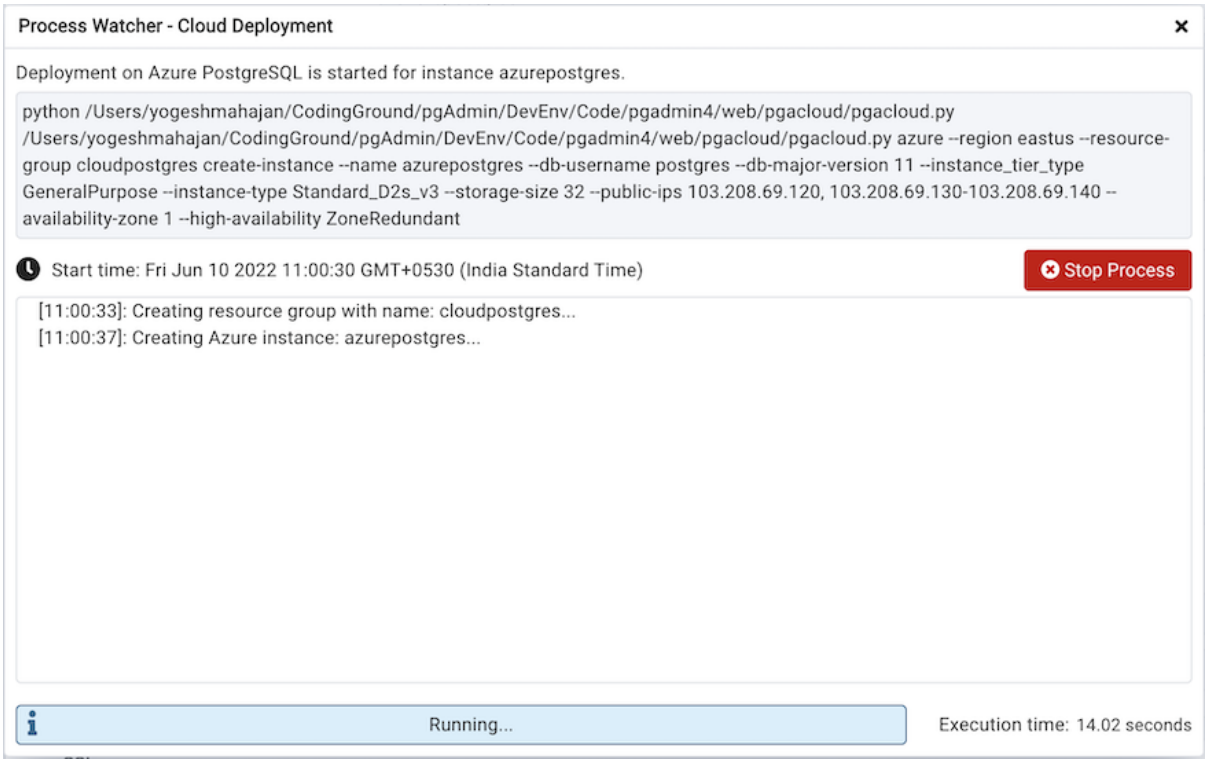
- *pgAdmin* **サーバーグループ** フィールドのドロップダウンリストを使用して、サーバーの親ノードを選択します。サーバーは指定されたグループ内の **ブラウザー** ツリーコントロールに表示されます。
- *Adminusername* フィールドを使用して、PostgreSQL サーバーのデータベース名を追加します。
- *Password* フィールドを使用して、サーバーで認証するときに指定されるパスワードを入力します。
- **パスワードの確認** フィールドを使用して、パスワードを繰り返します。

次へボタンをクリックして続行します。

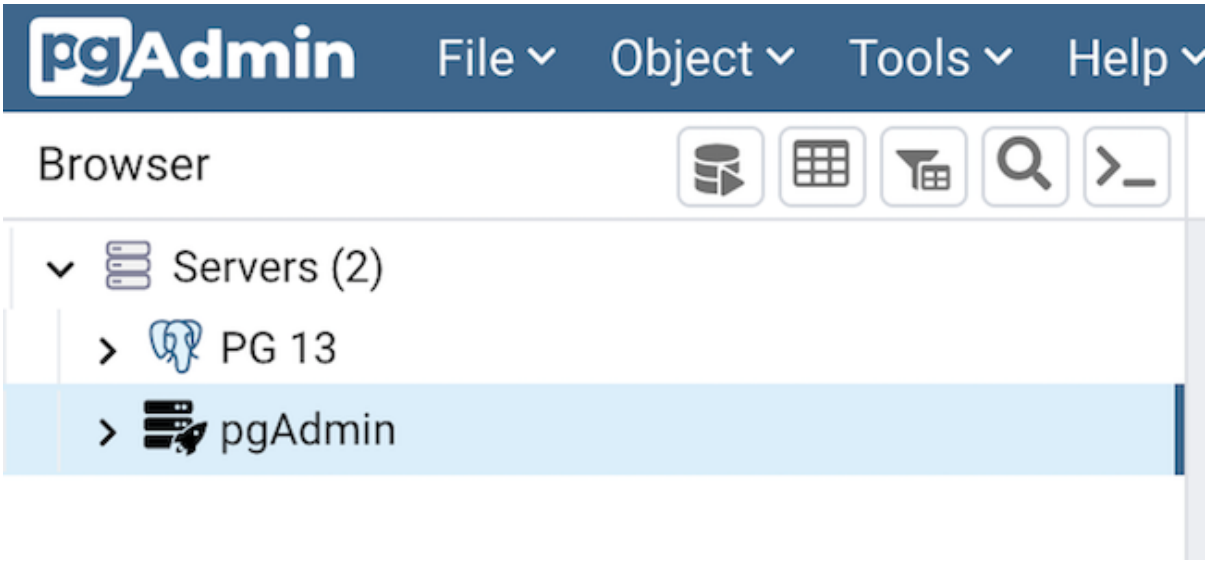


最後に、指定したインスタンスの詳細を確認します。Finish ボタンをクリックして、AzurePostgreSQL にインスタンスをデプロイします。

Finish をクリックすると、1 つのバックグラウンドプロセスが開始され、クラウドにインスタンスがデプロイされ、デプロイの進行状況が監視されます。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます



サーバーは、クラウド展開アイコンでツリーに追加されます。展開が完了すると、サーバーの詳細が更新されます。

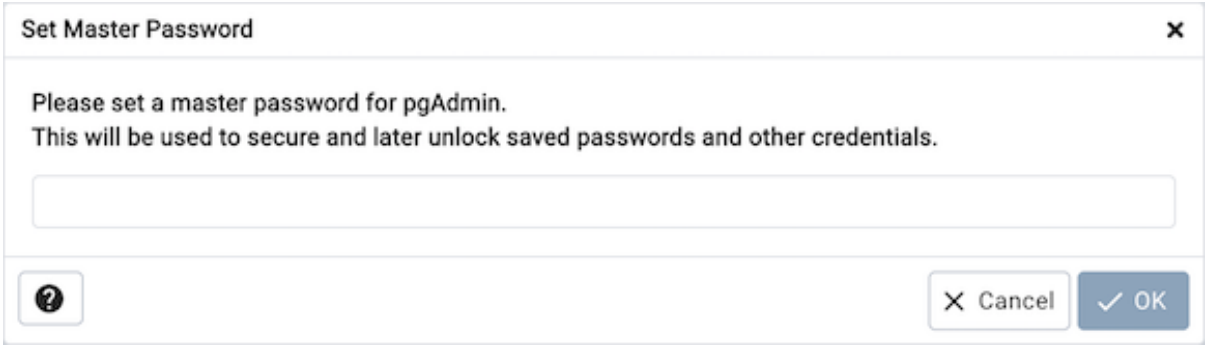


保存したサーバーパスワードを保護し、後でロックを解除するには、マスターパスワードが必要です。これはユーザーによって設定され、config を使用して無効にできます。

3.4 Master Password

保存したサーバーパスワードを保護し、後でロックを解除するには、マスターパスワードが必要です。これは、デスクトップモードのユーザーにのみ適用されます。

- アプリケーションを起動して初めてウィンドウを開くと、マスターパスワードの入力を求められます。
- マスターパスワードを設定すると、既存の保存されたパスワードはすべてマスターパスワードを使用して再暗号化されます。
- SQLiteDB ファイルまたは外部データベースに保存されているサーバーパスワードは、マスターパスワードを使用して暗号化および復号化されます。



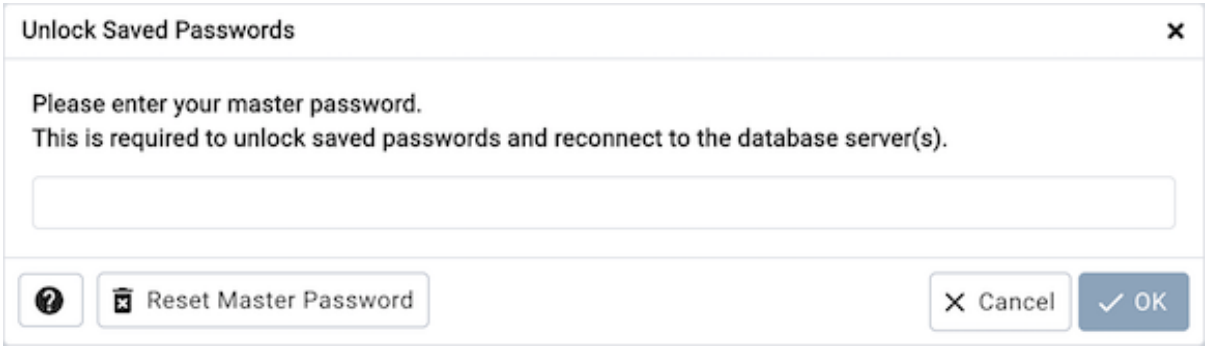
注釈: pgAdmin は **secure by default** を目指していますが、構成パラメーター *MASTER_PASSWORD_REQUIRED=False* を設定することにより、マスターパスワードを無効にできます。構成パラメーターと、それらを組織全体で変更または適用する方法の詳細については、 **secure by default** を参照してください。

注釈: マスターパスワードが無効になっている場合、保存されているパスワードはすべて削除されます。

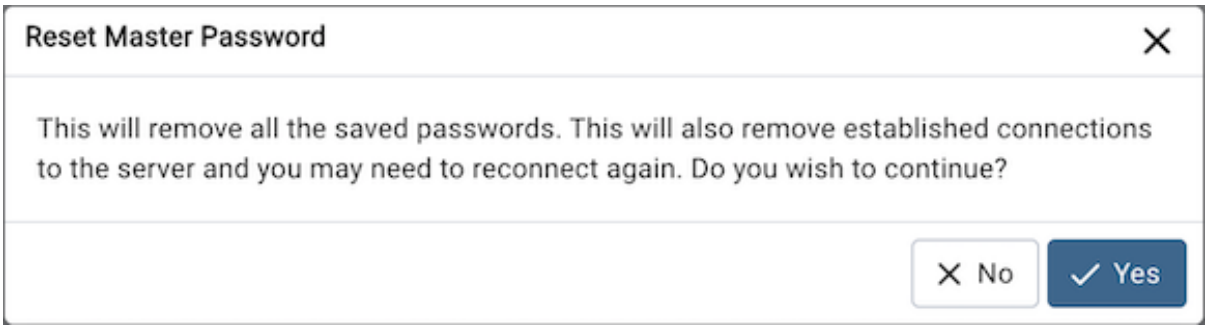
警告: マスターパスワードが無効になっている場合、保存されたパスワードは構成データベース内の情報から派生したキーを使用して暗号化されます。マスターパスワードを使用すると、暗号化キーをどこにも保存する必要がないため、攻撃者が構成データベースを利用できるようになった場合にサーバーの資格情報へのアクセスが防止されます。

SavePassword オプションを使用する場合は、マスターパスワードを使用することをお勧めします。

- マスターパスワードは物理ストレージのどこにも保存されません。アプリケーションのメモリに一時的に保存され、アプリケーションを再起動しても保存されません。
- pgAdmin サーバーを再起動すると、マスターパスワードの入力を求められます。



- マスターパスワードを忘れた場合は、 **マスターパスワードのリセット** ボタンを使用してパスワードをリセットできます。



警告: マスターパスワードをリセットすると、保存されているすべてのパスワードが削除され、既存の確立された接続がすべて閉じます。

サーバー接続を定義したら、サーバー名を右クリックし、[**サーバーに接続**] を選択してサーバーで認証し、pgAdmin を使用してサーバー上にあるオブジェクトの管理を開始します。

3.5 Connect to Server

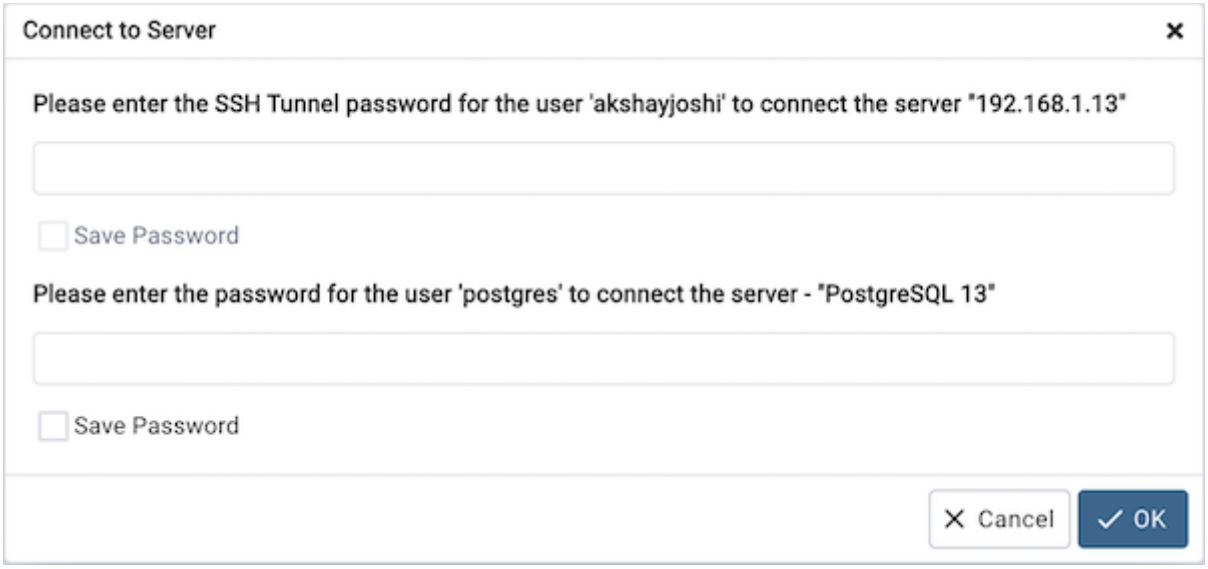
サーバーに接続 ダイアログを使用して、定義されたサーバーで認証し、pgAdmin ツリーコントロールを介してサーバーに保存されているオブジェクトにアクセスします。ダイアログにアクセスするには、*pgAdmin* ツリーコントロールでサーバー名を右クリックし、コンテキストメニューから *ConnectServer...* を選択します。



選択したサーバーの認証情報を提供します。

- *Password* フィールドを使用して、定義されたサーバーに関連付けられたユーザーのパスワードを入力します。
- *SavePassword* の横にあるチェックボックスをオンにして、今後の接続のためにパスワードを保存するようにサーバーに指示します。パスワードを保存すると、このサーバー定義でデータベースサーバに再接続するときにプロンプトが表示されます。

SSH トンネリングを使用する場合、 **サーバーに接続** ダイアログで、SSH トンネルとデータベースサーバーのパスワードの入力が求められます。



選択したサーバーの認証情報を提供します。

- *Password* フィールドを使用して、定義されたサーバーに関連付けられたユーザーのパスワードを入力します。
- それぞれの *SavePassword* の横にあるチェックボックスをオンにして、今後の接続のためにパスワードを保存するようにサーバーに指示します。パスワードを保存すると、このサーバー定義でデータベースサーバに再接続するときにプロンプトが表示されます。

サーバーが正常に接続すると、pgAdmin クライアントは右下隅の緑色のステータスバーにメッセージを表示します。

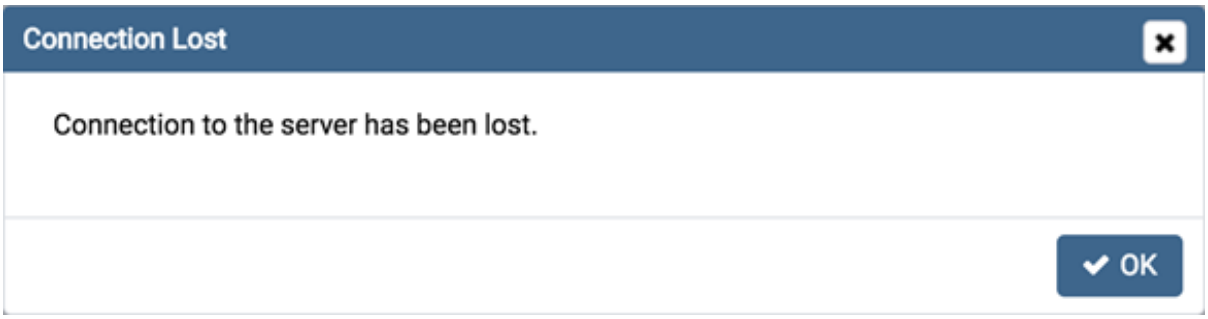
接続中にエラーメッセージを受け取った場合は、ネットワークで pgAdmin ホストとデータベースサーバーのホストが通信できることを確認してください。特定のエラーメッセージの詳細については、 *Connection Error* ヘルプページを参照してください。

接続の詳細を確認または変更するには、サーバーの名前を右クリックし、コンテキストメニューから **プロパティ...** を選択します。

3.6 Connection Error

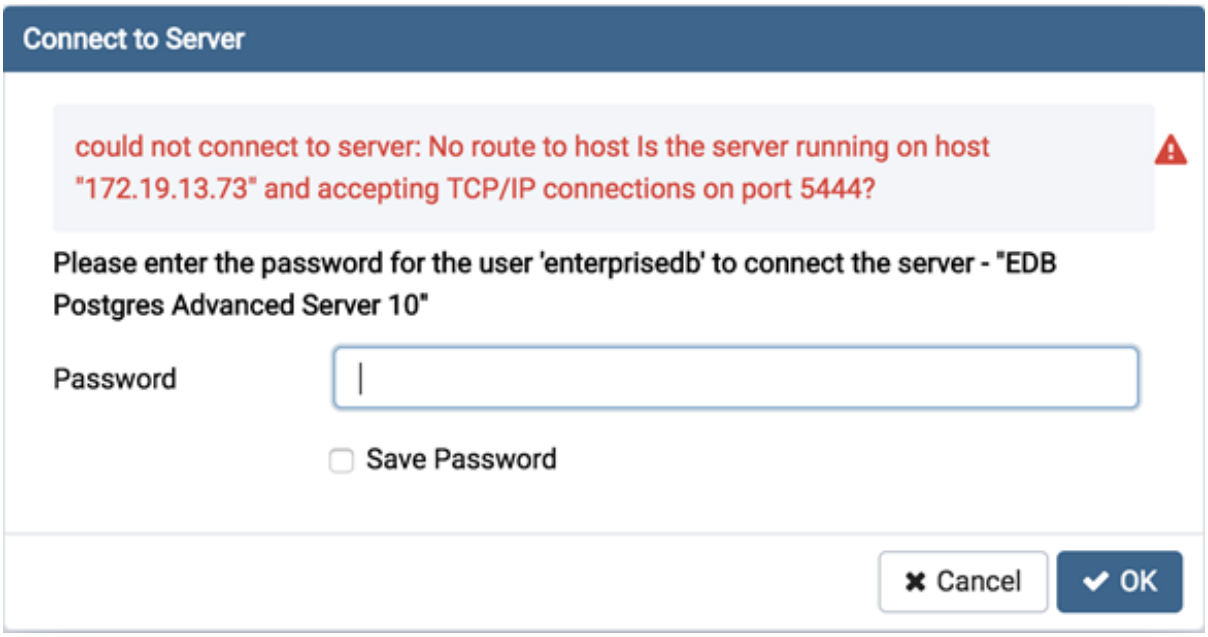
PostgreSQL サーバーに接続すると、エラーメッセージが表示される場合があります。エラーメッセージが表示された場合は、メッセージを注意深く確認してください。各エラーメッセージには、問題の解決に必要な情報が含まれています。特定のエラーの詳細については、以下のリストでエラーメッセージを見つけてください。

Connection to the server has been lost



このエラーメッセージは、接続試行に指定されたしきい値よりも時間がかかったことを示します。 **サーバー** ダイアログで提供される接続プロパティに問題があるか、ネットワーク接続に問題があるか、サーバーが実行されていない可能性があります。

could not connect to Server: Connection refused



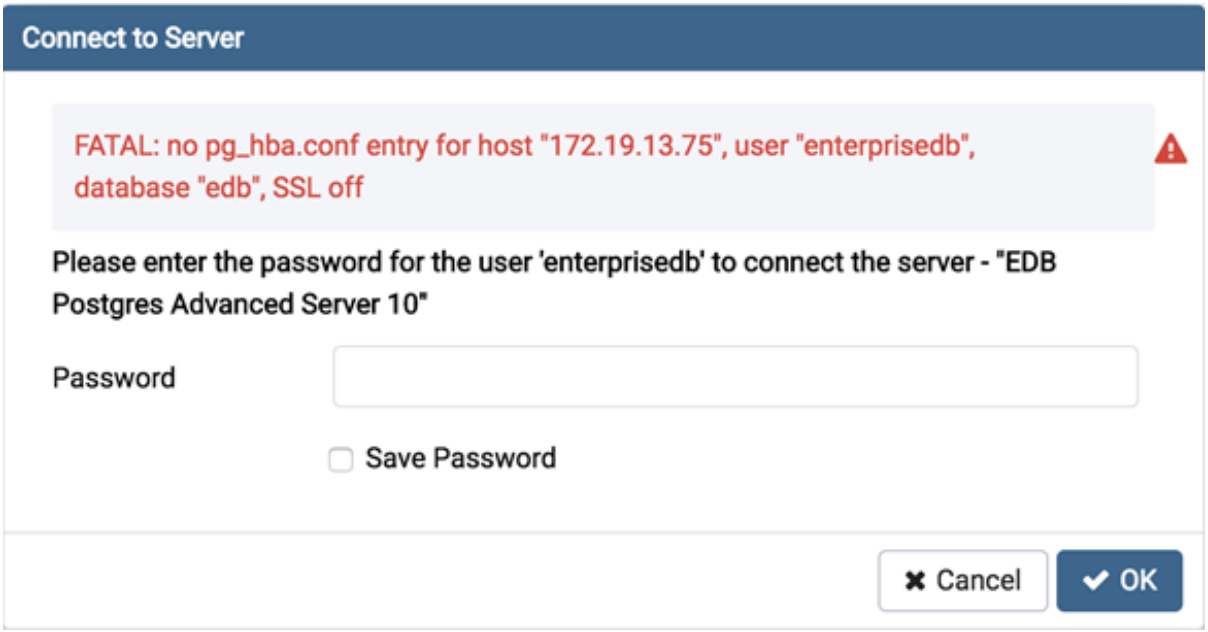
pgAdmin がこのメッセージを表示する場合、これには 2 つの理由が考えられます。

- データベースサーバーが実行されていません。起動してください。
- サーバーは、表示されているアドレスで TCP/IP 要求を受け入れるように構成されていません。

セキュリティ上の理由から、「すぐに使える」PostgreSQL サーバーは TCP/IP ポートでリッスンしません。代わりに、TCP/IP 要求をリッスンするように有効にする必要があります。これは、 ** listen_addresses=' ' * を追加することで実行できます。これにより、サーバーは任意の IP インターフェイスでの接続を受け入れます。

詳細については、 runtime configuration に関する PostgreSQL ドキュメントを参照してください。

FATAL: no pg_hba.conf entry



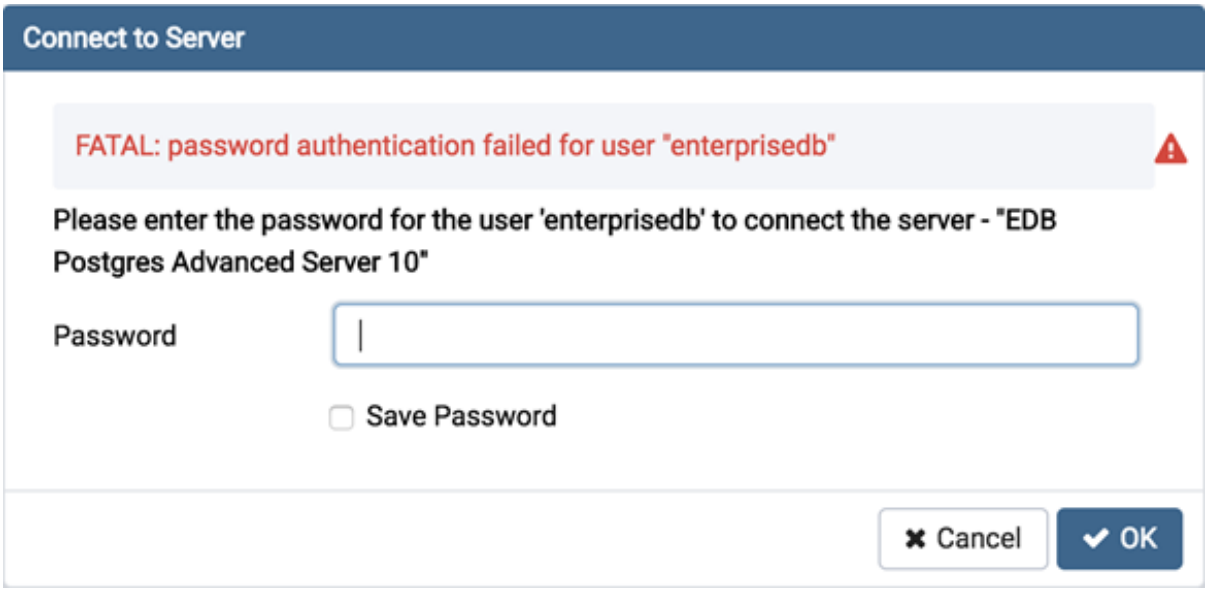
接続時に pgAdmin がこのメッセージを表示する場合、サーバーにはネットワークを介して正しく接続できますが、接続を受け入れるように構成されていません。あなたのクライアントはデータベースの正当なユーザーとして検出されませんでした。

サーバーに接続するには、データベースサーバー上の pg_hba.conf ファイルを pgAdmin クライアントのホストからの接続を受け入れるように構成する必要があります。データベースサーバホスト上の pg_hba.conf ファイルを変更し、次の形式でエントリを追加します。

- IPV4 ネットワークの場合は **host template1 postgres 192.168.0.0/24 md5**
- IPV6 ネットワークの場合は **host template1 postgres ::ffff:192.168.0.0/120 md5**

詳細については、 client authentication に関する PostgreSQL ドキュメントを参照してください。

FATAL: password authentication failed



passwordauthenticationfailedforuser エラーメッセージは、入力したパスワードに問題がある可能性があることを示しています。パスワードを再試行して、正しく入力したことを確認します。エラーメッセージが返された場合は、正しいパスワードを使用していること、サーバーへのアクセスが許可されていること、およびサーバーの postgresql.conf 構成ファイルでアクセスが正しく構成されていることを確認してください。

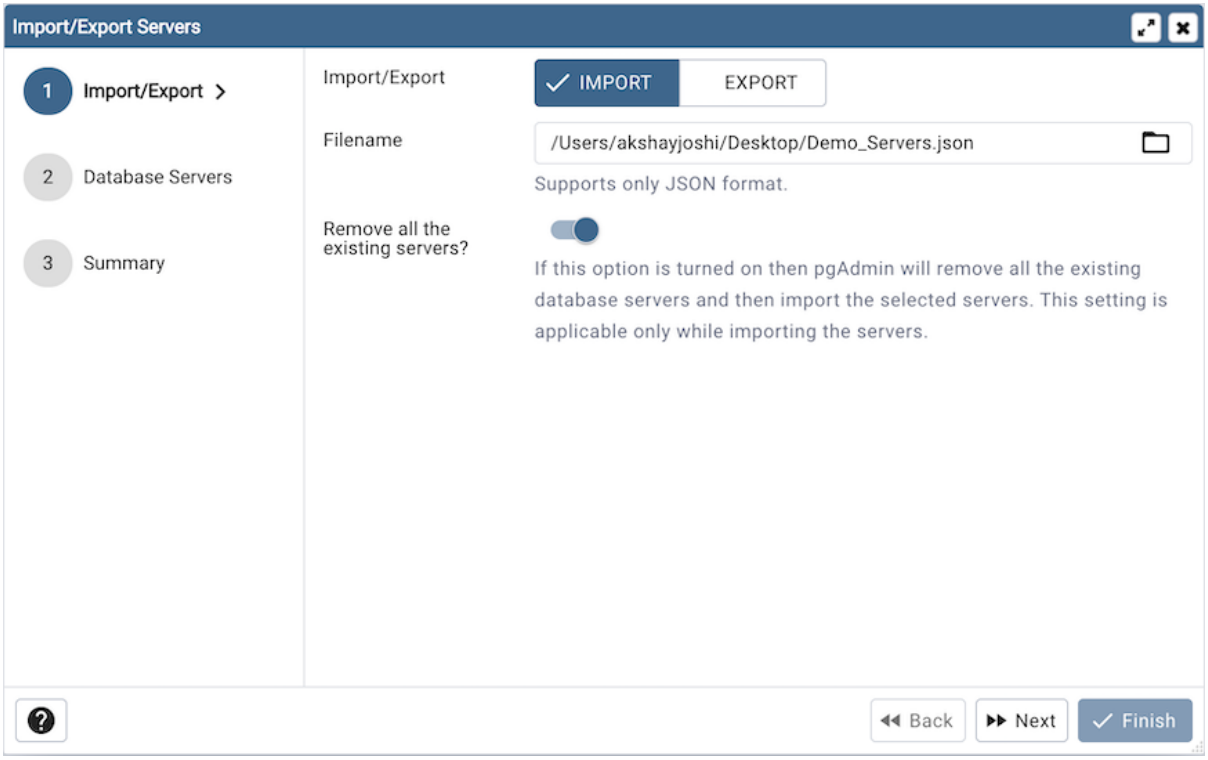
サーバー定義（およびそのグループ）を JSON ファイルにエクスポートし、同じまたは別のシステムに再インポートして、pgAdmin の簡単な事前構成を有効にできます。

3.7 Import/Export Servers

サーバー定義（およびそのグループ）を JSON ファイルにエクスポートし、同じまたは別のシステムに再インポートして、pgAdmin の簡単な事前構成を有効にできます。

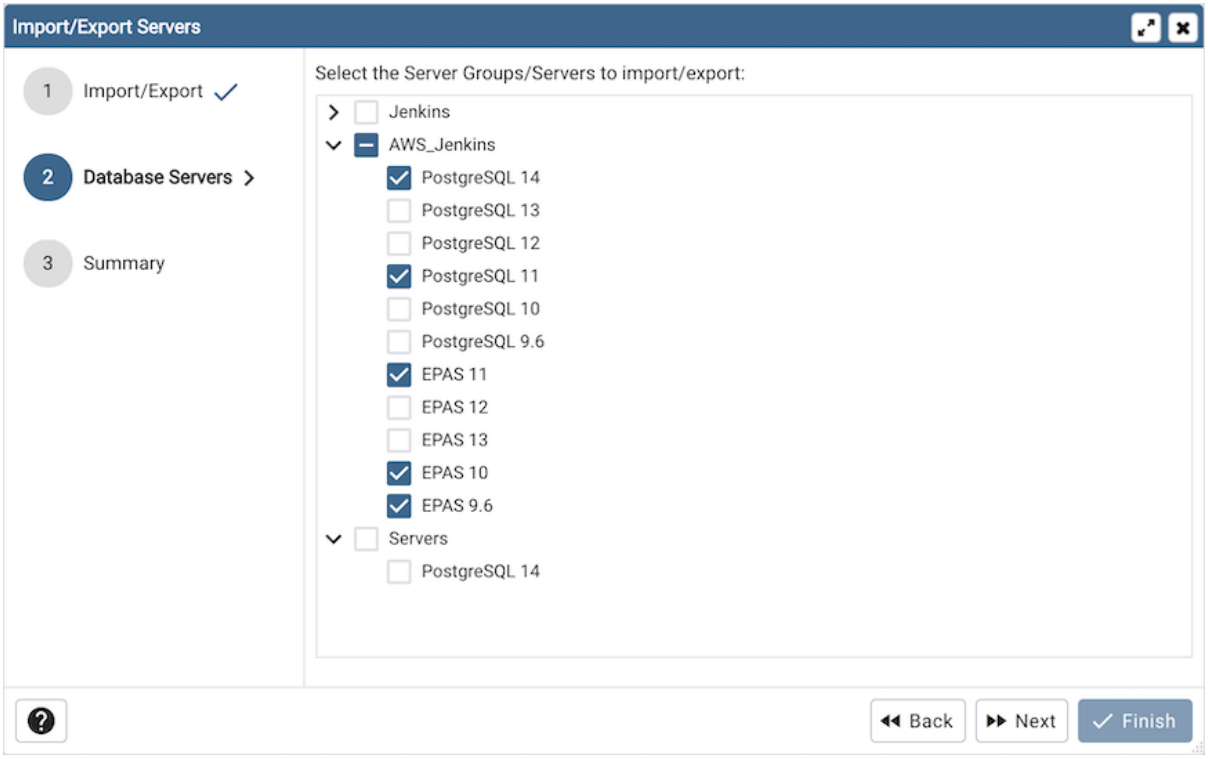
3.7.1 pgAdmin4GUI の使用

Import/ExportServers... ツールを起動するには、メニューバーの *Tools* に移動して、*Import/ExportServers* オプションをクリックします。



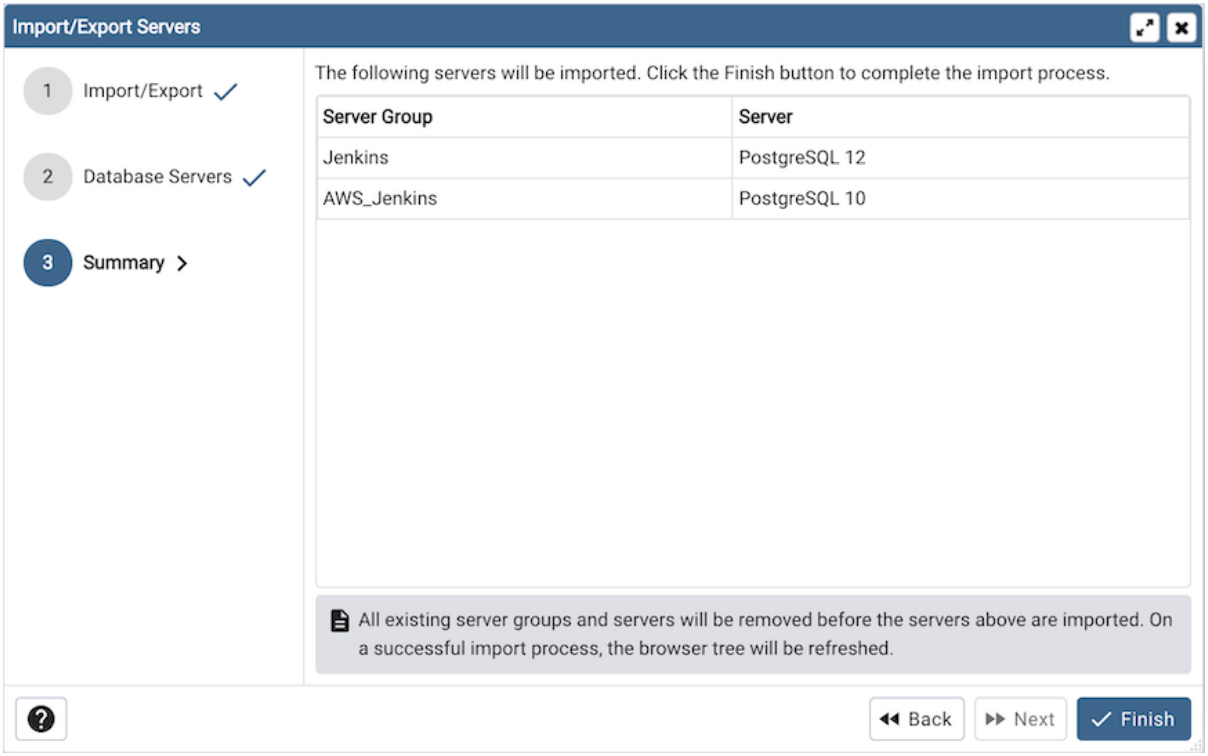
- *Import/Export* フィールドを使用して、インポートまたはエクスポートするサーバーグループ/サーバーを選択します。
- **ファイル名** フィールドを使用して、サーバーをインポートする JSON ファイルを選択するか、サーバーを JSON 形式でエクスポートするエクスポートの場合は新しいファイルを作成します。
- **既存のサーバーをすべて削除しますか**? フィールドを使用して、新しく選択したサーバーをインポートする前に既存のサーバーをすべて削除するかどうかを指定します。このフィールドは、インポートサーバーの場合にのみ適用されます。

[**次へ**] ボタンをクリックして続行するか、[*X*] ボタンをクリックしてウィザードを閉じます。



- インポート/エクスポートするサーバーグループ/サーバーを選択します。

[**次へ**] ボタンをクリックして続行するか、[*X*] ボタンをクリックしてウィザードを閉じます。



[概要] ページでインポート/エクスポートするサーバーの概要を確認します。

Finish ボタンをクリックしてウィザードを閉じます。

3.7.2 「setup.py」コマンドラインスクリプトの使用

注釈: setup.py スクリプトを使用してサーバーをエクスポートまたはインポートするには、通常 pgAdmin の実行に使用される Python インタープリターを使用して、必要な Python パッケージが利用可能であることを確認する必要があります。ほとんどのパッケージでは、これはインストールディレクトリにある PythonVirtualEnvironment にあります。プラットフォームネイティブパッケージを使用する場合、Python のシステムインストールが pgAdmin によって使用される場合があります。

サーバーのエクスポート

インストールで定義されたサーバーをエクスポートするには、 --dump-servers コマンドラインオプションに続けて名前（および必要な場合はパス）を指定して setup.py を呼び出します。デフォルトでは、デスクトップモードユーザーが所有するサーバーがダンプされます（デフォルトでは pgadmin4@pgadmin.org-config.py の DESKTOP_USER 設定を参照）。これは、 --user コマンドラインオプションでオーバーライドできます。同じシステムに pgAdmin の複数の構成がある場合があります。特定の pgAdmin 構成 DB ファイルからサーバーをダンプするには、 --sqlite-path オプションを使用できます。デスクトップモードで pgAdmin を実行するときにもこのオプションを使用することをお勧めします。デフォルトでは、config.py の SQLITE_PATH 設定が使用されます。例：

```
/path/to/python /path/to/setup.py --dump-servers output_file.json

# or, to specify a non-default user name:

/path/to/python /path/to/setup.py --dump-servers output_file.json --user user@example.
↪com

# to specify a pgAdmin config DB file:

/path/to/python /path/to/setup.py --dump-servers output_file.json --sqlite-path /path/
↪to/pgadmin4.db
```

特定のサーバーのみをエクスポートするには、 --servers オプションを使用して、1 つ以上のサーバー ID をリストします。例：

```
/path/to/python /path/to/setup.py --dump-servers output_file.json --server 1 2 5
```

サーバーのインポート

JSON ファイルで定義されたサーバーをインポートするには、 `--load-servers` コマンドラインオプションに続けて、サーバー定義を含む JSON ファイルの名前（および必要な場合はパス）を指定します。サーバーはデスクトップモードユーザーが所有します（デフォルトでは pgadmin4@pgadmin.org- `config.py` の DESKTOP_USER 設定を参照）。これは、 `--user` コマンドラインオプションでオーバーライドできます。同じシステムに pgAdmin の複数の構成がある場合があります。デフォルトの動作では、インポートされたサーバーは既存のリストに追加されます。これにより、重複が発生する場合があります。これは、サーバーのリストを新しくインポートしたものに置き換える `--replace` コマンドラインオプションでオーバーライドできます。サーバーを特定の pgAdmin 構成 DB ファイルにロードするには、 `--sqlite-path` オプションを使用できます。デスクトップモードで pgAdmin を実行するときにもこのオプションを使用することをお勧めします。デフォルトでは、 `config.py` の SQLITE_PATH 設定が使用されます。例：

```
/path/to/python /path/to/setup.py --load-servers input_file.json

# or, to replace the list of servers with the newly imported one:

/path/to/python /path/to/setup.py --load-servers input_file.json --replace

# or, to specify a non-default user name to own the new servers:

/path/to/python /path/to/setup.py --load-servers input_file.json --user user@example.
↪com

# to specify a pgAdmin config DB file:

/path/to/python /path/to/setup.py --load-servers input_file.json --sqlite-path /path/
↪to/pgadmin4.db
```

構成データベースにまだ存在しないサーバーグループでサーバーが定義されている場合、必要なグループが作成されます。

JSON 形式

サーバーをインポートまたはエクスポートするときに使用される JSON ファイル形式は非常に簡単で、多くの属性を持つサーバーのリストが含まれています。次の属性はすべてのサーバー定義に存在する必要があります。Name、Group、Port、Username、SSLMode、MaintenanceDB と、Host、HostAddr、または Service のいずれか。

パスワードフィールドはインポートまたはエクスポートできません。

次の例は、最小限に定義されたサーバーと完全に定義されたサーバーの両方を示しています。

```
{
    "Servers": {
        "1": {
```

(次のページに続く)

(前のページからの続き)

```
            "Name": "Minimally Defined Server",
            "Group": "Server Group 1",
            "Port": 5432,
            "Username": "postgres",
            "Host": "localhost",
            "SSLMode": "prefer",
            "MaintenanceDB": "postgres"
        },
        "2": {
            "Name": "Fully Defined Server",
            "Group": "Server Group 2",
            "Host": "host.domain.com",
            "HostAddr": "192.168.1.2",
            "Port": 5432,
            "MaintenanceDB": "postgres",
            "Username": "postgres",
            "Role": "my_role_name",
            "SSLMode": "require",
            "Comment": "This server has every option configured in the JSON",
            "DBRestriction": "live_db test_db",
            "PassFile": "/path/to/pgpassfile",
            "SSLCert": "/path/to/sslcert.crt",
            "SSLKey": "/path/to/sslcert.key",
            "SSLRootCert": "/path/to/sslroot.crt",
            "SSLCrl": "/path/to/sslcrl.crl",
            "SSLCompression": 1,
            "Shared": false,
            "BGColor": "#ff9900",
            "FGColor": "#000000",
            "Service": "postgresql-10",
            "Timeout": 60,
            "UseSSHTunnel": 1,
            "TunnelHost": "192.168.1.253",
            "TunnelPort": 22,
            "TunnelUsername": "username",
            "TunnelAuthentication": 0
        }
    }
}
```

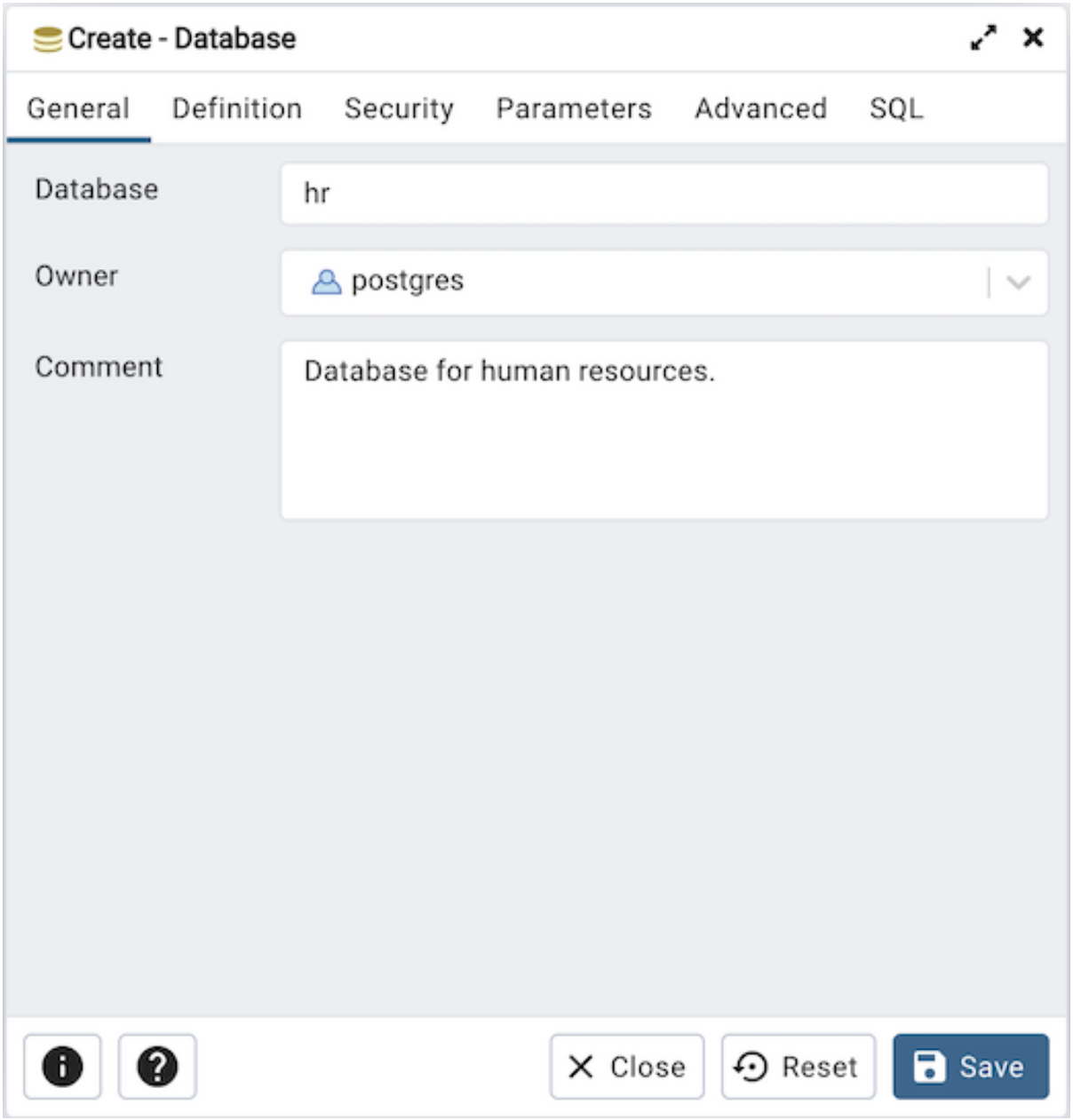
第4章　Managing Cluster Objects

一部のオブジェクト定義はクラスターレベルにあります。pgAdmin4 は、これらのオブジェクトを作成、管理、および相互の関係を制御できるダイアログを提供します。データベースオブジェクトを作成できるダイアログにアクセスするには、pgAdmin ツリーコントロールでオブジェクトタイプを右クリックし、そのオブジェクトの *Create* オプションを選択します。たとえば、新しいデータベースを作成するには、*Databases* ノードを右クリックし、 *CreateDatabase...* を選択します。

4.1 Database Dialog

データベース ダイアログを使用して、データベースを定義または変更します。データベースを作成するには、データベースのスーパーユーザーであるか、CREATE 権限を持っている必要があります。

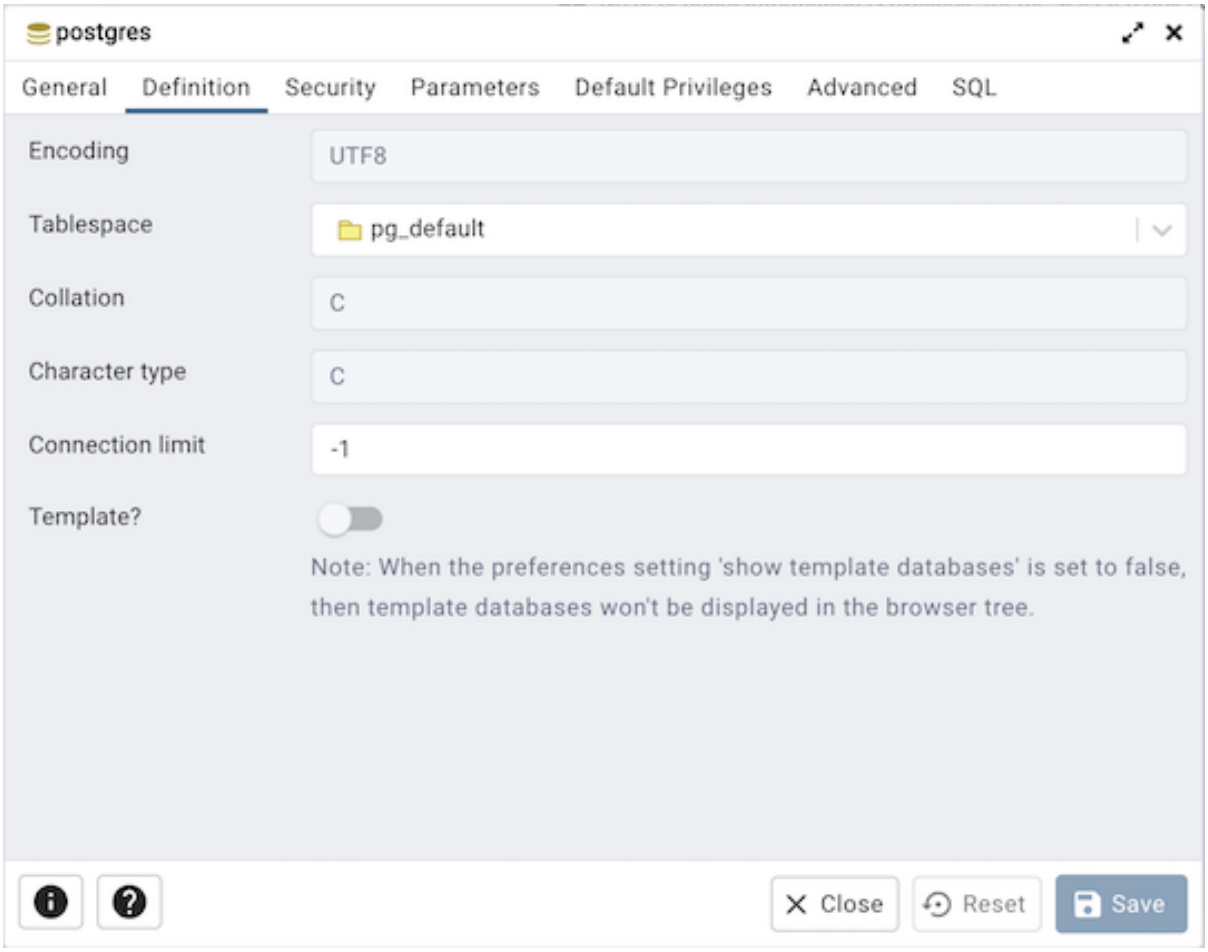
データベース ダイアログは、 **一般** 、 **定義** 、 **セキュリティ** 、および **パラメーター** のダイアログタブを使用してデータベースの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、データベースを識別します。

- *Database* フィールドを使用して、データベースの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** フィールドのドロップダウンリストボックスからデータベースの所有者を選択します。
- データベースに関するメモを **コメント** フィールドに保存します。

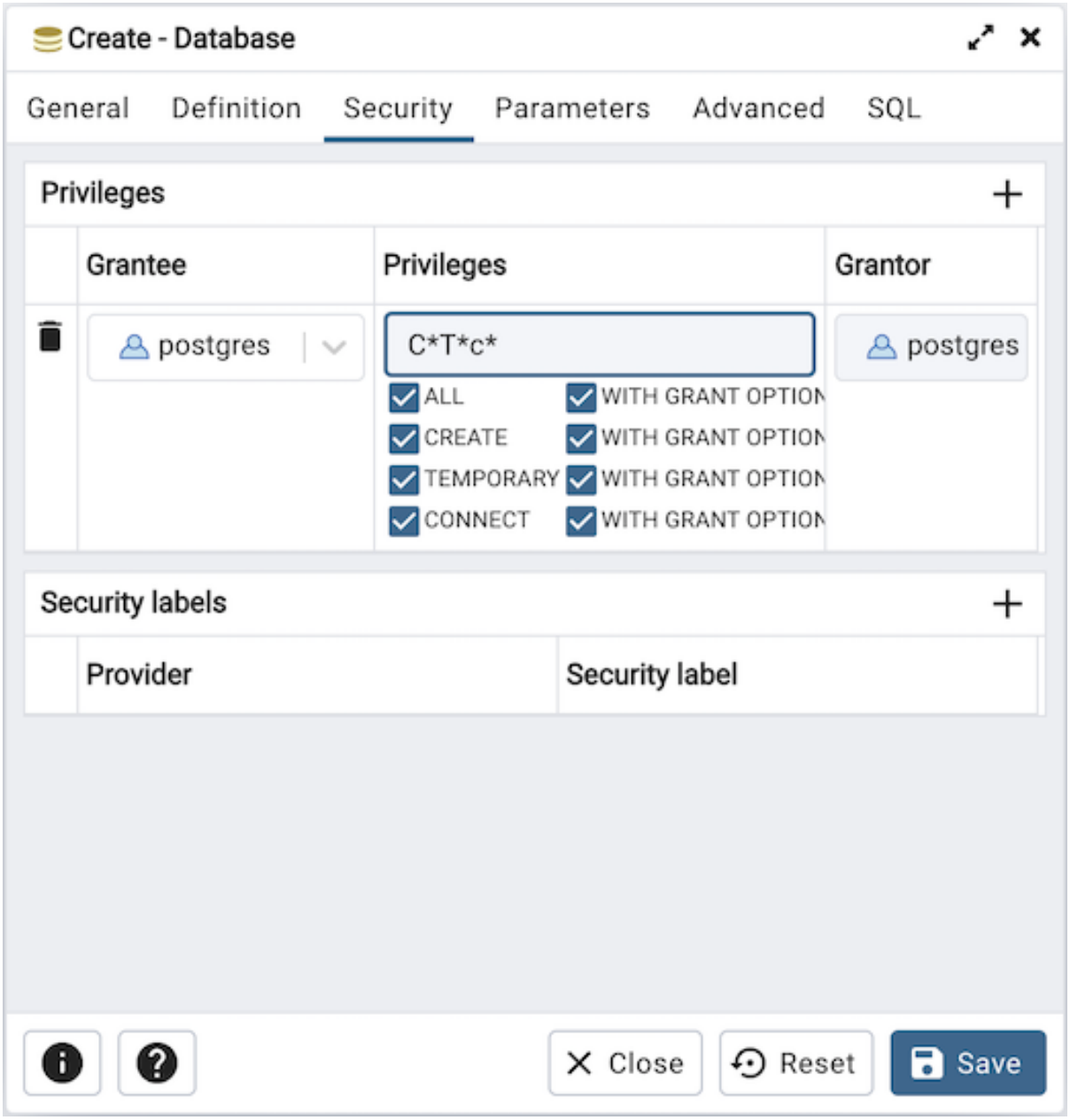
[**定義**] タブをクリックして続行します。



[**定義**] タブを使用して、データベースのプロパティを設定します。

- **エンコーディング** フィールドのドロップダウンリストボックスから文字セットを選択します。デフォルトは *UTF8* です。
- **テンプレート** フィールドのドロップダウンリストボックスからテンプレートを選択します。テンプレートを指定しない場合、データベースは template1 を使用します。
- *Tablespace* フィールドのドロップダウンリストボックスからテーブルスペースを選択します。選択した表領域は、データベースオブジェクトを含めるために使用されるデフォルトの表領域になります。
- **照合順序** フィールドのドロップダウンリストボックスから照合順序を選択します。
- *CharacterType* フィールドのドロップダウンリストボックスから文字分類を選択します。これは、文字の分類に影響します。下、上、数字。デフォルト、または空白のフィールドでは、テンプレートデータベースの文字分類を使用します。
- *ConnectionLimit* フィールドに接続制限を指定して、接続要求の最大数を構成します。デフォルト値（ *-1* ）では、データベースへの無制限の接続が許可されます。
- *templates?* が true に設定されている場合、database はテンプレートデータベースになります。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、データベースオブジェクトの特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

add をクリックして追加の特権を設定します。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

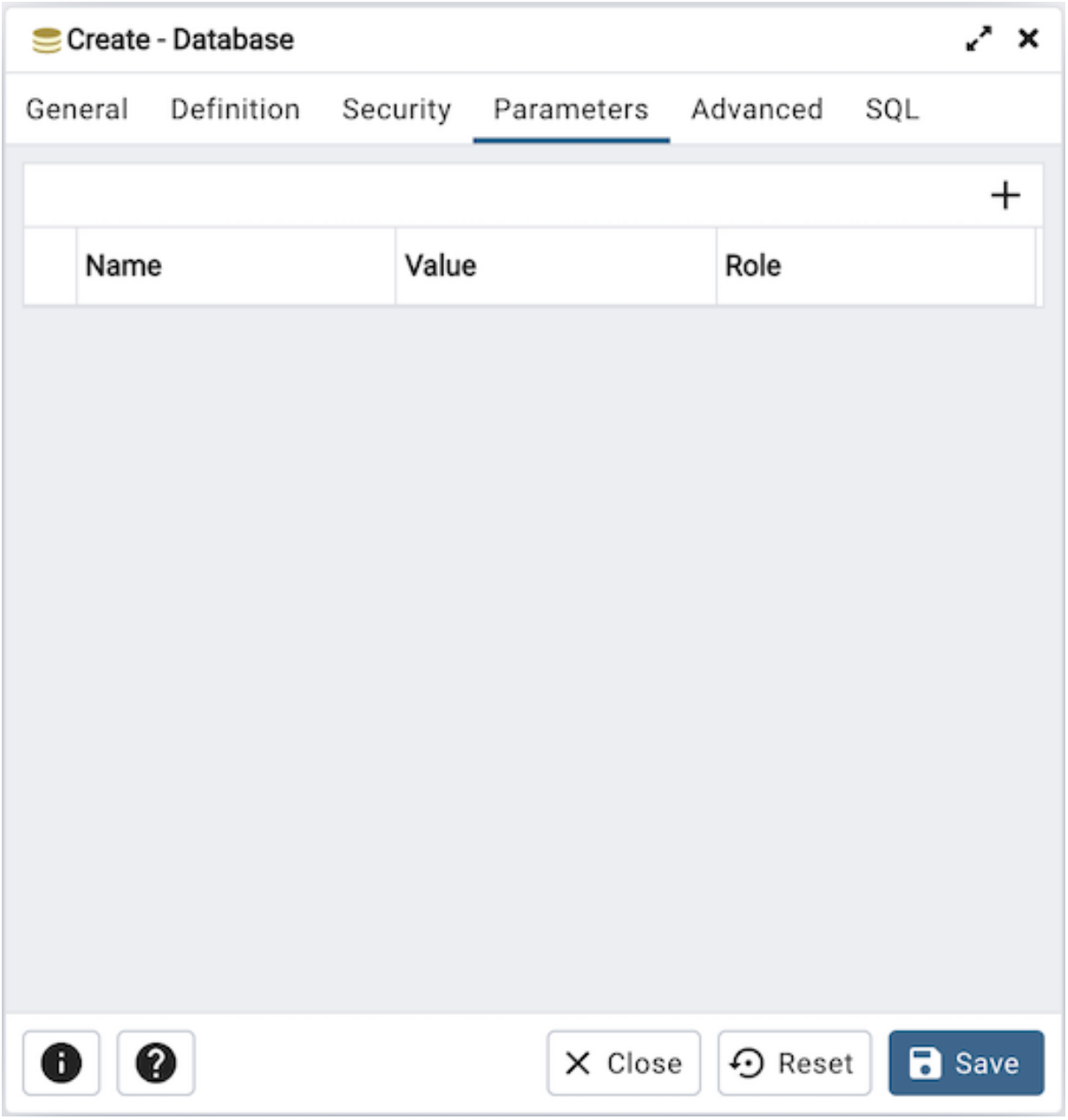
セキュリティラベル パネルを使用して、データベースに適用するセキュリティラベルを定義します。 **追加**

アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

[**パラメーター**] タブをクリックして続行します。

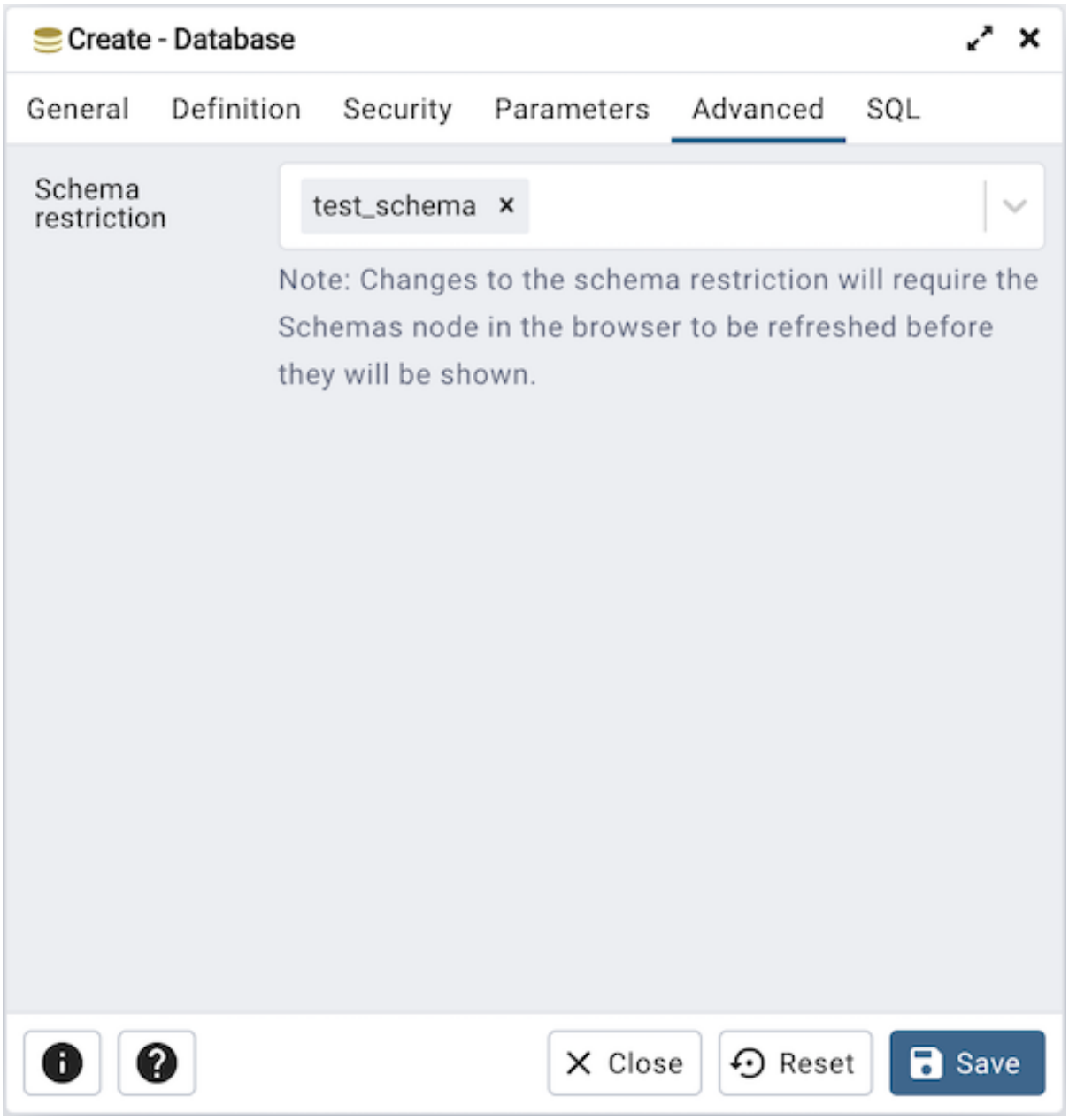


[**パラメーター**] タブを使用して、データベースのパラメーターを設定します。 **追加** アイコン（+）をクリックして、各パラメーターを追加します。

- **名前** フィールドのドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Value* フィールドを使用して、パラメーターの値を設定します。
- **ロール** の横にあるドロップダウンリストボックスを使用して、指定したパラメーター設定を適用するロールを選択します。

次の手順に従って、パラメーター値の定義を追加します。パラメーターを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**詳細**] タブをクリックして続行します。



[**詳細**] タブを使用して、データベースの詳細パラメーターを設定します。

- **スキーマ制限** フィールドを使用して、表示されるスキーマを制限するために pg_namespace テーブルに対して使用される SQL 制限を提供します。たとえば、次のように入力します。 *public* のみが pgAdmin ブラウザに表示されるようにします。入力するときはエントリをコンマまたはタブで区切

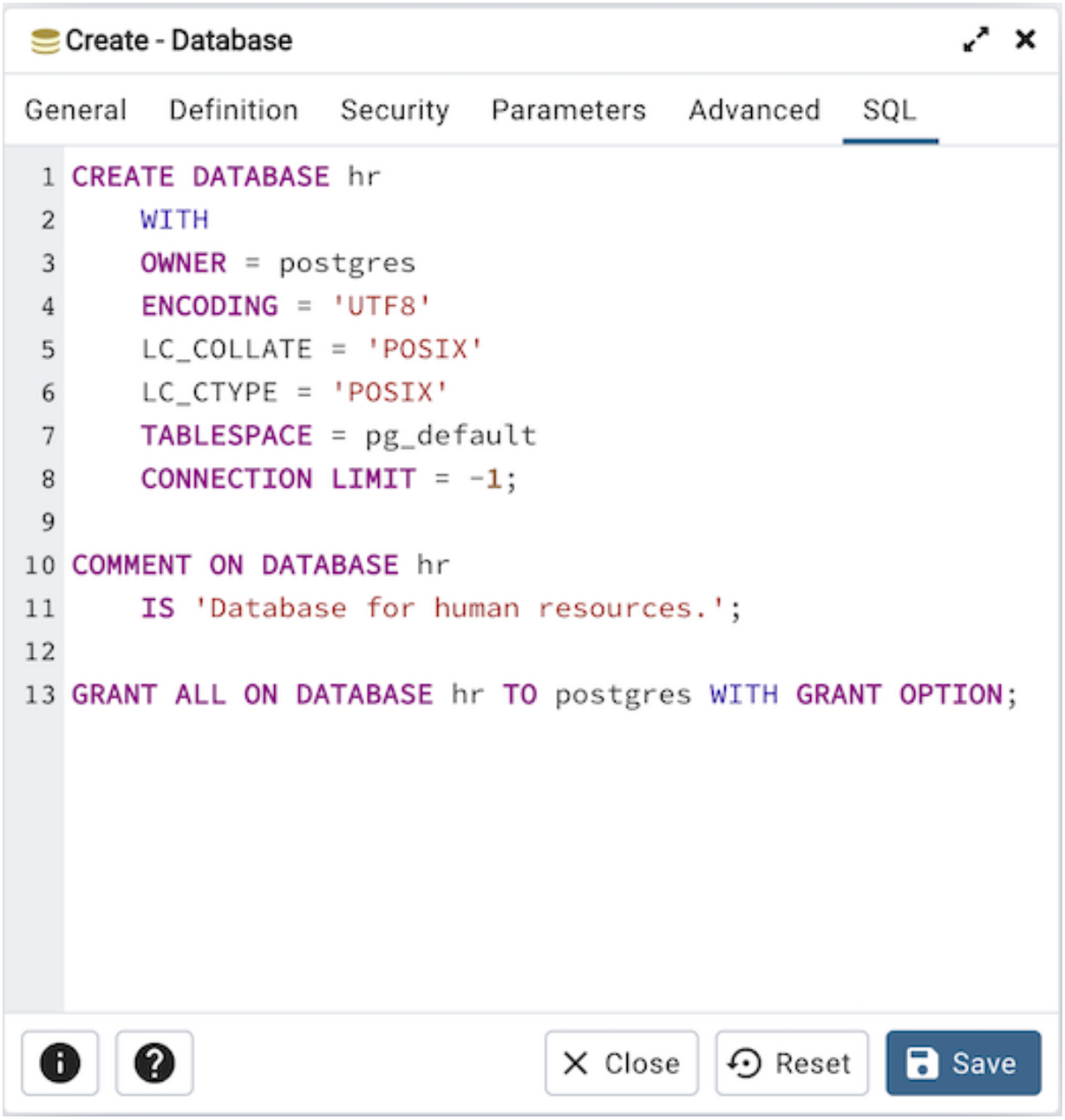
ります。

SQL タブをクリックして続行します。

データベース ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

4.1.1 例

以下は、 **データベース** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



この例では、 *postgres* が所有する *hr* という名前のデータベースを作成します。無制限の接続を許可し、すべての認証済みユーザーが利用できます。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。

- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

4.2 Resource Group Dialog

[**リソースグループ**] ダイアログを使用して、リソースグループを作成し、そのリソースの値を設定します。リソースグループは、さまざまなリソース使用量の制限を定義できる名前付けのグローバルグループです。リソースグループには、クラスター内のすべてのデータベースからアクセスできます。 **リソースグループ** ダイアログを使用するには、スーパーユーザー権限が必要です。EDBPostgresAdvancedServer に接続するとリソースグループがサポートされることに注意してください。リソースグループの使用の詳細については、次の場所にある EDBPostgresAdvancedServer ガイドを参照してください。

http://www.enterprisedb.com/

リソースグループの作成に使用されるフィールドは、[**全般**] タブにあります。 *SQL* タブには、 **リソースグループ** ダイアログでの選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、リソースグループの属性を指定します。

- *Name* フィールドを使用して、リソースグループの説明的な名前を追加します。この名前はツリーコントロールに表示されます。
- *CPU* **レート制限** *(%)* フィールドを使用して、リソースグループに割り当てられる CPU レート制限リソースタイプの値を設定します。CPU レート制限の有効な範囲は 0〜1.67772e+07 です。デフォルト値は 0 です。
- **ダーティレート制限（***KB***）** フィールドを使用して、リソースグループに割り当てられるダーティレート制限リソースタイプの値を設定します。ダーティレート制限の有効な範囲は 0〜1.67772e+07 です。デフォルト値は 0 です。

SQL タブをクリックして続行します。

リソースグループ ダイアログに入力すると、SQL コマンドが生成されます。確認には *SQL* タブを使用します。[**全般**] タブに再度アクセスして、SQL コマンドを変更します。

4.2.1 例

以下は、 **リソースグループ** ダイアログでの選択によって生成される sql コマンドの例です。



この例では、 *cpu_rate_limit* を *2* に設定し、 *dirty_rate_limit* を *6144* に設定する *acctg* という名前のリソースグループを作成します。

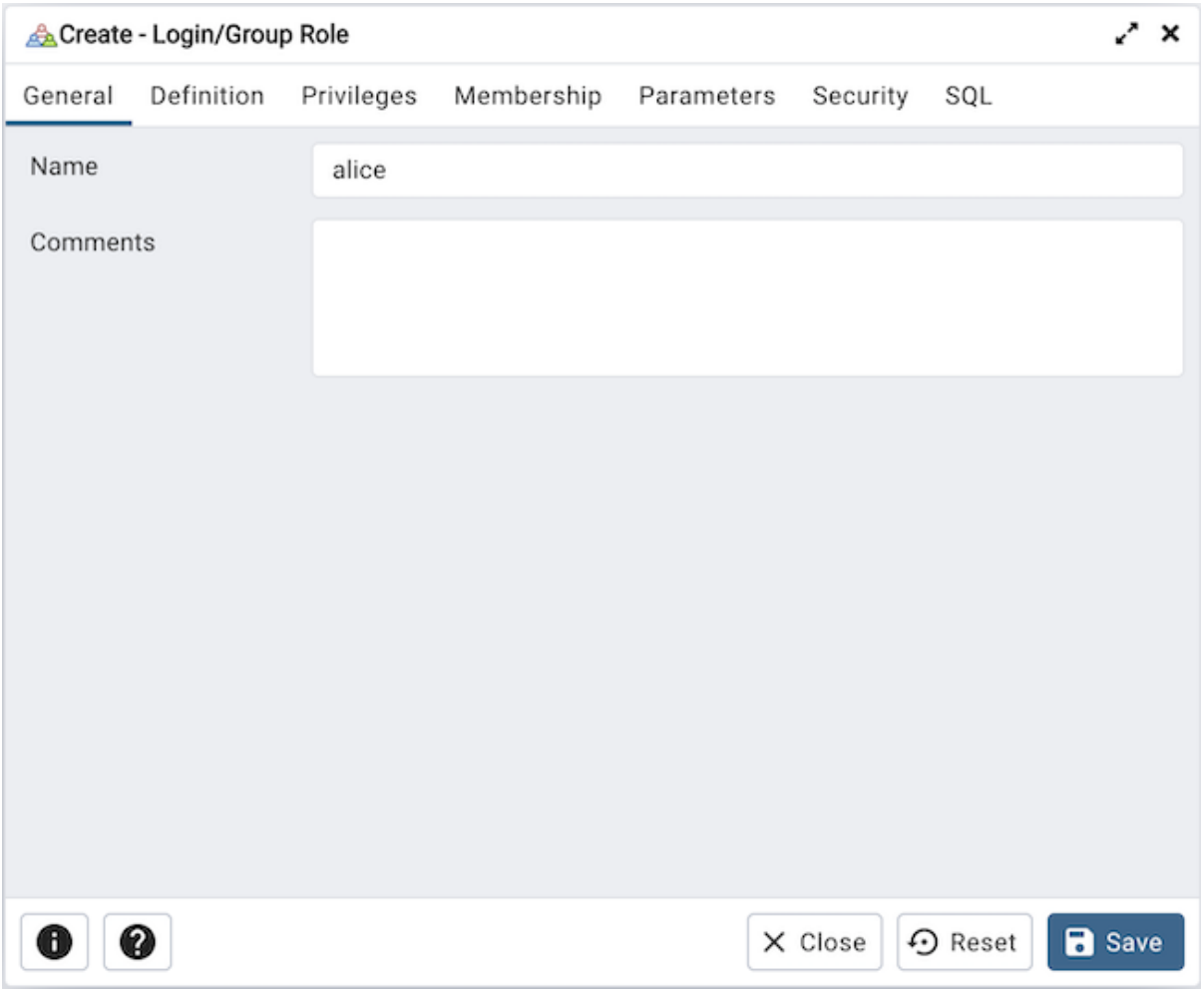
- Info ボタン（ *i* ）をクリックして、オンラインの SQL 構文リファレンスにアクセスします。
- Help ボタン（ *?* ）をクリックして、リソースグループに関するオンラインドキュメントにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。

- *Reset* ボタンをクリックして、構成パラメーターを復元します。

4.3 Login/Group Role Dialog

Login/GroupRole ダイアログを使用して、ロールを定義します。ロールは、個々のユーザー（ログイン権限の有無にかかわらず）またはユーザーのグループです。クラスターレベルで定義されたロールは、クラスター内のすべてのデータベースで共有されることに注意してください。

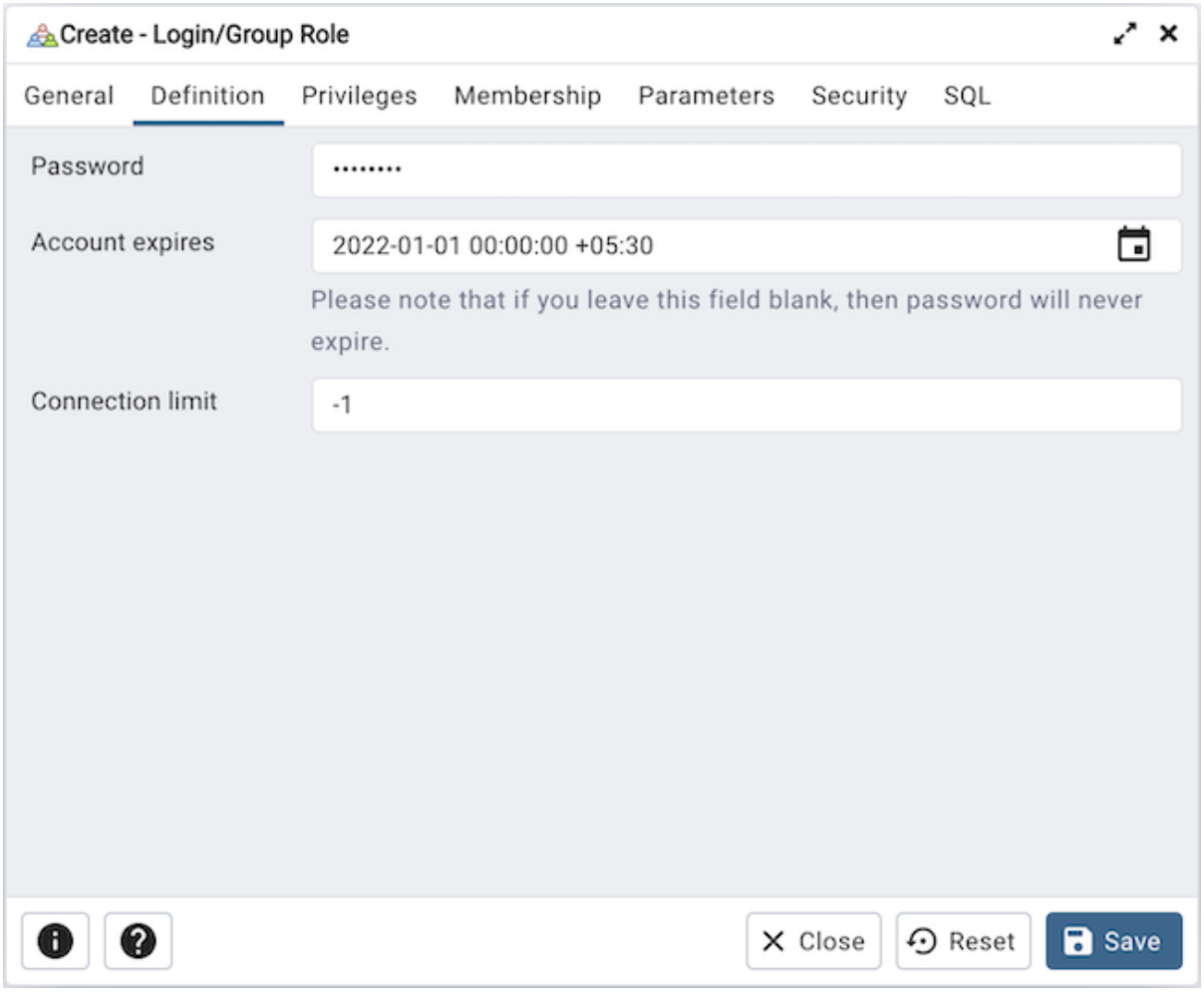
ログイン/グループロール ダイアログは、次のダイアログタブを使用してロールの作成と管理を編成します。 **一般** 、 **定義** 、 **特権** 、 **メンバーシップ パラメーター** 、および **セキュリティ** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、ロールを識別します。

- *Name* フィールドを使用して、ロールの名前を指定します。名前がツリーコントロールに表示されます。
- **コメント** フィールドにロールに関するメモを入力します。

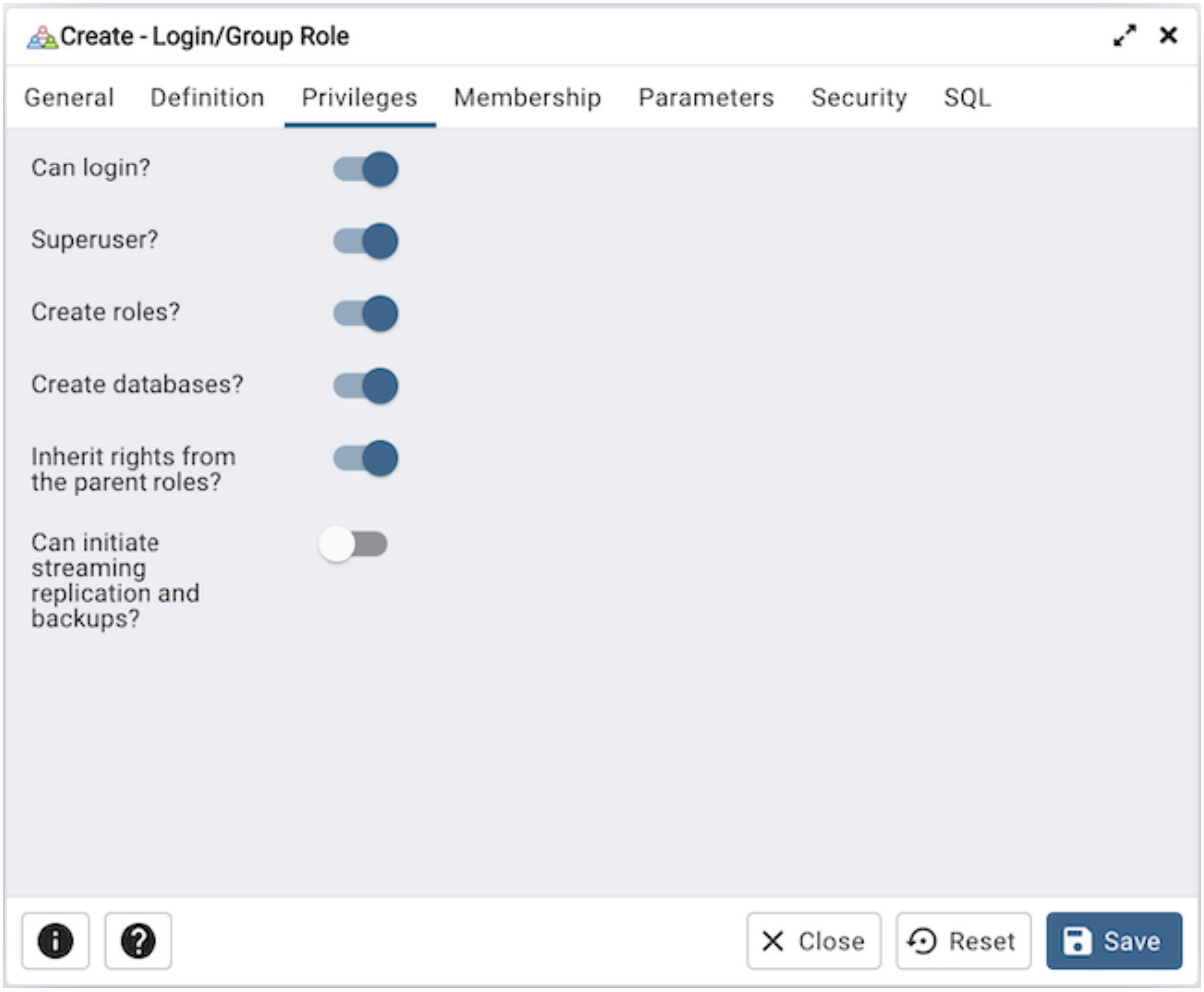
[**定義**] タブをクリックして続行します。



[**定義**] タブを使用して、パスワードを設定し、接続ルールを構成します。

- ロールに関連付けられるパスワードを *Password* フィールドに入力します。
- **アカウントの有効期限** フィールドにパスワードの有効期限を入力します（ロールに有効期限はありません）。ユーザーがパスワードベース以外の認証方法でログインする場合、有効期限は適用されません。
- ロールがログインロールの場合、ロールが作成できる同時接続数を *ConnectionLimit* フィールドに指定します。デフォルト値（ *-1* ）では、無制限の接続が許可されます。

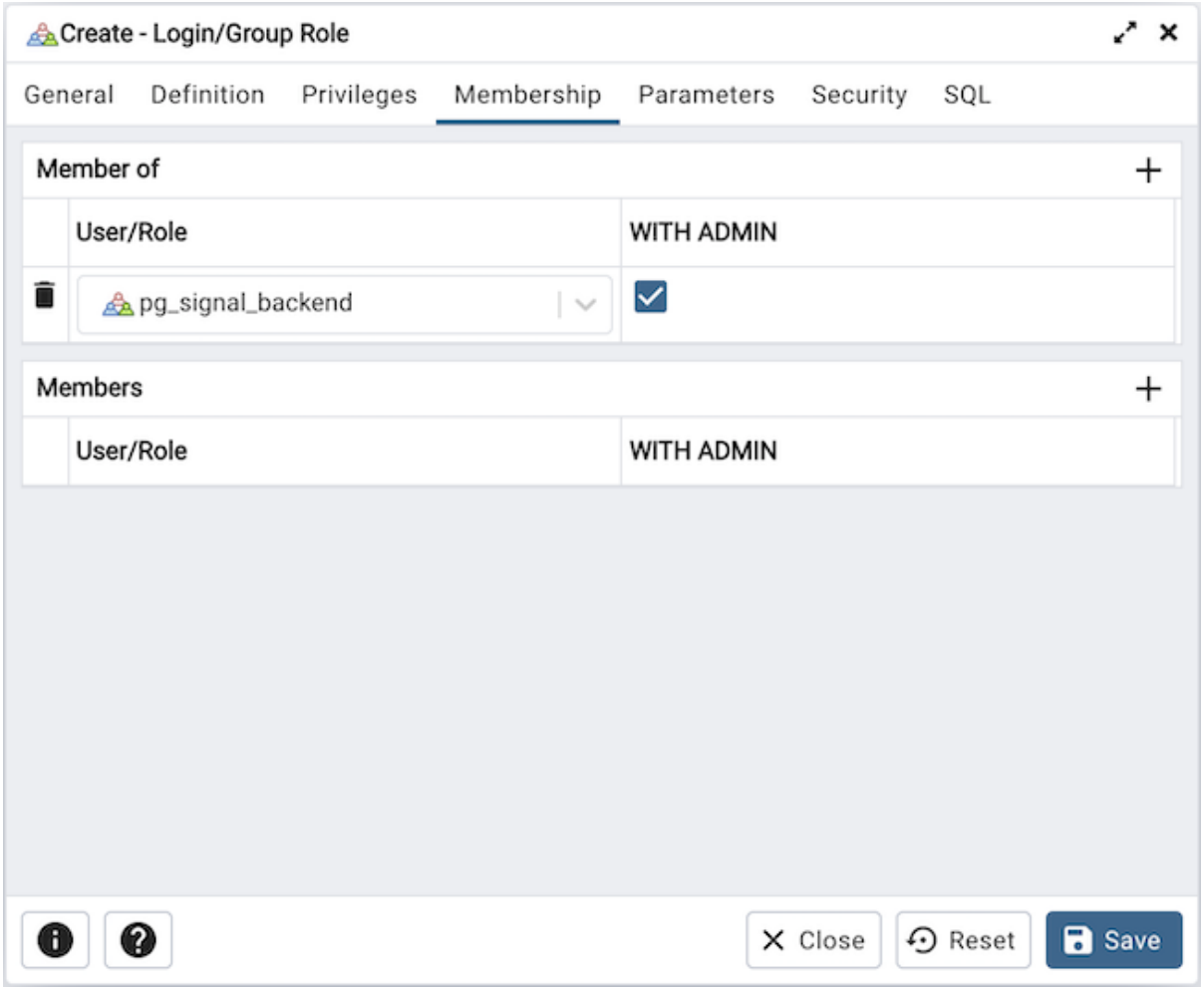
[**特権**] タブをクリックして続行します。



[*Privileges*] タブを使用して、ロールに特権を付与します。

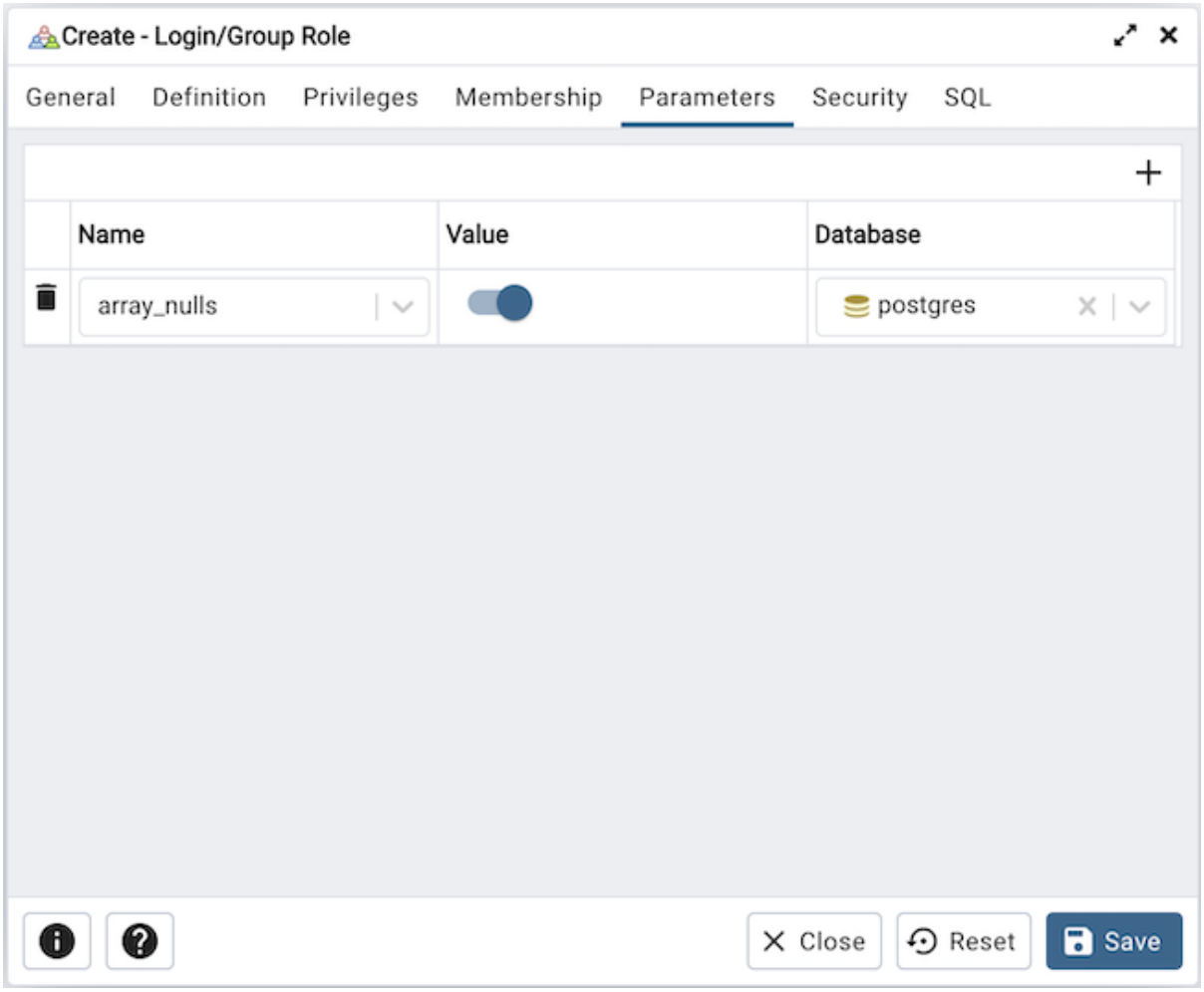
- ロールにログイン権限がある場合は、 *Canlogin?* スイッチを *Yes* の位置に移動します。デフォルト値は **いいえ** です。
- ロールがデータベース内のスーパーユーザーの場合は、 **スーパーユーザー** スイッチを **はい** の位置に移動します。デフォルト値は **いいえ** です。
- *Createroles?* スイッチを *Yes* の位置に移動して、ロールにロールの作成を許可するかどうかを指定します。この権限を持つロールは、ロールを変更および削除できます。デフォルト値は **いいえ** です。
- **データベースの作成** スイッチを **はい** の位置に移動して、ロールがデータベースを作成できるかどうかを制御します。デフォルト値は **いいえ** です。
- ロールにスーパーユーザ権限が付与されるまで、 *Updatecatalog?* スイッチは無効になっています。 **カタログを更新しますか**? スイッチを **いいえ** の位置に移動して、ロールがカタログを更新できるかどうかを制御します。 **スーパーユーザー** スイッチが **はい** の位置にある場合、デフォルト値は **はい** です。
- ロールが特権を継承しない場合は、 **親ロールから権限を継承しますか**? スイッチを **いいえ** の位置に移動します。デフォルト値は **はい** です。
- **ストリーミングレプリケーションとバックアップを開始できますか**? スイッチを **はい** の位置に移動して、ロールがストリーミングレプリケーションを開始できるか、システムをバックアップモードに

切り替えられるかを制御します。デフォルト値は **いいえ** です。



- *Memberof* フィールドでロールのメンバーを指定し、 *Member* フィールドでメンバーを指定します。ロール名の右側にあるチェックボックスをオンにして、各選択を確認します。ロール名の左側にある *x* をクリックして選択を削除します。メンバーシップは、指定されたロールに付与された特権を各メンバーに伝えます。

[**パラメーター**] タブをクリックして続行します。

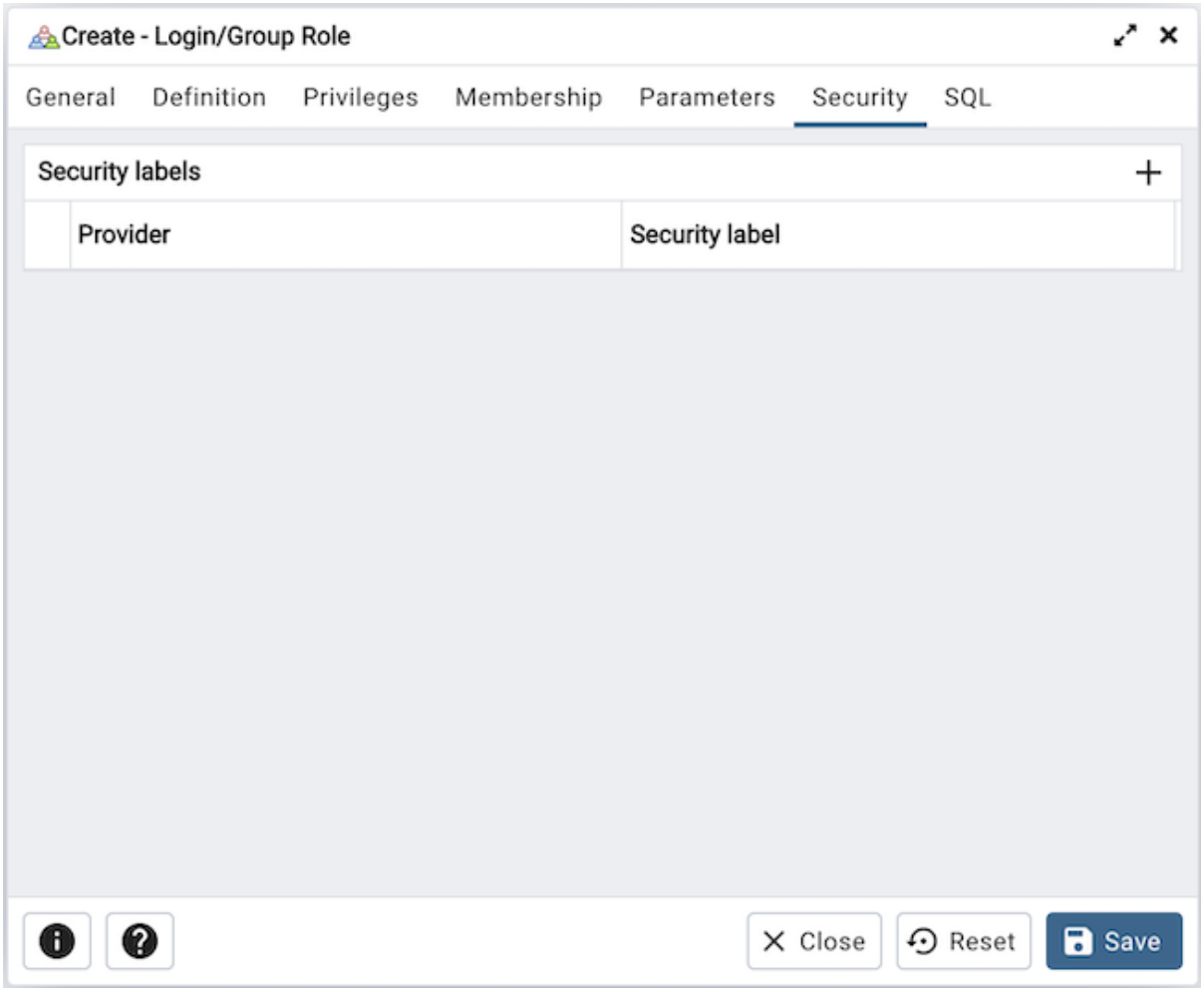


[**パラメーター**] タブのフィールドを使用して、ロールが指定されたデータベースに接続されているときに、選択した構成パラメーターのセッションのデフォルトを設定します。このタブは、ALTER-ROLE...SETconfiguration_parameter 構文を呼び出します。 **追加** アイコン（+）をクリックして、パラメーターに値を割り当てます。

- **名前** フィールドのドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Value* フィールドを使用して、パラメーターの値を指定します。
- **データベース** フィールドのドロップダウンリストボックスを使用して、データベースを選択します。

追加 アイコン（+）をクリックして、追加のパラメーターを指定します。パラメーターを破棄するには、行の左側にあるごみ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、ロールに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

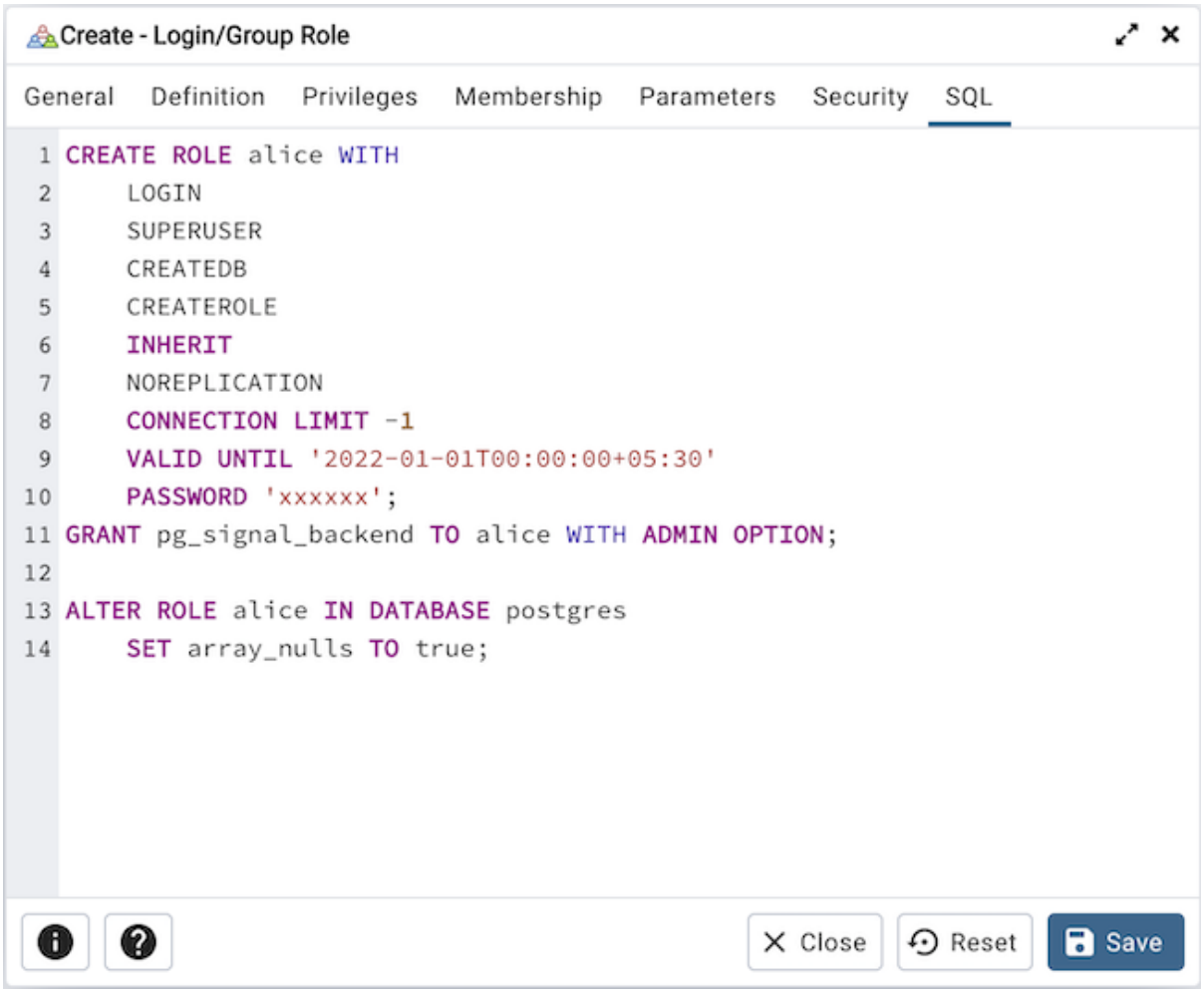
セキュリティラベルを破棄するには、行の左側にあるごみ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

ログイン/グループロール ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

4.3.1 例

以下は、**ログイン/グループロール** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



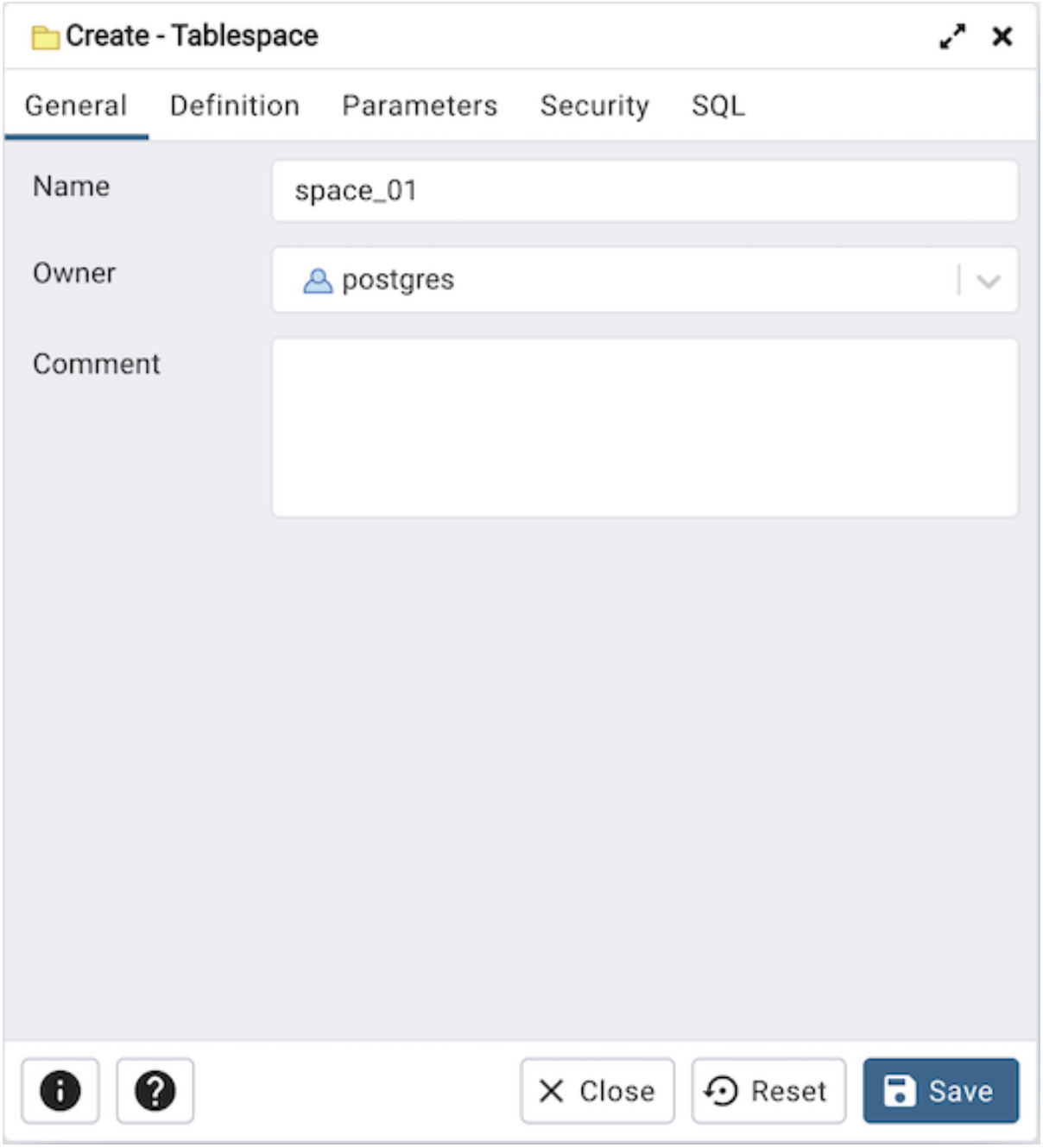
この例では、 *pg_signal_backend* 特権を持つ *alice* という名前のログインロールを作成します。このロールは、いつでもサーバーへの無制限の接続を確立できます。

- Info ボタン（ *i* ）をクリックして、オンライン SQL ヘルプにアクセスします。
- Help ボタン（ *?* ）をクリックして、ダイアログのドキュメントにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

4.4 Tablespace Dialog

Tablespace ダイアログを使用して、テーブルスペースを定義します。スーパーユーザーは、テーブルスペースを使用すると、データベースオブジェクト（テーブルやインデックスなど）を含むデータファイルが存在するファイルシステム上の代替の場所を定義できます。表領域は、シンボリックリンクをサポートするシステムでのみサポートされています。表領域は、それが定義されているクラスターとは無関係に使用できないことに注意してください。

Tablespace ダイアログは、 *General* 、 *Definition* 、 *Parameters* 、および *Security* のタブを介してテーブルスペースの定義を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



- *Name* フィールドを使用して、わかりやすい名前でテーブルスペースを識別します。名前を pg/_で始

めることはできません。これらの名前はシステムテーブルスペース用に予約されています。

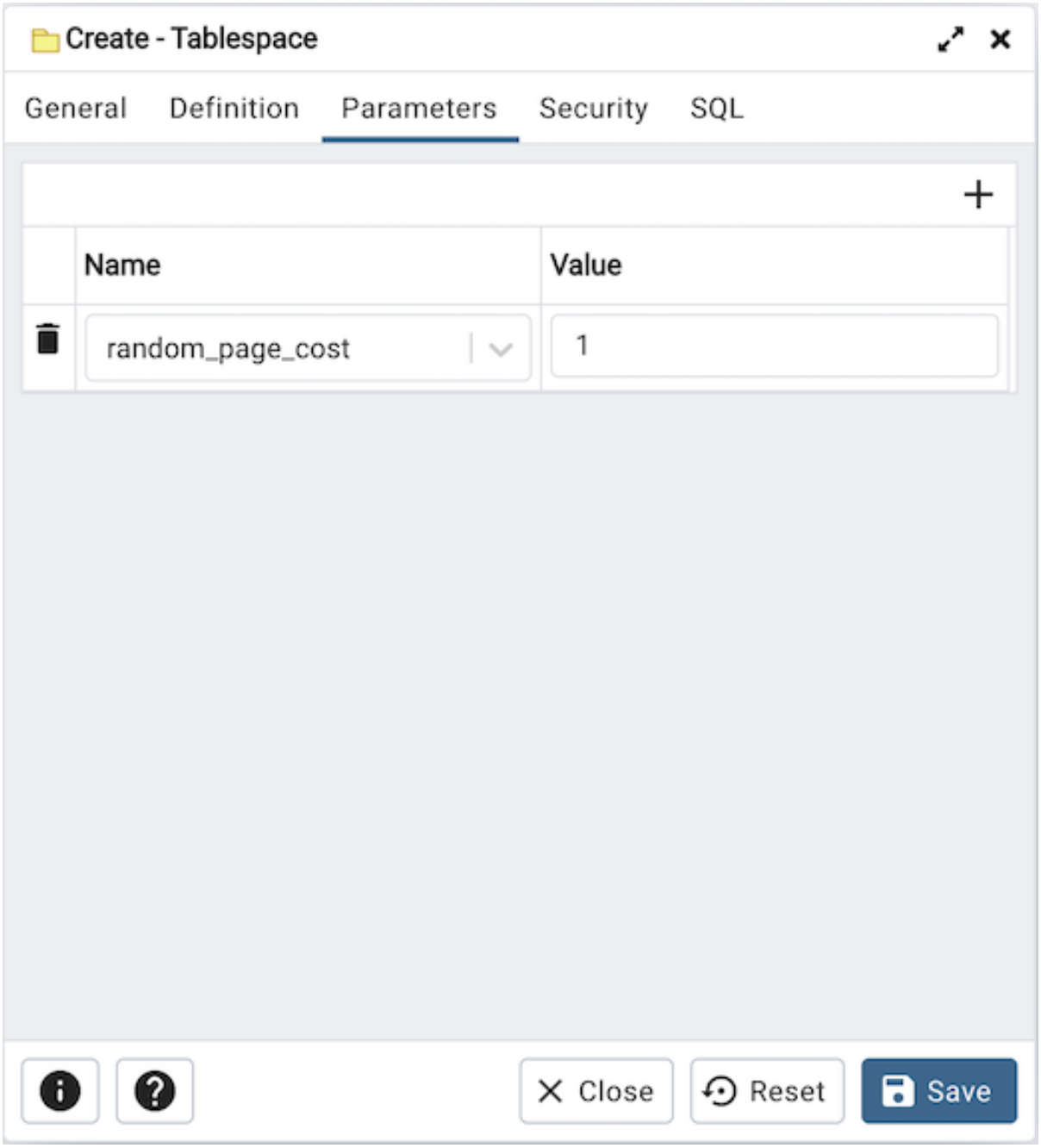
- **所有者** フィールドのドロップダウンリストボックスからテーブルスペースの所有者を選択します。
- **コメント** フィールドにテーブルスペースに関するメモを保存します。

[**定義**] タブをクリックして続行します。



- *Location* フィールドを使用して、テーブルスペースを含むディレクトリへの絶対パスを指定します。

[**パラメーター**] タブをクリックして続行します。

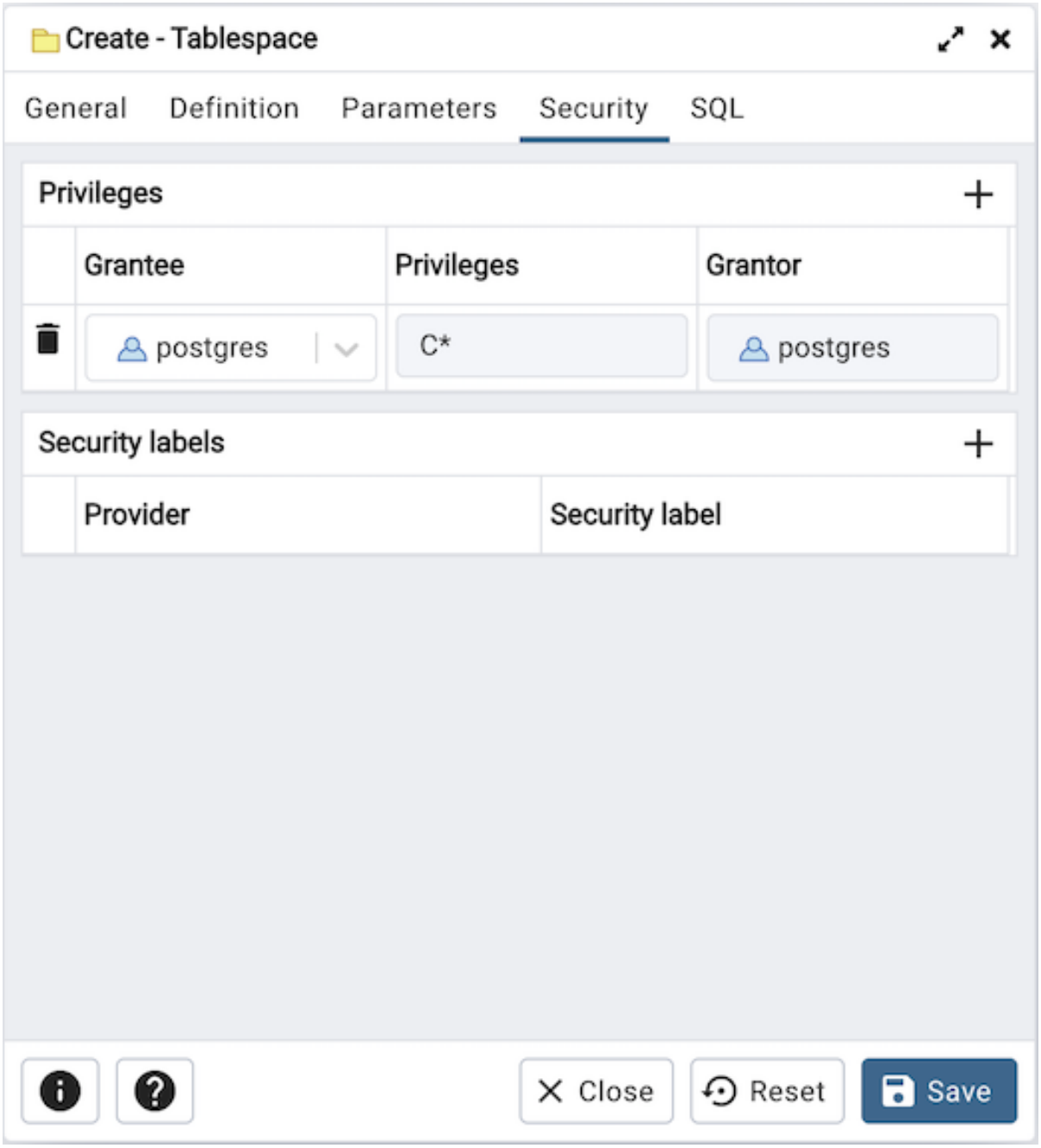


[**パラメーター**] タブを使用して、テーブルスペースのパラメーターを設定します。 **追加** アイコン（+）をクリックして、下のテーブルに行を追加します。

- **名前** の横にあるドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Value* フィールドを使用して、パラメーターの値を設定します。

追加 アイコン（+）をクリックして、追加のパラメーターを指定します。パラメーターを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 **行の削除** ダイアログで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、テーブルスペースのセキュリティラベルを定義します。

[*Privileges*] パネルを使用して、セキュリティ特権を割り当てます。 **追加** アイコン（+）をクリックして、一連の特権を割り当てます。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。

追加 アイコンをクリックして、追加の特権セットを割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、表領域に適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

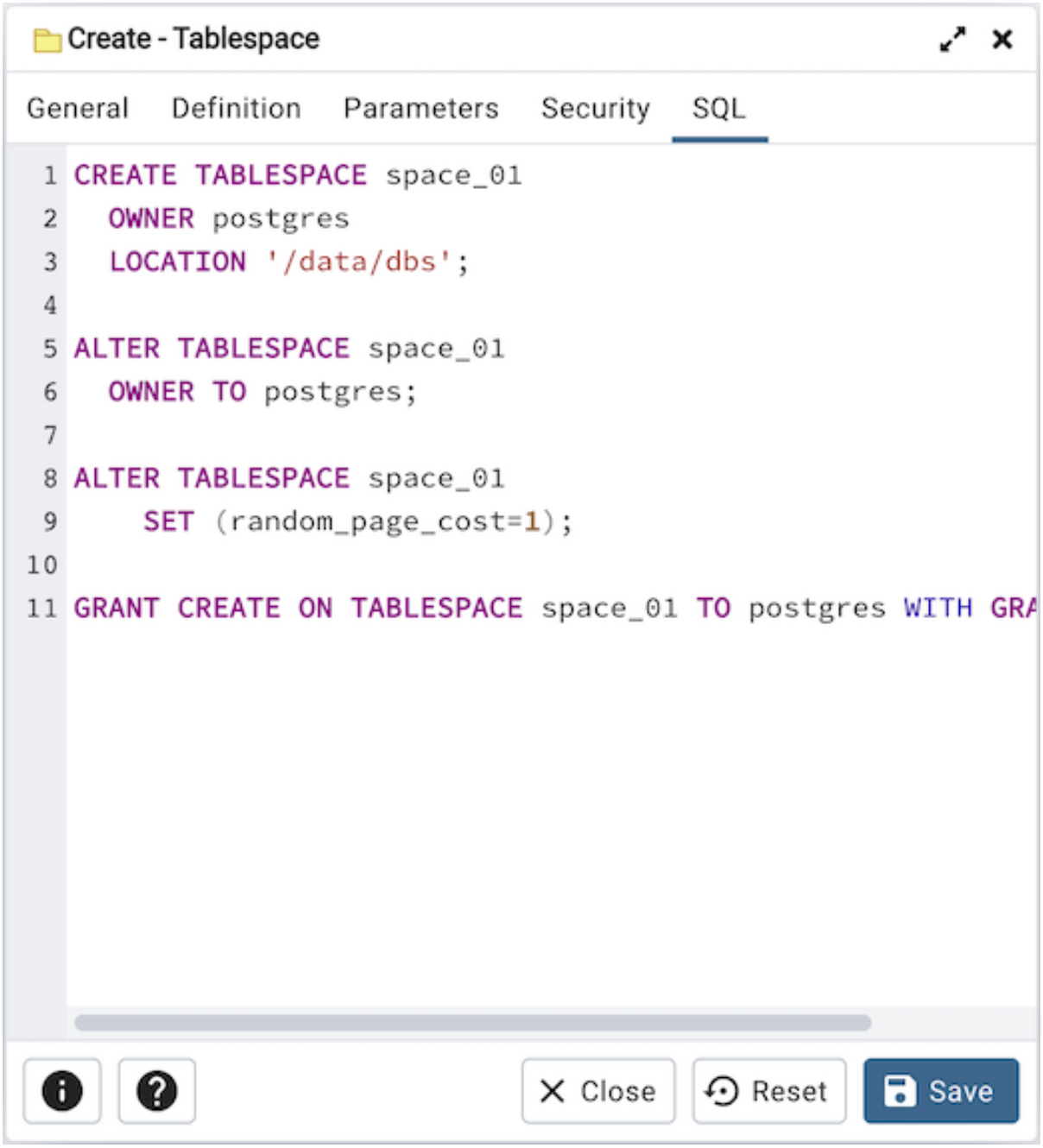
セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Tablespace ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

4.4.1 例

以下は、 **テーブルスペース** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



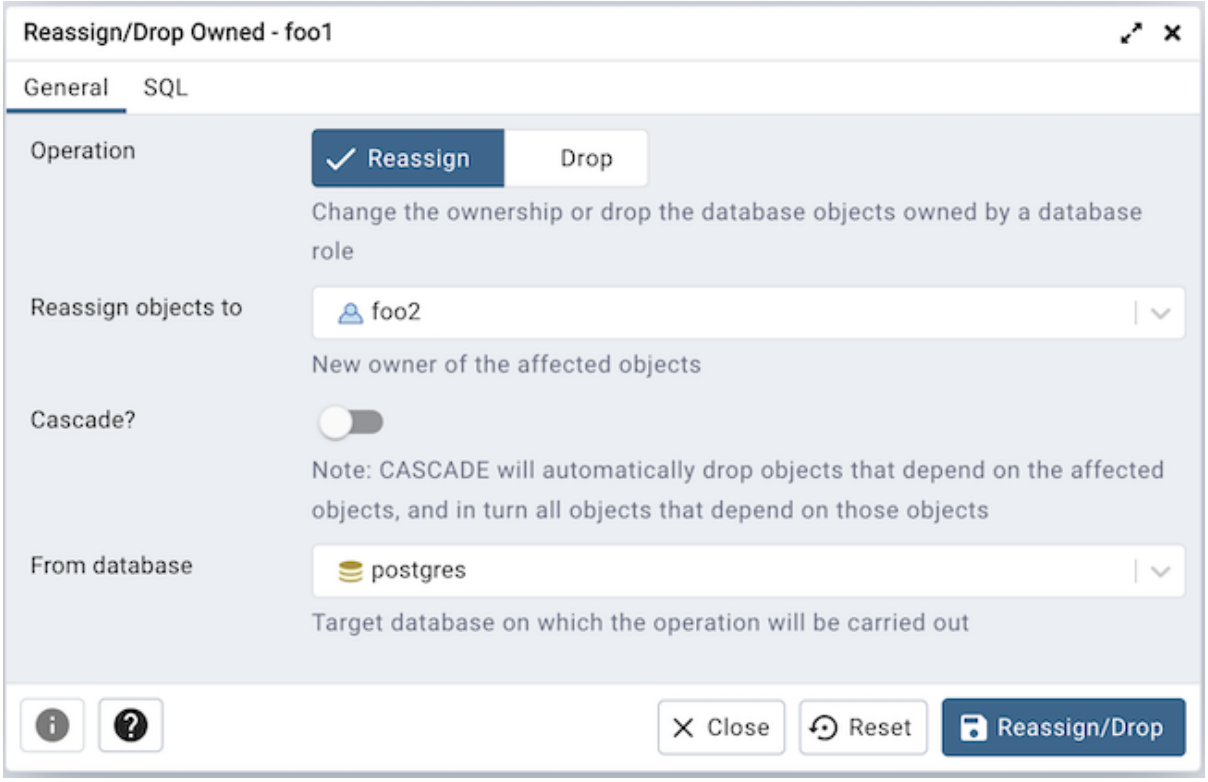
この例は、 *space_01* という名前のテーブルスペースの作成を示しています。 *1* に等しい *random_page_cost* 値があります。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

4.5 Role Reassign/Drop Own Dialog

Reassign/DropOwn ダイアログを使用して、データベースロールが所有するデータベースオブジェクトの所有権を変更します。このダイアログは、 *old_roles* が所有するデータベースオブジェクトの所有権を *new_role* に変更するようにシステムに指示します。

Reassign/DropOwn ダイアログは、General タブを介して Reassign&Drop ロールを編成します。



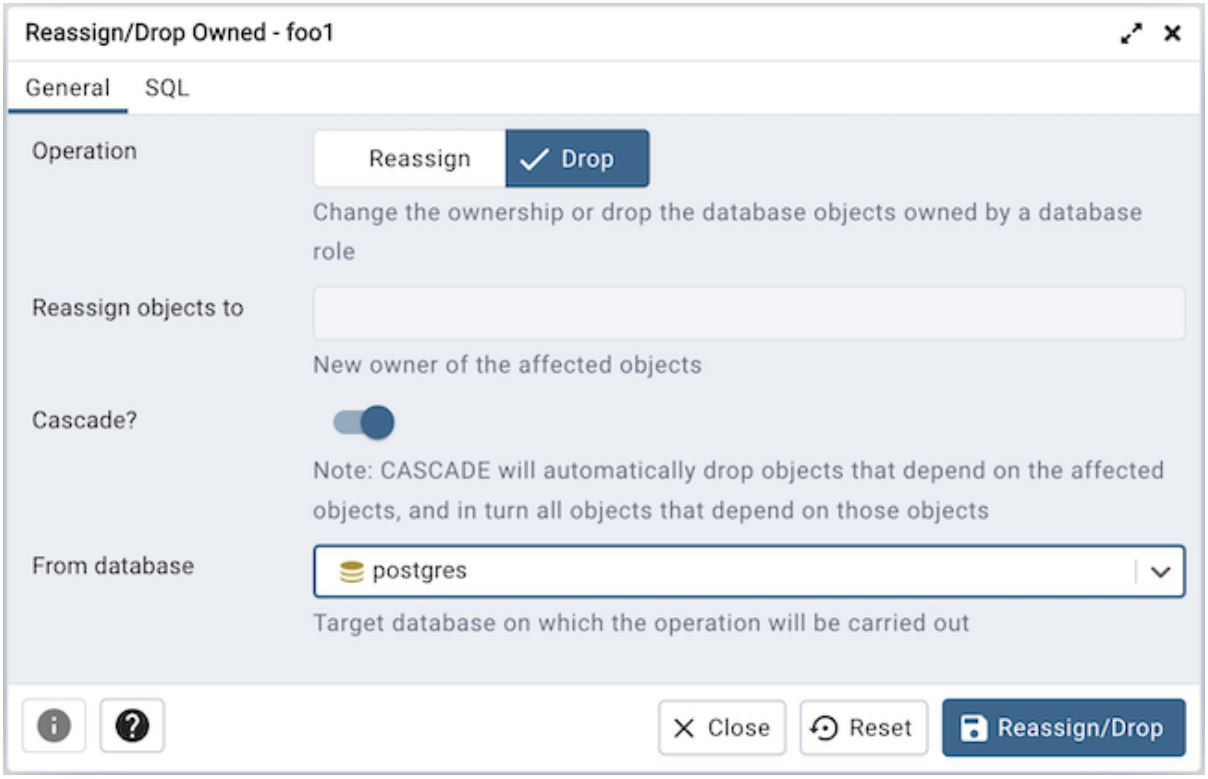
- *Operation* フィールドを使用して、再割り当てオプションを提供します。
- *ReassignObjectsto* フィールドに新しいロールを入力します。選択したデータベース内のすべてのオブジェクト、および *old_role* が所有するすべての共有オブジェクト（データベース、テーブルスペース）の所有権は、 *new_role* に再割り当てされます。
- 再割り当てを実行するデータベースを提供します。

上記の例は、 *old_role* を *new_role* に再割り当てする方法を示しています。

SQL タブをクリックして続行します。



データベースロールが所有するデータベースオブジェクトを削除しています。



- *Operation* フィールドを使用して、ドロップオプションを提供します。
- *Cascade?* フィールドを使用して Yes を指定します。デフォルトは No です。
- オブジェクトのドロップが実行されるデータベースを提供します。

SQL タブをクリックして続行します。



上記の例は、 *role* が所有するドロップを示しています。

- オンラインヘルプにアクセスするには、 **ヘルプ** ボタン（?）をクリックします。
- *OK* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 *Cancel* ボタンをクリックします。

第5章　Managing Database Objects

pgAdmin4 は、データベースオブジェクトの設計と作成に使用できるシンプルで強力なダイアログを提供します。各ダイアログには、ダイアログによって作成されるオブジェクトを説明するために使用する一連のタブが含まれています。[SQL] タブには、オブジェクトの作成時にサーバーが実行する SQL コマンドが表示されます。

データベースオブジェクトを作成できるダイアログにアクセスするには、pgAdmin ツリーコントロールでオブジェクトタイプを右クリックし、そのオブジェクトの *Create* オプションを選択します。たとえば、新しいキャストを作成するには、 *Casts* ノードを右クリックし、 *CreateCast...* を選択します。

5.1 Cast Dialog

キャスト ダイアログを使用して、キャストを定義します。キャストは、値をあるデータ型から別のデータ型に変換する方法を指定します。

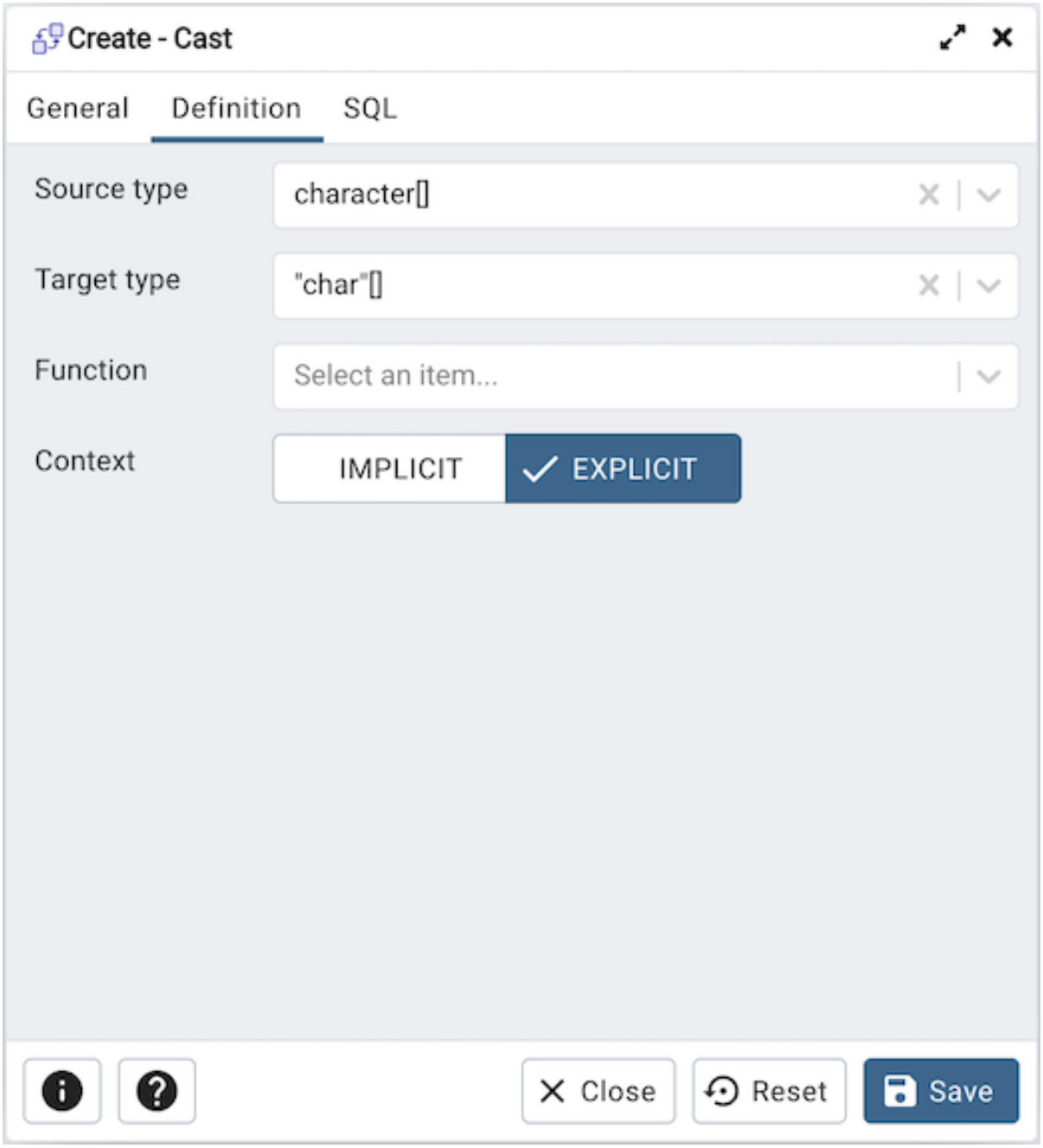
キャスト ダイアログは、次のダイアログタブを使用してキャストの開発を編成します。 **一般** および **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、キャストを識別します。

- **名前** フィールドは無効になっています。 *pgAdmin* ツリーコントロールに表示される名前は、 **ターゲット** タイプと連結された **ソース** タイプであり、 **キャスト** ダイアログ *[***定義**] タブで選択すると自動的に生成されます。
- キャストに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、パラメーターを定義します。

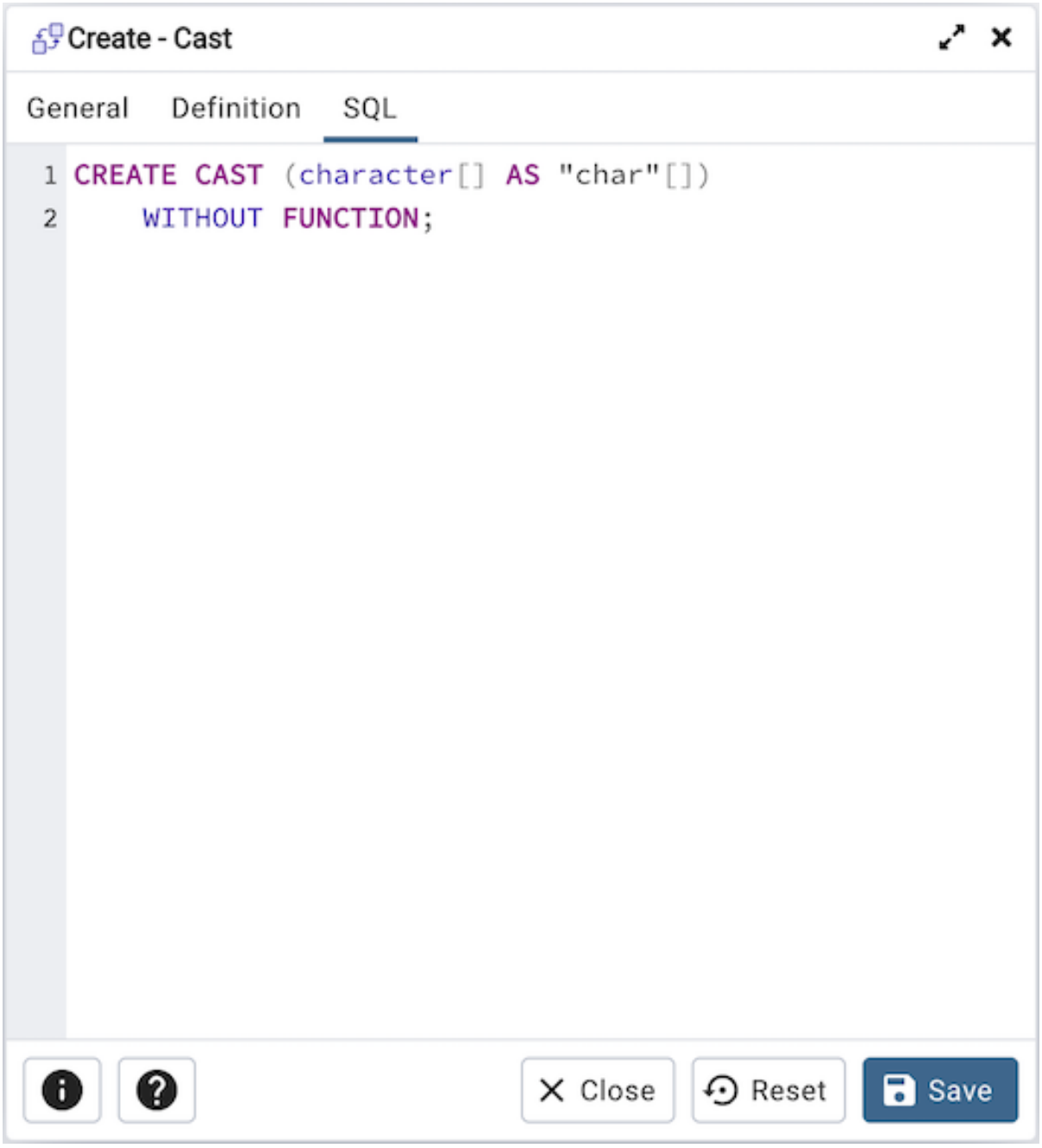
- **ソースの種類** の横にあるドロップダウンリストボックスを使用して、キャストのソースデータ型の名前を選択します。
- **ターゲットタイプ** の横にあるドロップダウンリストボックスを使用して、キャストのターゲットデータ型の名前を選択します。
- **ファンクション** の横にあるドロップダウンリストボックスを使用して、キャストの実行に使用するファンクションを選択します。関数の結果のデータ型は、キャストのターゲット型と一致する必要があります。
- キャストが暗黙的である場合は、 *Context* トグルを *Implicit* に選択します。デフォルトでは、キャストは明示的なキャスト要求によってのみ呼び出すことができます。キャストが *Implicit* とマークされている場合、割り当てまたは式の内部で、任意のコンテキストで暗黙的に呼び出すことができます。

SQL タブをクリックして続行します。

Cast ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

Example

以下は、 **キャスト** ダイアログでのユーザー選択によって生成された sql コマンドの例です。

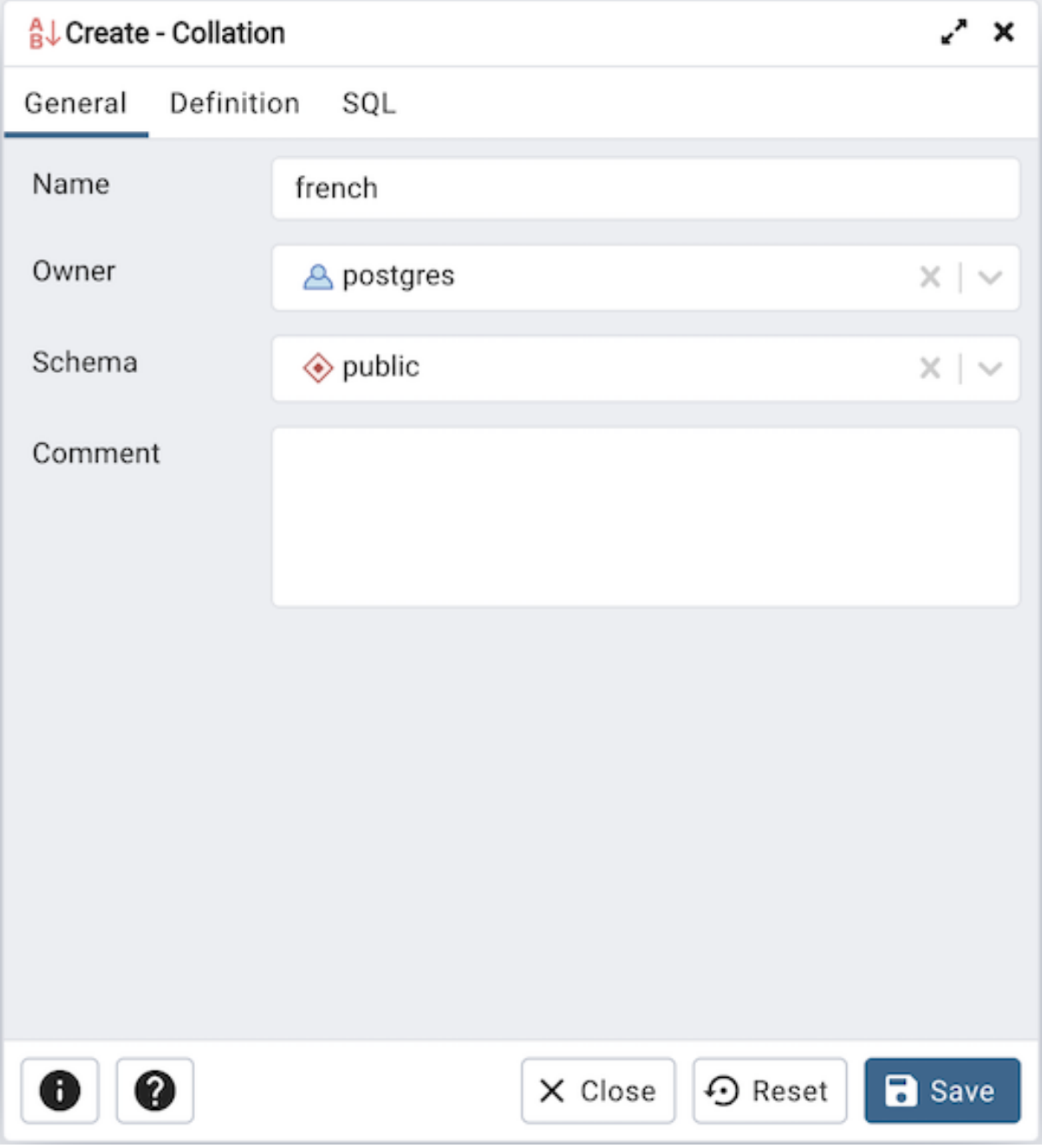


- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.2 Collation Dialog

Collation ダイアログを使用して、照合を定義します。照合は、SQL 名をオペレーティングシステムのロケールにマップする SQL スキーマオブジェクトです。照合を作成するには、宛先スキーマに対する CREATE 権限が必要です。

照合 ダイアログは、次のダイアログタブを介して照合の展開を編成します。 **一般** と **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、照合を識別します。

- *Name* フィールドを使用して、照合順序の名前を指定します。照合名はスキーマ内で一意である必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。

- **所有者** フィールドのドロップダウンリストボックスから所有者の名前を選択します。
- **スキーマ** フィールドのドロップダウンリストボックスから、照合が存在するスキーマの名前を選択します。
- **コメント** フィールドに照合に関するメモを保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、オペレーティングシステムのロケール設定を指定します。

- *Copycollation* の横にあるドロップダウンリストボックスを使用して、コピーする既存の照合の名前を選択します。新しい照合は既存の照合と同じプロパティを持ちますが、独立したオブジェクトになります。既存の照合をコピーすることを選択した場合、このタブに表示される照合プロパティを変更することはできません。

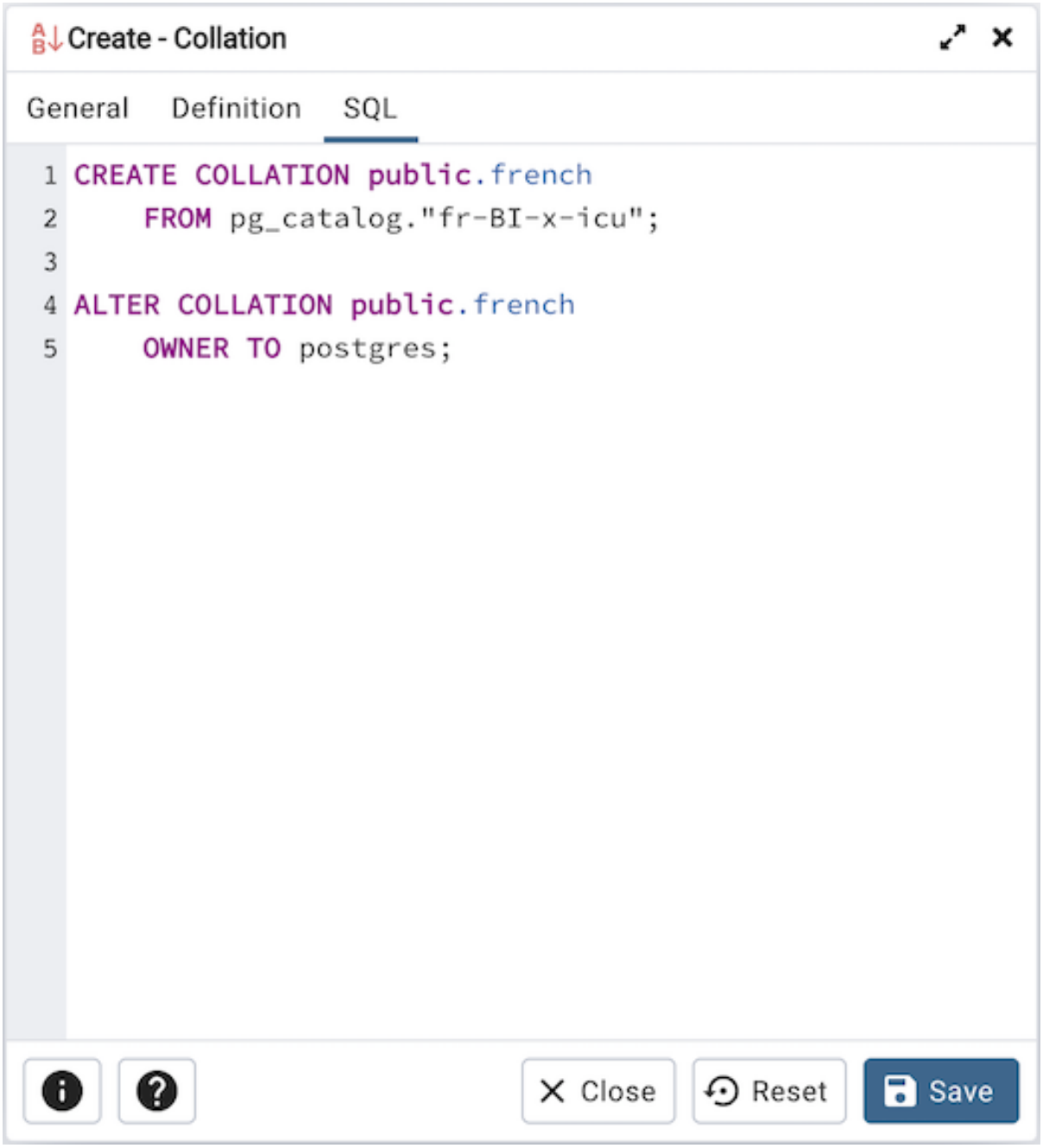
- *Locale* フィールドを使用してロケールを指定します。ロケールは、言語と言語のフォーマット特性を指定します。これを指定した場合、以下のパラメーターは指定できません。Linux システムでサポートされているロケールのリストを表示するには、コマンド *locale-a* を使用します。
- *LC_COLLATE* フィールドを使用して、指定した文字列の並べ替え順序でロケールを指定します。ロケールは、現在のデータベースエンコーディングに適用できる必要があります。(詳細については、CREATEDATABASE を参照してください。)
- *LC_CTYPE* フィールドを使用して、指定された文字分類のロケールを指定します。ロケールは、現在のデータベースエンコーディングに適用できる必要があります。(詳細については、CREATEDATABASE を参照してください。)

SQL タブをクリックして続行します。

Collation ダイアログに入力すると、SQL コマンドが生成されます（例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.2.1 例

以下は、 **照合** *ダイアログでのユーザー選択によって生成された sql コマンドの例です。



この例は、ロケール *fr-BI-x-icu* に指定されたルールを使用する french という名前の照合を作成する方法を示しています。照合は postgres*が所有しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。ロケールの設定の詳細については、PostgreSQL コアドキュメントの 22.1 章ロケールサポートを参照してください。

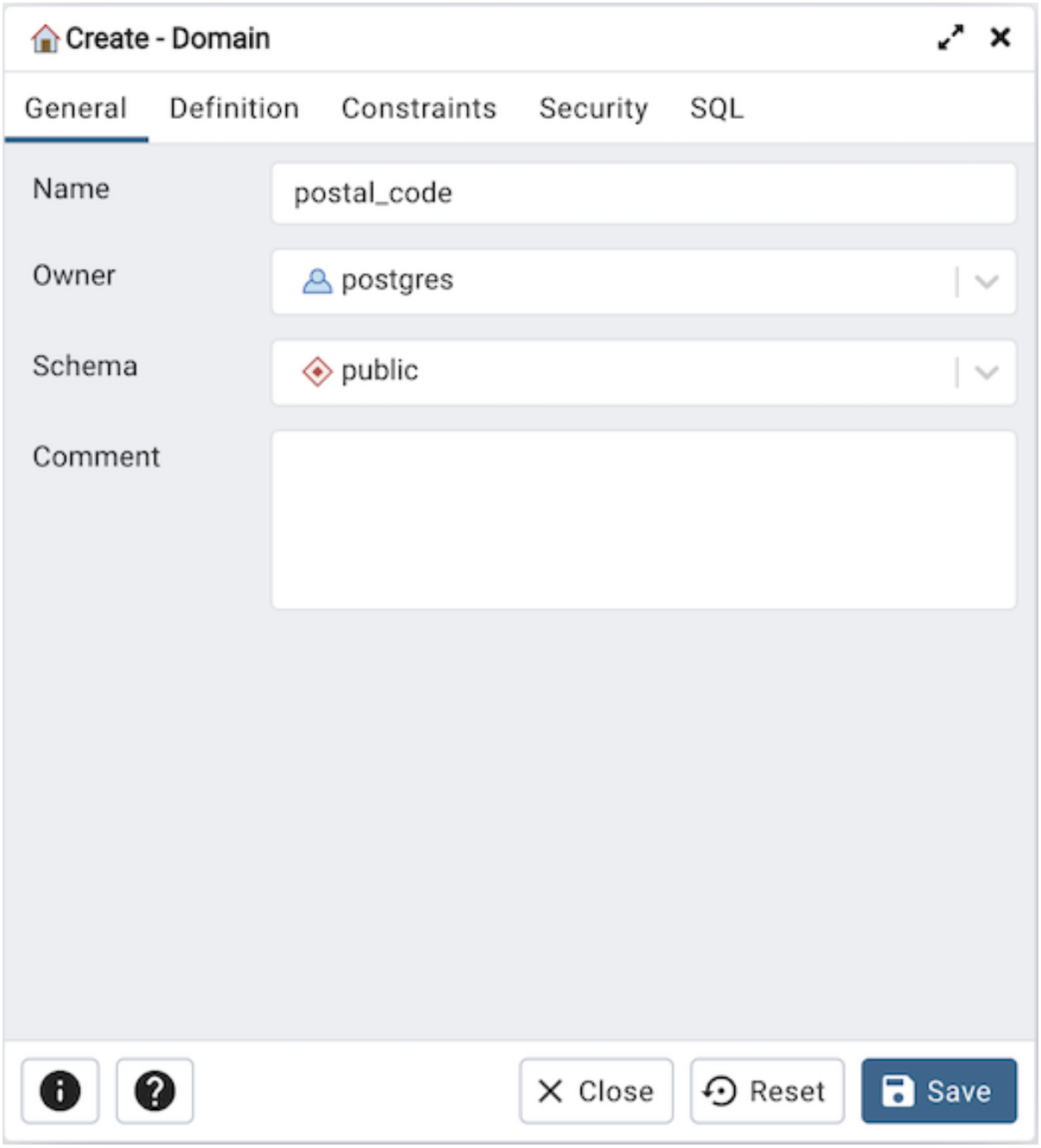
  https://www.postgresql.org/docs/current/locale.html

- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.3 Domain Dialog

ドメインを定義するには、 *Domain* ダイアログを使用します。ドメインは、許容値を制限する可能性のあるデータ型定義です。ドメインは、比較可能な列を含む複数のテーブルを作成する場合に役立ちます。各制約セットを個別に定義するのではなく、列に共通の制約を定義するドメインを作成し、列を作成するときにドメイン定義を再利用できます。

[**ドメイン**] ダイアログは、 **一般** 、 **定義** 、 **制約** 、および **セキュリティ** のタブを使用してドメインの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、ドメインを識別します。

- *Name* フィールドを使用して、ドメインの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

- **所有者** の横にあるドロップダウンリストボックスを使用して、ドメインを所有するロールを選択します。
- **スキーマ** フィールドのドロップダウンリストボックスから、ドメインが存在するスキーマの名前を選択します。
- ドメインに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。

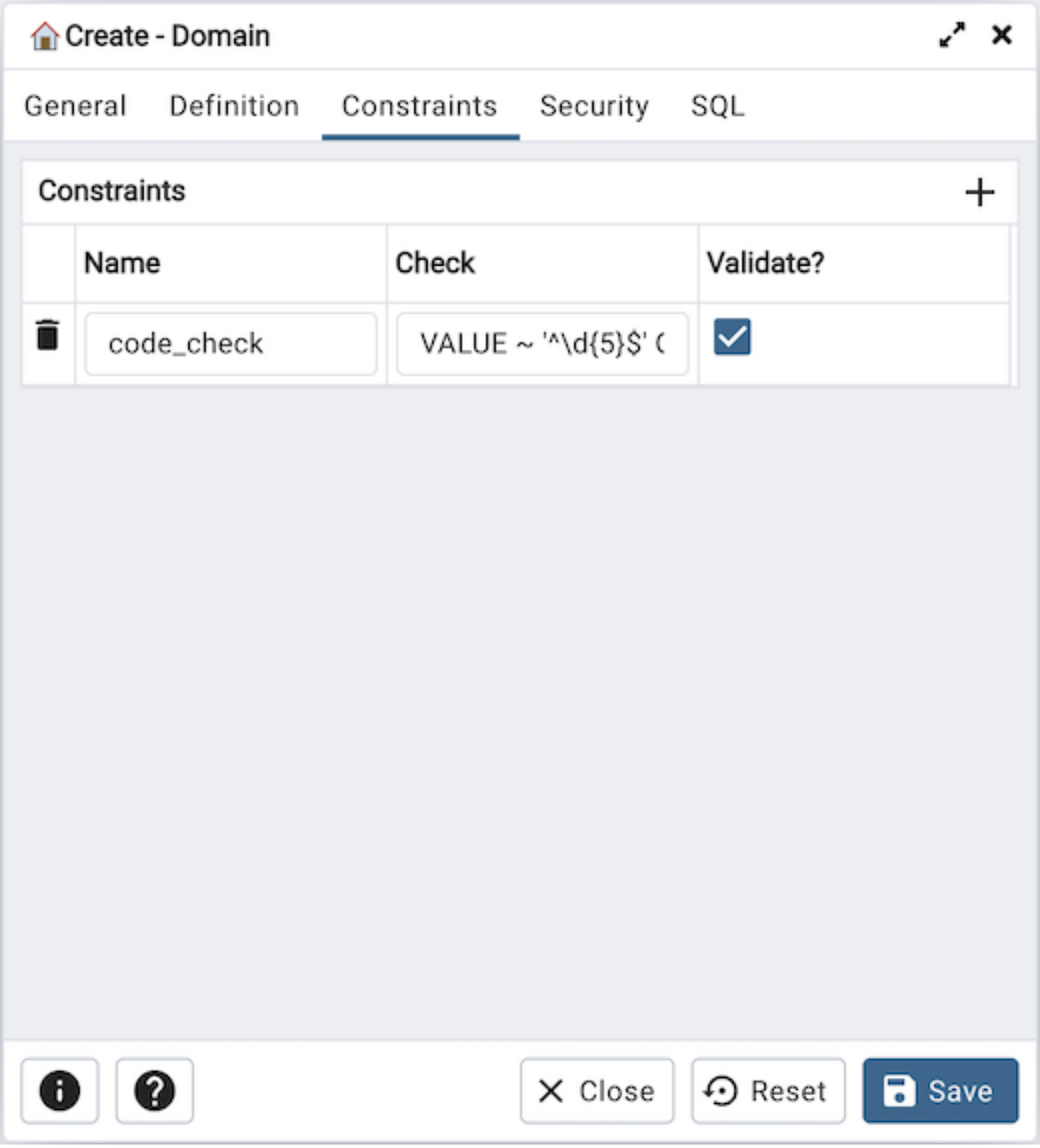


[**定義**] タブのフィールドを使用して、ドメインを説明します。

- *Basetype* の横にあるドロップダウンリストボックスを使用して、データ型を指定します。
- 状況依存の *Length* フィールドを使用して、数値型の数値長を指定します。

- 状況依存の **精度** フィールドを使用して、数値型の有効桁数の合計を指定します。
- *Default* フィールドにドメインデータ型のデフォルト値を指定します。デフォルトの式のデータ型は、ドメインのデータ型と一致する必要があります。デフォルト値が指定されていない場合、デフォルト値は null 値です。
- *NotNull* スイッチを移動して、このドメインの値が null にならないように指定します。
- *Collation* の横にあるドロップダウンリストボックスを使用して、照合キャストを適用します。照合が指定されていない場合、基になるデータ型の既定の照合が使用されます。COLLATE が指定されている場合、基になる型は照合可能である必要があります。

[*Constraints*] タブをクリックして続行します。



[*Constraints*] タブのフィールドを使用して、ドメインのルールを指定します。 **追加** アイコン（+）をク

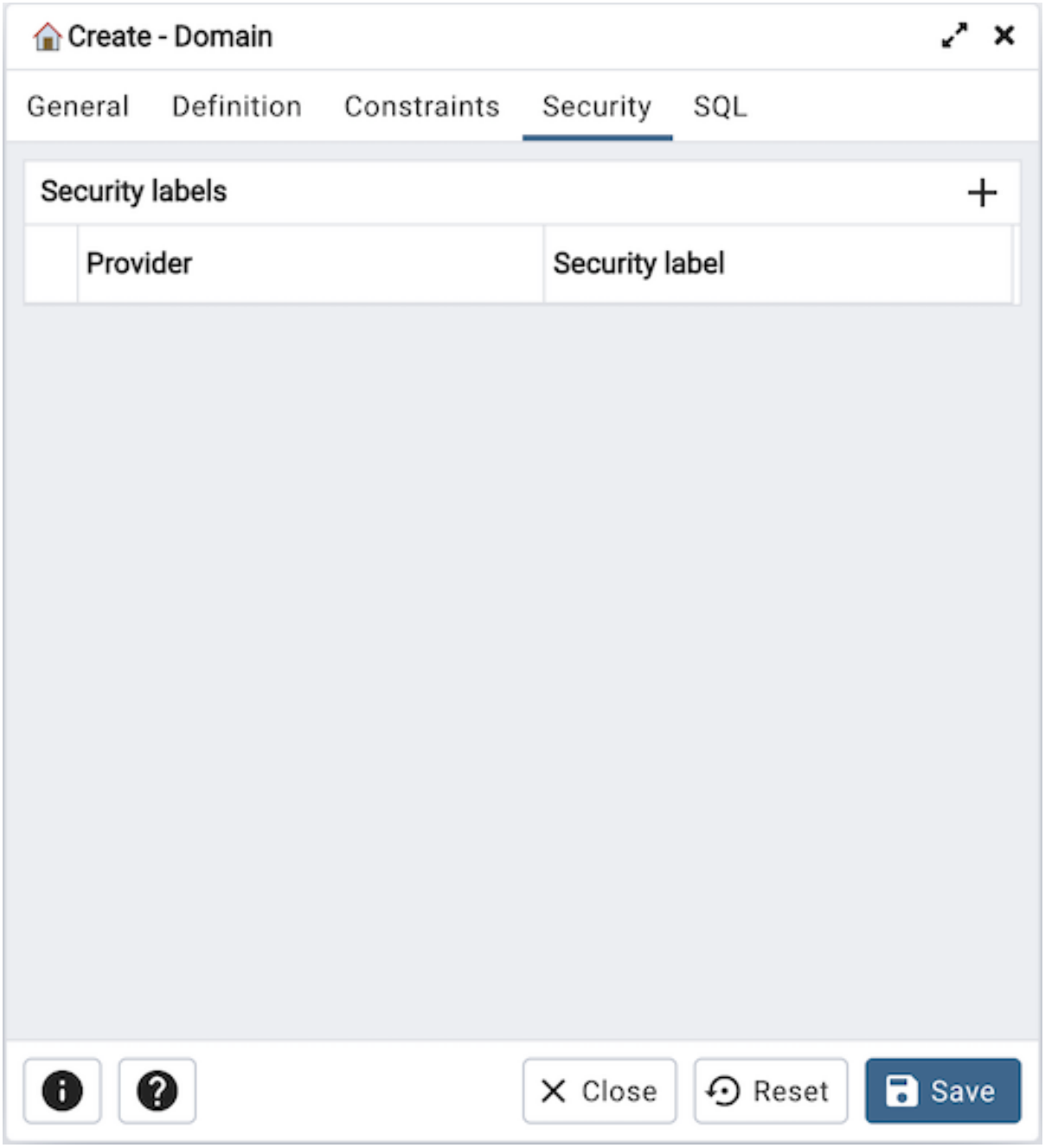
リックして、制約を設定します。

- *Name* フィールドを使用して、制約の名前を指定します。
- *Check* フィールドを使用して、制約の式を提供します。
- *Validate* チェックボックスを使用して、制約を検証するかどうかを決定します。デフォルトのチェックボックスはオンになっており、検証要件を設定します。

CHECK 句は、ドメインの値が満たす必要がある整合性テストを指定します。各制約は、ブール結果を生成する式である必要があります。テストする値を参照するには、キーワード VALUE を使用します。TRUE または UNKNOWN と評価される式は成功します。式が FALSE の結果を生成した場合、エラーが報告され、値をドメイン型に変換できません。CHECK 式にサブクエリを含めたり、VALUE 以外の変数を参照したりすることはできません。ドメインに複数の CHECK 制約がある場合、名前のアルファベット順にテストされます。

追加 アイコン（+）をクリックして、追加の制約を設定します。制約を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



セキュリティラベル パネルを使用して、セキュリティラベルを割り当てます。 **追加** アイコン（+）をクリックして、ラベルを追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

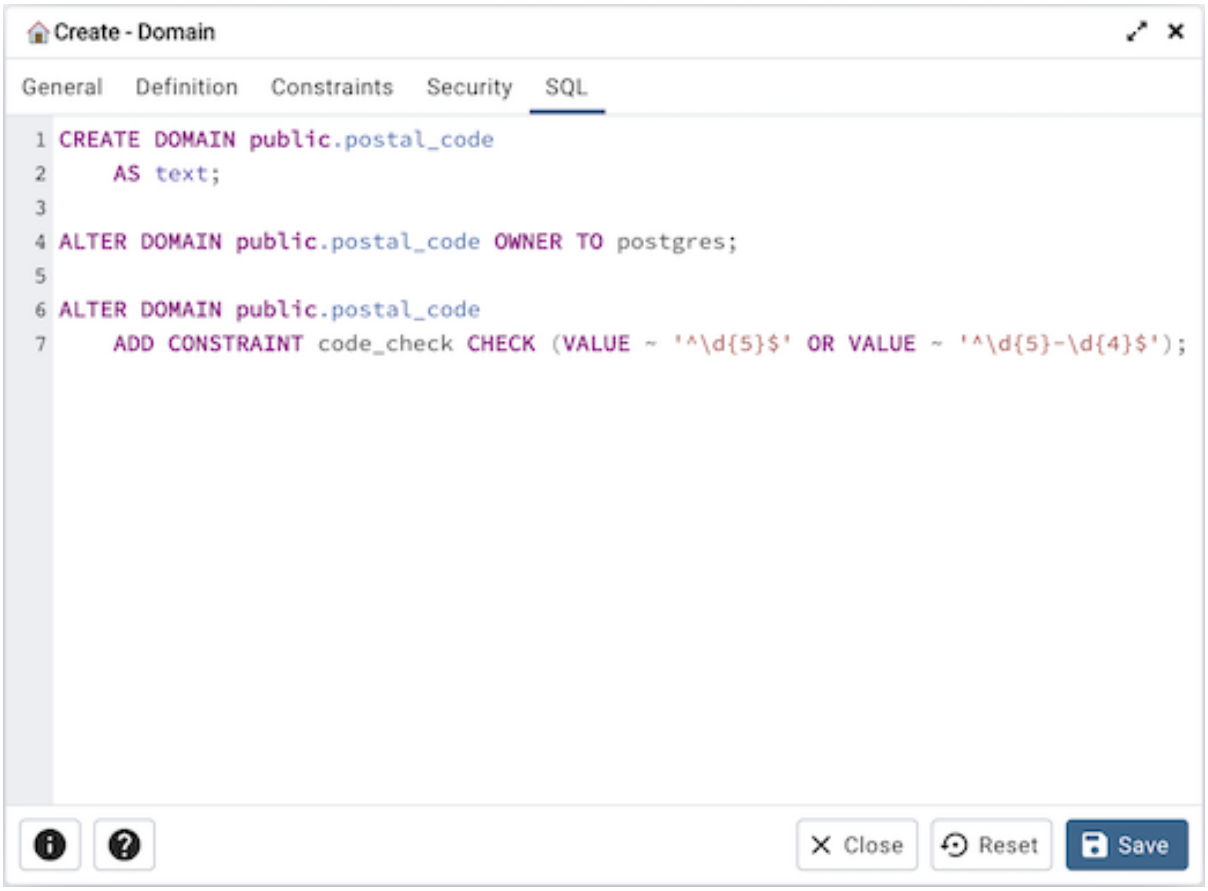
追加 アイコン（+）をクリックして、追加の各ラベルを指定します。ラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Domain ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.3.1 例

以下は、 **ドメイン** ダイアログでの選択によって生成される sql コマンドの例です。



この例は、入力された値が適切な形式であることを確認する *postal_code* という名前のドメインを作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.4 Domain Constraints Dialog

DomainConstraints ダイアログを使用して、ドメイン制約を作成または変更します。ドメイン制約は、ドメインに指定された値が定義された基準を満たしていることを確認します。 *DomainConstraints* ダイアログは、ALTERDOMAIN コマンドのオプションを実装します。

ドメイン制約 ダイアログは、次のダイアログタブを介してドメイン制約の開発を編成します。 **一般** および **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、ドメイン制約を識別します。

- *Name* フィールドを使用して、制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

- 制約に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、ドメイン制約を定義します。

- *Check* フィールドを使用して、CHECK 式を提供します。CHECK 式は、ドメインが満たさなければならない制約を指定します。制約はブール結果を生成する必要があります。テストする値を参照するには、キーワード VALUE を含めます。TRUE または UNKNOWN に評価される式のみが成功します。CHECK 式にサブクエリを含めたり、VALUE 以外の変数を参照したりすることはできません。ドメインに複数の CHECK 制約がある場合、それらはアルファベット順にテストされます。
- *Validate?* スイッチを *No* の位置に移動して、制約を無効にします。制約が NOTVALID とマークされている場合、制約は既存の列データに適用されません。デフォルト値は **はい** です。

SQL タブをクリックして続行します。

DomainConstraints ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.4.1 例

以下は、 **ドメインの制約** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、 *weekday* という名前のドメイン **タイムシート** にドメイン制約を作成する方法を示しています。 *Friday* に等しい値を制限します。

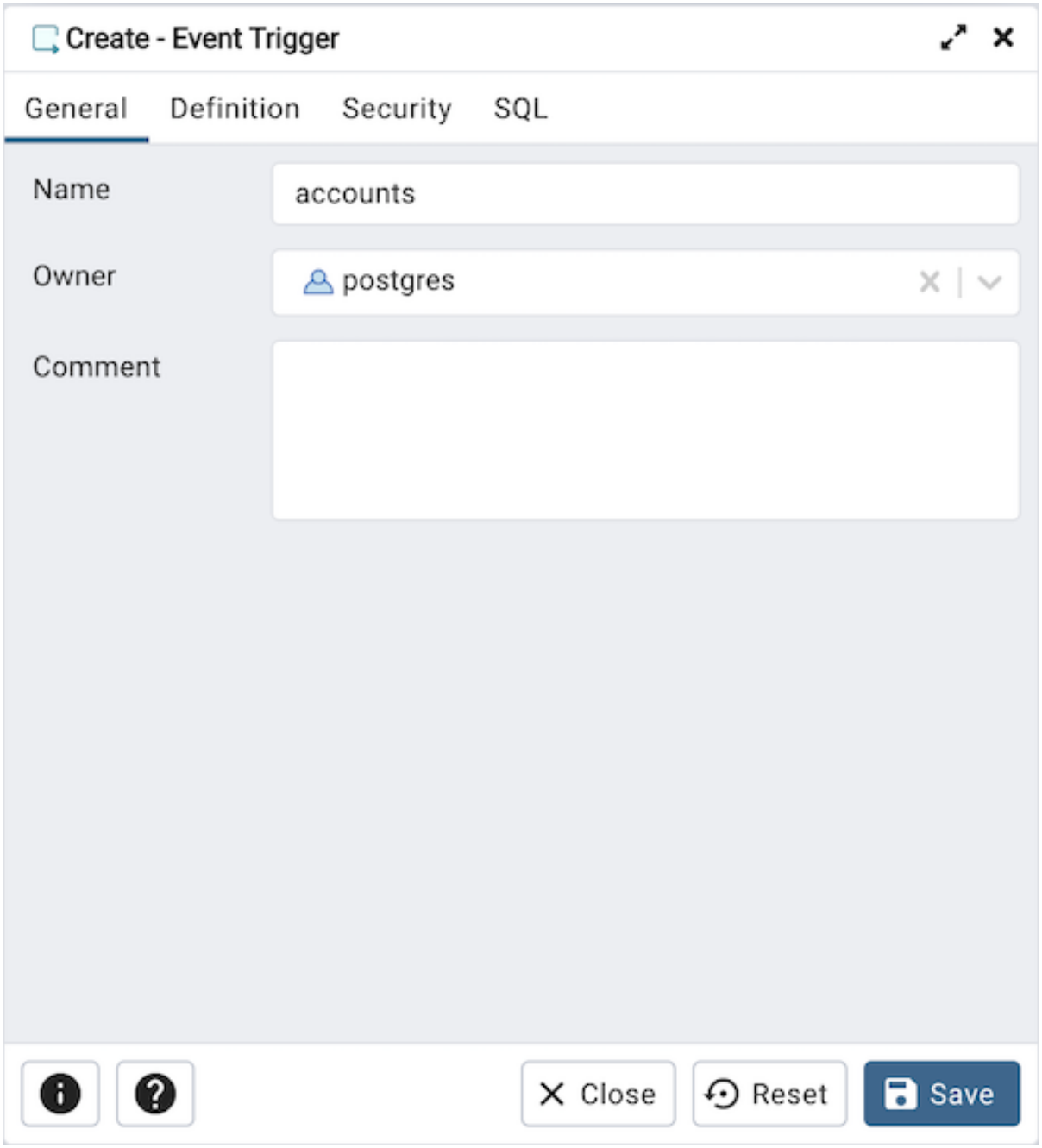
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。

- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.5 Event Trigger Dialog

イベントトリガー ダイアログを使用して、イベントトリガーを定義します。単一のテーブルにアタッチされ、DML イベントのみをキャプチャする通常のトリガーとは異なり、イベントトリガーは特定のデータベースに対してグローバルであり、DDL イベントをキャプチャできます。通常のトリガーと同様に、イベントトリガーは、イベントトリガーのサポートを含む手続き型言語または C で作成できますが、SQL 作成できません。

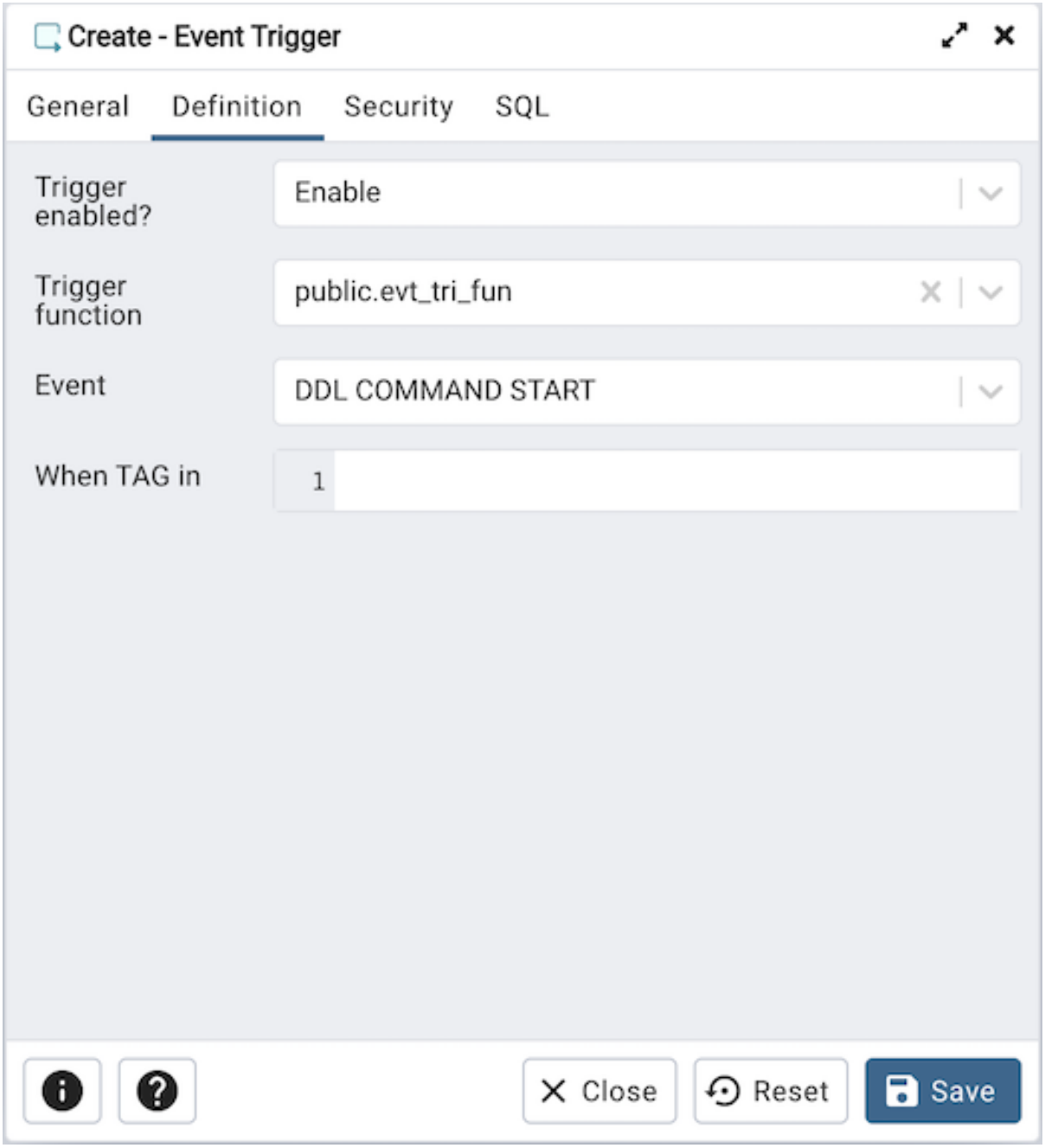
イベントトリガー ダイアログは、次のダイアログタブを介してイベントトリガーの開発を整理します： **一般** 、 **定義** 、および **セキュリティラベル** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、イベントトリガーを識別します。

- *Name* フィールドを使用して、イベントトリガーの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、イベントトリガーの所有者を指定します。
- イベントトリガーに関するメモを **コメント** フィールドに保存します。

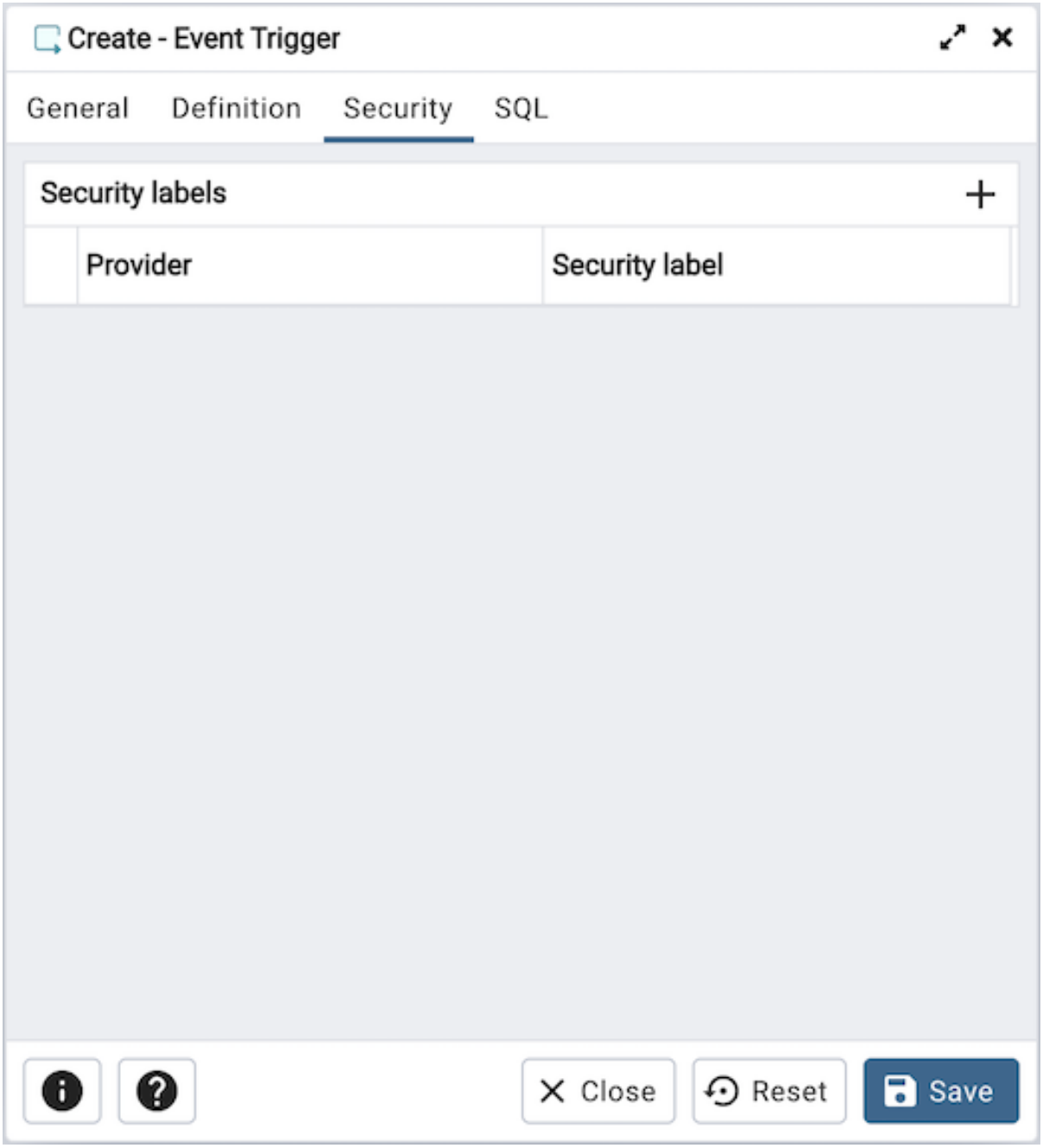
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、イベントトリガーを定義します。

- **トリガー有効** フィールドのドロップダウンから値を選択して、トリガーのステータスを指定します：**有効** **無効** 、 **レプリカ** **常に** 。
- **トリガー関数** の横にあるドロップダウンリストボックスを使用して、既存の関数を指定します。トリガー関数は空の引数リストを取り、event_trigger タイプの値を返します。
- *Events* フィールドのドロップダウンから値を選択して、イベントトリガーを起動するタイミングを指定します。 *DDLCOMMANDSTART* 、 *DDLCOMMANDEND* 、または *SQLDROP* 。
- *WhenTAGin* フィールドを使用して、トリガーが実行される TAG のフィルター値を入力します。値はコンマで区切って一重引用符で囲む必要があります。

[**セキュリティラベル**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、トリガーに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルを追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

イベントトリガー ダイアログに入力すると、SQL コマンドが生成されます。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.5.1 例

以下は、 **イベントトリガー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



このコマンドは、 *evt_tri_fun* という名前のプロシージャーを呼び出す *accounts* という名前のイベントトリガーを作成します。

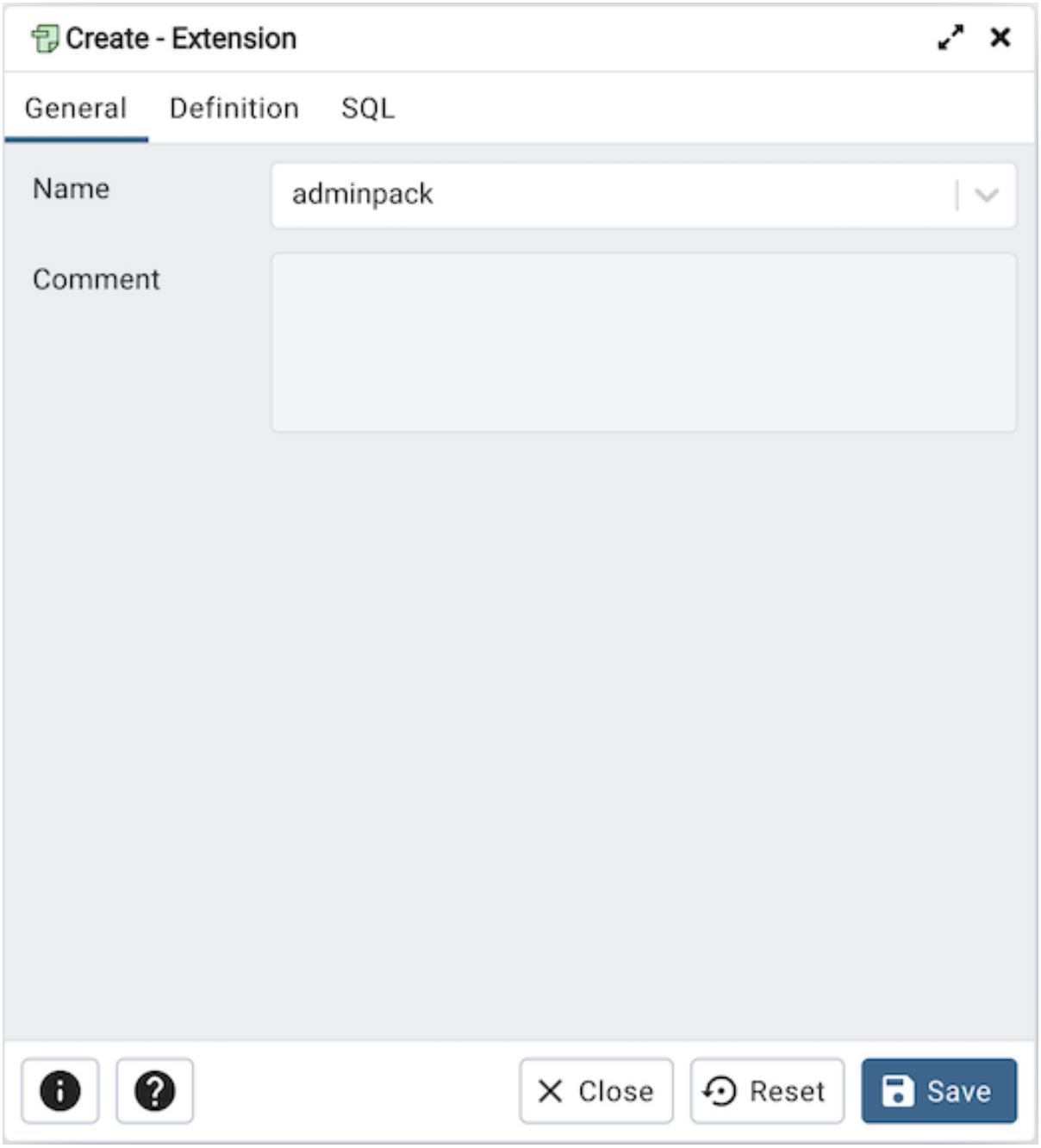
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。

- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.6 Extension Dialog

Extension ダイアログを使用して、現在のデータベースに新しい拡張機能をインストールします。拡張機能は、Postgres インストールに対象の機能を追加する SQL オブジェクトのコレクションです。 **拡張機能** ダイアログは、拡張機能の機能を現在のデータベースにのみ追加します。拡張機能を使用する各データベースに拡張機能を登録する必要があります。拡張機能をデータベースにロードする前に、前提条件のファイルがインストールされていることを確認する必要があります。

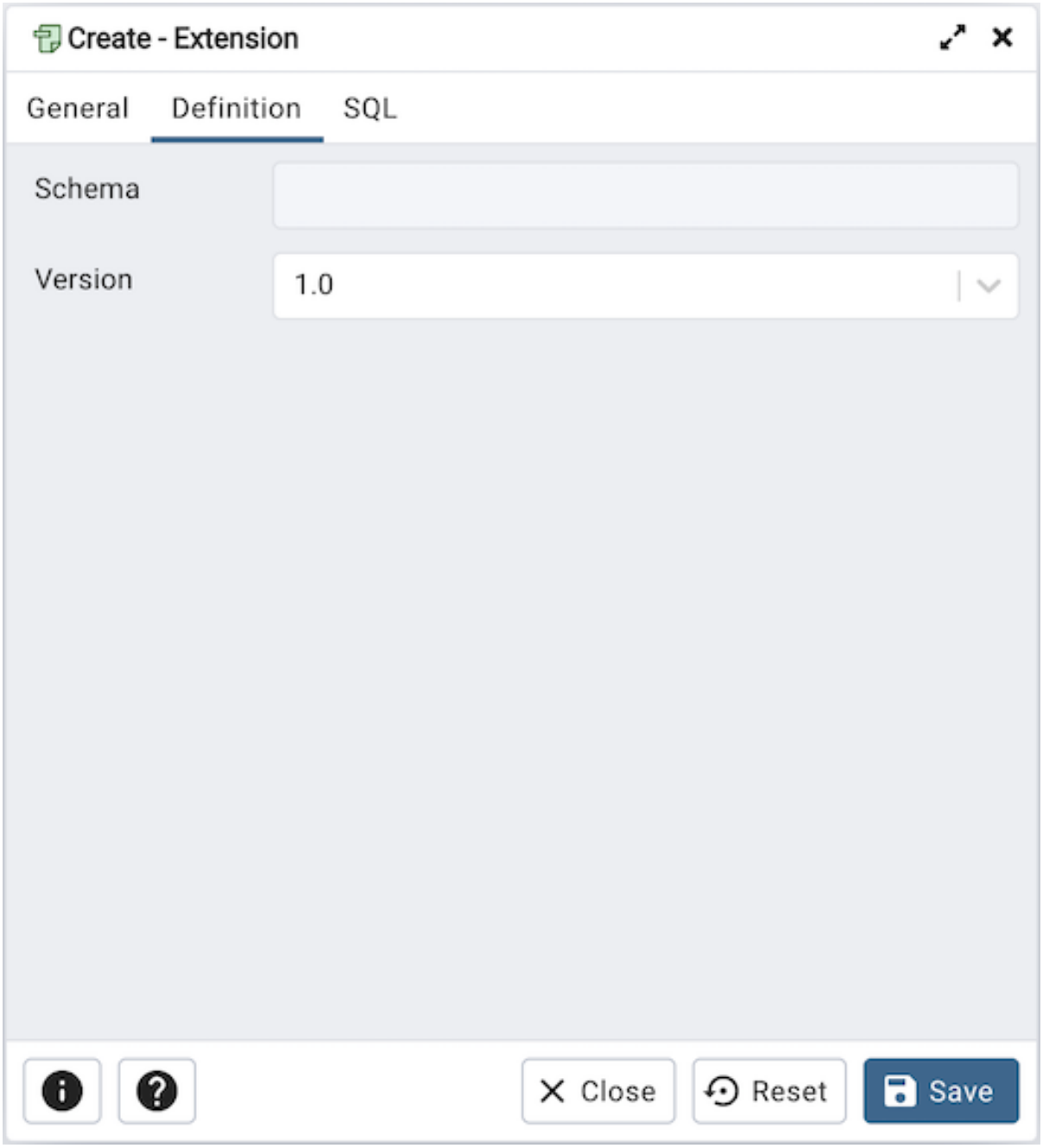
Extension ダイアログでは、次のダイアログタブを介して CREATEEXTENSION コマンドのオプションを実装できます。 *General* および *Definition* 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、拡張機能を識別します。

- **名前** フィールドのドロップダウンリストボックスを使用して、拡張子を選択します。各拡張機能には一意の名前が必要です。
- 拡張機能に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブを使用して、 **スキーマ** と **バージョン** を選択します。

- **スキーマ** の横にあるドロップダウンリストボックスを使用して、拡張機能のオブジェクトをインストールするスキーマの名前を選択します。
- **バージョン** の横にあるドロップダウンリストボックスを使用して、インストールする拡張機能のバージョンを選択します。

SQL タブをクリックして続行します。

Extension ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.6.1 例

以下は、 *Extension* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



このコマンドは、 **パブリック** スキーマに *adminpack* 拡張機能を作成します。 *adminpack* のバージョン *1.0* です。

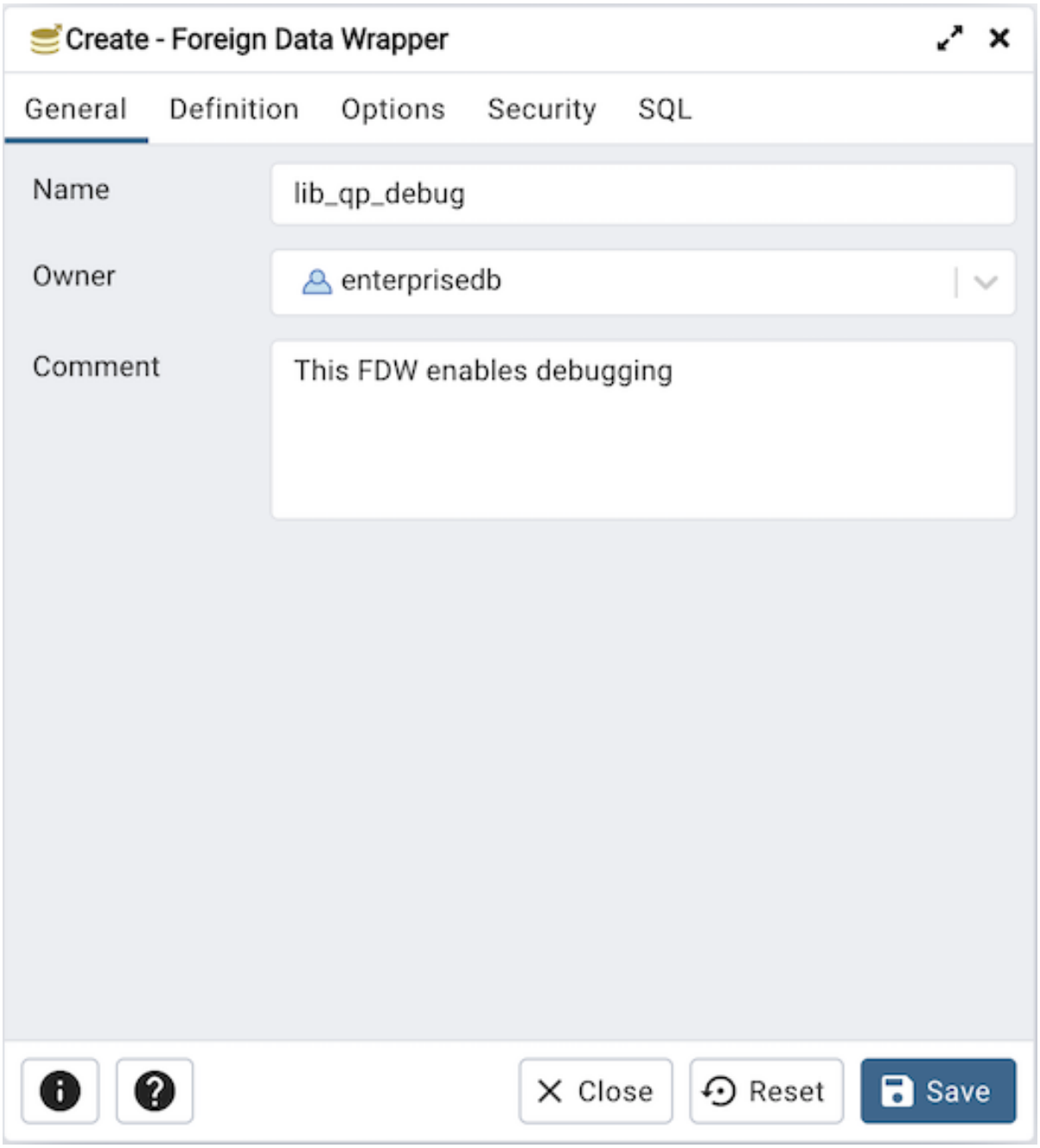
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.7 Foreign Data Wrapper Dialog

外部データラッパー ダイアログを使用して、外部データラッパーを作成または変更します。外部データラッパーは、Postgres データベースと別のデータソースに保存されているデータ間のアダプターです。

外部データラッパーを作成するにはスーパーユーザーである必要があります。

外部データラッパー ダイアログは、 **一般** 、 **定義** 、 **オプション** 、および **セキュリティ** のダイアログタブを介して外部データラッパーの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



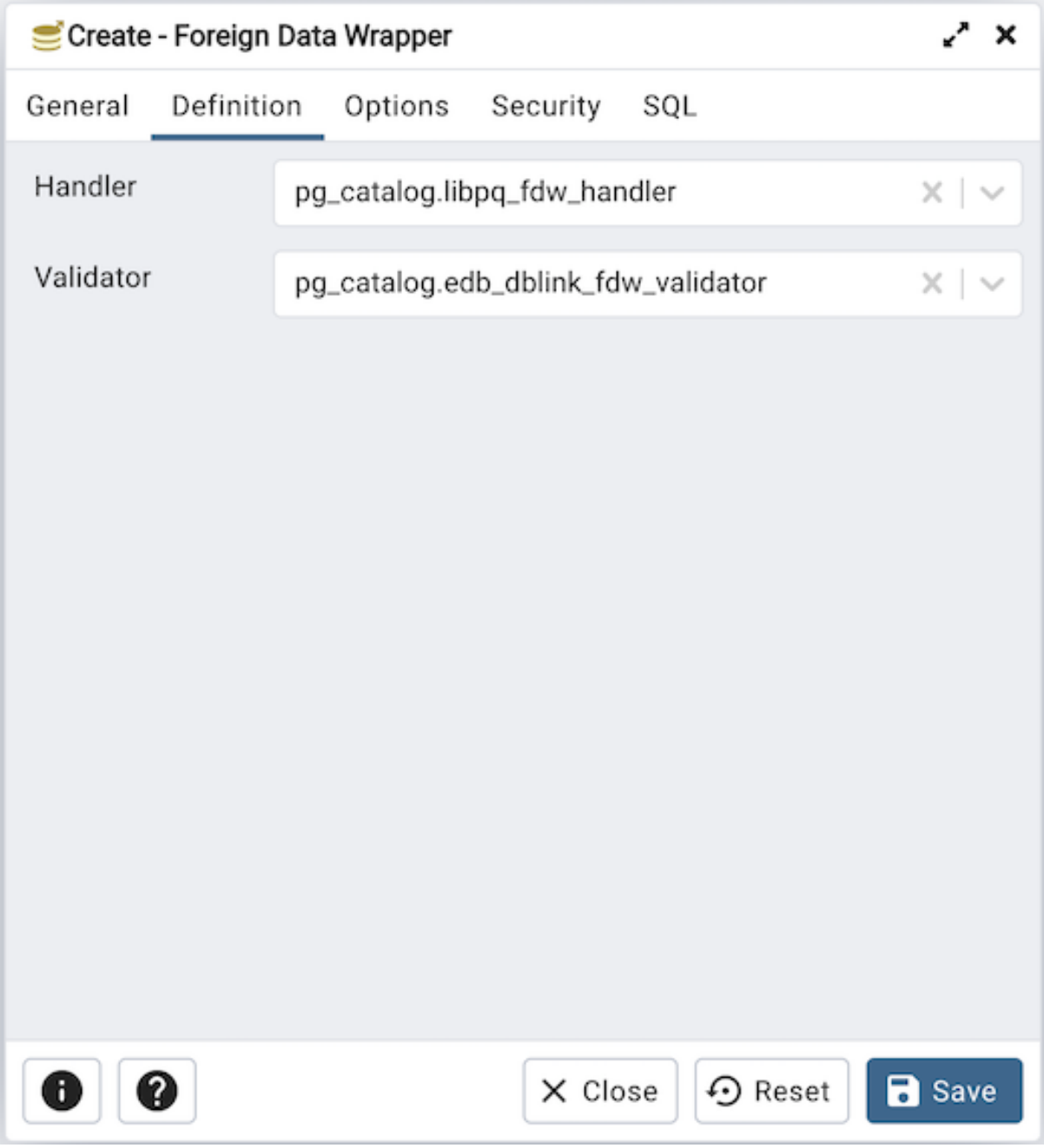
[**全般**] タブのフィールドを使用して、外部データラッパーを識別します。

- *Name* フィールドを使用して、外部データラッパーの説明的な名前を追加します。外部データラッ

パー名はデータベース内で一意である必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。

- **所有者** の横にあるドロップダウンリストボックスを使用して、外部データラッパーを所有するロールの名前を選択します。
- 外部データラッパーに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。

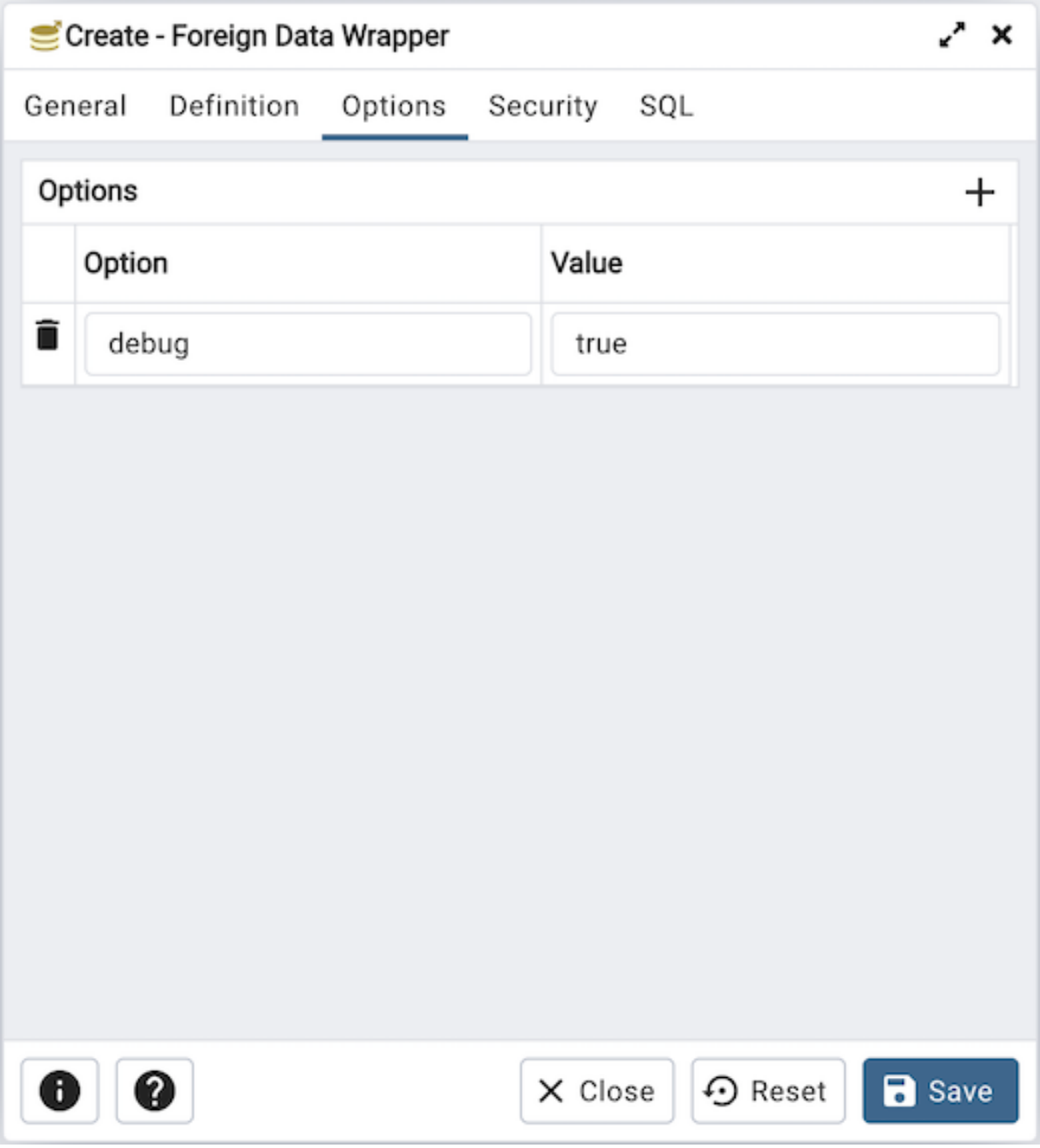


[**定義**] タブのフィールドを使用して、パラメーターを設定します。

- *Handler* フィールドのドロップダウンリストボックスからハンドラー関数の名前を選択します。これは、外部テーブルの実行ファンクションを取得するために呼び出される既存のファンクションの名前です。

- **バリデーター** フィールドのドロップダウンリストボックスからバリデーター関数の名前を選択します。これは、外部データラッパーに指定された汎用オプション、および外部データラッパーを使用する外部サーバー、ユーザーマッピング、および外部テーブルのオプションを確認するために呼び出される既存のファンクションの名前です。

[**オプション**] タブをクリックして続行します。



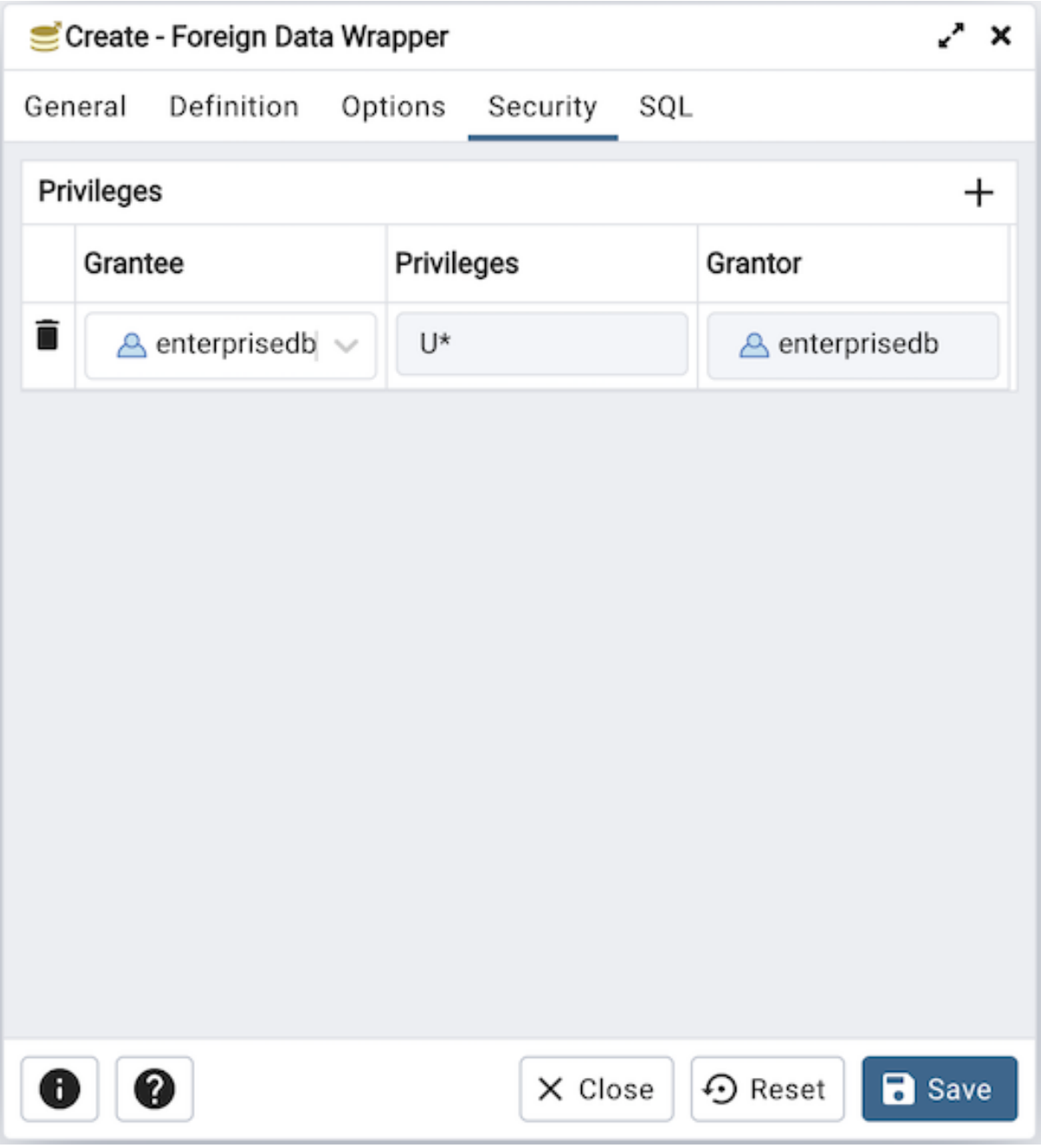
[**オプション**] タブのフィールドを使用して、オプションを指定します。

- **追加** アイコン（+）ボタンをクリックして、外部データラッパーのオプション/値のペアを追加します。サポートされるオプション/値のペアは、選択した外部データラッパーに固有です。
- *Option* フィールドにオプション名を指定し、対応する値を *Value* フィールドに入力します。

追加 アイコン（+）をクリックして、追加の各ペアを指定します。オプションを破棄するには、行の左側

にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、セキュリティ特権を割り当てます。 **追加** アイコン（+）をクリックして、一連の特権を割り当てます。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

add をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコン

をクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

ForeignDataWrapper ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.7.1 例

以下は、 *ForeignDataWrapper* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



この例では、既存のバリデーターとハンドラー関数、 *dblink_fdw_validator* および *libpg_fdw_handler* を使用する *lib_qp_debug* という名前の外部データラッパーを作成します。[**オプション**] タブでの選択は *debug*

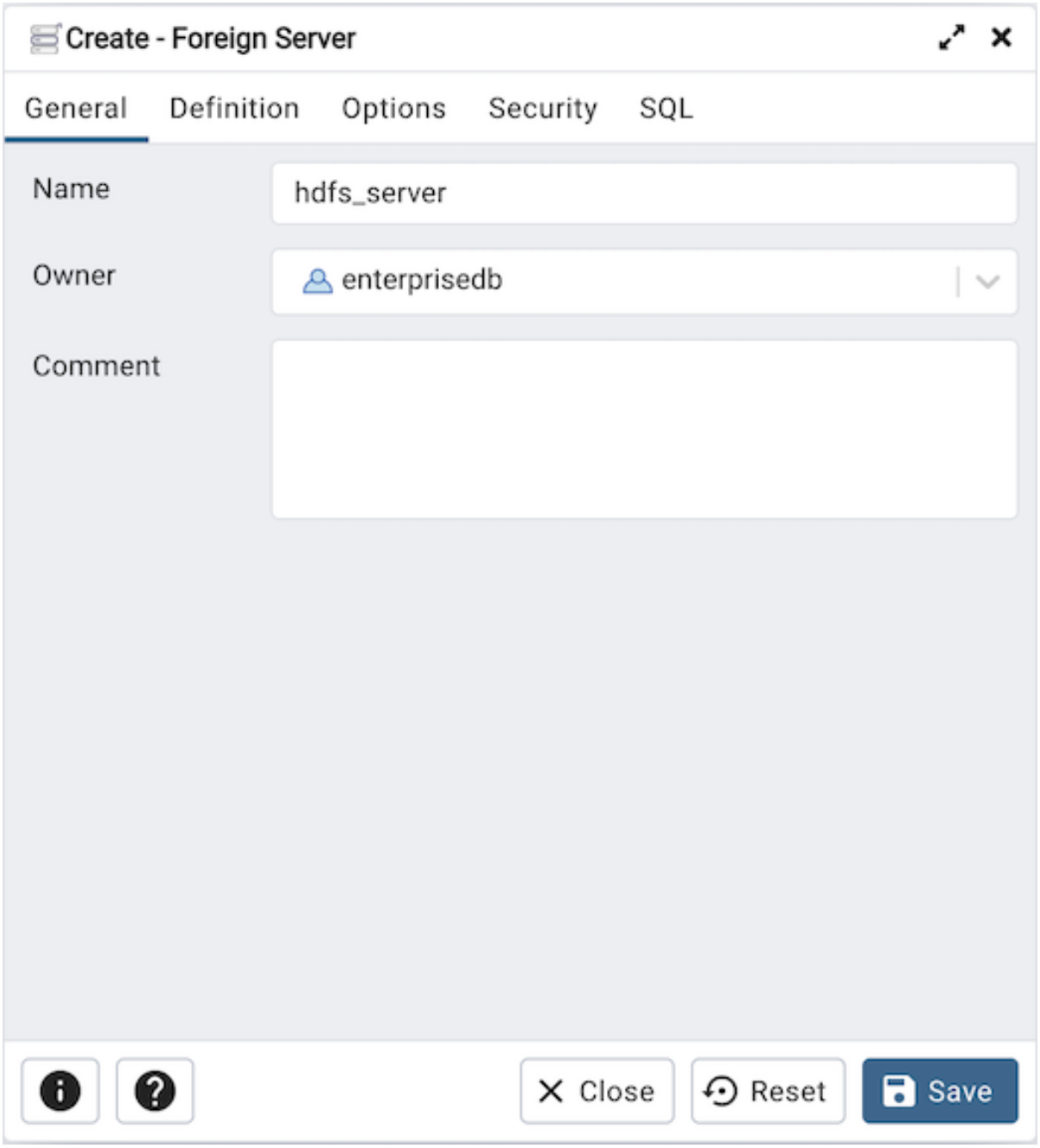
を *true* に設定します。外部データラッパーは *postgres* が所有しています。

- オンラインヘルプにアクセスするには、 **ヘルプ** ボタン（?）をクリックします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.8 Foreign Server Dialog

外部サーバー ダイアログを使用して、外部サーバーを作成します。通常、外部サーバーは、外部データラッパーが外部データリソースにアクセスするために使用する接続情報をカプセル化します。各外部データラッパーは異なる外部サーバーに接続できます。 *pgAdmin* ツリーコントロールで、該当する外部データラッパーのノードを展開して *ForeignServer* ダイアログを起動します。

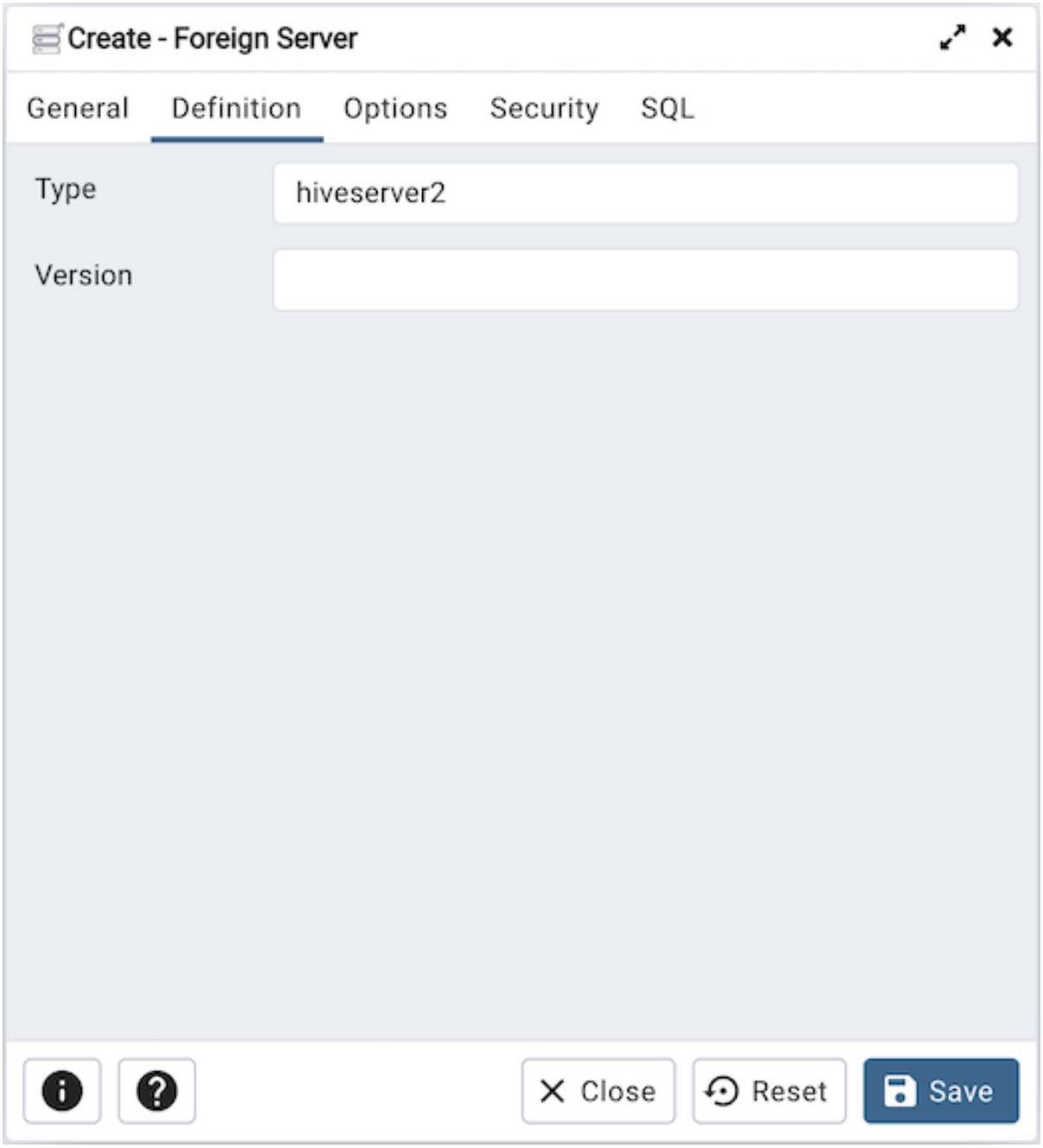
[**外部サーバー**] ダイアログは、 **一般** 、 **定義** 、 **オプション** 、および **セキュリティ** のダイアログタブを使用して外部サーバーの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、外部サーバーを識別します。

- *Name* フィールドを使用して、外部サーバーの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。データベース内で一意である必要があります。
- **所有者** の横にあるドロップダウンリストボックスを使用して、ロールを選択します。
- **コメント** フィールドに外部サーバーに関するメモを保存します。

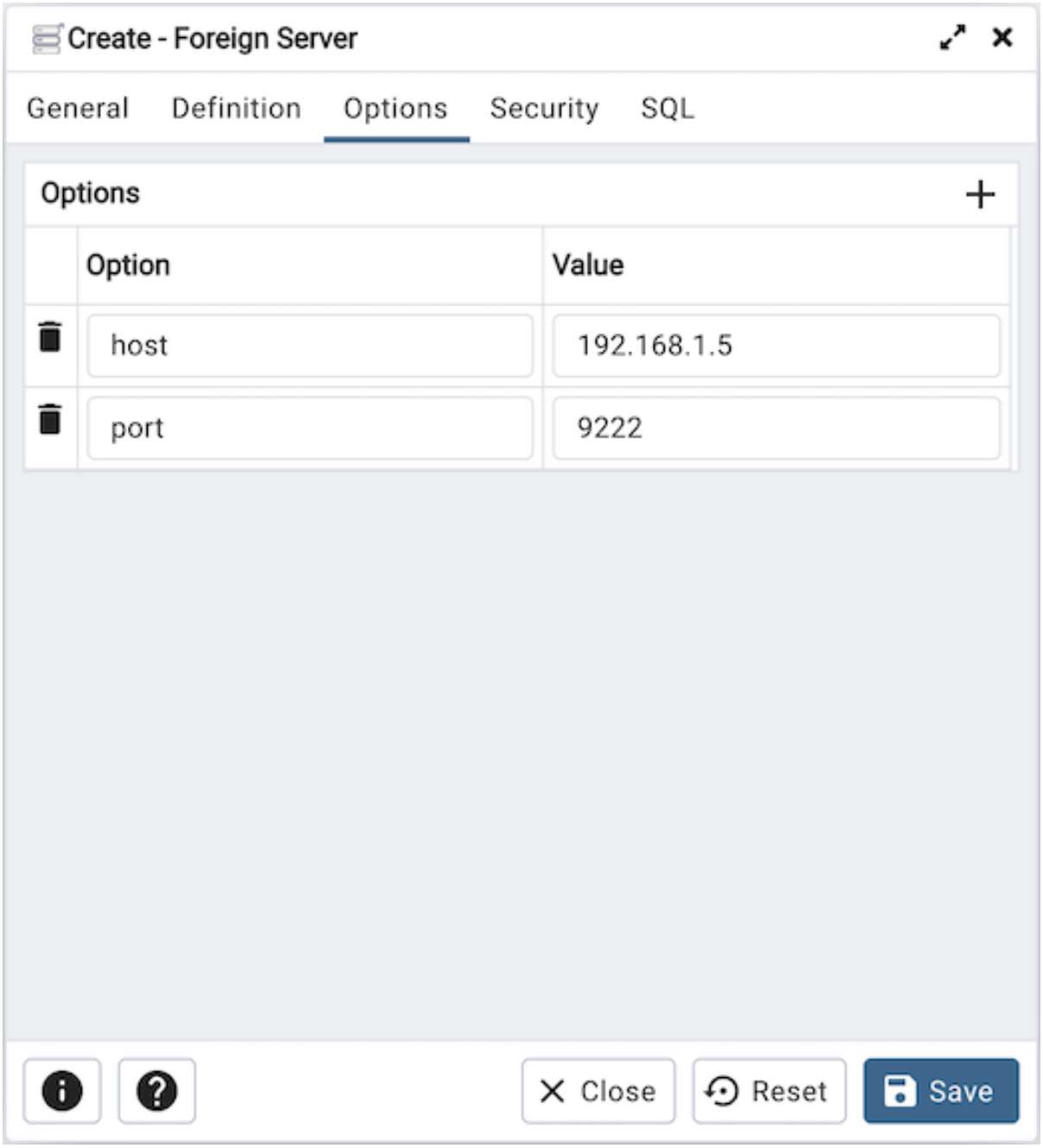
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、パラメーターを設定します。

- *Type* フィールドを使用して、サーバーの種類を指定します。
- *Version* フィールドを使用して、サーバーバージョンを指定します。

[**オプション**] タブをクリックして続行します。

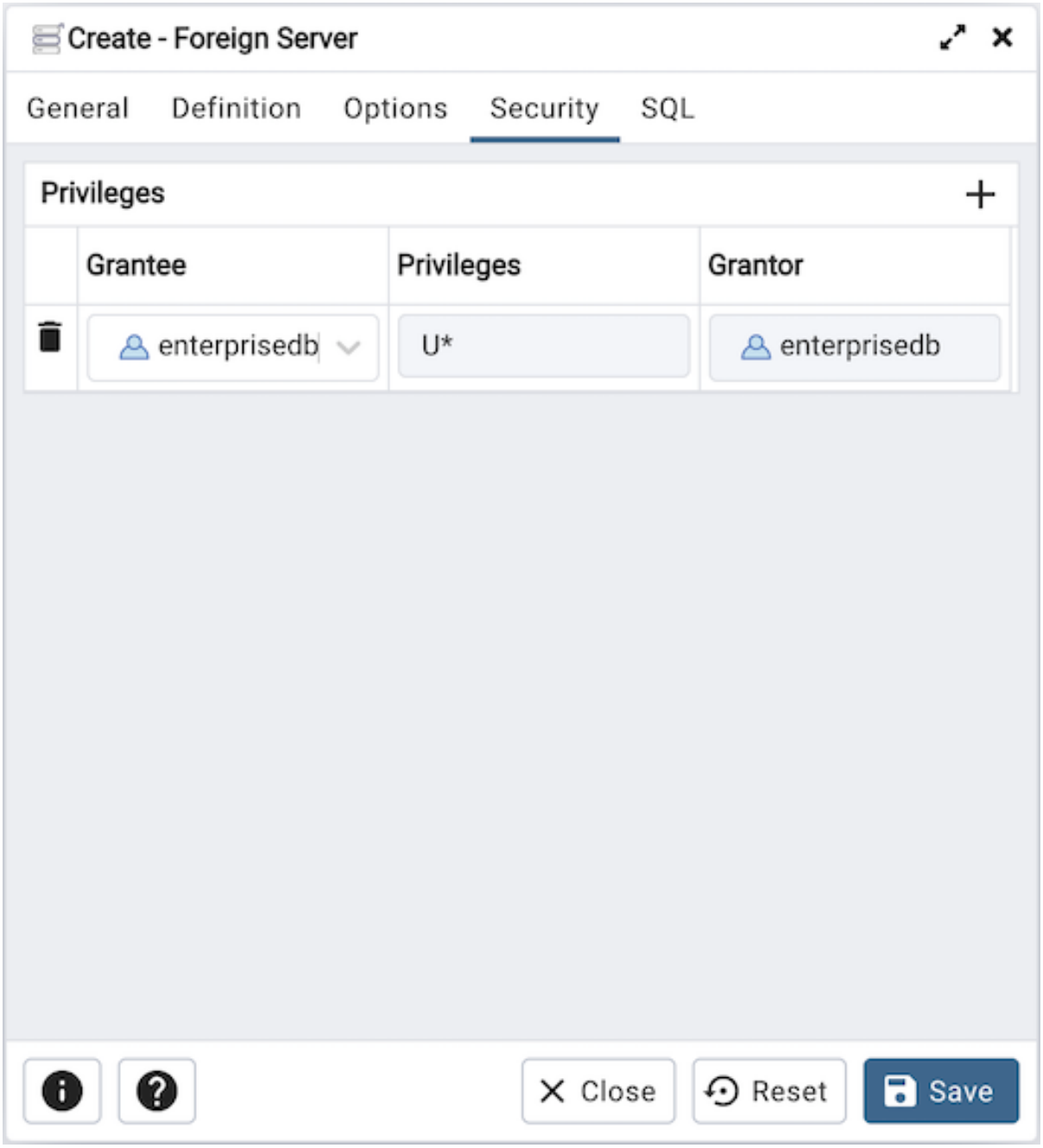


[**オプション**] タブのフィールドを使用して、オプションを指定します。 *Add* ボタンをクリックして、外部サーバーのオプション句を作成します。

- *Option* フィールドにオプション名を指定します。
- *Value* フィールドに対応する値を入力します。

Add をクリックして、追加の各句を作成します。オプションを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



セキュリティ タブを使用して、外部サーバーにセキュリティ特権を割り当てます。一連の特権を割り当てる前に、[**追加**] をクリックします。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

[**追加**] をクリックして、新しい特権セットを割り当てます。特権を破棄するには、行の左側にあるごみ箱アイコンをクリックし、 **行の削除** ダイアログで削除を確認します。

SQL タブをクリックして続行します。

ForeignServer ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.8.1 例

以下は、 **外部サーバー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、外部データラッパー *hdfs_fdw* の外部サーバーの作成を示しています。名前は *hdfs_server* です。そのタイプは *hiveserver2* です。外部サーバーのオプションには、ホストとポートが含まれます。

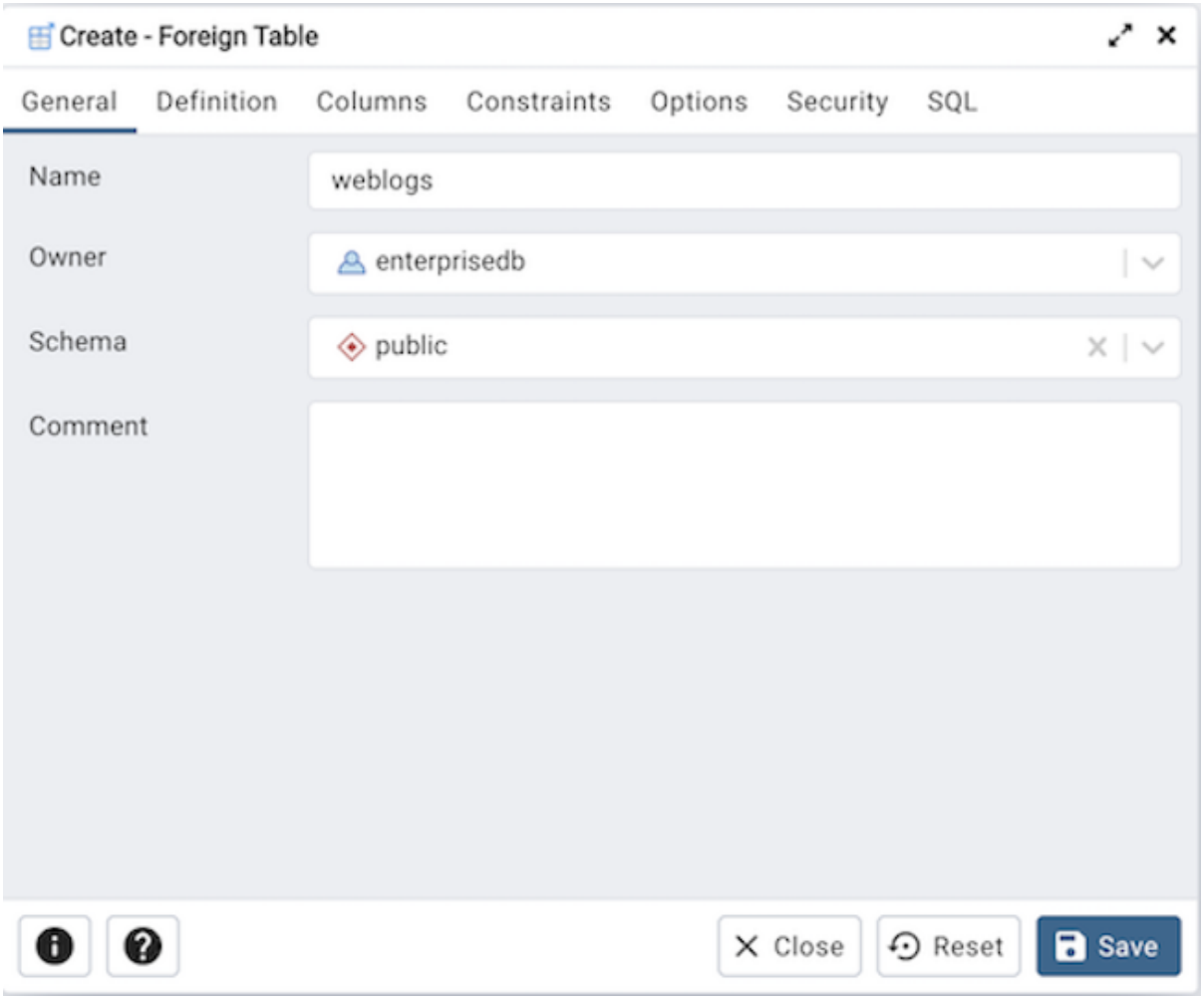
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。

- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.9 Foreign Table Dialog

外部テーブル ダイアログを使用して、現在のデータベースに外部テーブルを定義します。外部テーブルは、外部サーバーにある外部データソースの構造を定義します。

外部テーブル ダイアログは、 **一般** 、 **定義** 、 **列** 、 **制約** 、 **オプション** 、および **セキュリティ** のダイアログタブを使用して外部テーブルの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

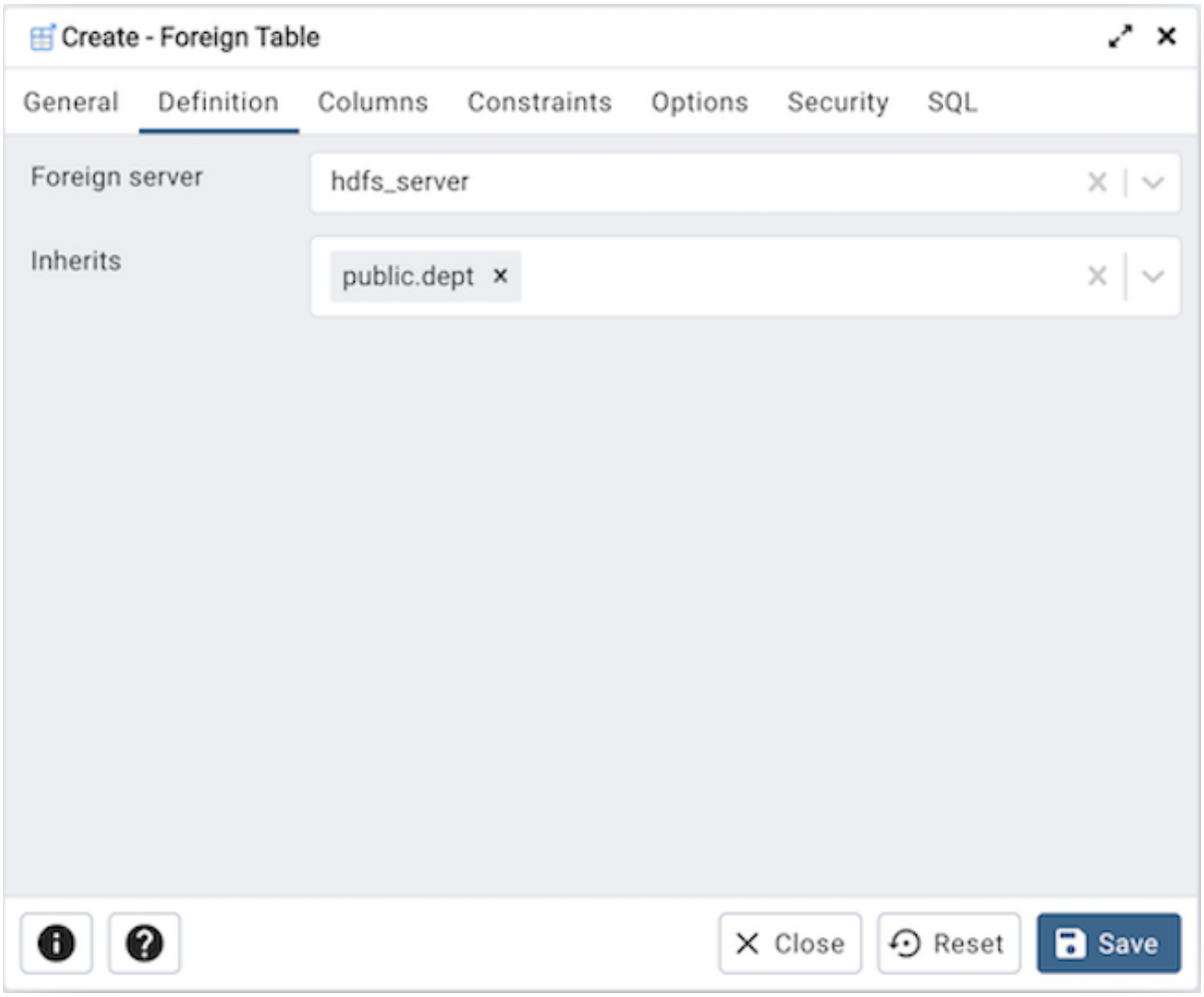


[*General*] タブのフィールドを使用して、外部テーブルを識別します。

- *Name* フィールドを使用して、外部テーブルに説明的な名前を追加します。外部テーブルの名前は、同じスキーマ内の他の外部テーブル、テーブル、シーケンス、インデックス、ビュー、既存のデータ型またはマテリアライズドビューの名前とは異なる必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、外部テーブルを所有するロールの名前を選択します。

- **スキーマ** フィールドのドロップダウンリストボックスから、外部テーブルが存在するスキーマの名前を選択します。
- **コメント** フィールドに外部テーブルに関するメモを保存します。

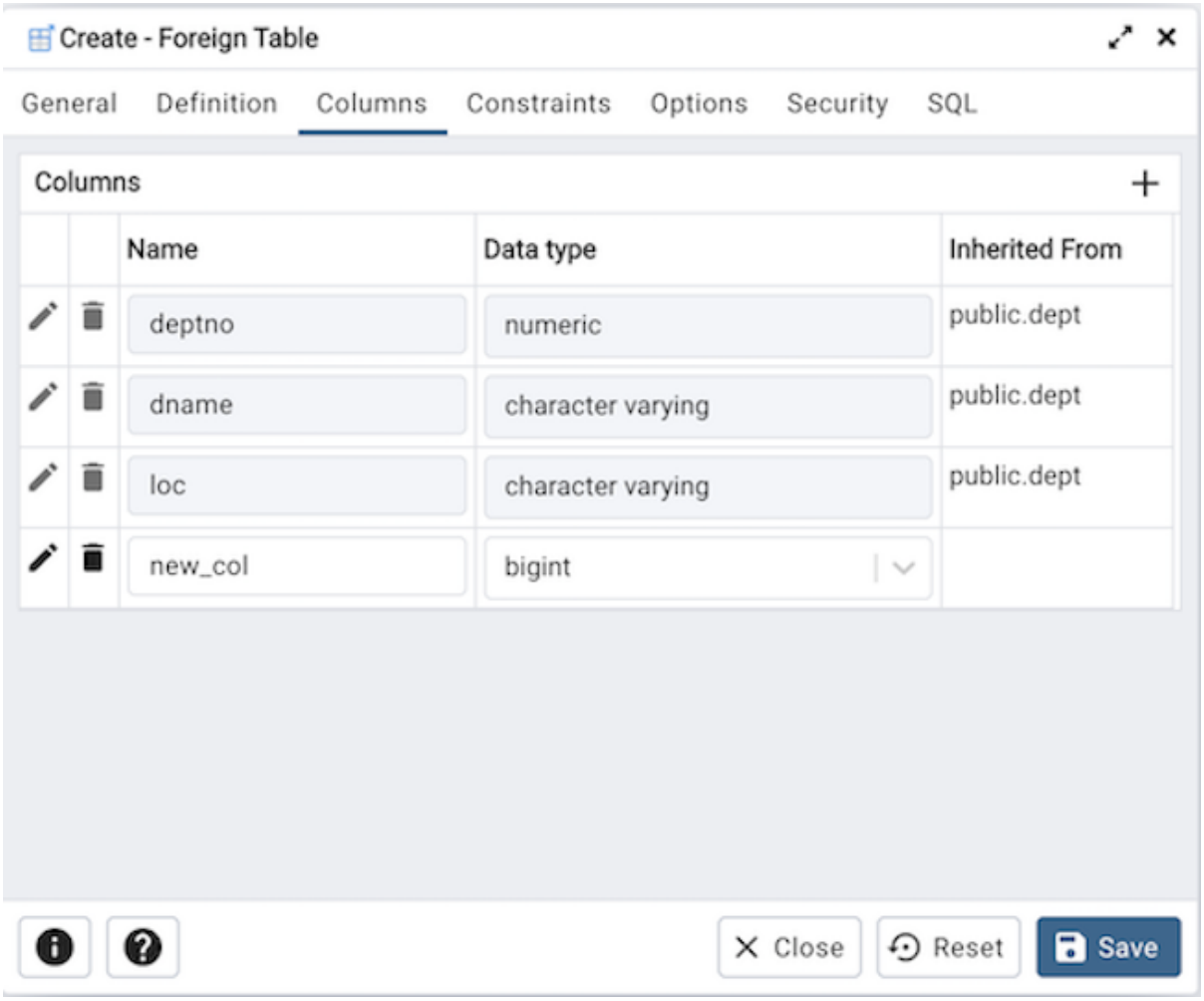
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、外部データソースを定義します。

- **外部サーバー** の横にあるドロップダウンリストボックスを使用して、外部サーバーを選択します。このリストには、 **外部サーバー** ダイアログで定義されたサーバーが入力されています。
- *Inherits* の横にあるドロップダウンリストボックスを使用して、親テーブルを指定します。外部テーブルはそのすべての列を継承します。このフィールドはオプションです。

[**列**] タブをクリックして続行します。

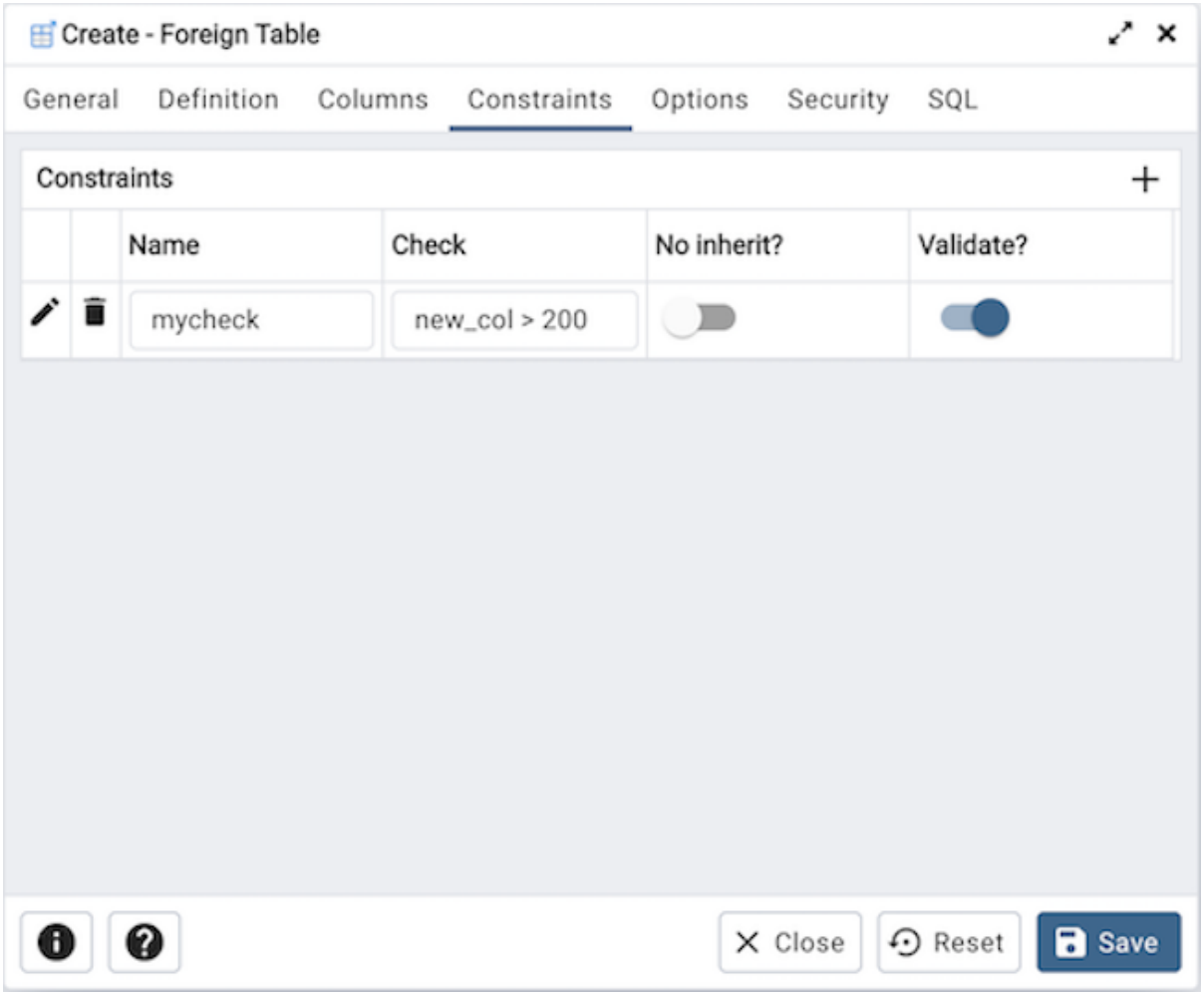


[**列**] タブのフィールドを使用して、列とその属性をテーブルに追加します。 **追加** アイコン（+）をクリックして、列を定義します。

- *Name* フィールドを使用して、列にわかりやすい名前を追加します。
- *DataType* フィールドのドロップダウンリストボックスを使用して、列のデータ型を選択します。これには配列指定子を含めることができます。PostgreSQL でサポートされているデータ型の詳細については、コアドキュメントの 8 章を参照してください。

追加 アイコン（+）をクリックして、追加の列を指定します。列を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[*Constraints*] タブをクリックして続行します。

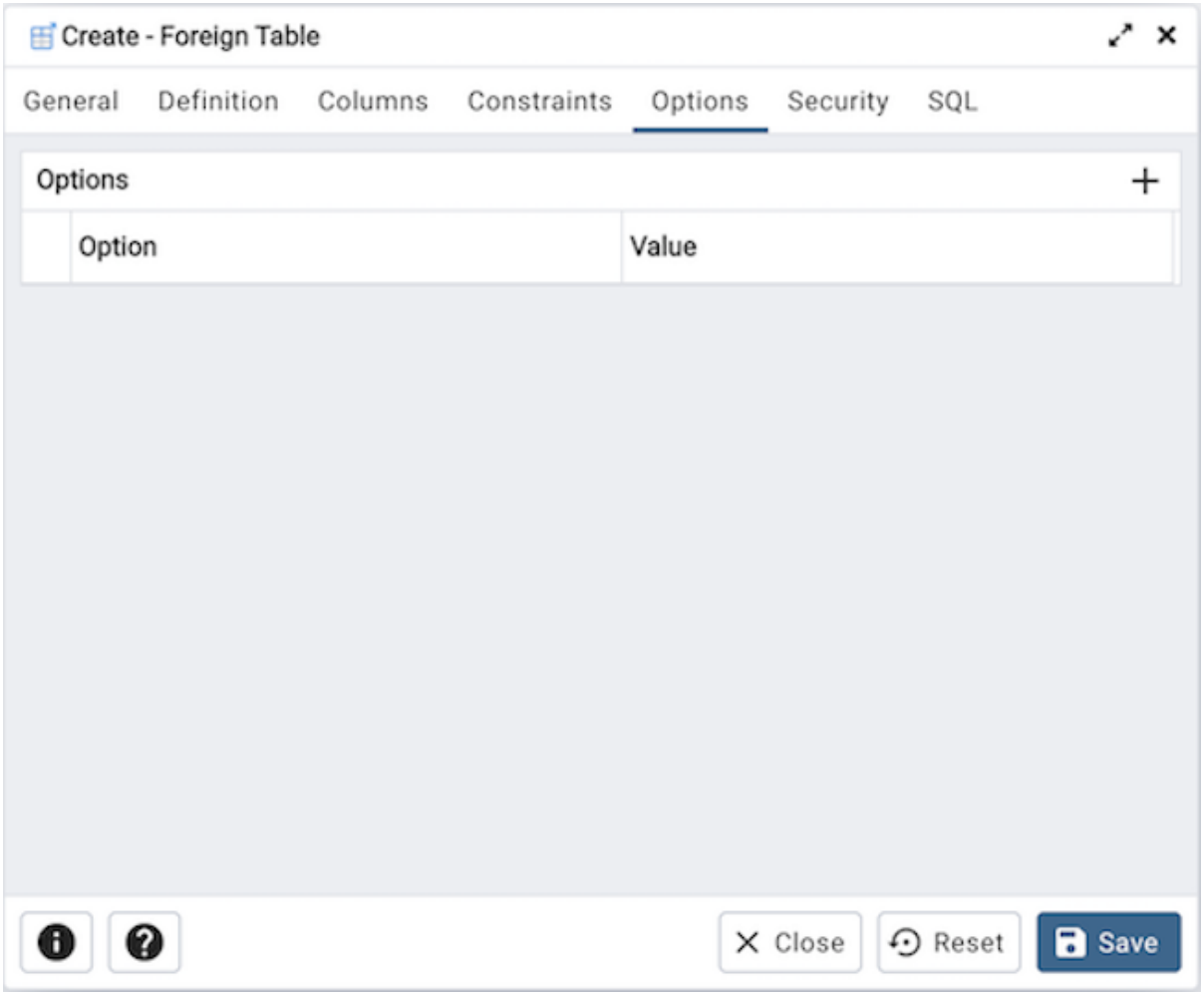


[*Constraints*] タブのフィールドを使用して、外部テーブルにテーブル制約を適用します。 **追加** アイコン（+）をクリックして、制約を定義します。

- *Name* フィールドを使用して、制約の説明的な名前を追加します。制約に違反している場合、エラーメッセージに制約名が表示されるため、 *col* **は正でなければなりません** のような制約名を使用して、役立つ情報を伝えることができます。
- *Check* フィールドを使用して、ブール結果を生成するチェック式を記述します。外部テーブルの各行は、チェック式を満たす必要があります。
- *NoInherit* チェックボックスをオンにして、制約が子テーブルに伝播しないことを指定します。
- 検証を無効にするには、 **検証** チェックボックスをオフにします。データベースは、テーブル内のすべての行に制約が適用されるとは想定していません。

追加 アイコン（+）をクリックして、追加の各制約を指定します。制約を破棄するには、行の左側にあるごみ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**オプション**] タブをクリックして続行します。

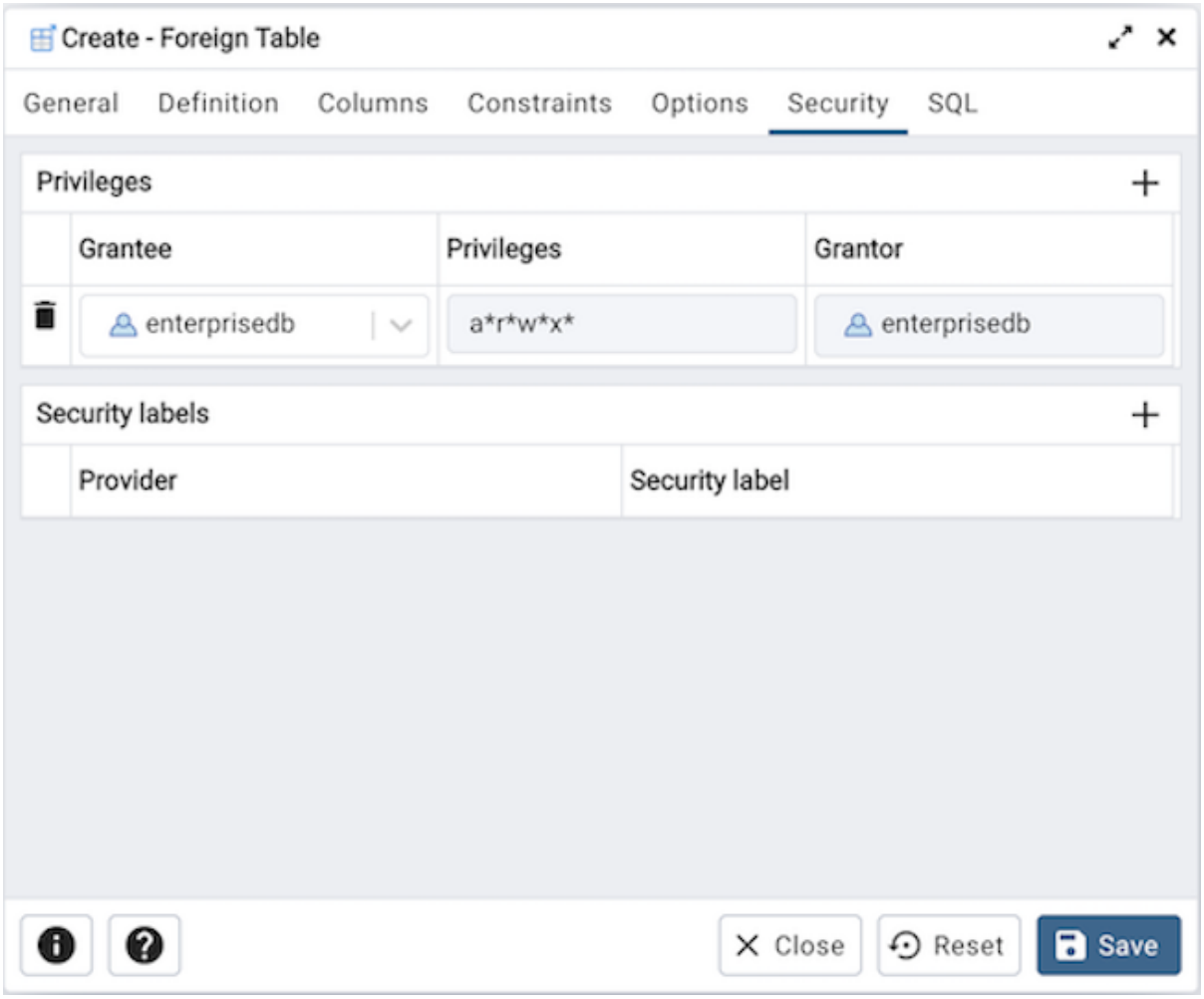


[**オプション**] タブのフィールドを使用して、新しい外部テーブルまたはその列の 1 つに関連付けるオプションを指定します。受け入れられるオプションの名前と値は、外部サーバーに関連付けられた外部データラッパーに固有です。 **追加** アイコン（+）をクリックして、オプション/値のペアを追加します。

- *Option* フィールドにオプション名を指定します。重複するオプション名は許可されません。
- *Value* フィールドに対応する値を入力します。

追加 アイコン（+）をクリックして、追加のオプション/値のペアを指定します。オプションを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、データベースオブジェクトの特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスから、特権が割り当てられるロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、ファンクションに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解

釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

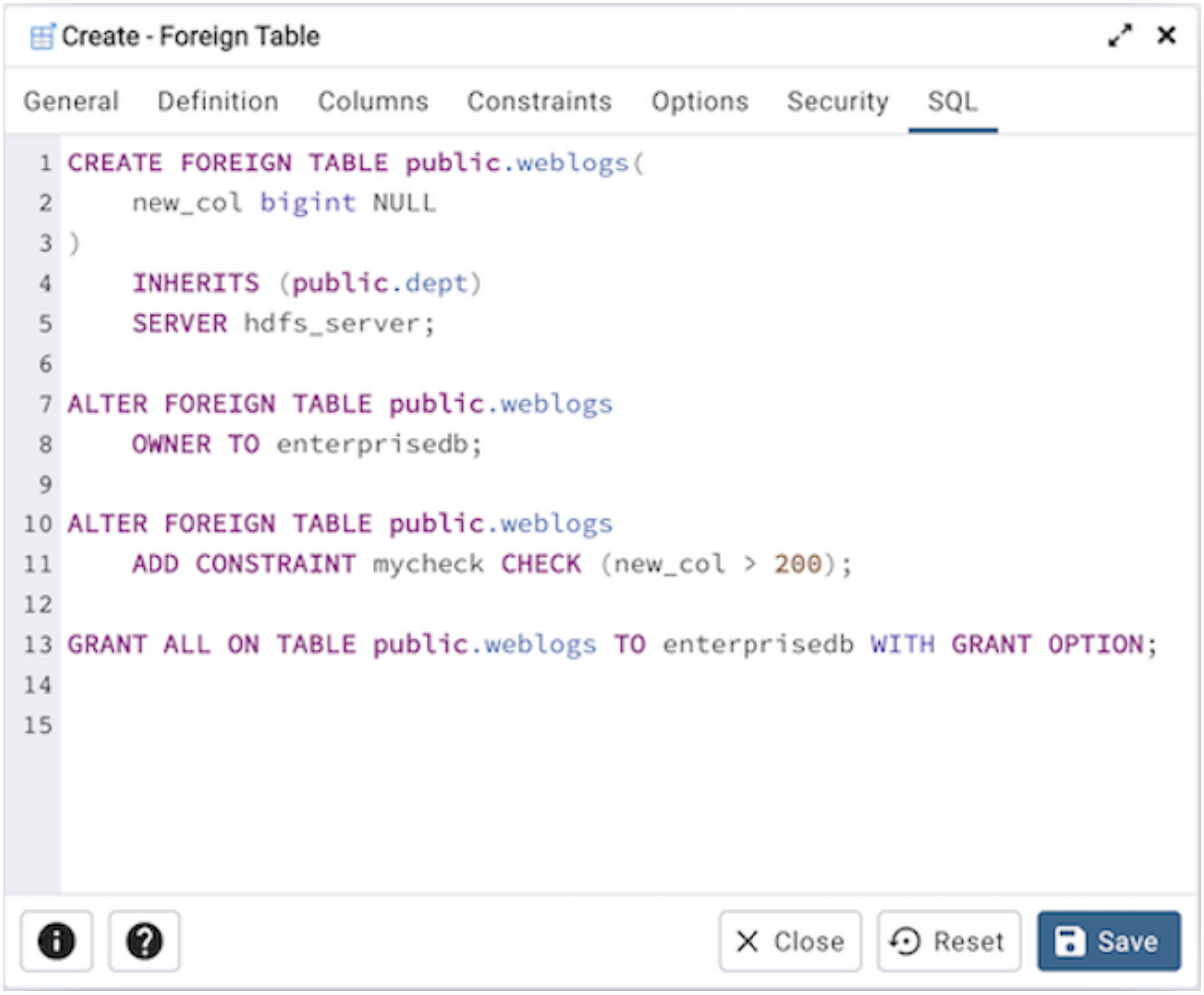
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

外部テーブル ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.9.1 例

以下は、 **外部テーブル** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、複数の列を持つ外部テーブル *weblogs* を作成する方法を示しています。

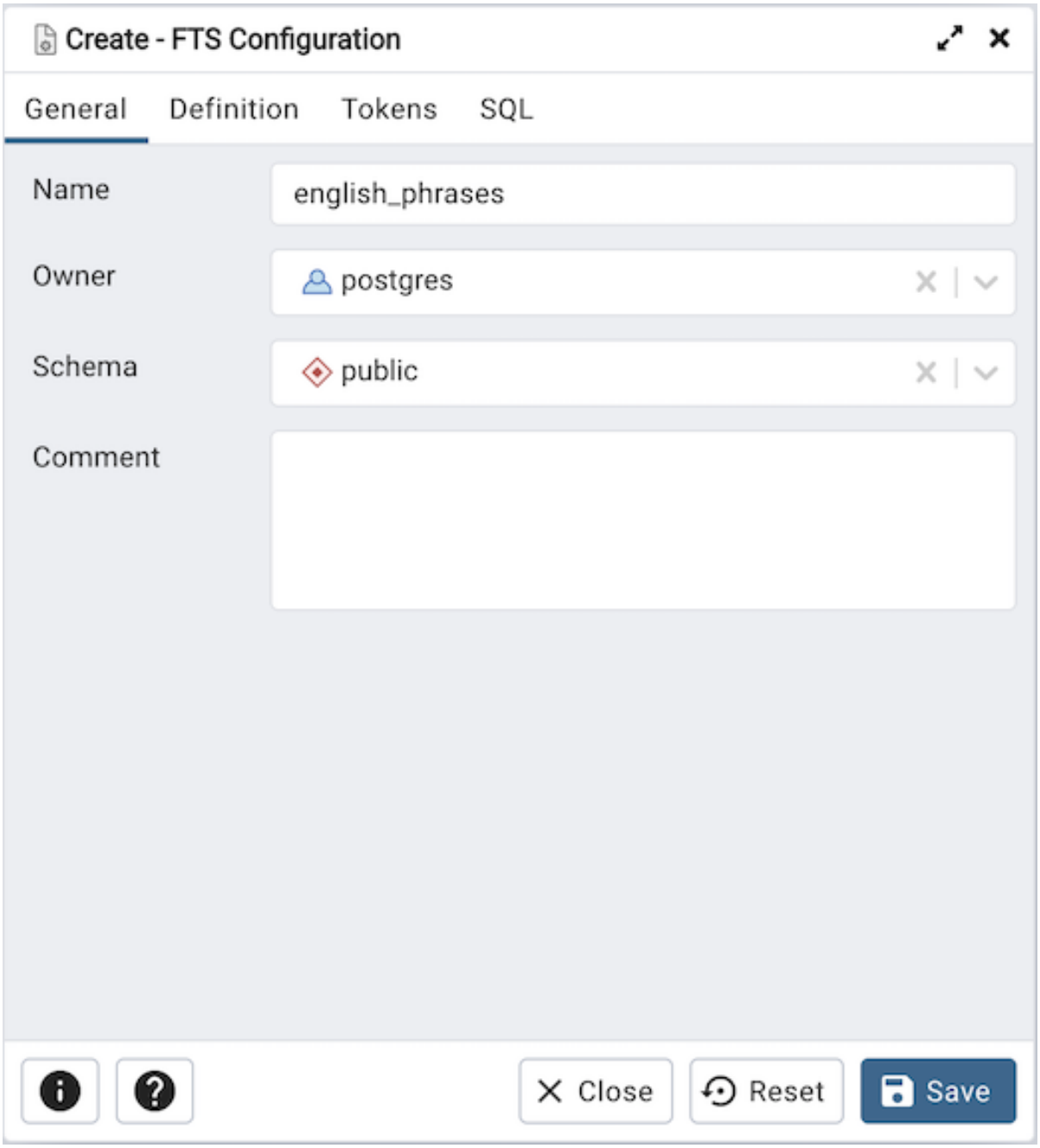
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.10 FTS Configuration Dialog

FTSConfiguration ダイアログを使用して、全文検索を構成します。テキスト検索構成は、文字列をトークンに分割できるテキスト検索パーサーと、検索可能なトークンを識別できるディクショナリを指定します。

FTSConfiguration ダイアログは、次のダイアログタブを介して FTS 構成の開発を編成します。/" *General* 、*Definition* 、および *Tokens* 。 *SQL* タブには、ダイアログの選択によって生成された SQL コードが表示されます。

[**全般**] タブをクリックして開始します。



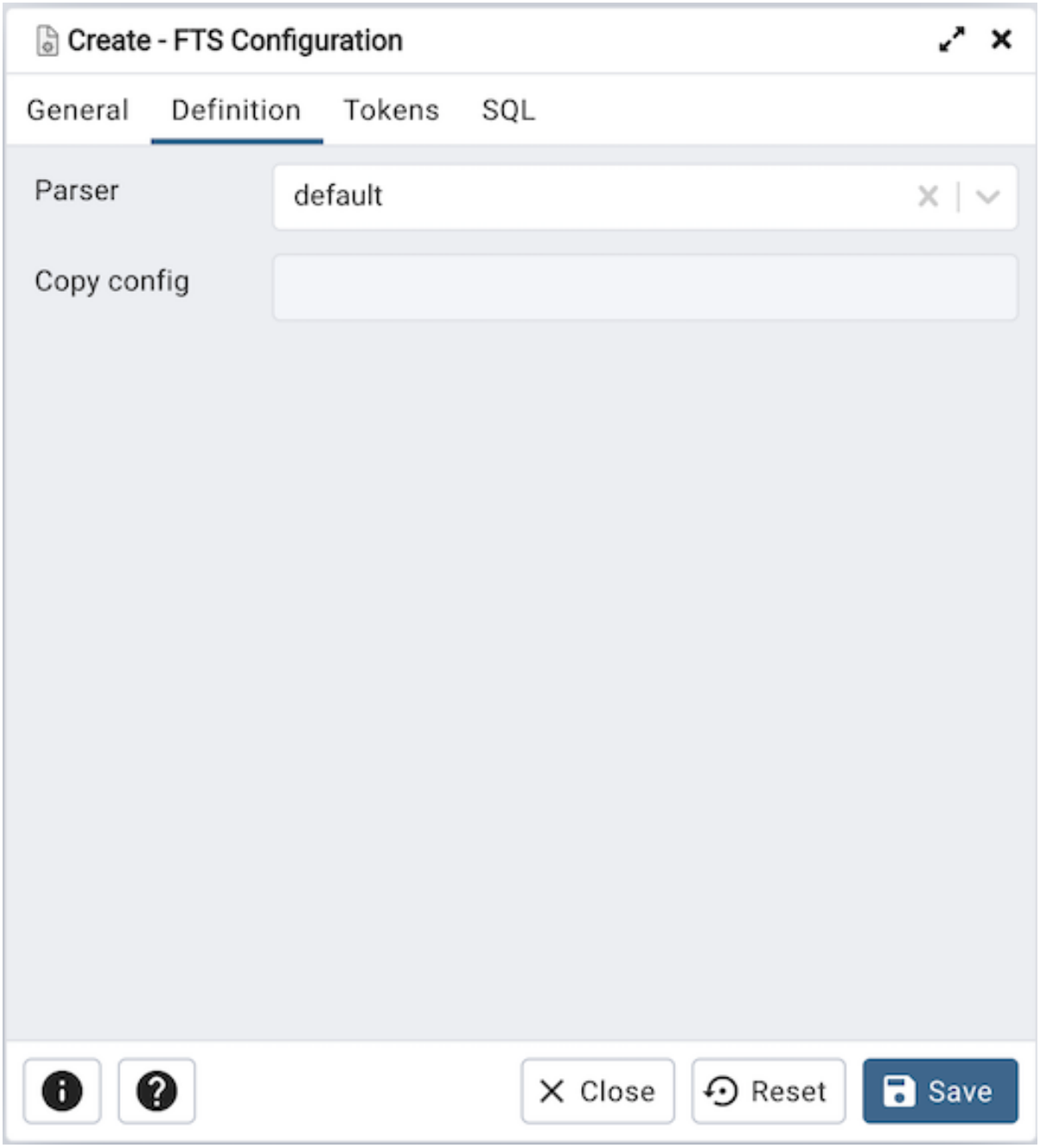
[**全般**] タブのフィールドを使用して、FTS 構成を識別します。

- *Name* フィールドを使用して、FTS 構成の説明的な名前を追加します。名前は *pgAdmin* ツリーコント

ロールに表示されます。

- **所有者** の横にあるドロップダウンリストボックスを使用して、構成を所有するロールを指定します。
- **スキーマ** フィールドのドロップダウンリストボックスから、FTS 構成が存在するスキーマの名前を選択します。
- **コメント** フィールドに FTS 構成に関するメモを保存します。

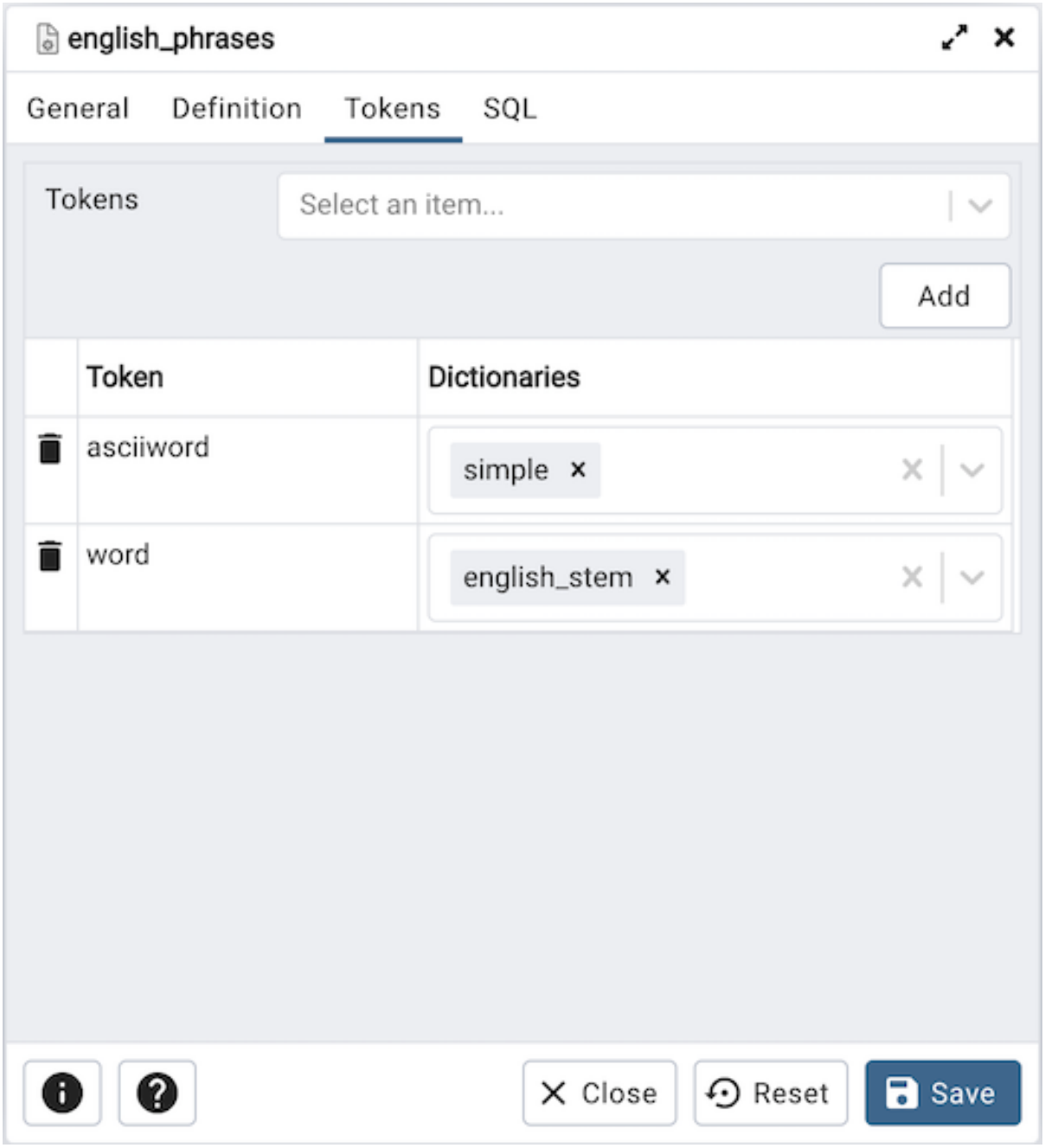
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、パラメーターを定義します。

- **パーサー** フィールドのドロップダウンリストボックスからテキスト検索パーサーの名前を選択します。
- **構成のコピー** フィールドのドロップダウンリストボックスから言語を選択します。

[**トークン**] タブをクリックして続行します。



[**トークン**] タブのフィールドを使用して、トークンを追加します。

- *Tokens* フィールドを使用して、トークンの名前を指定します。
- *Add* ボタンをクリックしてトークンを作成します。
- *Dictionaries* フィールドを使用して、辞書を指定します。

これらの手順を繰り返して、さらにトークンを追加します。トークンを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

FTSConfiguration ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には

SQL タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.10.1 例

以下は、 *FTSConfiguration* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



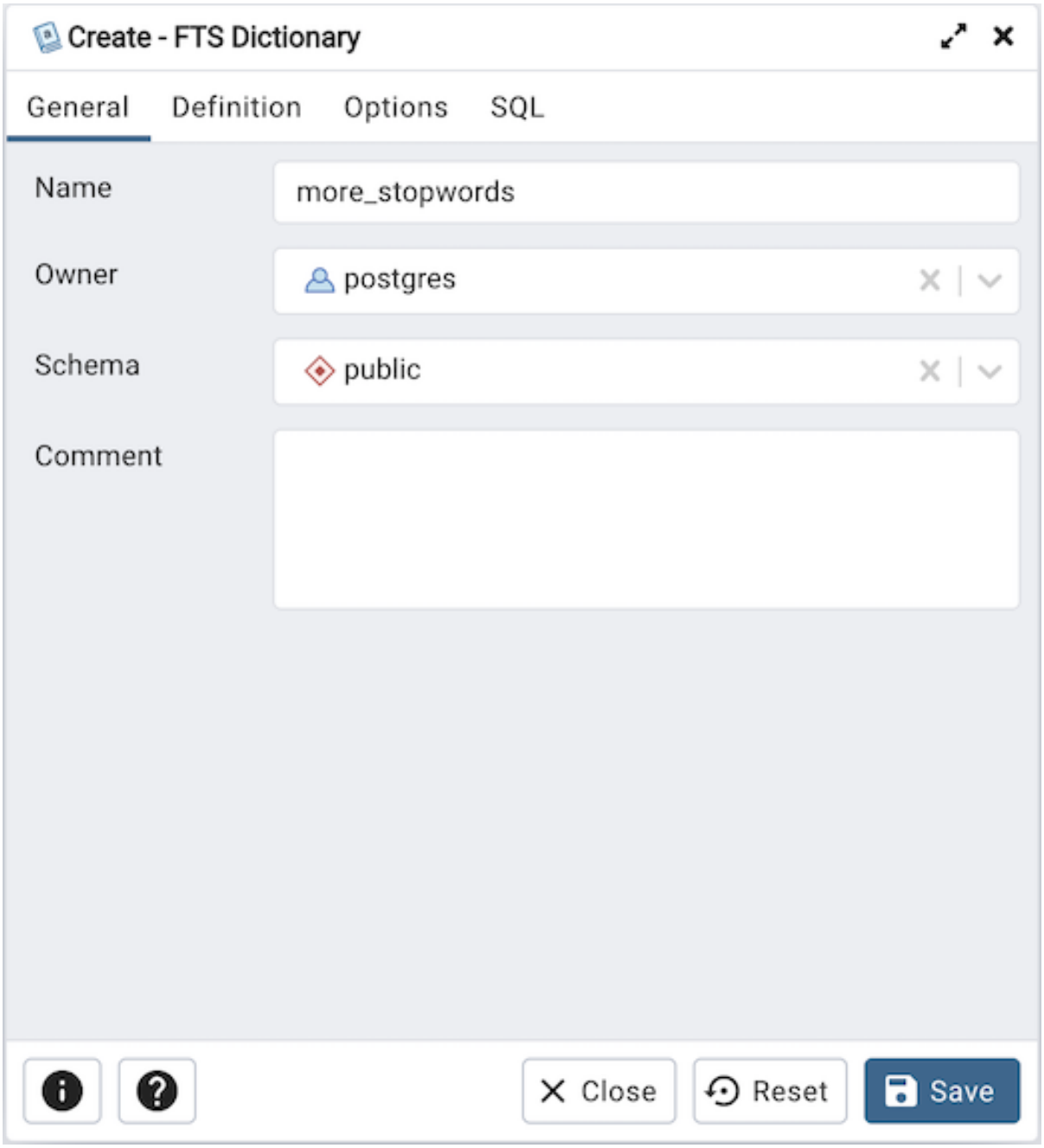
示されている例は、 *english_phrases* という名前の FTS 構成を作成する方法を示しています。 **デフォルト** のパーサーを使用します。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.11 FTS Dictionary Dialog

FTSDictionary ダイアログを使用して、全文検索辞書を作成します。事前定義されたテンプレートを使用するか、カスタムパラメーターで新しい辞書を作成できます。

FTS **辞書** ダイアログは、 **一般** 、 **定義** 、および **オプション** のダイアログタブを介して FTS 辞書の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



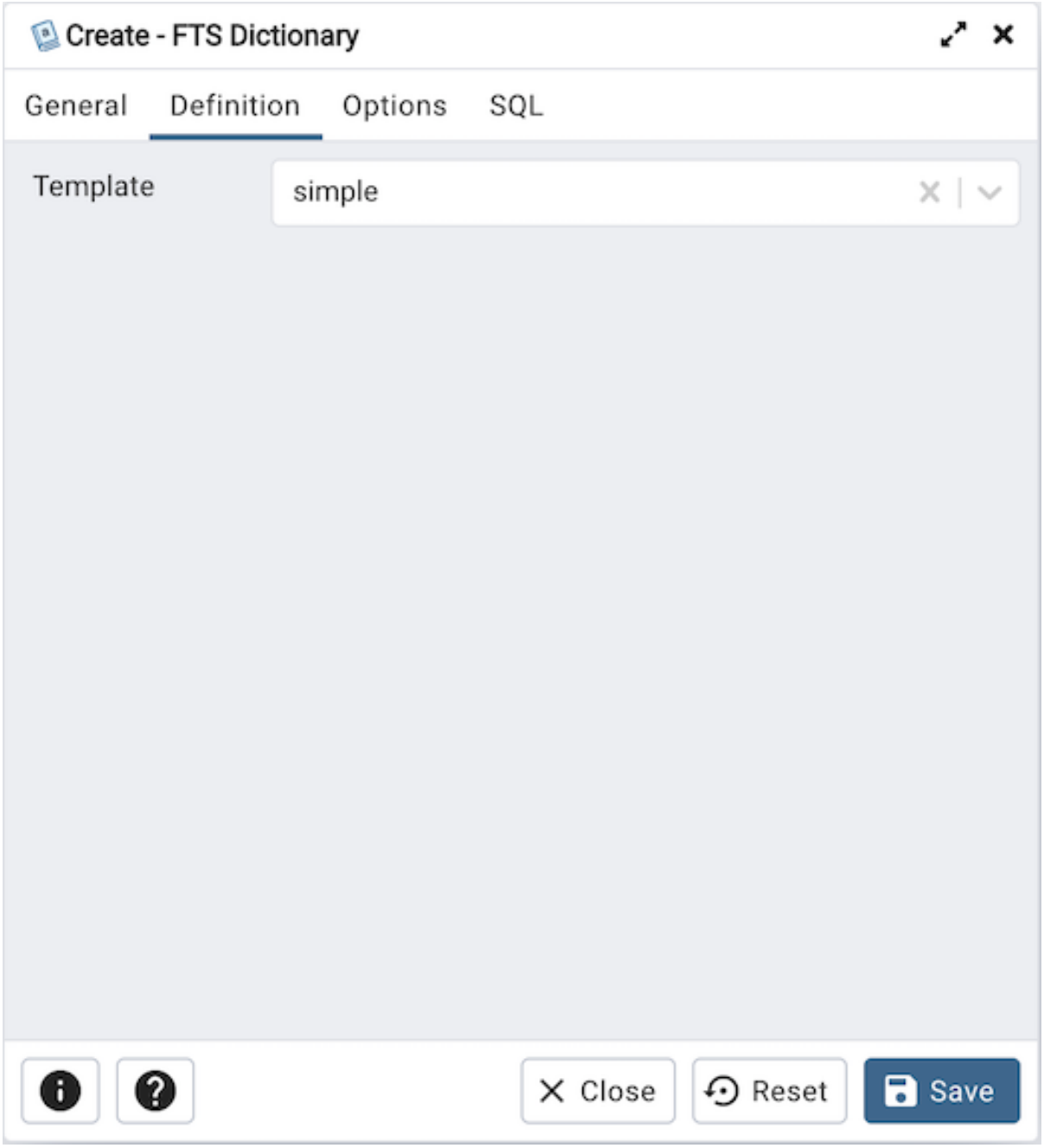
[**全般**] タブのフィールドを使用して、辞書を識別します。

- *Name* フィールドを使用して、辞書の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、FTS 辞書を所有するロールを選択し

ます。

- **スキーマ** フィールドのドロップダウンリストボックスから、ディクショナリを配置するスキーマの名前を選択します。
- 辞書に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、ドロップダウンリストボックスからテンプレートを選択します。

- *ispell* を選択して Ispell テンプレートを選択します。Ispell 辞書テンプレートは、形態素辞書をサポートしており、単語のさまざまな言語形式を同じ語彙素に正規化できます。たとえば、英語の Ispell 辞書は、検索用語 bank（banking、banked、banks、banks'、bank's など）のすべての変化形と活用形に一致します。Ispell 辞書は通常、限られた単語を認識するため、その後に別のより広い辞書を続ける

必要があります。たとえば、すべてを認識する Snowball 辞書です。

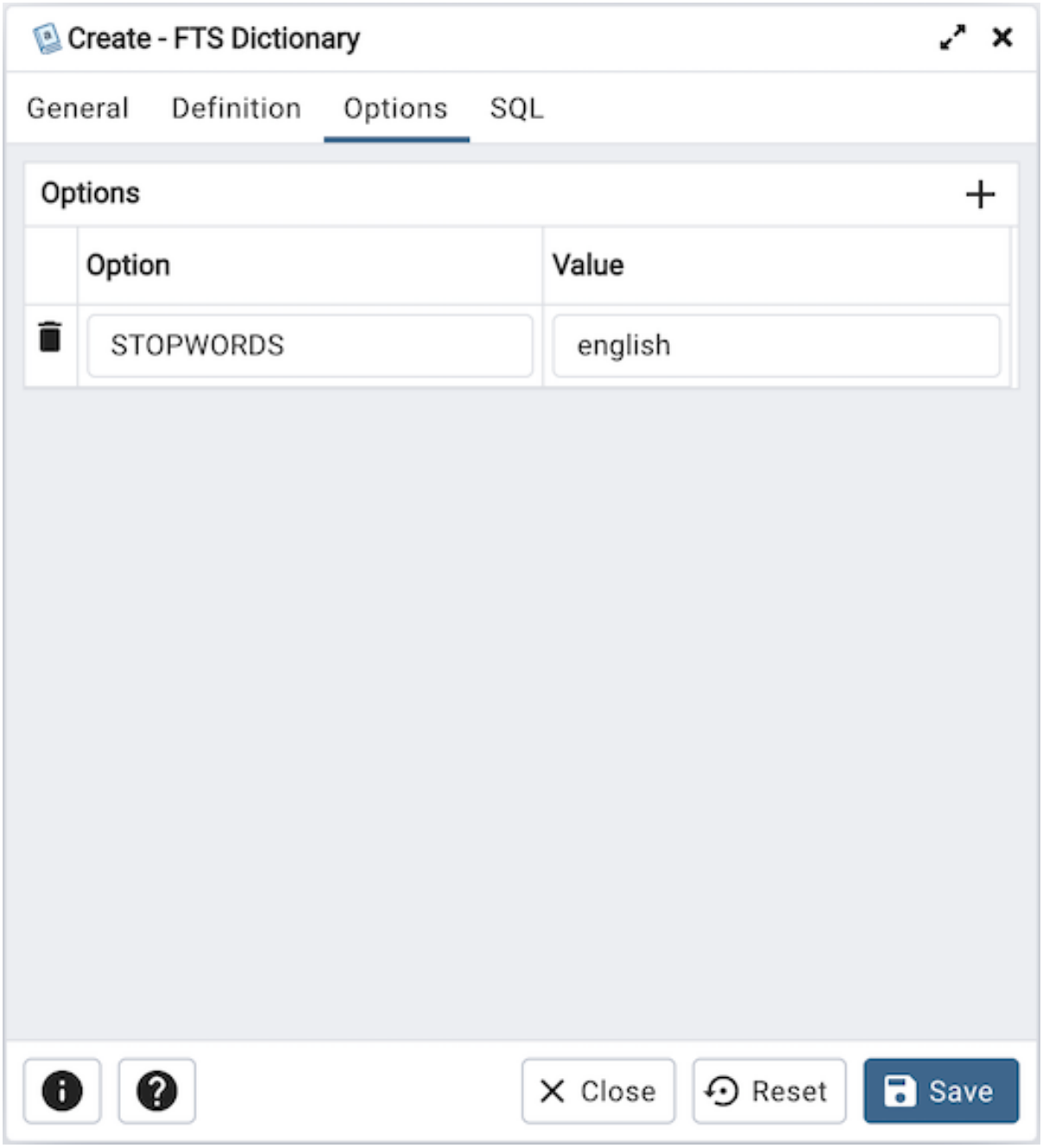
- *simple* を選択して、シンプルなテンプレートを選択します。単純な辞書テンプレートは、入力トークンを小文字に変換し、ストップワードのファイルと照合することによって動作します。ファイルで見つかった場合は空の配列が返され、トークンが破棄されます。そうでない場合、単語の小文字が正規化された語彙素として返されます。または、ノンストップワードを認識できないと報告するように辞書を構成し、リスト内の次の辞書に渡すことができます。

- *snowball* を選択して Snowball テンプレートを選択します。Snowball 辞書テンプレートは、人気のある Porter の英語ステミングアルゴリズムの発明者である MartinPorter によるプロジェクトに基づいています。Snowball は、多くの言語のステミングアルゴリズムを提供するようになりました (詳細については、Snowball サイトを参照してください)。各アルゴリズムは、単語の一般的なバリアント形式をその言語内の基本または語幹に削減する方法を理解しています。Snowball 辞書は、単語を単純化できるかどうかにかかわらず、すべてを認識するため、辞書リストの最後に配置する必要があります。トークンは次の辞書に渡さないため、他の辞書の前に持っても意味がありません。

- *synonym* を選択して、同義語テンプレートを選択します。この辞書テンプレートは、単語を同義語に置き換える辞書を作成するために使用されます。フレーズはサポートされていません（シソーラステンプレート（セクション 12.6.4）を使用してください)。同義語辞書を使用すると、言語の問題を解決できます。たとえば、英語のステマー辞書が単語 Paris を pari に削減しないようにすることができます。

- **シソーラス** を選択して、シソーラステンプレートを選択します。シソーラス辞書は、すべての非優先用語を 1 つの優先用語に置き換え、オプションで、インデックス付けのために元の用語も保存します。PostgreSQL のシソーラス辞書の現在の実装は、フレーズサポートが追加された同義語辞書の拡張です。

[**オプション**] タブをクリックして続行します。



[**オプション**] タブのフィールドを使用して、テンプレート固有のオプションを提供します。 **追加** アイコン（+）をクリックして、オプション句を追加します。

- *Option* フィールドにオプションの名前を指定します
- *Value* フィールドにオプションの値を入力します。

追加 アイコン（+）をクリックして、追加のオプション/値のペアを指定します。オプションを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

FTSDictionary ダイアログに入力すると、SQL コマンドが生成されます。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.11.1 例

以下は、 *FTSDictionary* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



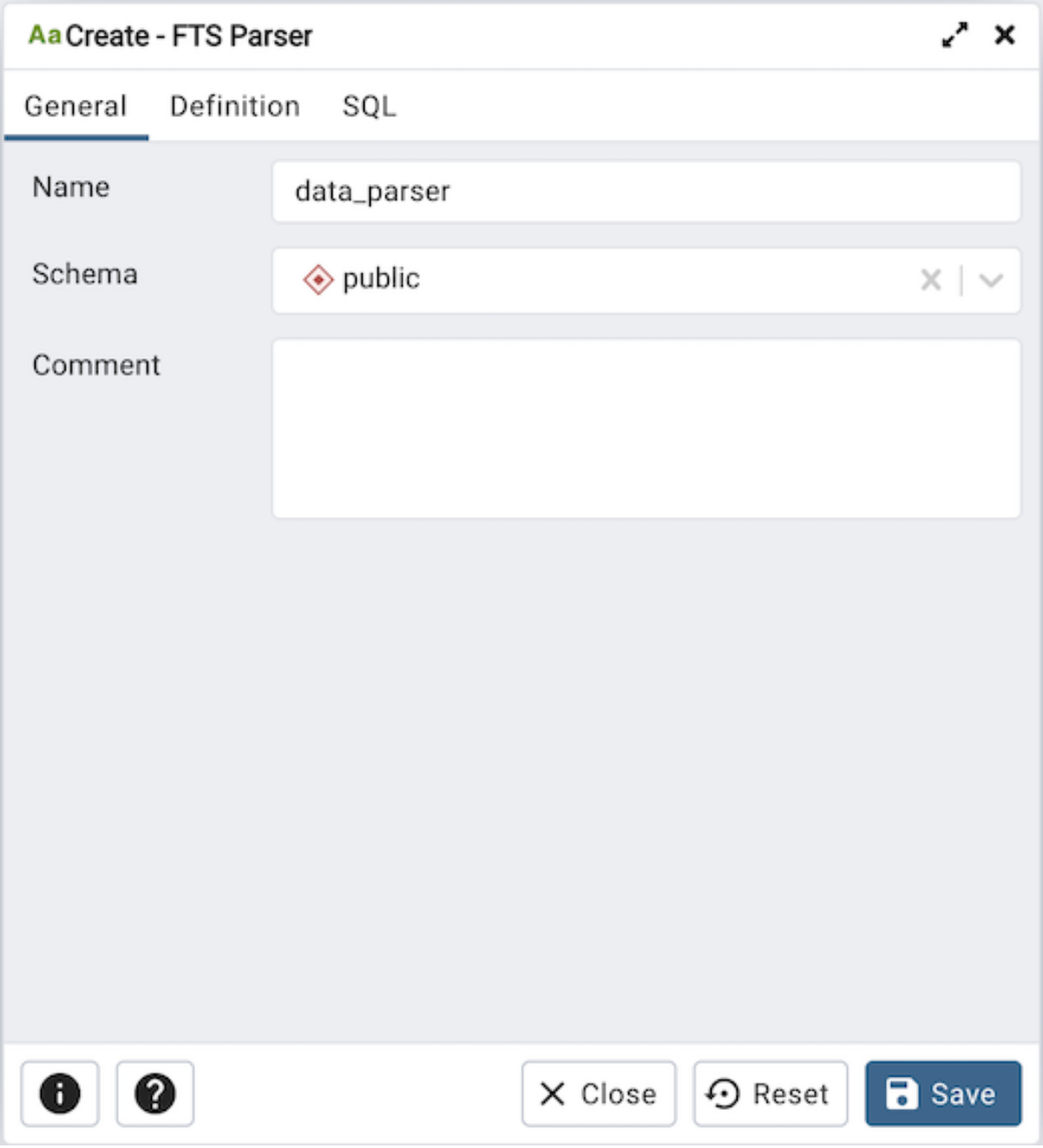
ここに示す例は、単純なテンプレートに基づいており、標準の英語を使用するように構成された *more_stopwords* という名前のカスタム辞書を作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.12 FTS Parser Dialog

FTS **パーサー** ダイアログを使用して、新しいテキスト検索パーサーを作成します。テキスト検索パーサーは、テキスト文字列をトークンに分割し、タイプ（カテゴリ）をトークンに割り当てる方法を定義します。

FTS **パーサー** ダイアログは、次のダイアログタブを介してテキスト検索パーサーの開発を編成します。 **一般** および **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



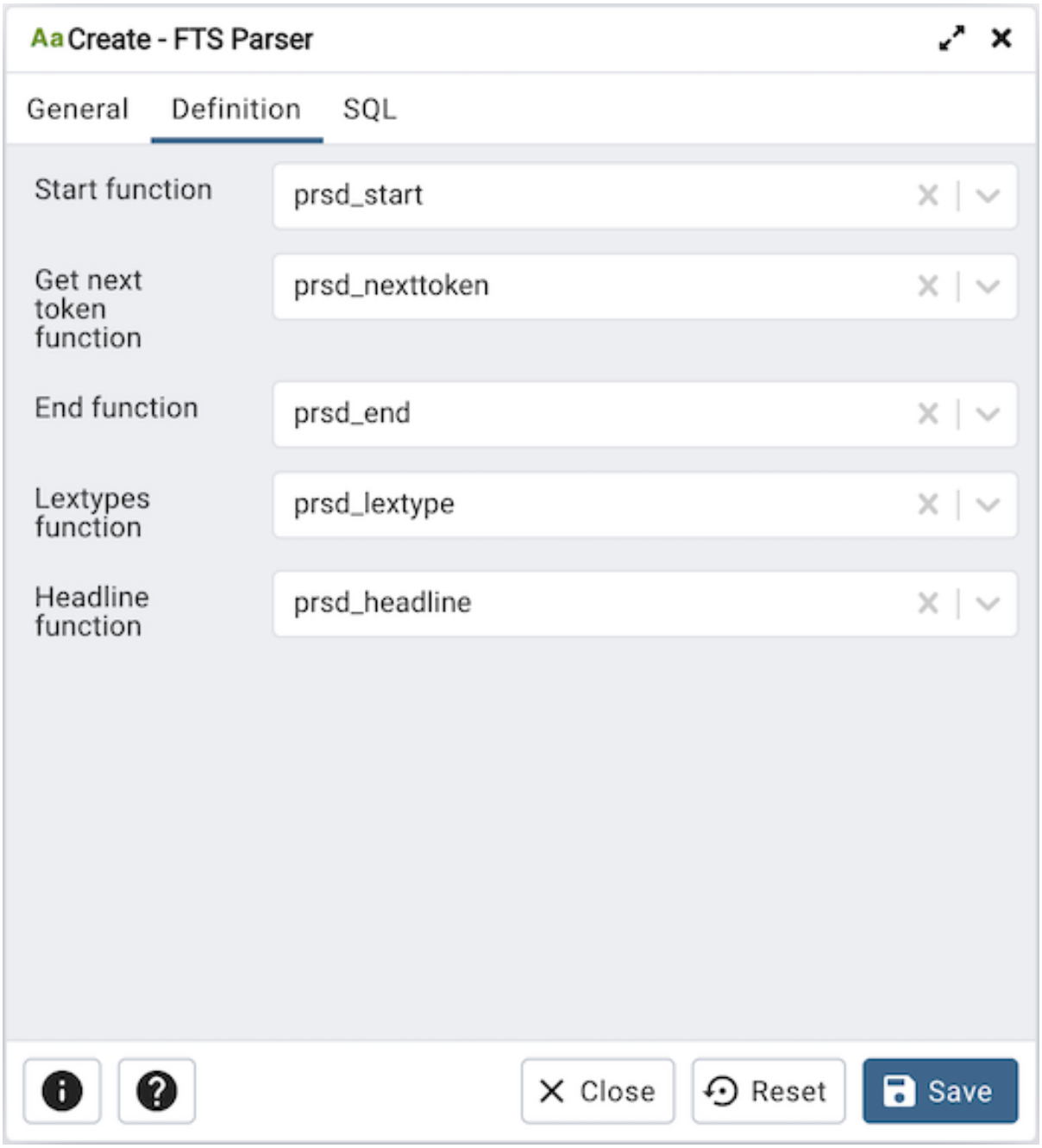
[*General*] タブのフィールドを使用して、テキスト検索パーサーを識別します。

- *Name* フィールドを使用して、パーサーの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **スキーマ** フィールドのドロップダウンリストボックスから、パーサーが存在するスキーマの名前を

選択します。

- ドメインに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、パラメーターを定義します。

- *Startfunction* の横にあるドロップダウンリストボックスを使用して、パーサーを初期化する関数の名前を選択します。
- **次のトークン関数を取得** の横にあるドロップダウンリストボックスを使用して、次のトークンを返す関数の名前を選択します。
- *Endfunction* の横にあるドロップダウンリストボックスを使用して、パーサーが終了したときに呼び出される関数の名前を選択します。

- *Lextypesfunction* の横にあるドロップダウンリストボックスを使用して、パーサーの lextypes ファンクションの名前を選択します。lextypes ファンクションは、パーサーが使用するトークンの ID、エイリアス、説明を含む配列を返します。
- **見出し関数** の横にあるドロップダウンリストボックスを使用して、パーサーの見出し関数の名前を選択します。headline 関数は、クエリの用語が強調表示されているドキュメントからの抜粋を返します。

SQL タブをクリックして続行します。



FTS **パーサー** ダイアログに入力すると、SQL コマンドが生成されます。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

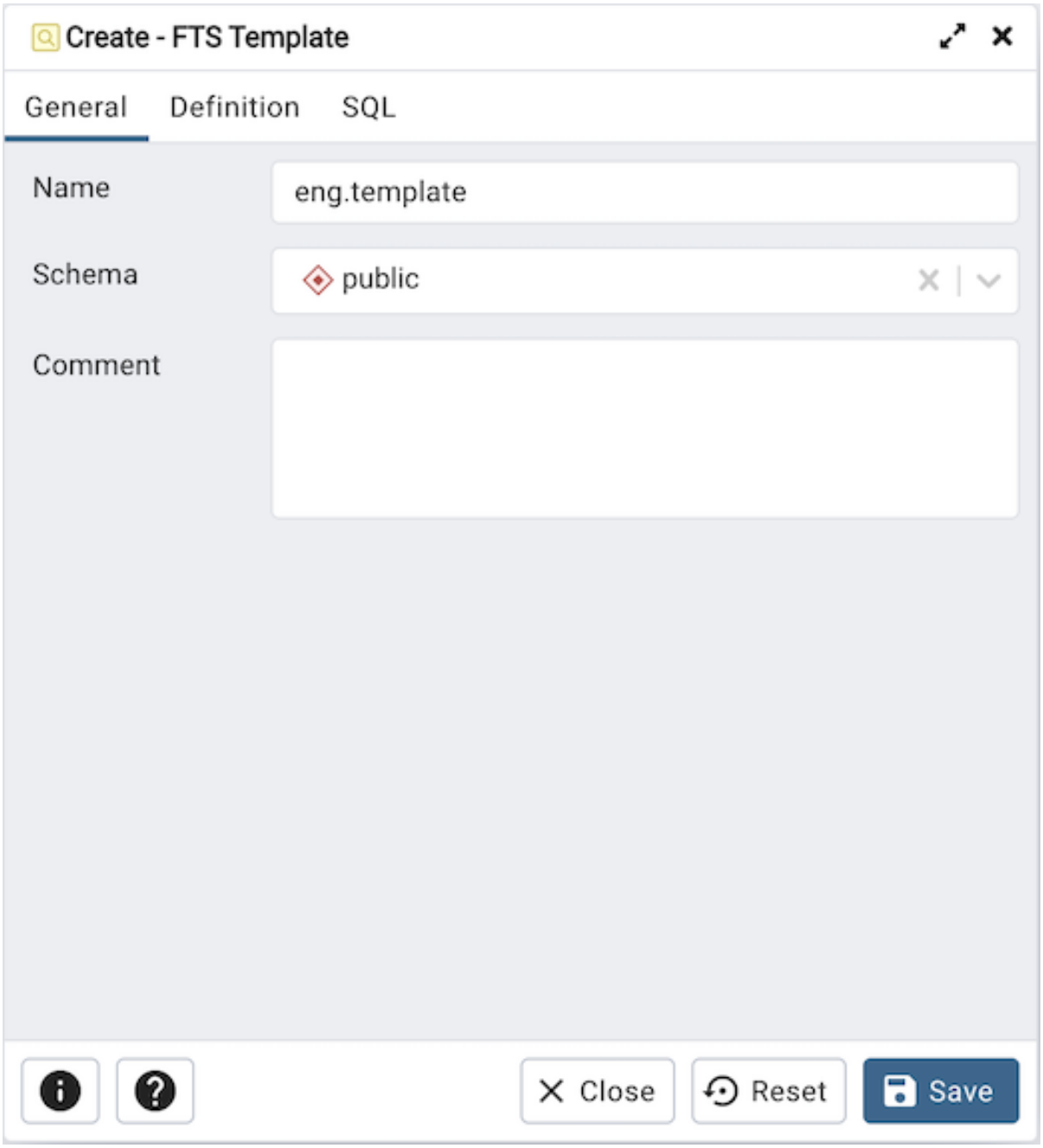
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。

- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.13 FTS Template Dialog

FTSTemplate ダイアログを使用して、新しいテキスト検索テンプレートを作成します。テキスト検索テンプレートは、テキスト検索辞書を実装する関数を定義します。

FTSTemplate ダイアログは、次のダイアログタブを介してテキスト検索テンプレートの開発を編成します。*General* 、および *Definition* 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、テンプレートを識別します。

- *Name* フィールドを使用して、テンプレートの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- テンプレートを配置するスキーマの名前を、 **スキーマ** フィールドのドロップダウンリストボックスから選択します。
- テンプレートに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、関数パラメーターを定義します。

- *Initfunction* の横にあるドロップダウンリストボックスを使用して、テンプレートの init 関数の名前を選択します。init 関数はオプションです。

- *Lexizefunction* の横にあるドロップダウンリストボックスを使用して、テンプレートの lexize 関数の名前を選択します。lexize 機能は必須です。

SQL タブをクリックして続行します。

FTSTemplate ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.13.1 例

以下は、 *FTSTemplate* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



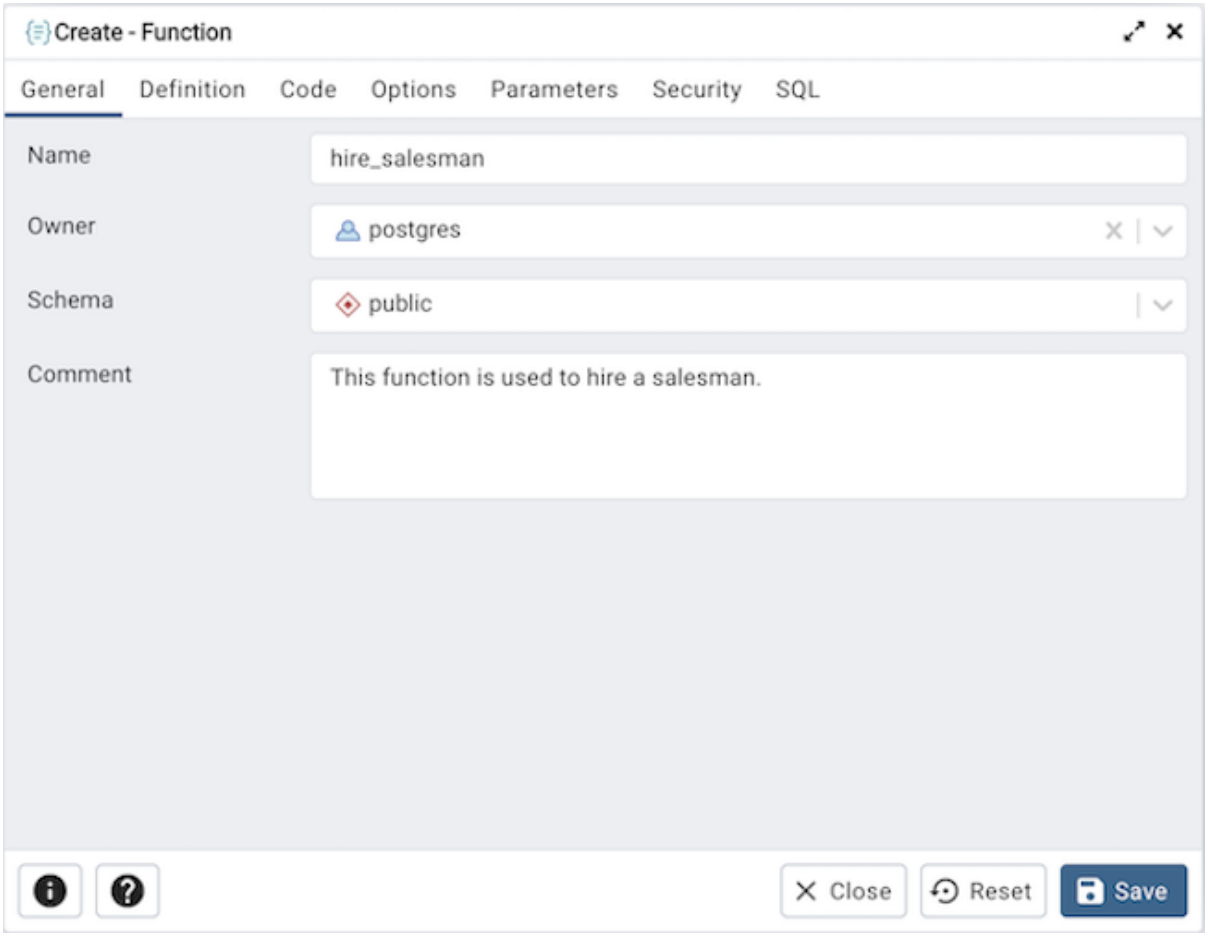
ここに示す例は、ispell 辞書を使用する *eng.template* という名前の fts テンプレートを作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.14 Function Dialog

Function ダイアログを使用して、関数を定義します。ファンクションを削除してから再作成した場合、新しいファンクションは古いエンティティと同じではありません。古い関数を参照する既存のルール、ビュー、トリガーなどを削除する必要があります。

[**ファンクション**] ダイアログは、 **一般** 、 **定義** 、 **コード** 、 **オプション** 、 **パラメーター** 、および **セキュリティ** のダイアログタブを使用して関数の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

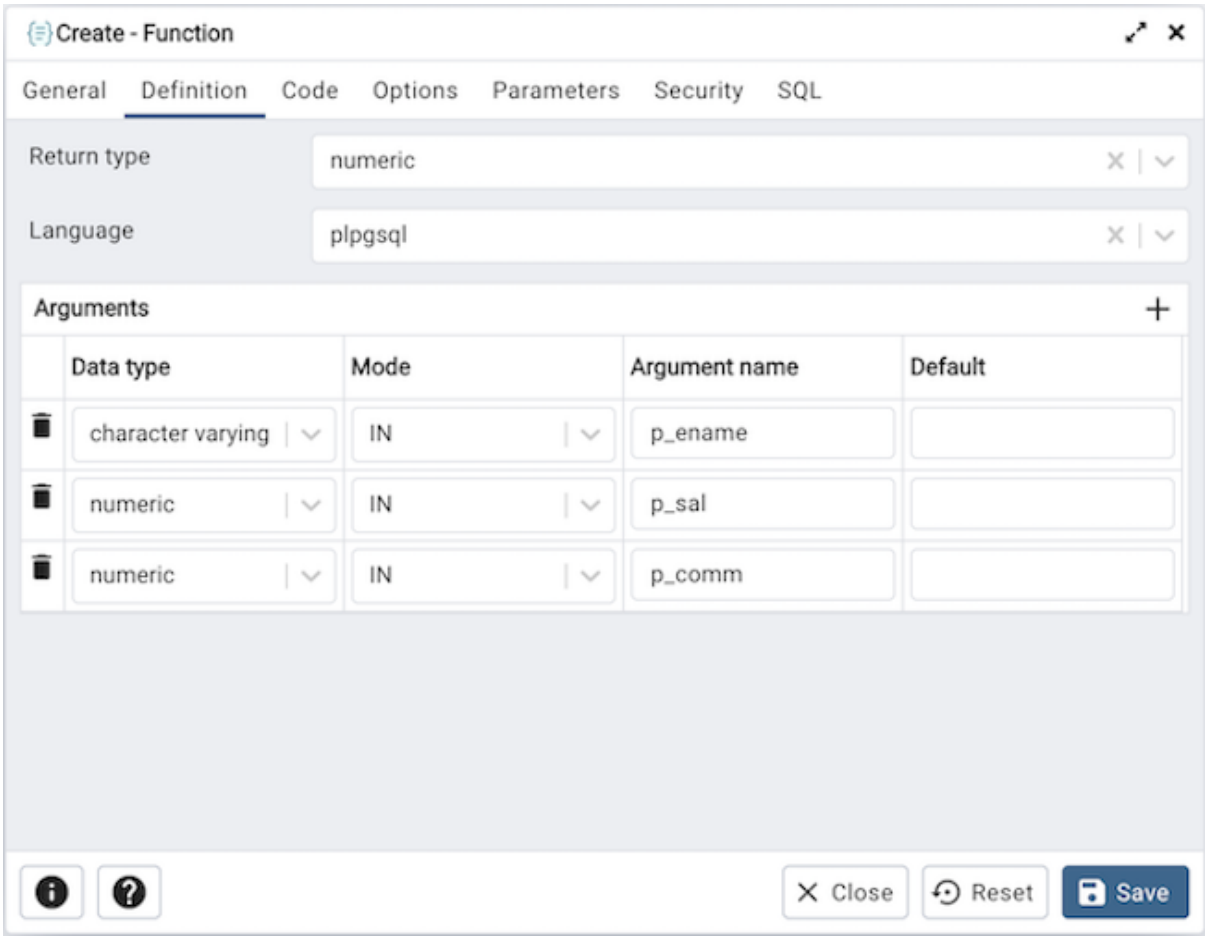


[**全般**] タブのフィールドを使用して、機能を識別します。

- *Name* フィールドを使用して、関数の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

- **所有者** の横にあるドロップダウンリストボックスを使用して、関数を所有するロールの名前を選択します。
- **スキーマ** の横にあるドロップダウンリストボックスを使用して、関数を作成するスキーマを選択します。
- 関数に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



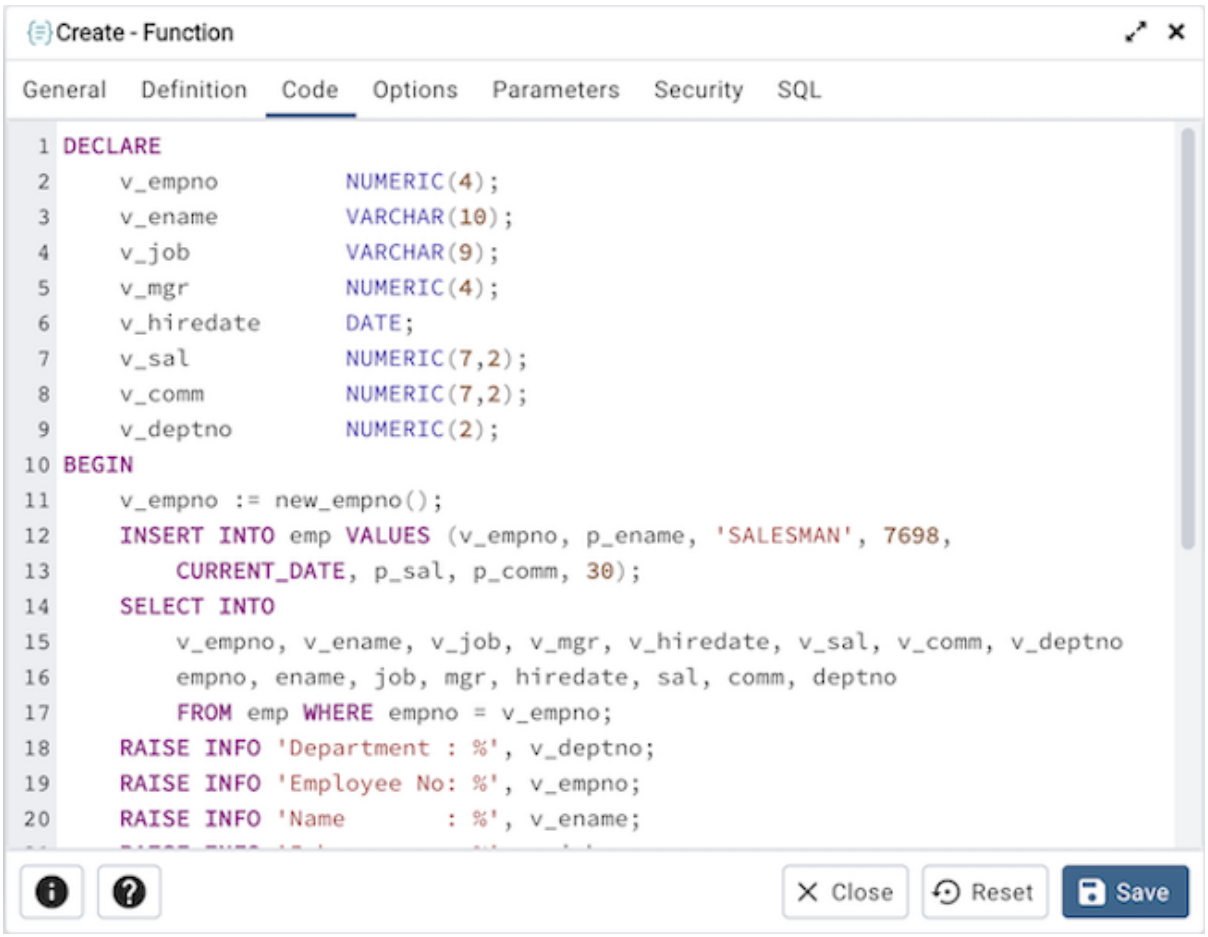
[**定義**] タブのフィールドを使用して、関数を定義します。

- *Returntype* の横にあるドロップダウンリストボックスを使用して、関数によって返されるデータ型を選択します（ある場合）。
- **言語** の横にあるドロップダウンリストボックスを使用して、実装言語を選択します。デフォルトは *sql* です。
- *Arguments* のフィールドを使用して、引数を定義します。 **追加** アイコン（+）をクリックして、引数のパラメーターと値を設定します。
  - **データ型** フィールドのドロップダウンリストボックスを使用して、データ型を選択します。
  - *Mode* フィールドのドロップダウンリストボックスを使用して、モードを選択します。入力パラメーターに *IN* を選択します。出力パラメーターに *OUT* を選択します。入力パラメーターと出力パラメーターの両方に *INOUT* を選択します。または、 *VARIADIC* を選択して VARIADIC パラメーターを指定します。

- **引数名** フィールドに引数の名前を入力します。
- **デフォルト値** フィールドに引数のデフォルト値を指定します。

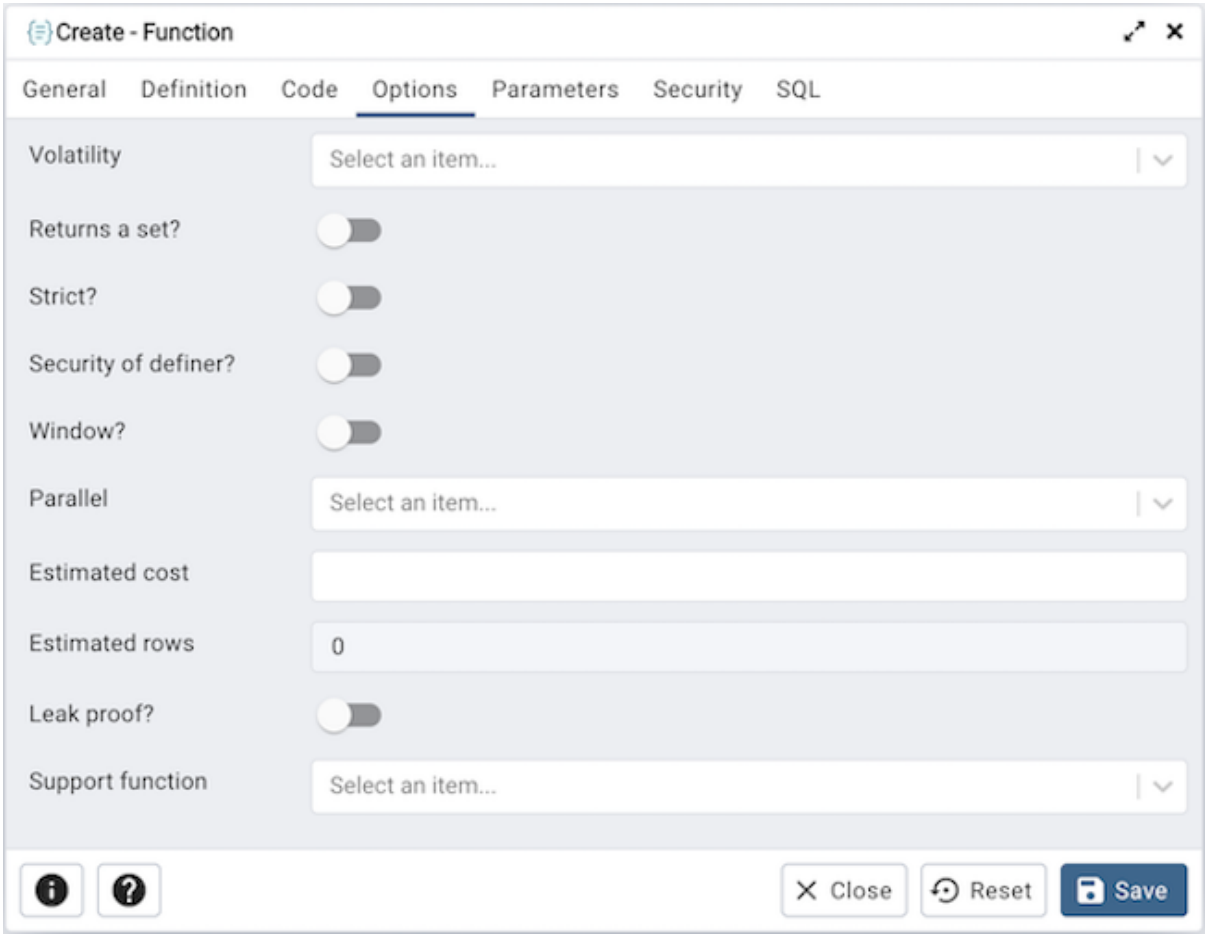
追加 アイコン（+）をクリックして、別の引数を定義します。引数を破棄するには、行の左側にあるごみ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**コード**] タブをクリックして続行します。



- *Code* フィールドを使用して、関数が呼び出されたときに実行されるコードを記述します。

[**オプション**] タブをクリックして続行します。



[**オプション**] タブのフィールドを使用して、関数のアクションを説明または変更します。

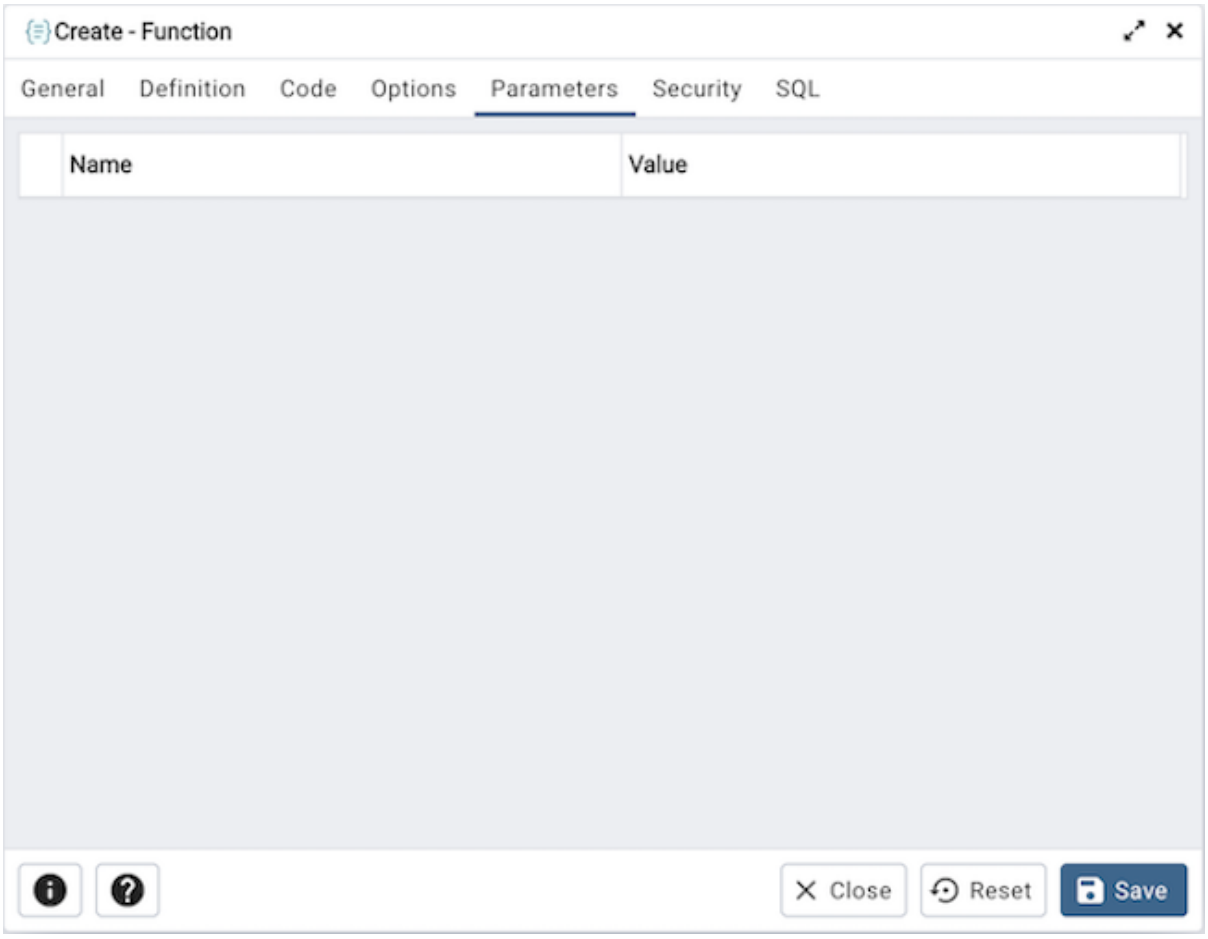
- **ボラティリティ** の横にあるドロップダウンリストボックスを使用して、次のいずれかを選択します。*VOLATILE* はデフォルト値です。
  - *VOLATILE* は、1 回のテーブルスキャン内でも関数値が変更される可能性があるため、最適化できないことを示します。
  - *STABLE* は、ファンクションがデータベースを変更できないこと、および 1 回のテーブルスキャン内で、同じ引数値に対して一貫して同じ結果を返すことを示します。
  - *IMMUTABLE* は、ファンクションがデータベースを変更できず、同じ引数値が指定された場合に常に同じ結果を返すことを示します。
- *ReturnsaSet?* スイッチを動かして、関数が複数の行を含むセットを返すかどうかを示します。デフォルトは **いいえ** です。
- *Strict?* スイッチを移動して、引数のいずれかが NULL の場合に常に関数が NULL を返すかどうかを示します。 *Yes* の場合、NULL 引数があると関数は実行されません。代わりに NULL 結果が自動的に想定されます。デフォルトは **いいえ** です。
- *Securityofdefiner?* スイッチを移動して、ファンクションを作成したユーザーの権限で実行することを指定します。デフォルトは **いいえ** です。
- *Window?* スイッチを動かして、関数がプレーン関数ではなくウィンドウ関数であることを示します。デフォルトは **いいえ** です。これは現在、C で記述された関数でのみ役立ちます。既存の関数定義を

置き換えるときに WINDOW 属性を変更することはできません。CREATEFUNCTION コマンドの詳細については、次の場所にある PostgreSQL コアドキュメントを参照してください。

https://www.postgresql.org/docs/current/functions-window.html

- *Estimatedcost* フィールドを使用して、関数の推定実行コストを表す正の数を cpu_operator_cost 単位で指定します。関数がセットを返す場合、これは返される行ごとのコストです。
- *Estimatedrows* フィールドを使用して、クエリプランナーが関数が返すと予想される行の推定数を示す正の数を指定します。これは、関数がセットを返すように宣言されている場合にのみ許可されます。デフォルトの想定は 1000 行です。
- **リーク防止**? スイッチを動かして、関数に副作用があるかどうかを示します。デフォルトは **いいえ** です。このオプションはスーパーユーザーのみが設定できます。
- **サポート機能** フィールドを使用して、機能に使用するプランナーサポート機能を指定します。

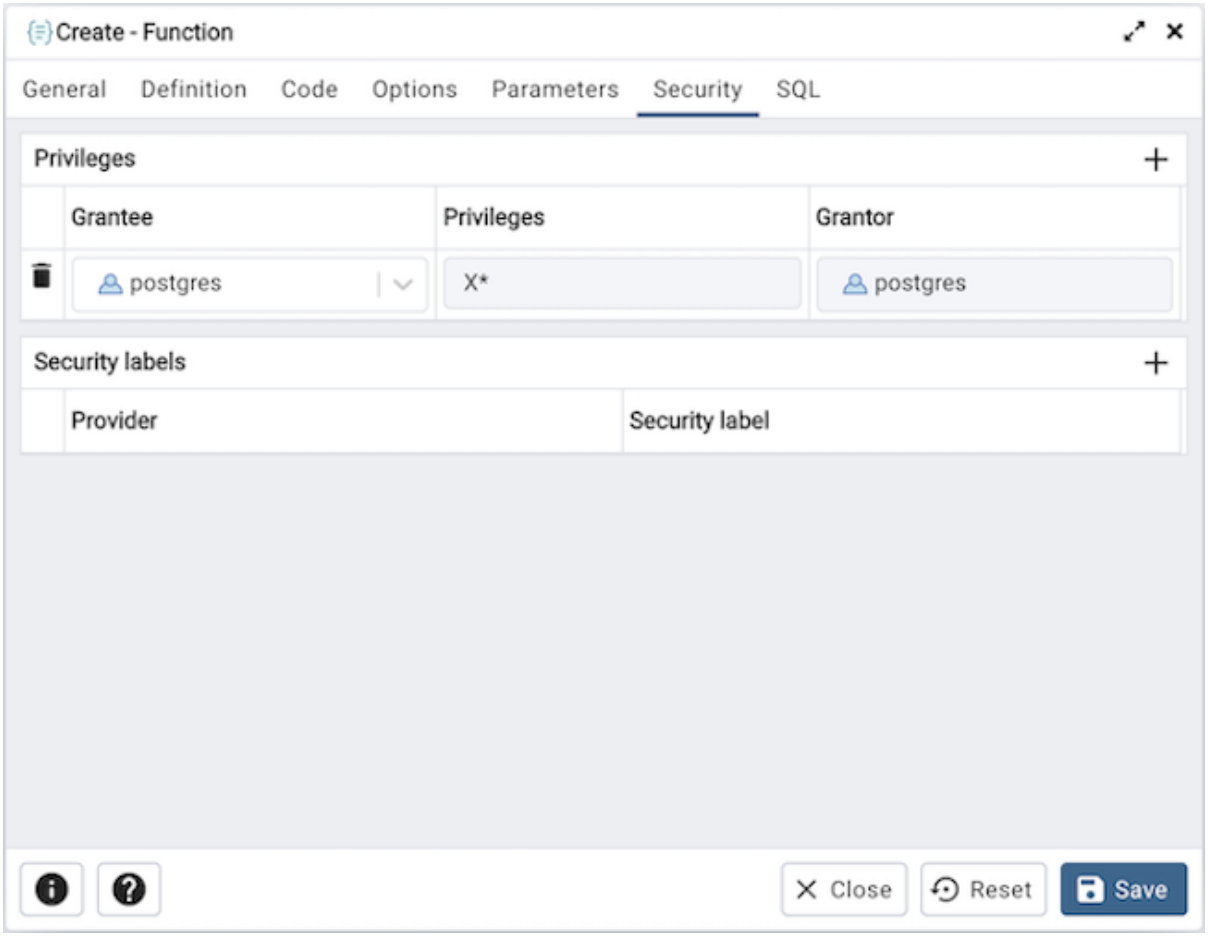
[**パラメーター**] タブをクリックして続行します。



[**パラメーター**] タブのフィールドを使用して、関数が呼び出されたときに適用される設定を指定します。**追加** アイコン（+）をクリックして、テーブルに **名前 / 値** フィールドを追加します。

- **パラメーター** パネルの **名前** 列のドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Value* フィールドを使用して、選択した変数に関連付けられる値を指定します。このフィールドは状況依存です。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

Privileges パネルを使用して、機能の使用権限をロールに割り当てます。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、ファンクションに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

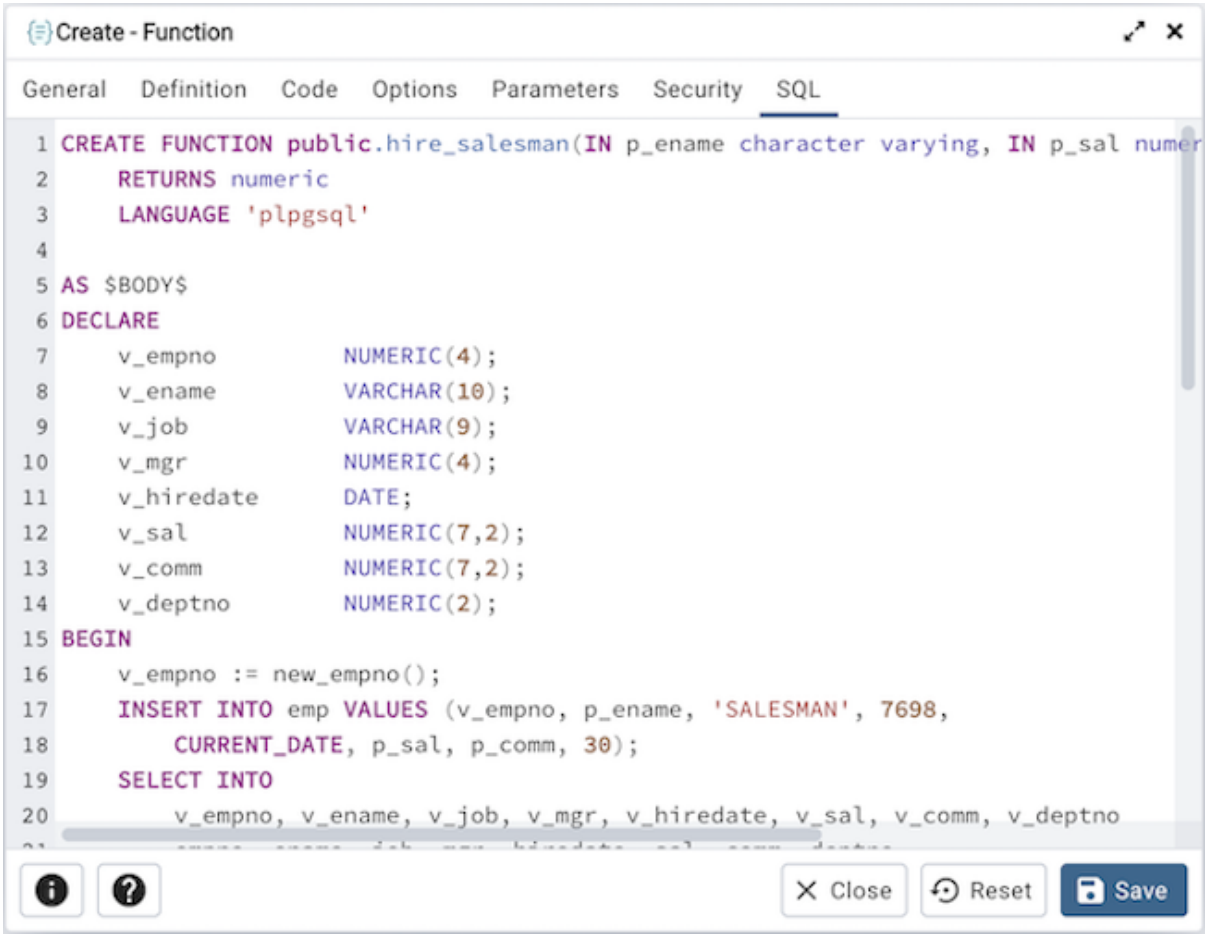
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Function ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.14.1 例

以下は、 **ファンクション** ダイアログでの選択によって生成される sql コマンドの例です。



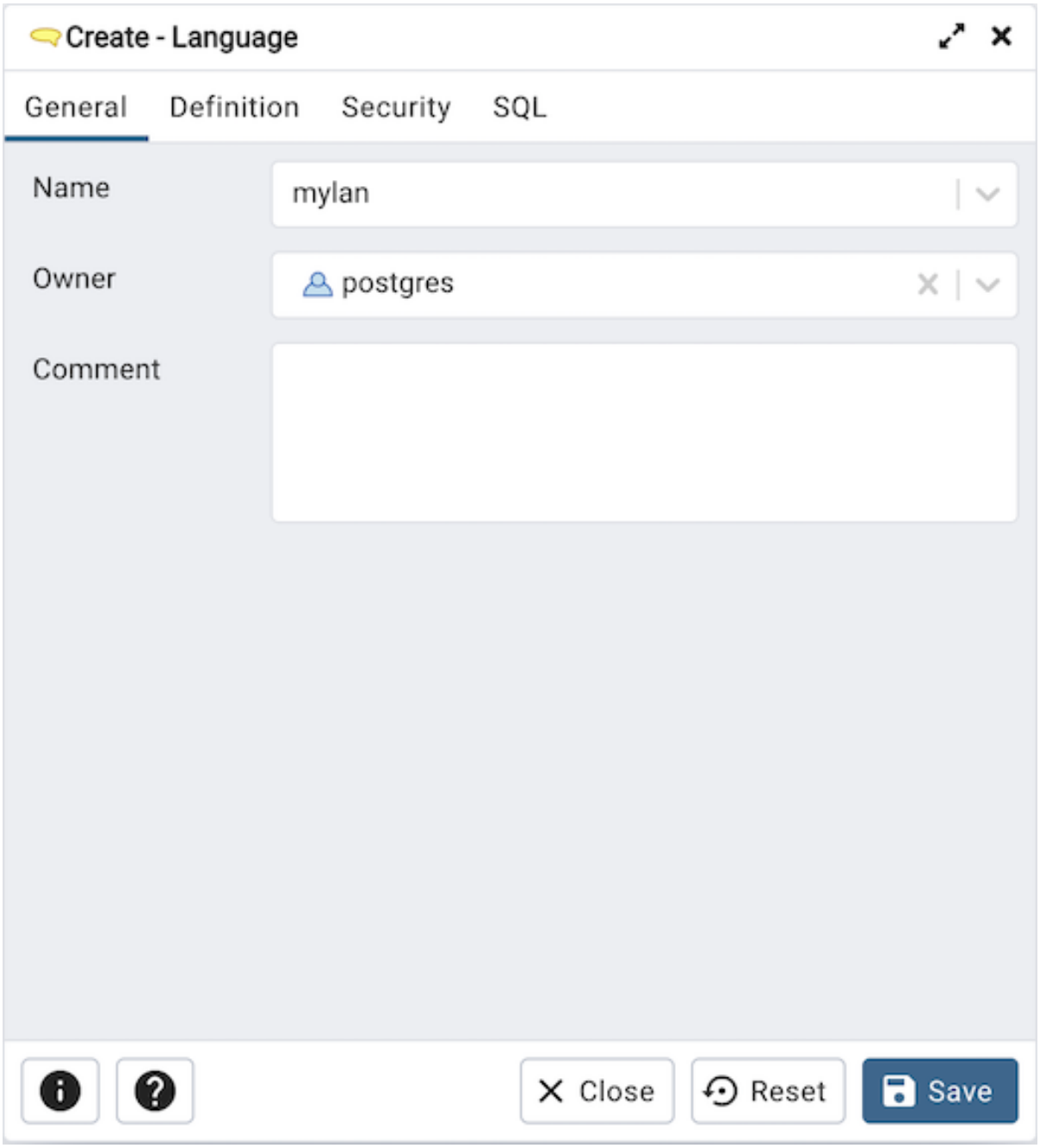
この例では、*hire_salesmen* という名前の *plpgsql* ファンクションの作成を示しています。このファンクションには 3 つの列（p_ename、p_sal、および p_comm）があります。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.15 Language Dialog

CREATELANGUAGE ダイアログを使用して、新しい手続き型言語を登録します。

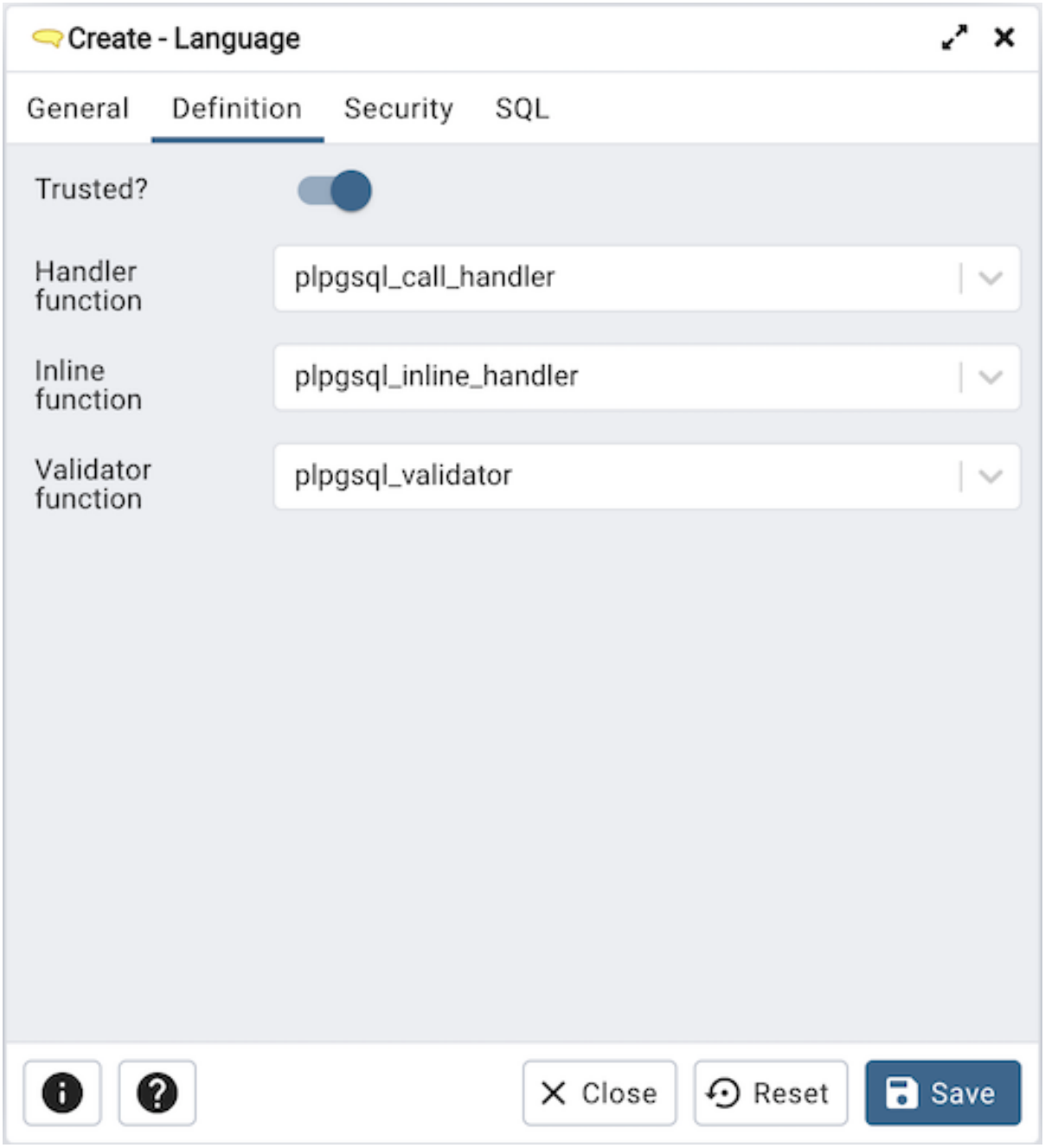
言語 ダイアログは、 **一般** 、 **定義** 、および **セキュリティ** のダイアログタブを介して手続き型言語の登録を構成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、言語を指定します。

- **名前** の横にあるドロップダウンリストボックスを使用して、言語スクリプトを選択します。
- **所有者** の横にあるドロップダウンリストボックスを使用して、ロールを選択します。
- **コメント** フィールドに言語に関するメモを保存します。

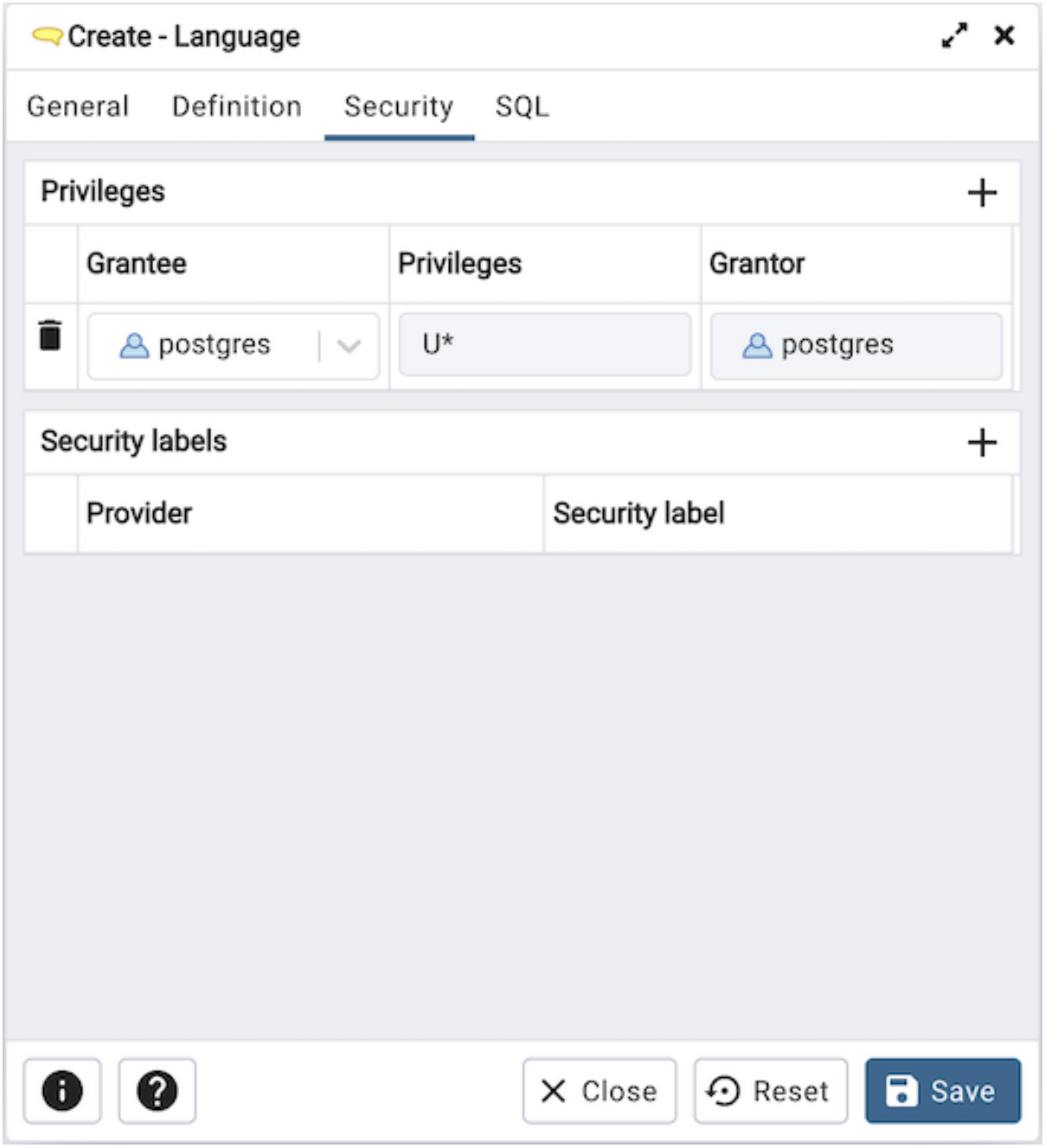
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、パラメーターを定義します。

- *Trusted?* スイッチを *No* の位置に移動して、PostgreSQL スーパーユーザー特権を持つユーザーのみがこの言語を使用できるように指定します。デフォルトは **はい** です。
- 有効にしたら、 *HandlerFunction* の横にあるドロップダウンリストボックスを使用して、言語の関数を実行するために呼び出される関数を選択します。
- 有効になっている場合、 **インライン関数** の横にあるドロップダウンリストボックスを使用して、この言語で匿名コードブロック（DO コマンド）を実行するために呼び出される関数を選択します。
- 有効になっている場合、 *ValidatorFunction* の横にあるドロップダウンリストボックスを使用して、その言語で新しい関数が作成されたときに呼び出される関数を選択し、新しい関数を検証します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、データベースオブジェクトの特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にある

ゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、ファンクションに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

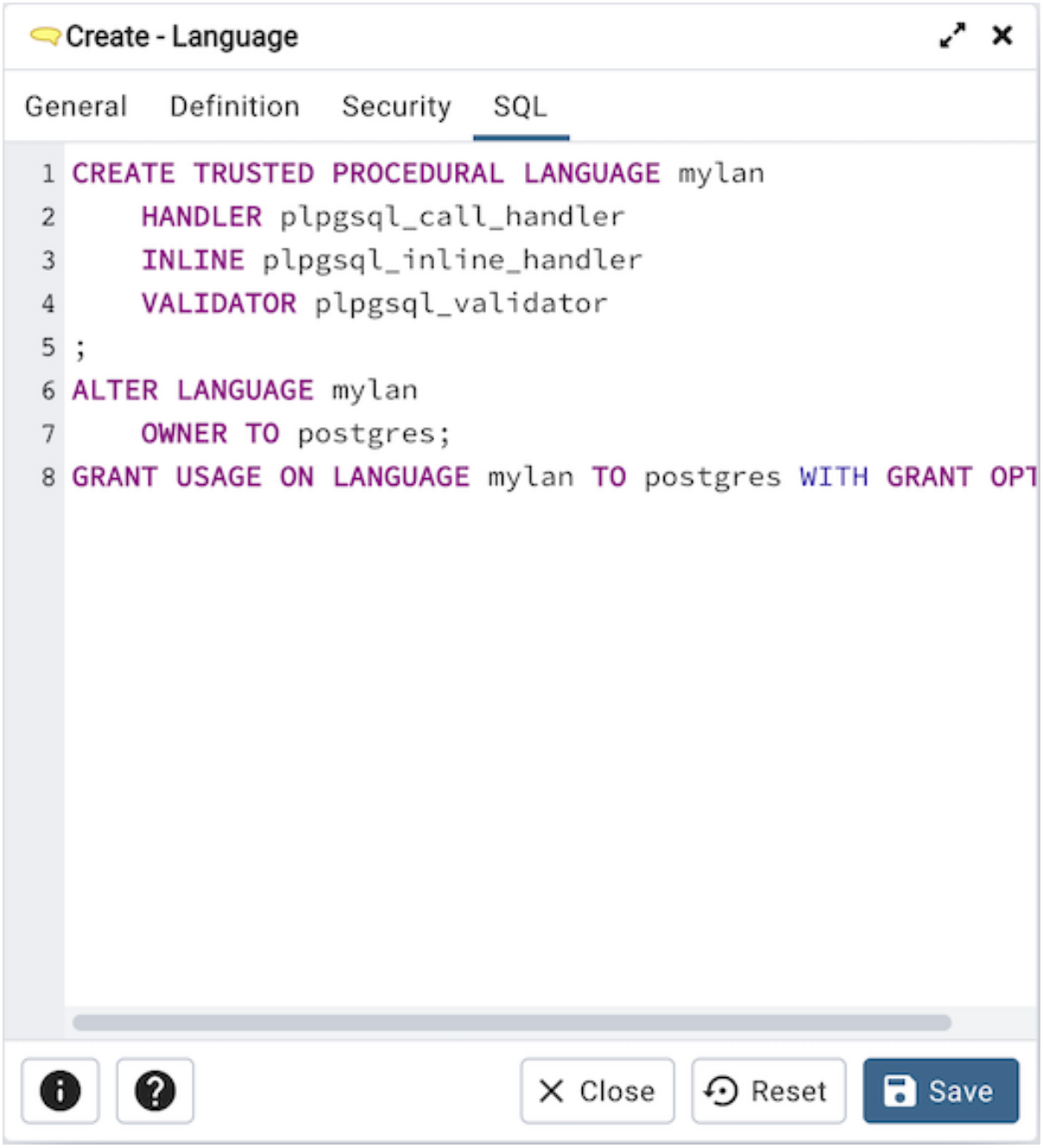
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

言語 ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.15.1 例

以下は、 **言語** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



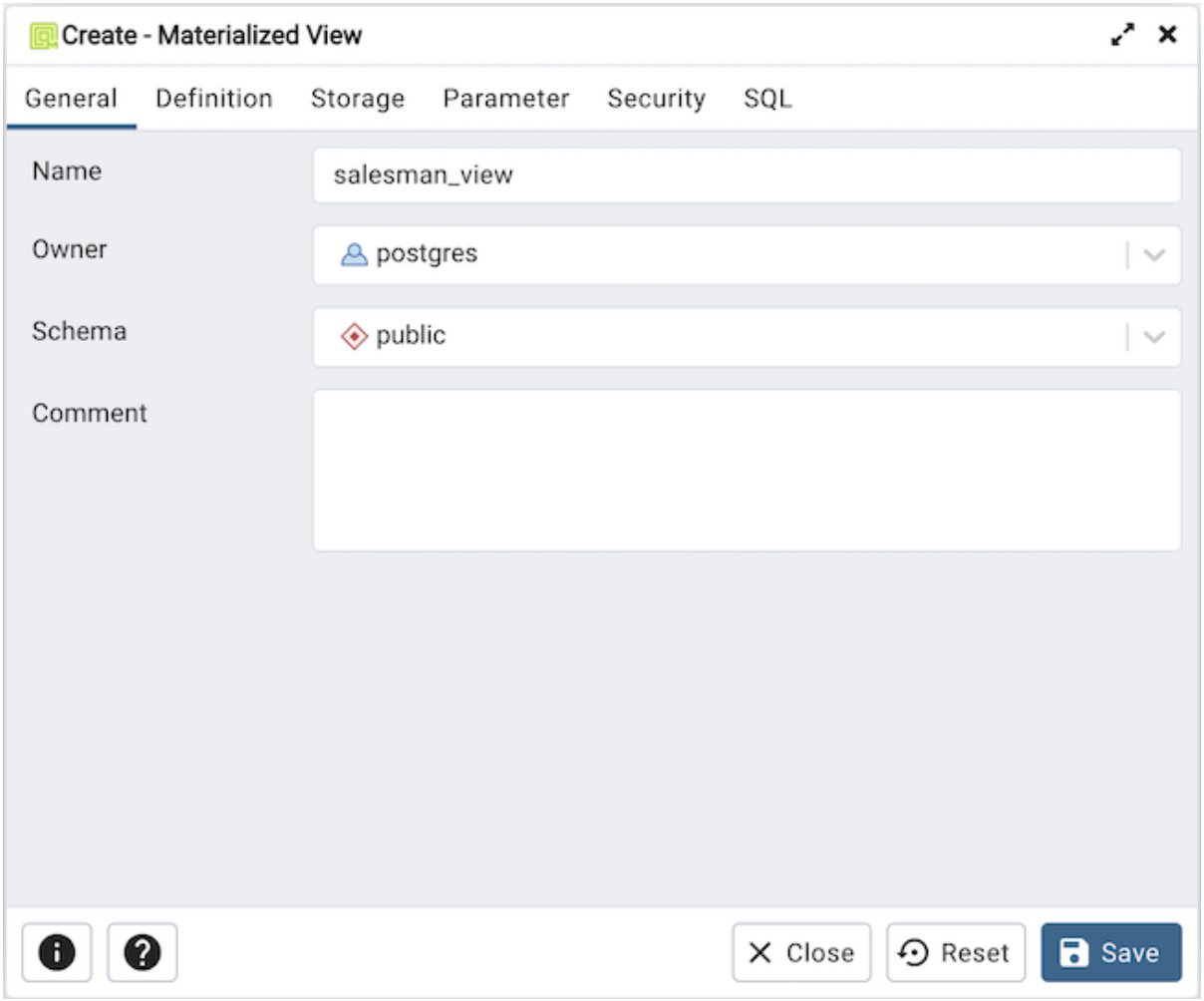
この例は、 *mylan* という名前の信頼できる手続き型言語の作成を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.16 Materialized View Dialog

マテリアライズドビュー ダイアログを使用して、マテリアライズドビューを定義します。マテリアライズドビューは、クエリの結果セットを含むストアドまたはキャッシュされたビューです。REFRESHMATERIALIZEDVIEW コマンドを使用して、マテリアライズドビューのコンテンツを更新します。

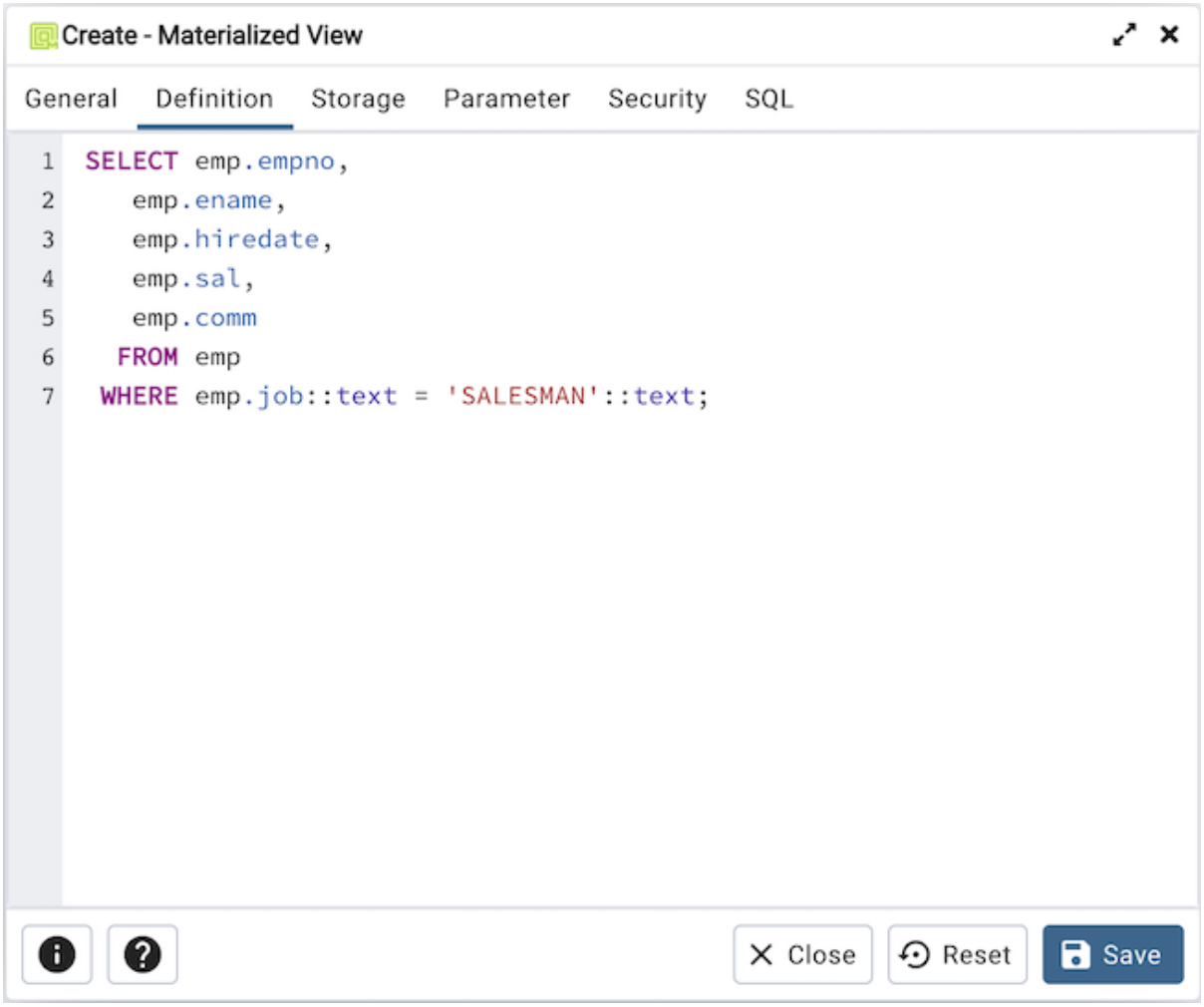
MaterializedView ダイアログは、 *General* 、 *Definition* 、 *Storage* 、 *Parameter* 、および *Security* のダイアログタブを使用して materialized_view の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、マテリアライズドビューを識別します。

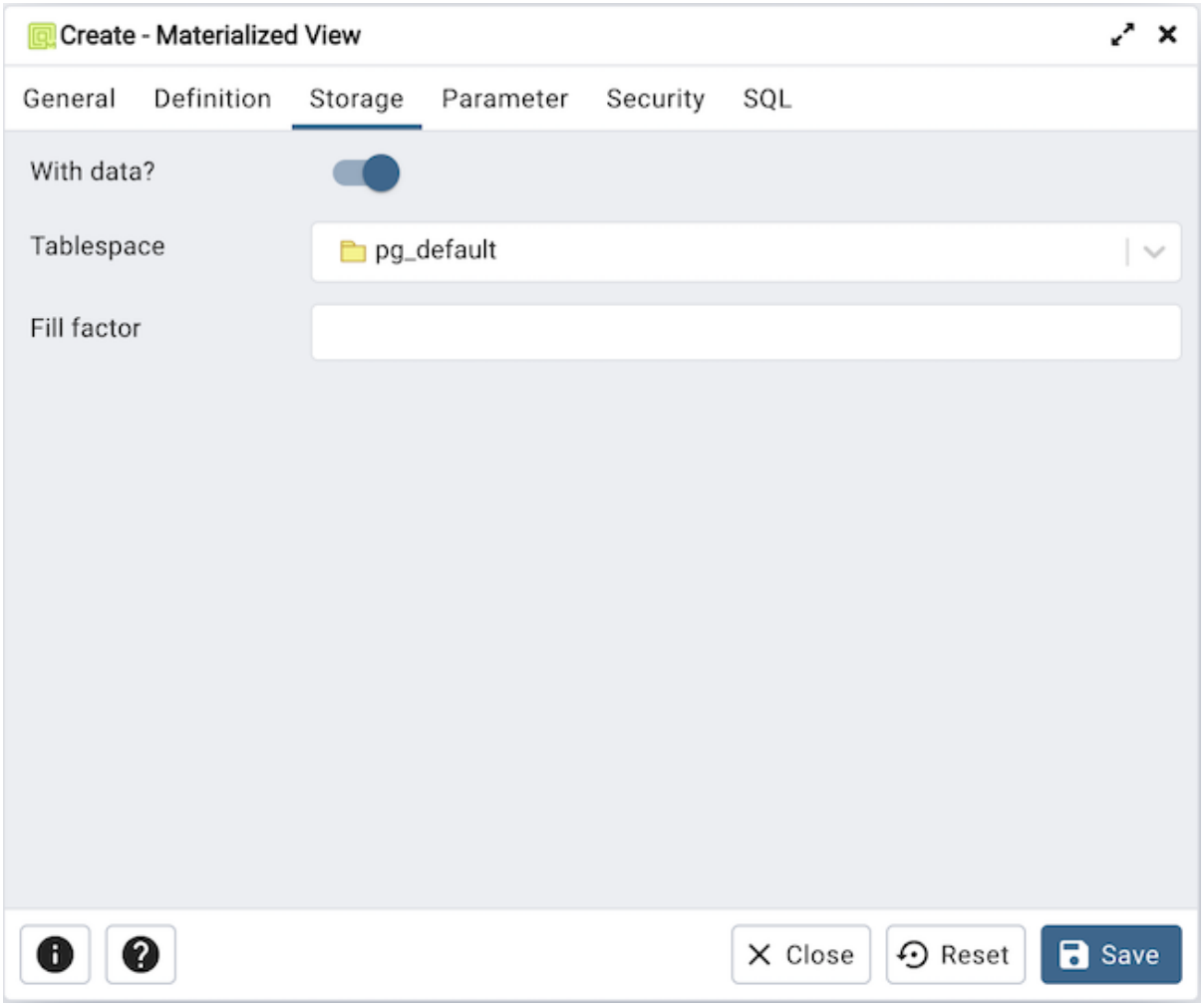
- *Name* フィールドを使用して、マテリアライズドビューに説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、マテリアライズドビューを所有するロールを選択します。
- *Schema* フィールドのドロップダウンリストボックスから、マテリアライズドビューが存在するスキーマの名前を選択します。
- マテリアライズドビューに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのテキストエディターフィールドを使用して、マテリアライズドビューを設定するクエリを提供します。既存のマテリアライズドビューの定義を更新すると、パラメーター（テーブル、トースト）、セキュリティ（特権とセキュリティラベル）、インデックス、およびその他の依存オブジェクトが失われることに注意してください。

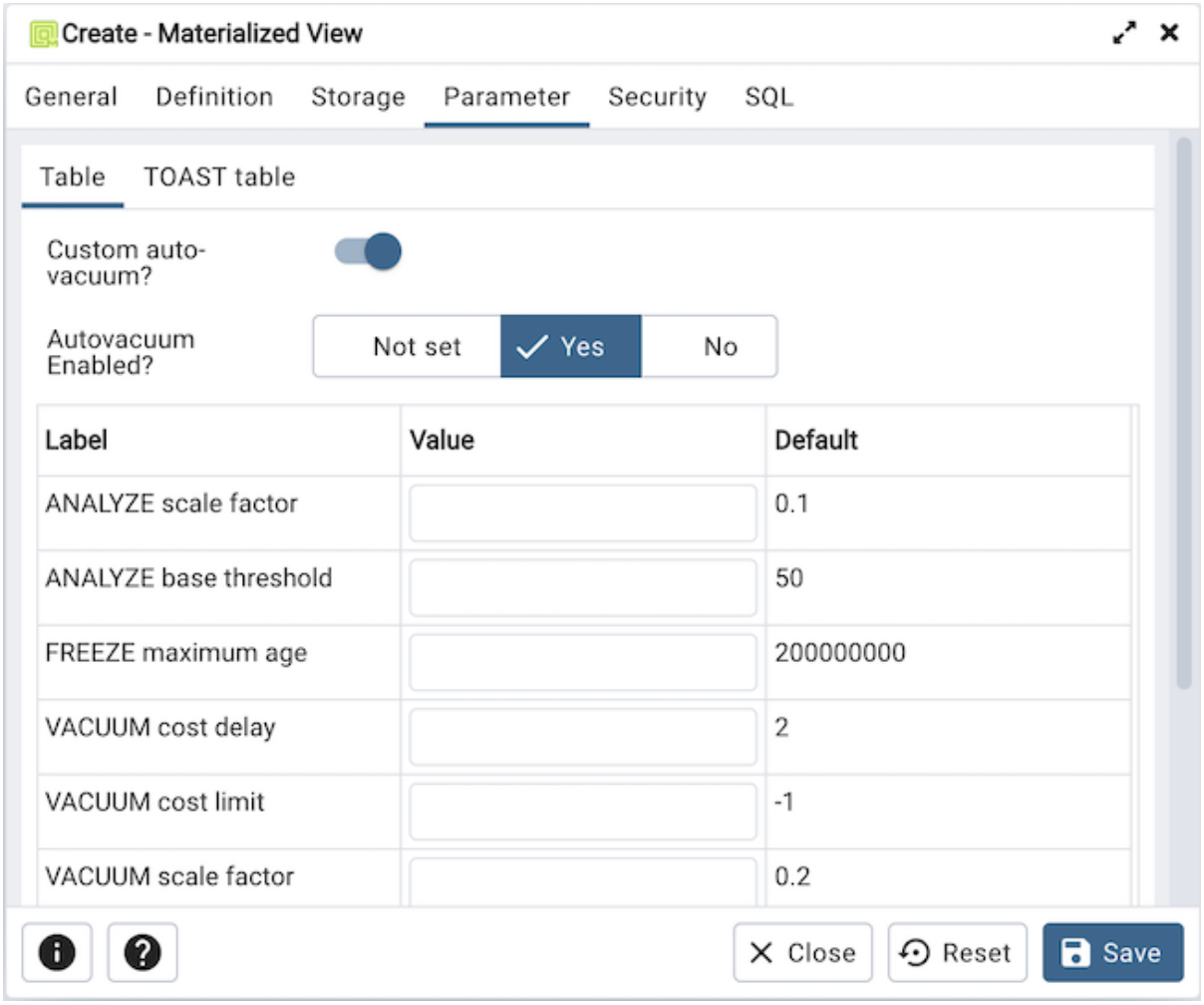
[**ストレージ**] タブをクリックして続行します。



[*Storage*] タブのフィールドを使用して、マテリアライズドビューを維持します。

- *WithData* スイッチを *Yes* の位置に移動して、作成時にマテリアライズドビューを移入することを指定します。そうでない場合、REFRESHMATERIALIZEDVIEW を呼び出すまでマテリアライズドビューを照会できません。
- *Tablespace* の横にあるドロップダウンリストボックスを使用して、マテリアライズドビューの場所を選択します。
- *FillFactor* フィールドを使用して、マテリアライズドビューのフィルファクターを指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。

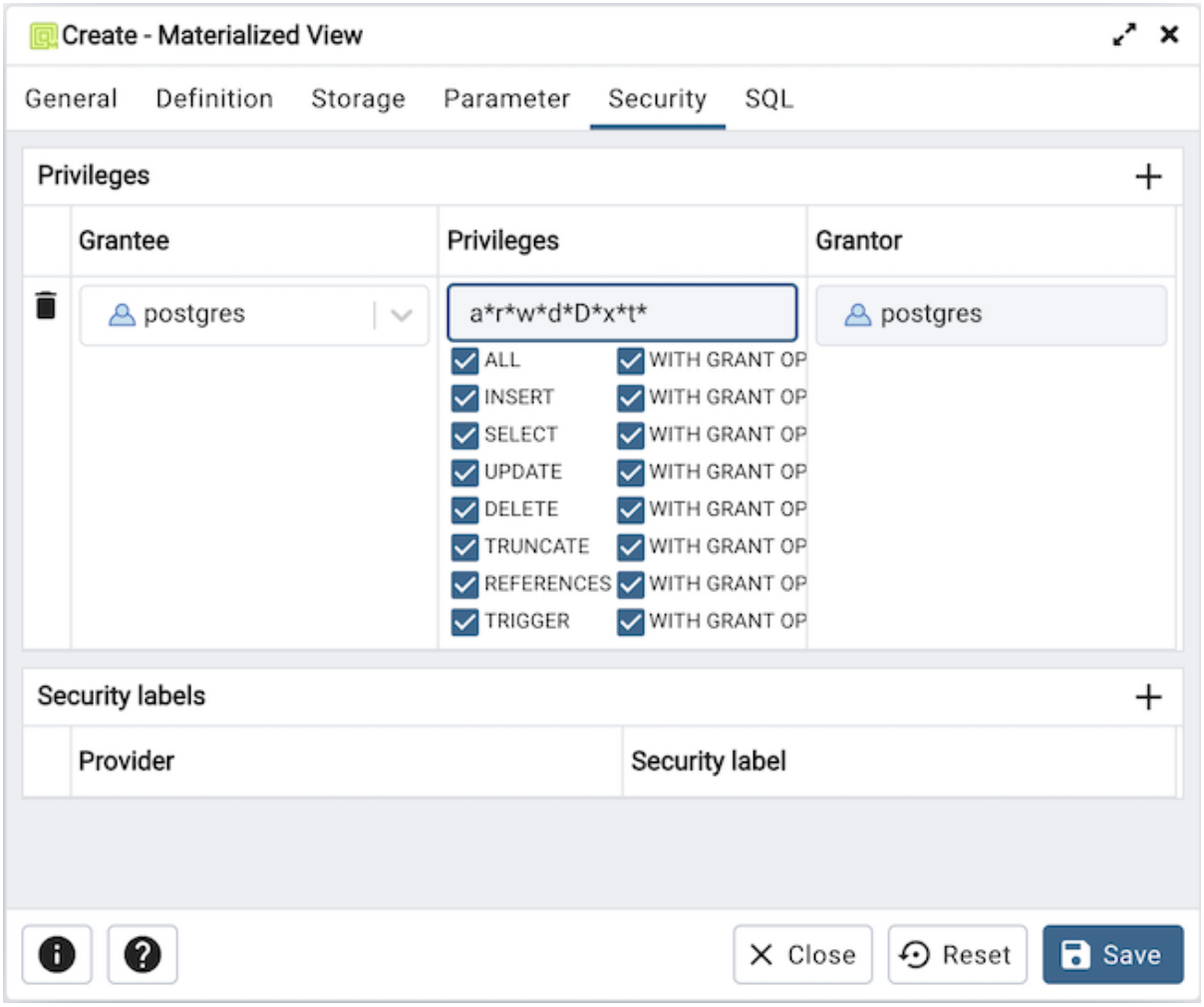
[**パラメーター**] タブをクリックして続行します。



パラメーター タブの中にネストされたタブを使用して、VACUUM および ANALYZE のしきい値を指定します。テーブルと関連するトーストテーブルの値をカスタマイズするには、**テーブル** タブと **トーストテーブル** タブを使用します。デフォルト値を変更するには：

- **カスタム自動バキューム**? スイッチを **はい** の位置に移動して、マテリアライズドビューでカスタムメンテナンスを実行し、**バキュームテーブル** で値を選択します。 **バキュームテーブル** は、メンテナンス操作のデフォルト値を提供します。
- *Autovacuumenabled?* を *Notset* に変更すると、autovacuum_enabled がリセットされます。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、マテリアライズドビューに権限を設定します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、マテリアライズドビューに適用するセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解

釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

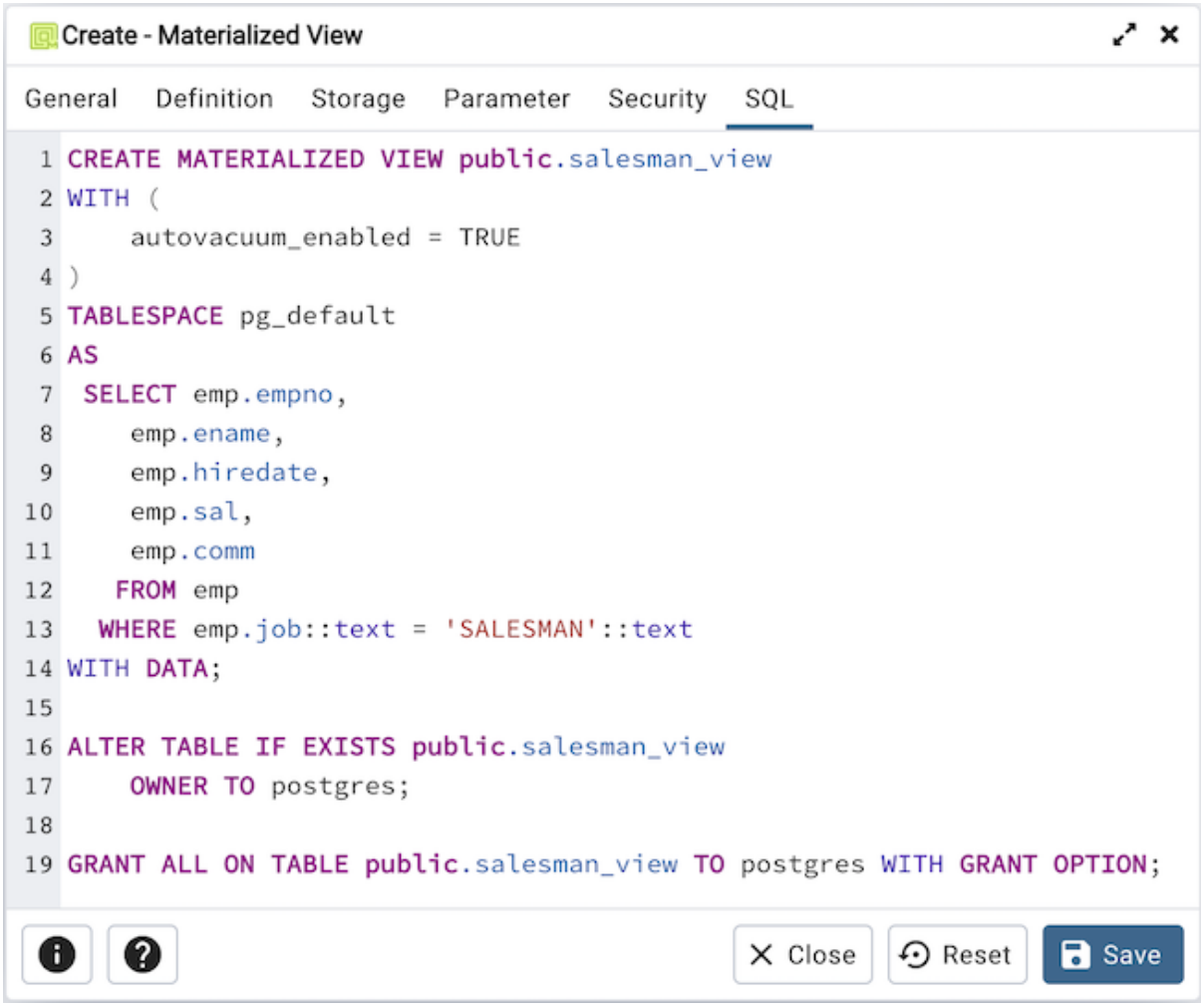
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

MaterializedView ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.16.1 例

以下は、**マテリアライズドビュー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



示されている例では、表示されたクエリの結果を *pg_default* テーブルスペースに保存する *new_hires* という名前のクエリを作成します。

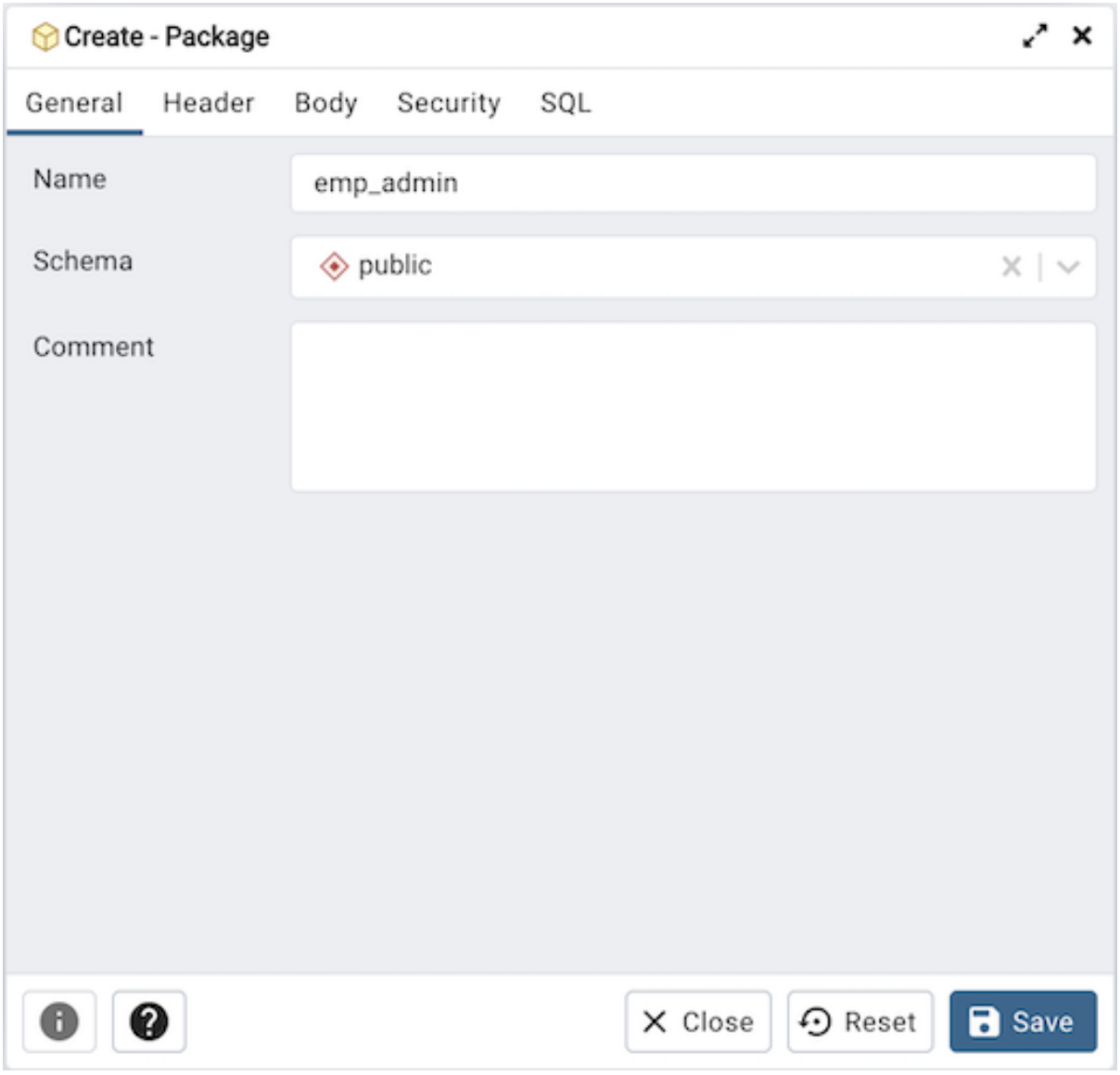
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、**閉じる** ボタンをクリックします。

- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.17 Package Dialog

Package ダイアログを使用して、（ユーザー定義の）パッケージ仕様書を作成します。

[**パッケージ**] ダイアログは、 **一般** 、 **ヘッダー** 、 **ボディ** 、および **セキュリティ** のダイアログタブを使用してパッケージの管理を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

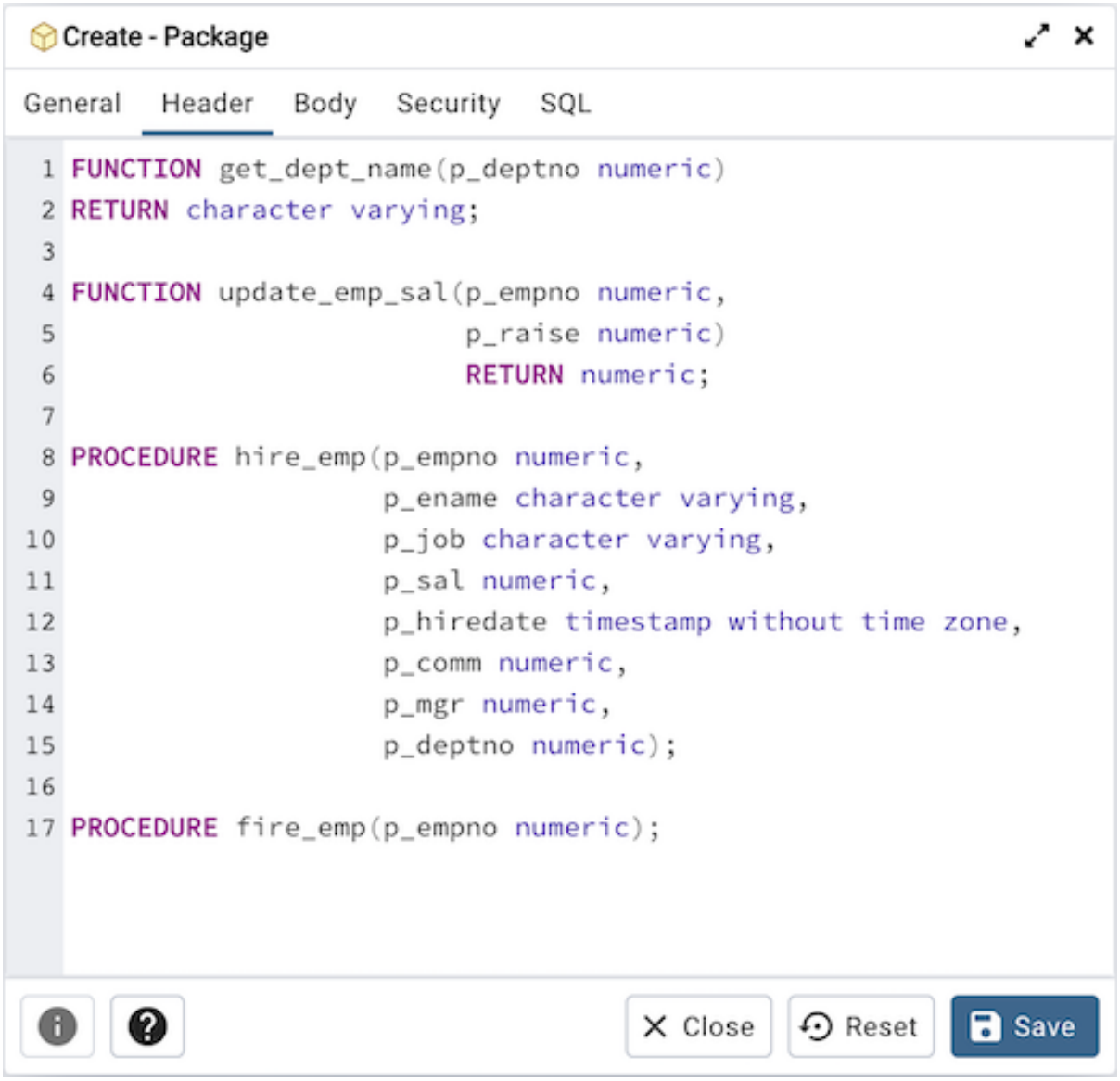


[**全般**] タブのフィールドを使用して、パッケージを識別します。

- *Name* フィールドを使用して、パッケージの説明的な名前を追加します。新しいパッケージの名前は、同じスキーマ内の既存のパッケージと一致してはなりません。
- **スキーマ** フィールドのドロップダウンリストボックスから、パッケージが存在するスキーマを選択します。

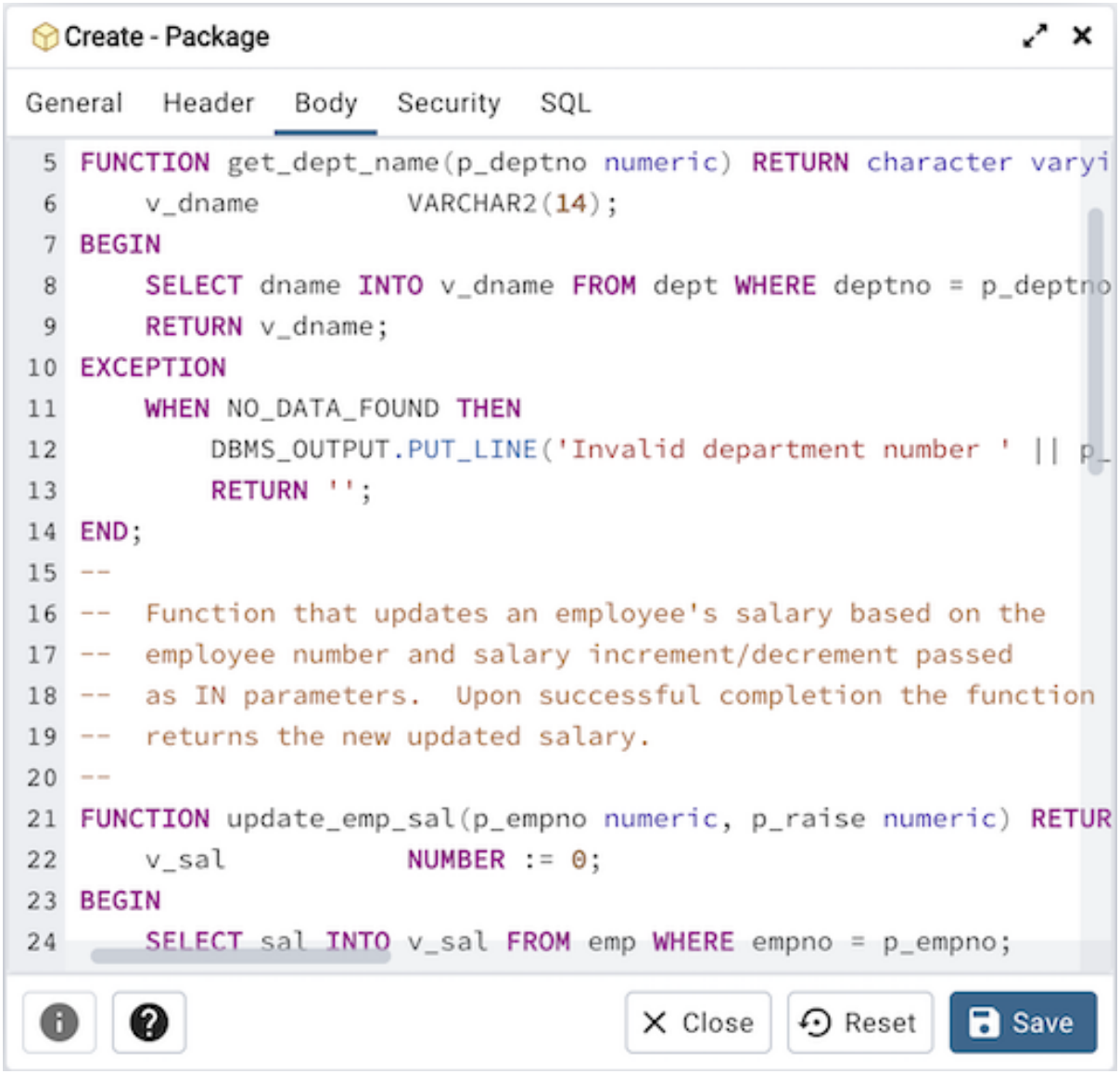
- パッケージに関するメモを **コメント** フィールドに保存します。

[**ヘッダー**] タブをクリックして続行します。



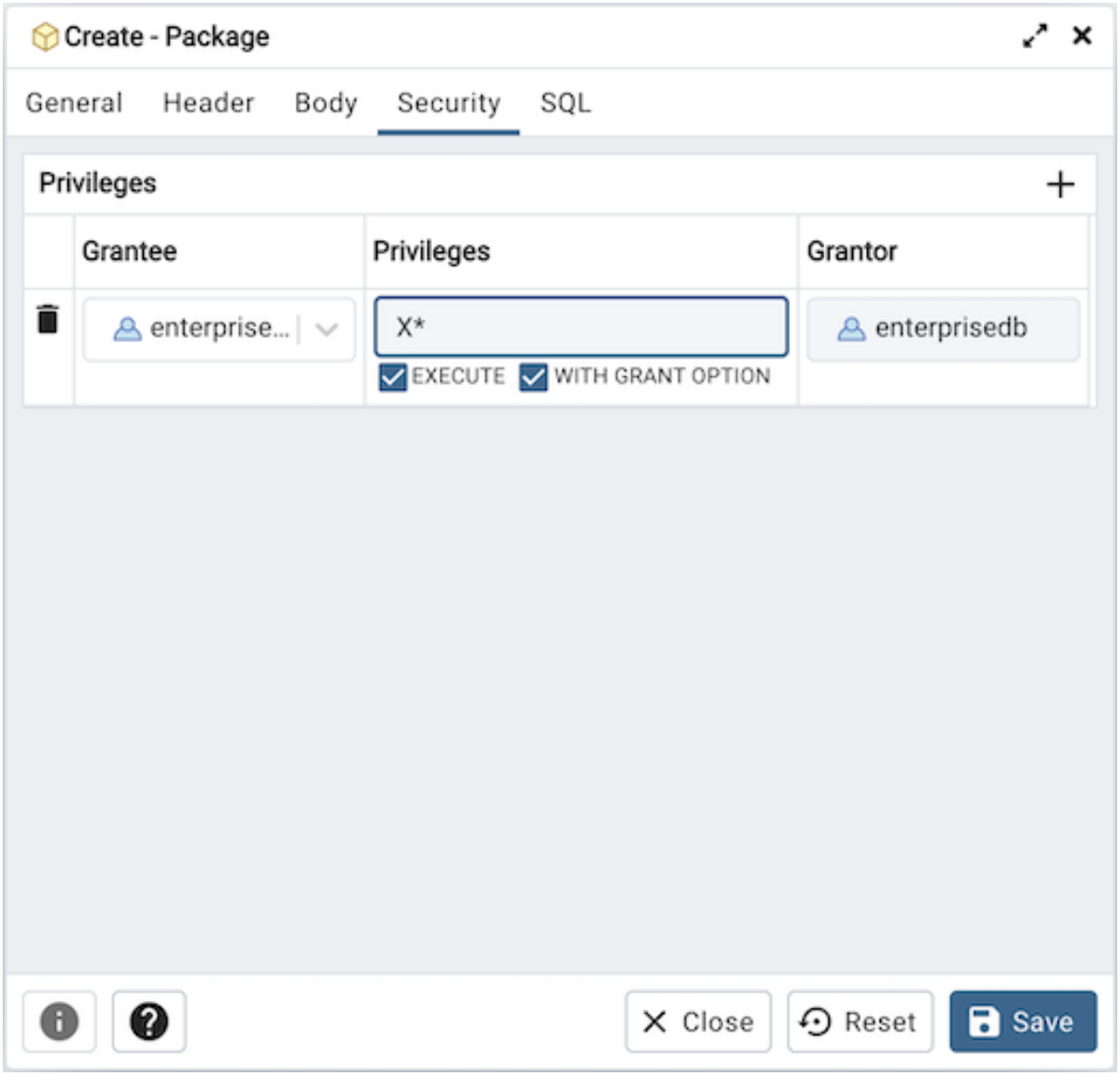
Header フィールドを使用して、パッケージのパブリックインターフェイスを定義します。

[**本文**] タブをクリックして続行します。



Body フィールドを使用して、各パッケージオブジェクトを実装するコードを提供します。

[**セキュリティ**] タブをクリックして続行します。



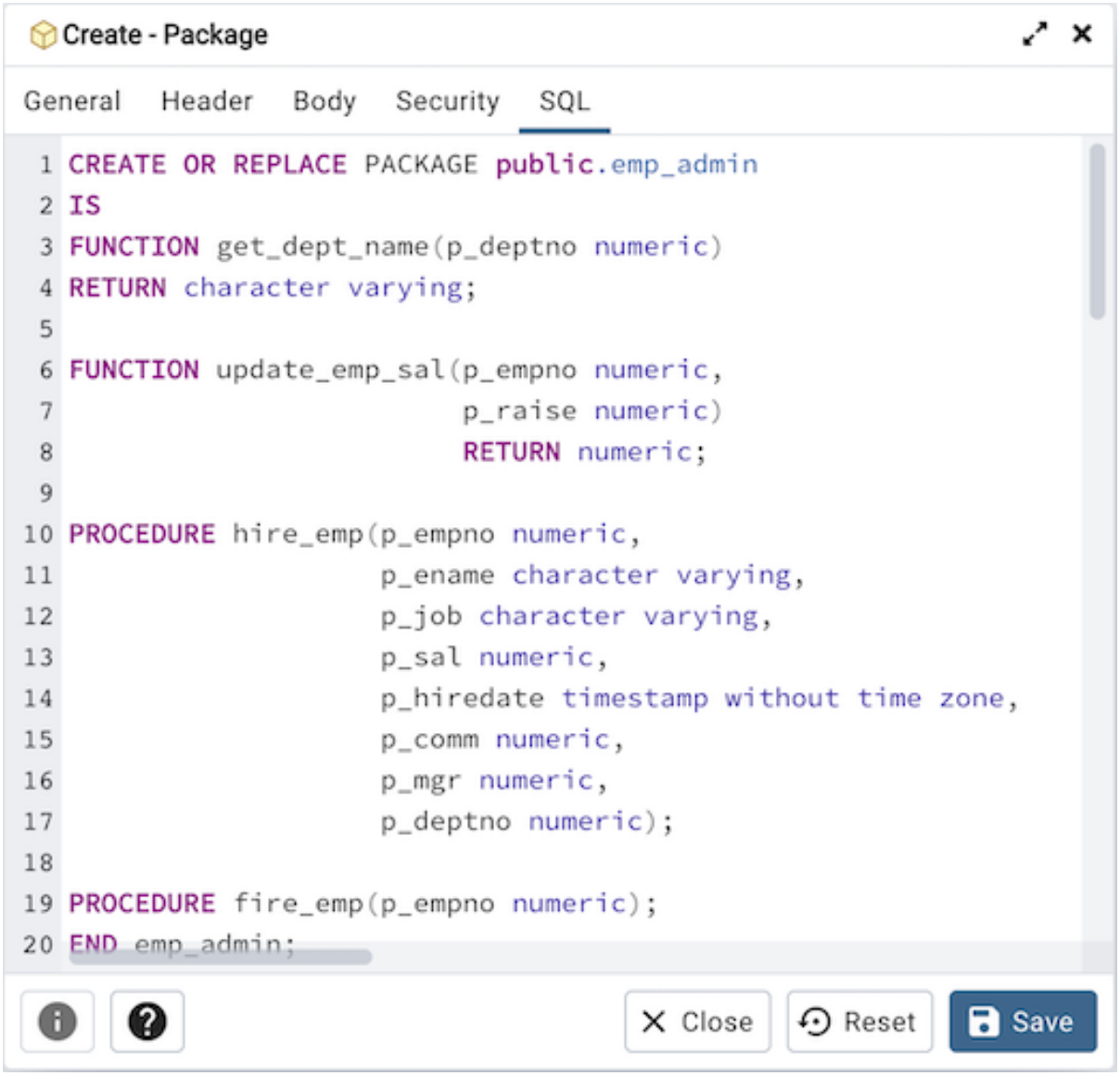
[**セキュリティ**] タブのフィールドを使用して、パッケージの EXECUTE 特権をロールに割り当てます。 **追加** アイコン（+）をクリックして、パッケージに特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。特権の左側にあるチェックボックスをオンにして、指定したユーザーに選択した特権を付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Package ダイアログに入力すると、パッケージ定義を作成または変更する SQL コマンドが生成されます。



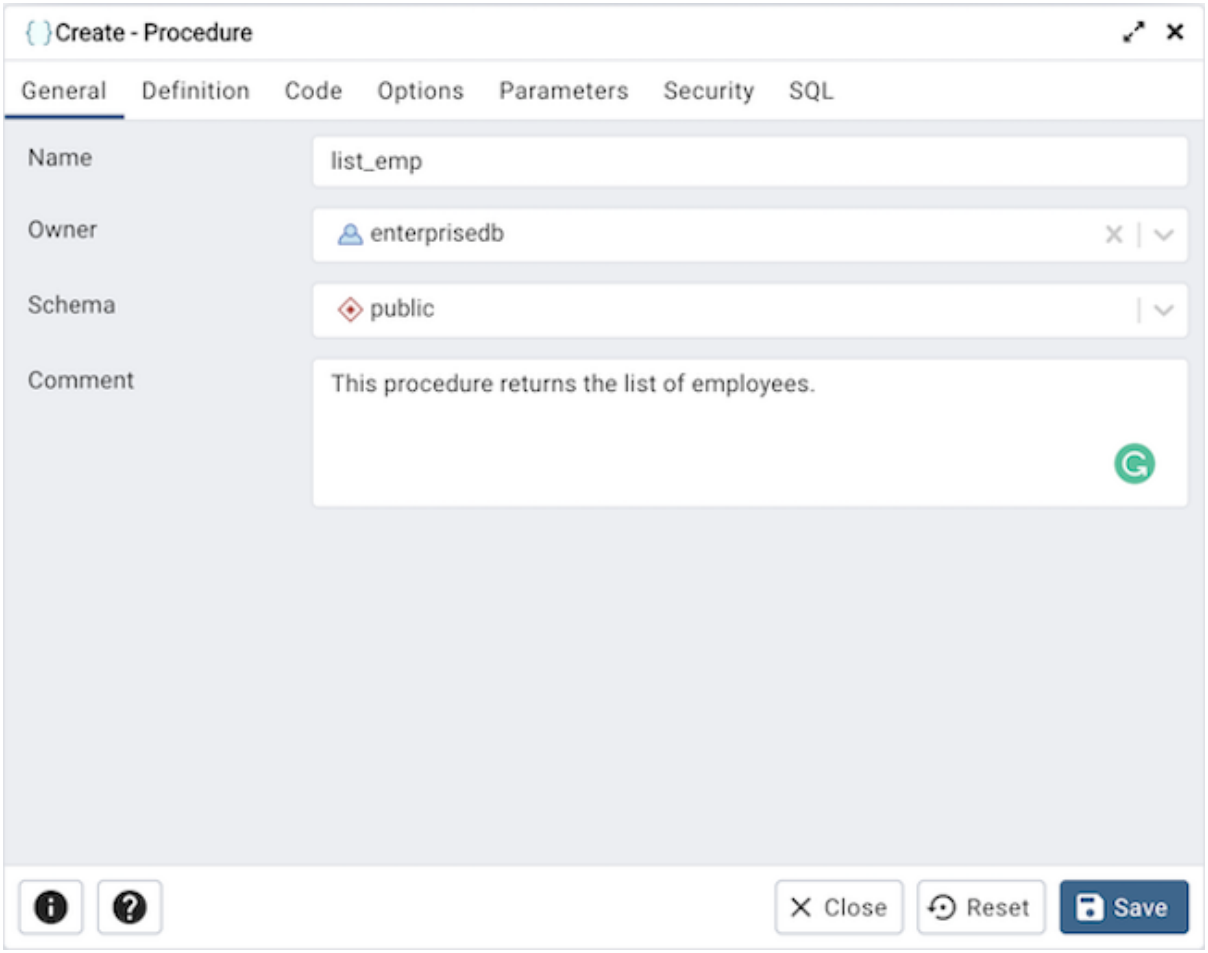
ここに示す例は、2 つのファンクションと 2 つのプロシージャを含む *empinfo* という名前のパッケージの作成を示しています。

- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- ダイアログへの変更を削除するには、 **リセット** ボタンをクリックします。

5.18 Procedure Dialog

プロシージャ ダイアログを使用して、プロシージャを作成します。プロシージャは PostgreSQLv11+および EDBPostgresAdvancedServer でサポートされています。 *Procedure* ダイアログでは、CREATEPROCEDURE コマンドのオプションを実装できます。

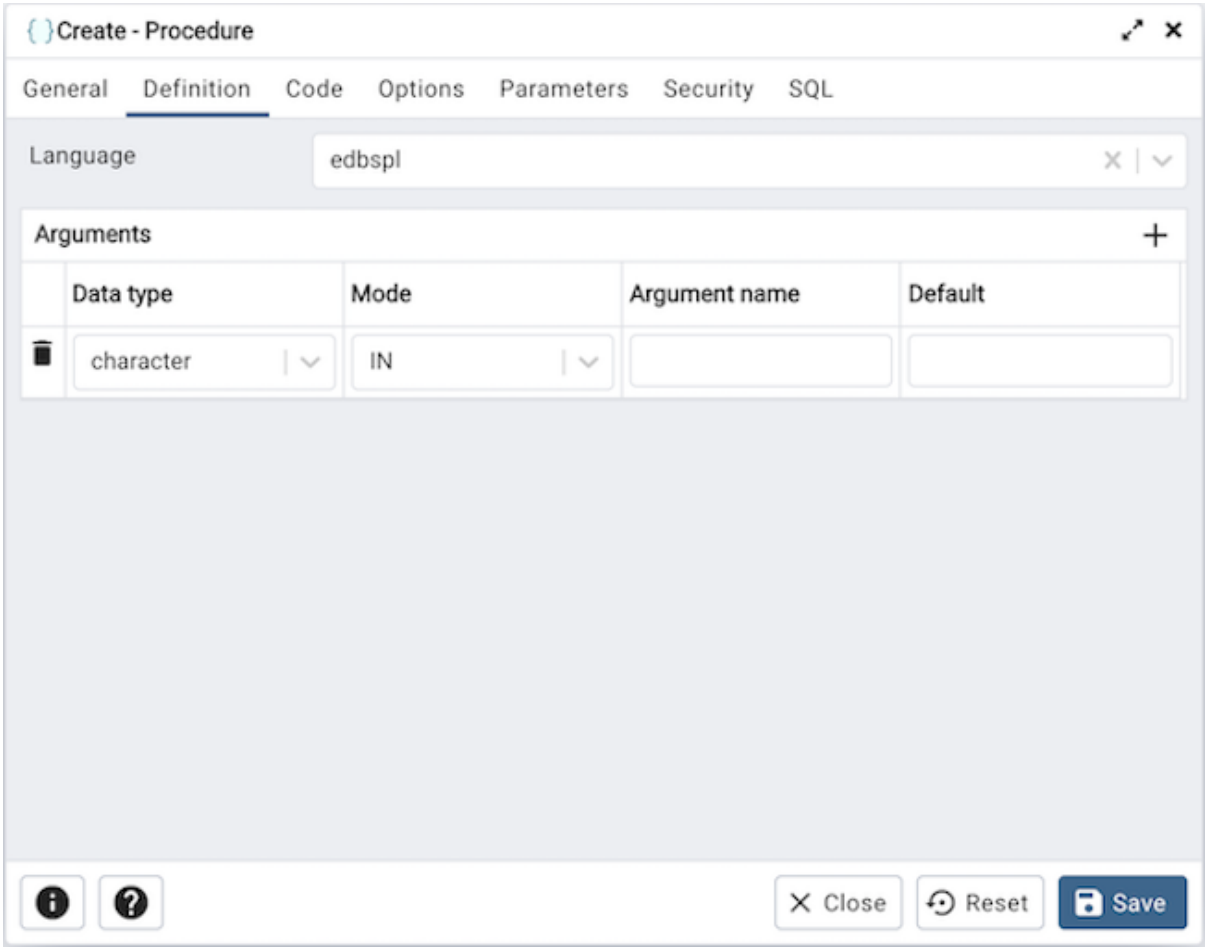
プロシージャ ダイアログは、 **一般** 、 **定義** 、 **コード** 、 **オプション** 、 **引数** 、 **パラメーター** 、および **セキュリティ** のダイアログタブを使用してプロシージャーの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、プロシージャを識別します。

- *Name* フィールドを使用して、プロシージャの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、ロールを選択します。
- **スキーマ** フィールドのドロップダウンリストボックスから、プロシージャが存在するスキーマの名前を選択します。
- 手順に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。

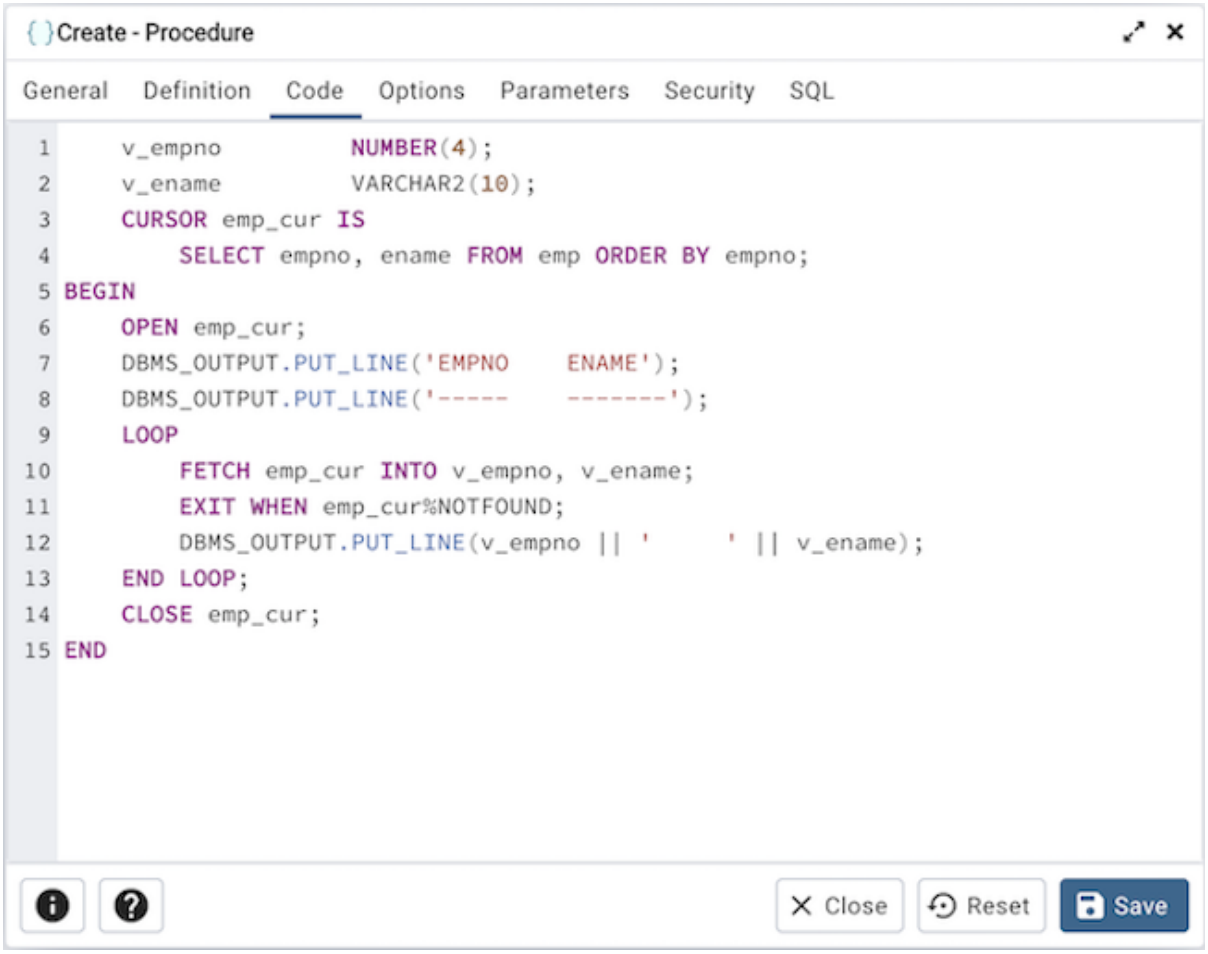


[**定義**] タブのフィールドを使用して、プロシージャを定義します。

- **言語** の横にあるドロップダウンリストボックスを使用して、言語を選択します。デフォルトは *edbspl* です。
- *Arguments* セクションのフィールドを使用して、引数を定義します。 *Add* をクリックして、引数のパラメーターと値を設定します。
- **データ型** の横にあるドロップダウンリストボックスを使用して、データ型を選択します。
- **モード** の横にあるドロップダウンリストボックスを使用して、モードを選択します。入力パラメーターに *IN* を選択します。出力パラメーターに *OUT* を選択します。入力パラメーターと出力パラメーターの両方に *INOUT* を選択します。または、 *VARIADIC* を選択して VARIADIC パラメーターを指定します。
- **引数名** フィールドに引数の名前を書き込みます。
- **デフォルト値** フィールドに引数のデフォルト値を指定します。

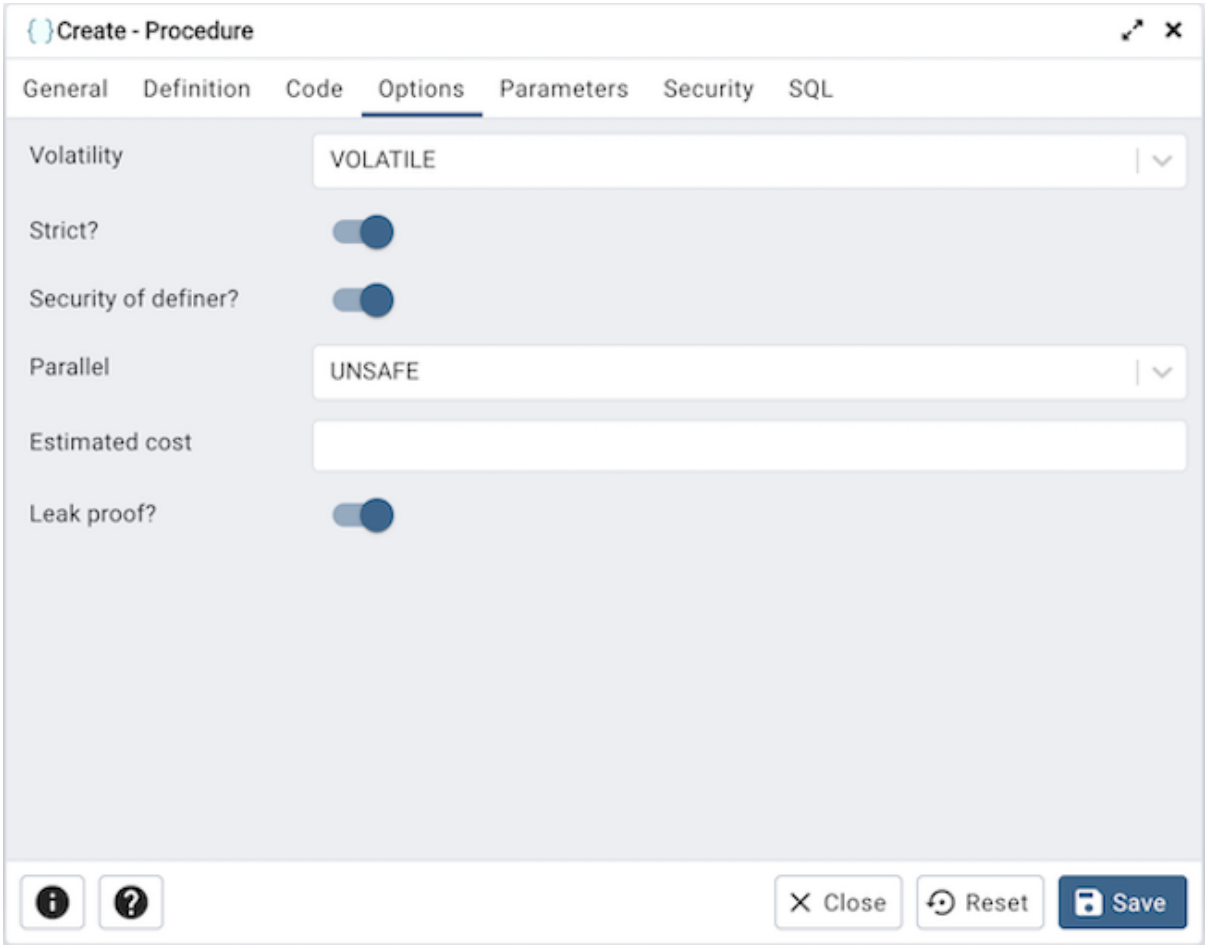
Add をクリックして別の引数を定義します。引数を破棄するには、行の左側にあるごみ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**コード**] タブをクリックして続行します。



- *Code* フィールドを使用して、プロシージャが呼び出されたときに実行されるコードを指定します。

[**オプション**] タブをクリックして続行します。

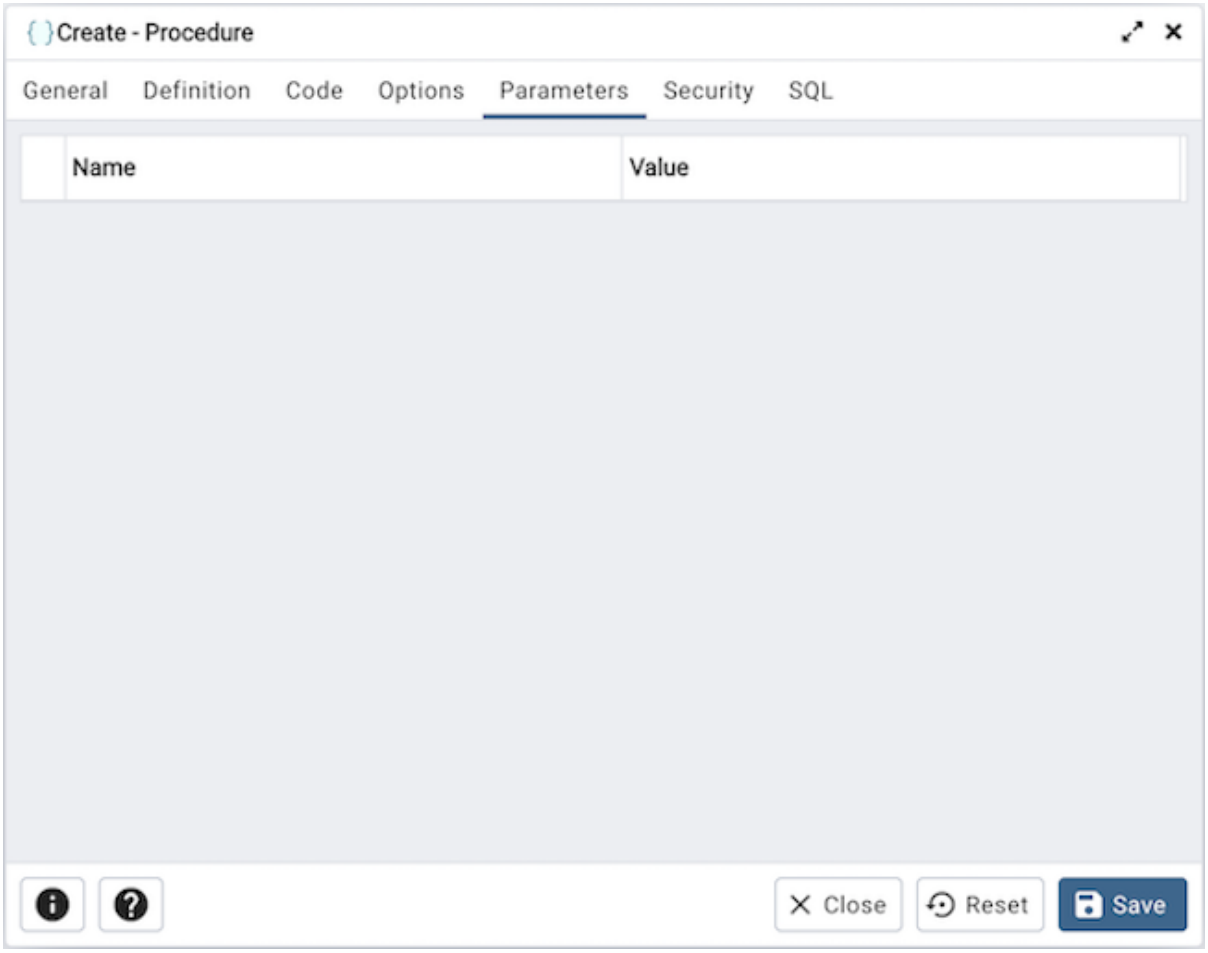


[**オプション**] タブのフィールドを使用して、プロシージャの動作を説明または変更します。

- **ボラティリティ** の下のドロップダウンリストボックスを使用して、次のいずれかを選択します。*VOLATILE* はデフォルト値です。
  - *VOLATILE* は、1 回のテーブルスキャン内でも値が変更される可能性があるため、最適化を行わないことを示します。
  - *STABLE* は、プロシージャーがデータベースを変更できないこと、および単一のテーブルスキャン内では、同じ引数値に対して一貫して同じ結果を返すが、その結果は SQL ステートメント全体で異なる可能性があることを示します。
  - *IMMUTABLE* は、プロシージャーがデータベースを変更できず、同じ引数値が指定された場合に常に同じ結果を返すことを示します。
- *Strict?* スイッチを移動して、引数のいずれかが NULL の場合にプロシージャが常に NULL を返すかどうかを示します。 *Yes* の場合、NULL 引数があるとプロシージャは実行されません。代わりに NULL 結果が自動的に想定されます。デフォルトは **いいえ** です。
- *Securityofdefiner?* スイッチを移動して、プロシージャを作成したユーザーの権限で実行することを指定します。デフォルトは **いいえ** です。
- **見積もりコスト** フィールドを使用して、プロシージャーの見積もり実行コストを表す正の数を cpu_operator_cost 単位で指定します。プロシージャーがセットを返す場合、これは返される行ごとのコストです。

- *Leakproof?* スイッチを移動して、プロシージャに副作用があるかどうかを示します。戻り値以外に引数に関する情報は明らかにしません。デフォルトは **いいえ** です。

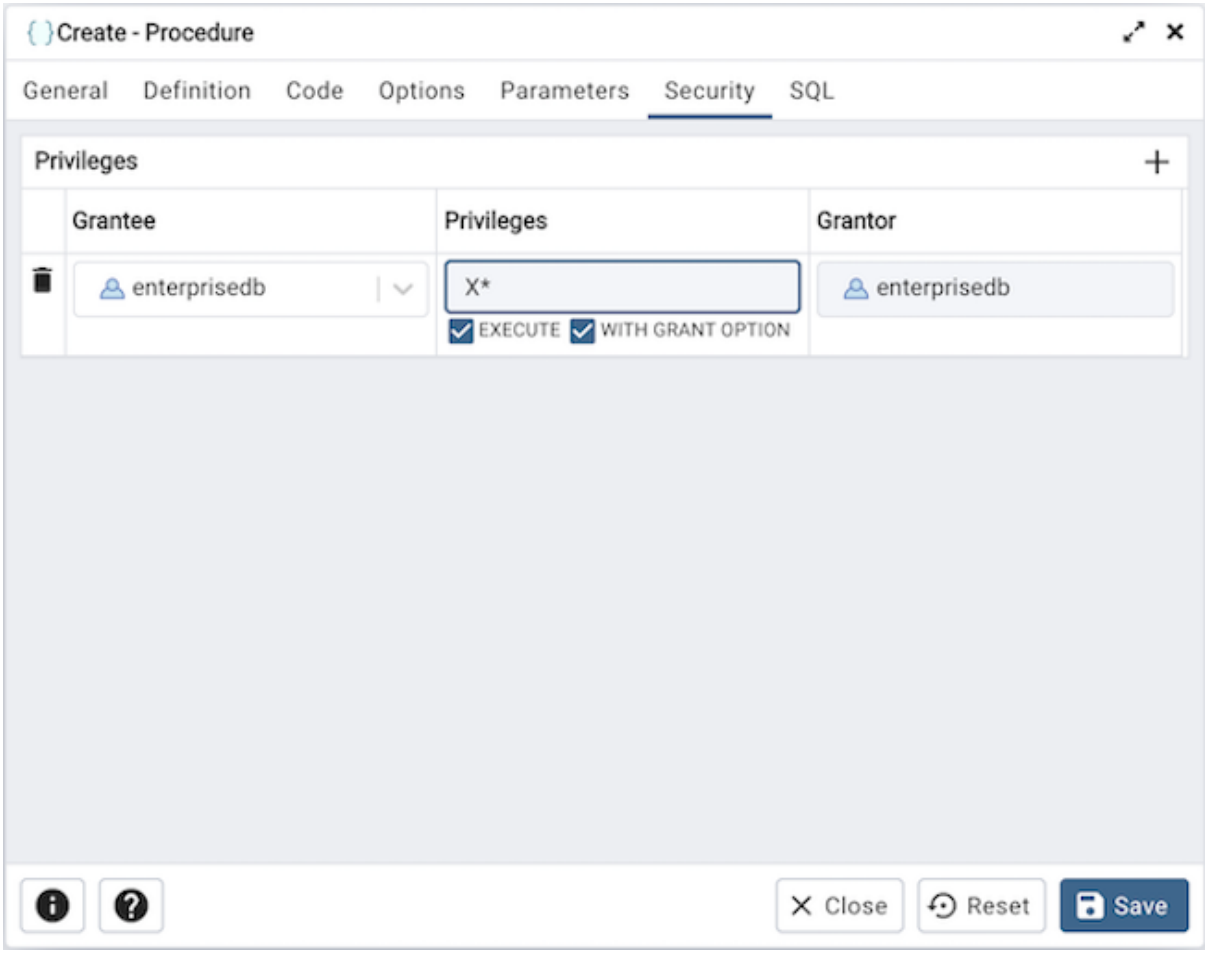
[**パラメーター**] タブをクリックして続行します。



[**パラメーター**] タブのフィールドを使用して、プロシージャが呼び出されたときに適用される設定を指定します。

- **パラメーター** パネルの **パラメーター名** の横にあるドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Add* ボタンをクリックして、変数をテーブルの *Name* フィールドに追加します。
- *Value* フィールドを使用して、選択した変数に関連付けられる値を指定します。このフィールドは状況依存です。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

Privileges パネルを使用して、プロシージャーの実行特権をロールに割り当てます。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

[**追加**] をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、プロシージャに適用されるセキュリティラベルを定義します。[**追加**] をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

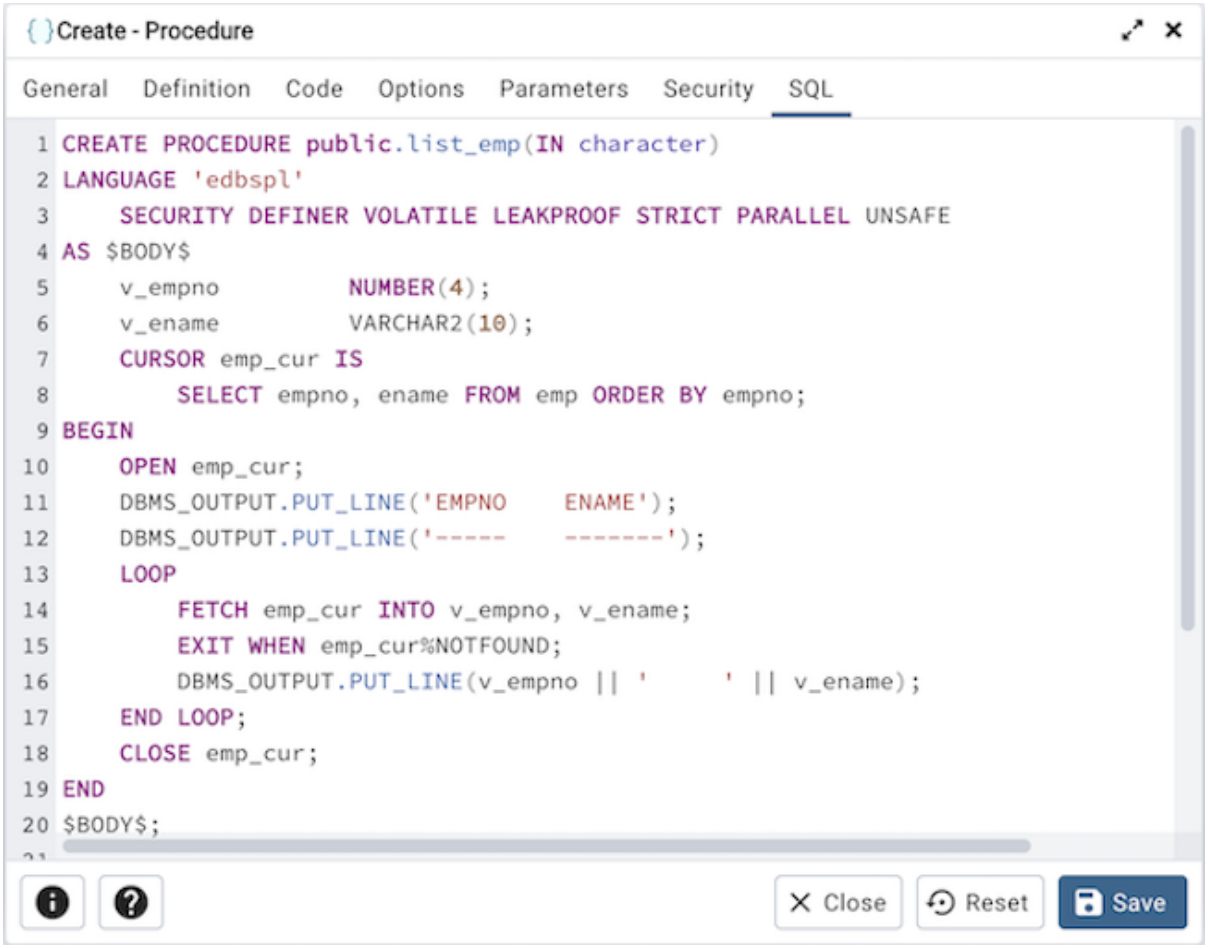
[*Add*] をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Procedure ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.18.1 例

以下は、[**プロシージャ**] ダイアログでの選択によって生成された sql コマンドの例です。



この例では、 *emp* という名前のテーブルから従業員のリストを返すプロシージャの作成を示しています。プロシージャーは SECURITYDEFINER であり、プロシージャーを定義したロールの特権で実行されます。

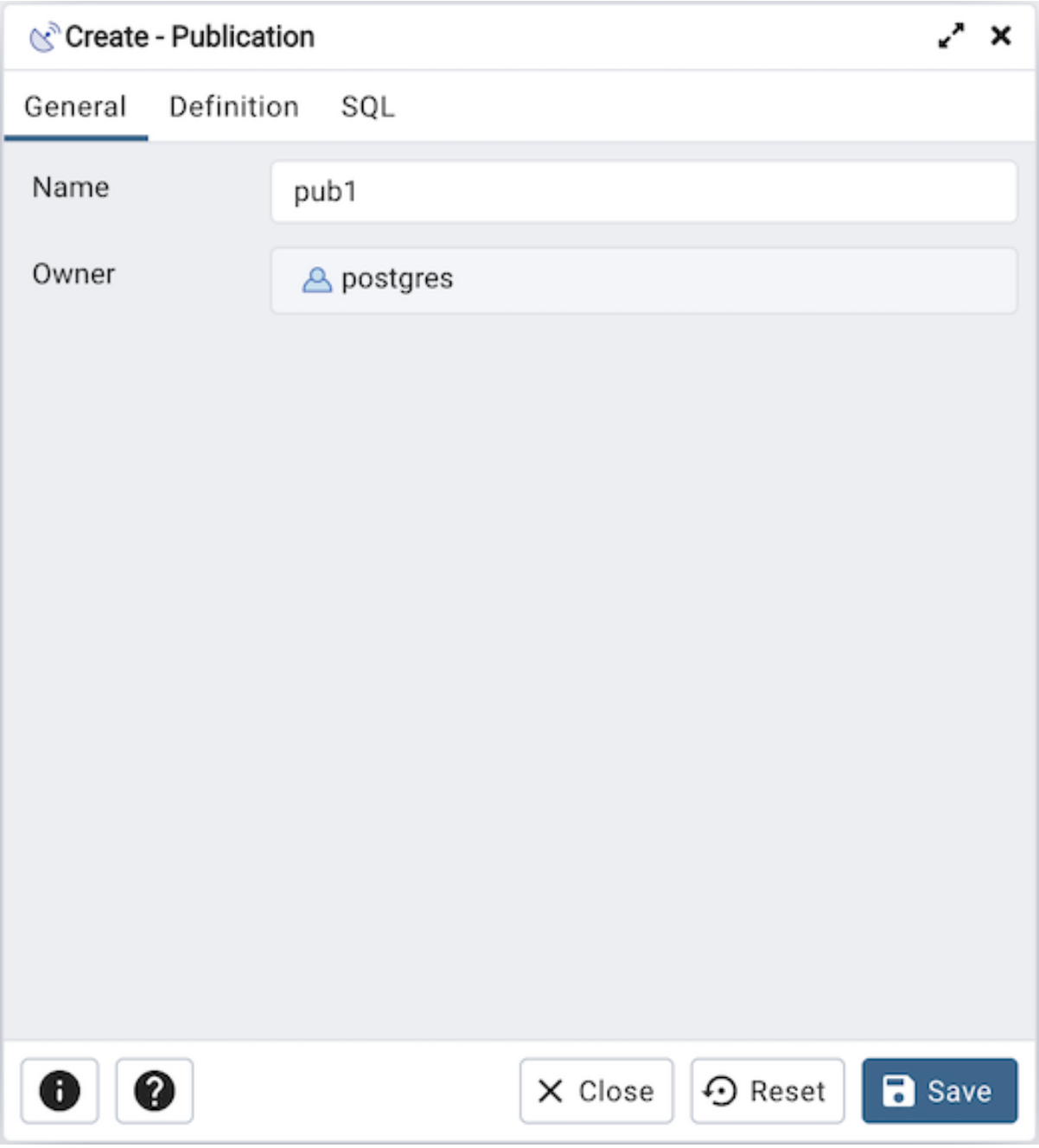
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.19 Publication Dialog

Logical replication は、パブリッシャーノード上の 1 つ以上の **パブリケーション** にサブスクライブする 1 つ以上の **サブスクライバー** で *publish* および *subscribe* モデルを使用します。

パブリケーションを作成するには、**パブリケーション** ダイアログを使用します。パブリケーションは、テーブルまたはテーブルのグループから生成された変更のセットであり、変更セットまたはレプリケーションセットと説明される場合があります。

パブリケーション ダイアログは、次のダイアログタブを介してパブリケーションの開発を編成します。 **一般** と **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、パブリケーションを識別します。

- *Name* フィールドを使用して、パブリケーションの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** フィールドは、現在接続しているユーザーとしてユーザーの名前を自動的に取得します。所有者は、alterPublication を使用してパブリケーションを作成した後に変更できます。

[**定義**] タブをクリックして続行します。



[**定義**] タブを使用して、パブリケーションのプロパティを設定します。

- 今後作成されるテーブルを含むデータベースのすべてのテーブルをレプリケートするには、 **すべてのテーブル***?* の横にあるスイッチを **はい** に移動します。
- *Onlytable?* の横にあるスイッチを *Yes* に移動して、そのすべての子孫テーブルを除くリストされたテーブルのみを複製します。

- リストされたすべてのテーブルを複製するには、 **テーブル** フィールドにテーブルまたはテーブルのリストを指定します。
- *With* セクションを使用して、新しいパブリケーションによってサブスクライバーにパブリッシュされる DML 操作を決定します。これらの DML 操作をパブリケーションからサブスクリプションに複製しない場合は、 *INSERT* 、 *UPDATE* 、 *DELETE* 、または *TRUNCATE* の横にあるスイッチを *No* に移動します。デフォルトでは、すべてのスイッチは *Yes* に設定されており、すべての DML 操作が許可されます。

注釈: パブリッシュされたテーブルには、UPDATE および DELETE 操作をレプリケートできるように構成された「レプリカ ID」が必要です。ALTERTABLE ステートメントで変更できます。レプリカ ID の詳細については、 Logical Replication Publication を参照してください。

SQL タブをクリックして続行します。

Publication ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

Example

以下は、 **パブリケーション** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



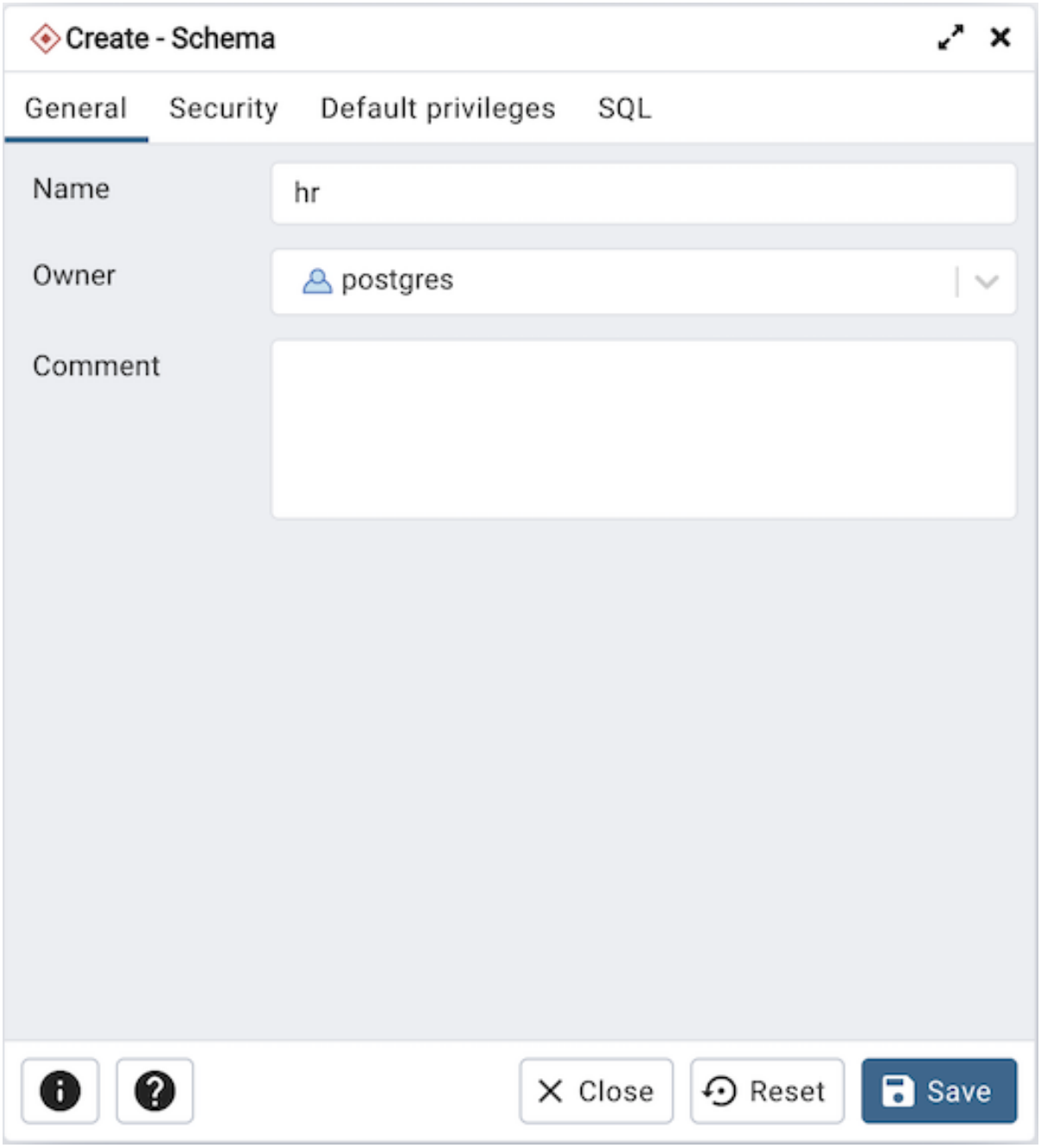
この例では、 *postgres* が所有する *pub1* という名前のパブリケーションを作成します。すべての DML 操作のレプリケーションが許可されます。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- **リセット** ボタンをクリックして、すべてのデフォルト設定を復元します。

5.20 Schema Dialog

スキーマを定義するには、 **スキーマ** ダイアログを使用します。スキーマは、データベースの組織的な主力であり、ディレクトリや名前空間に似ています。スキーマを作成するには、データベースのスーパーユーザーであるか、CREATE 権限を持っている必要があります。

スキーマ ダイアログは、次のダイアログタブを介してスキーマの開発を整理します： **一般** および **セキュリティ** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

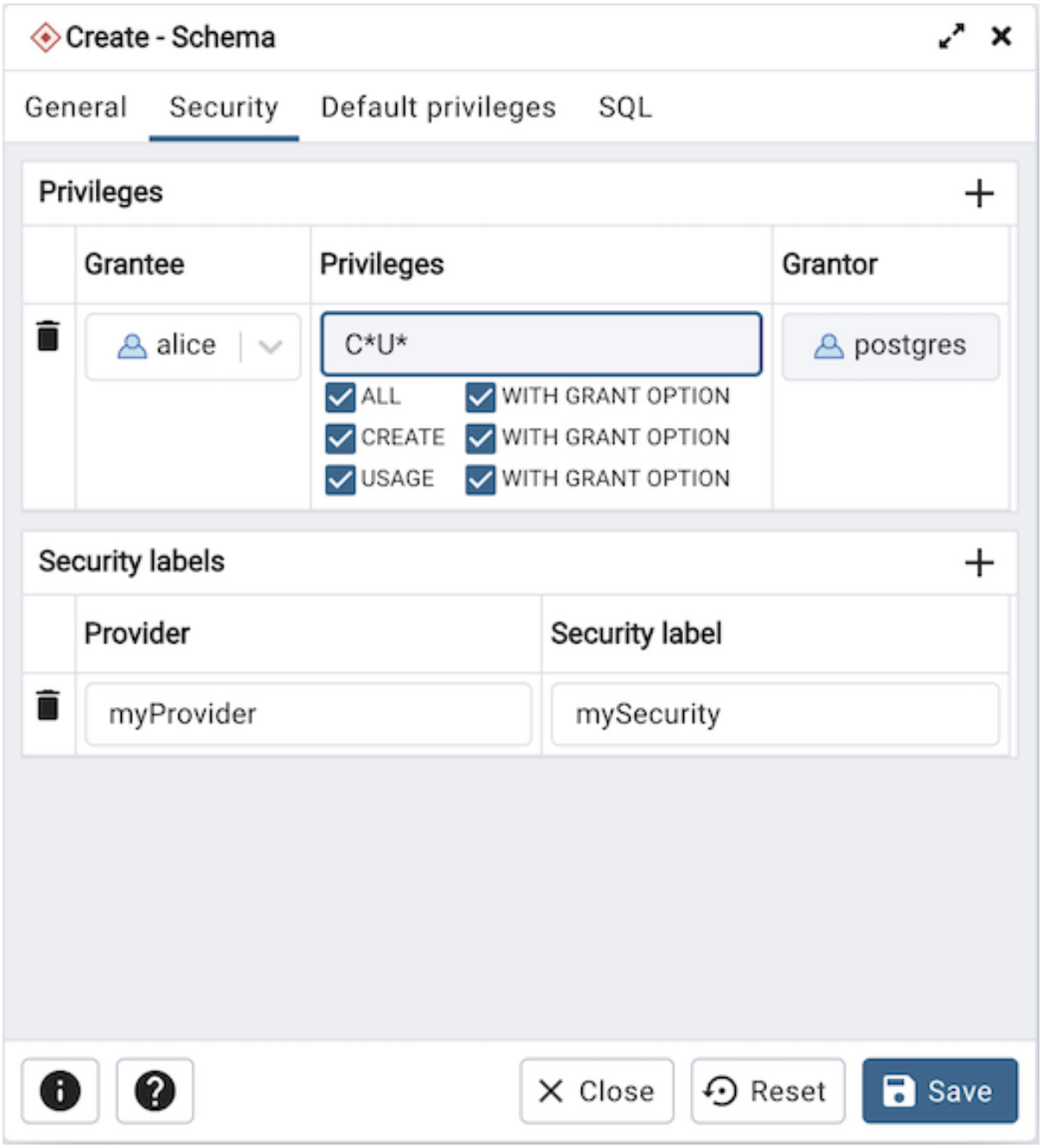


[**全般**] タブのフィールドを使用して、スキーマを識別します。

- *Name* フィールドを使用して、スキーマの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

- **所有者** フィールドのドロップダウンリストボックスからスキーマの所有者を選択します。
- スキーマに関するメモを **コメント** フィールドに保存します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、スキーマに特権とセキュリティラベルを割り当てます。

追加 アイコン（+）をクリックして、 **特権** パネルで一連の特権を割り当てます。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示され

ます。

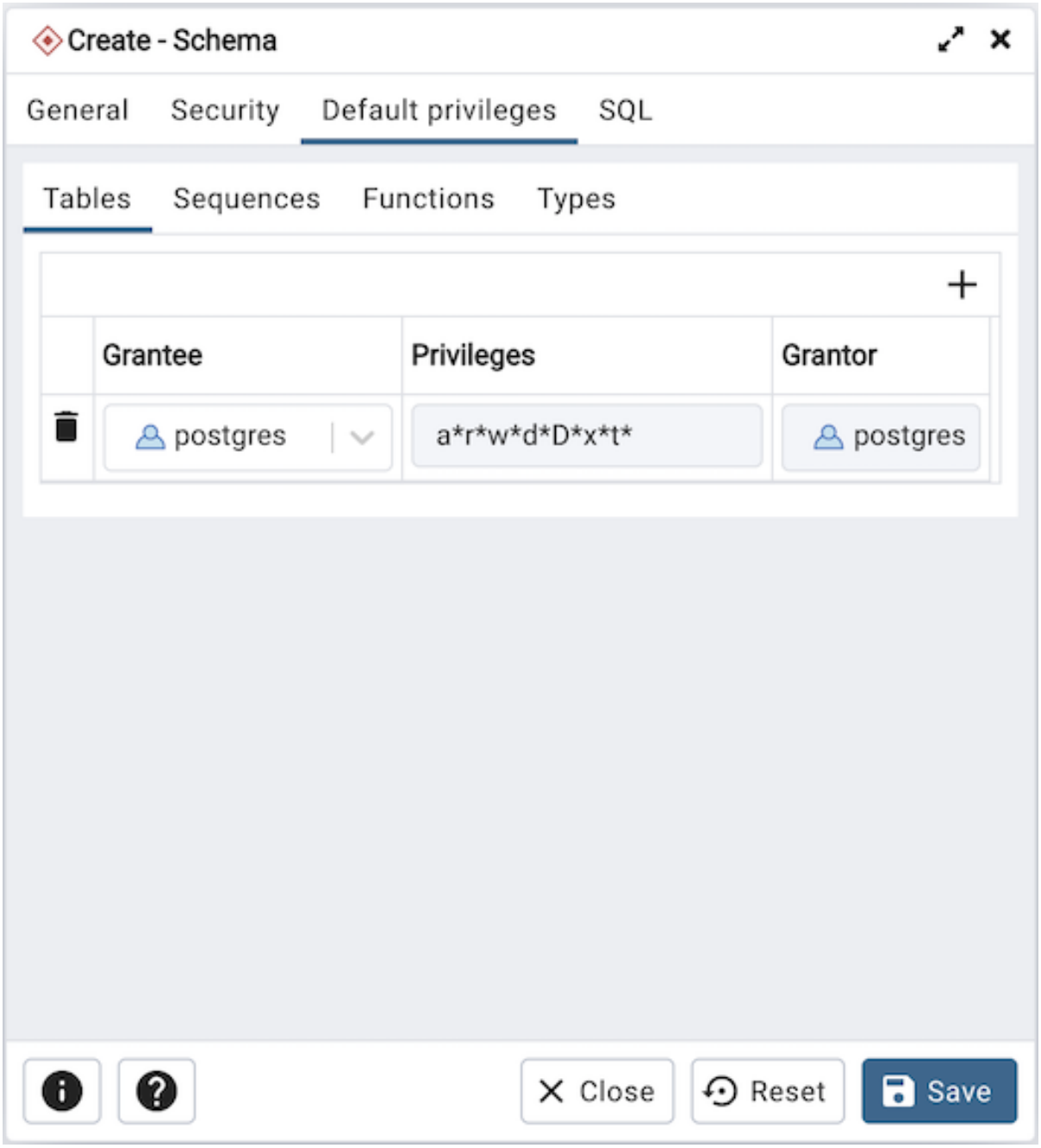
追加 アイコン（+）をクリックして、追加の特権セットを割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

追加 アイコン（+）をクリックして、 **セキュリティラベル** パネルでセキュリティラベルを割り当てます。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**デフォルトの権限**] タブをクリックして続行します。



[**デフォルトの特権**] タブを使用して、テーブル、シーケンス、ファンクション、およびタイプに対する特権を付与します。 *DefaultPrivileges* タブ内にネストされたタブを使用してデータベースオブジェクトを指定し、 *Add* アイコン（+）をクリックして一連の特権を割り当てます。

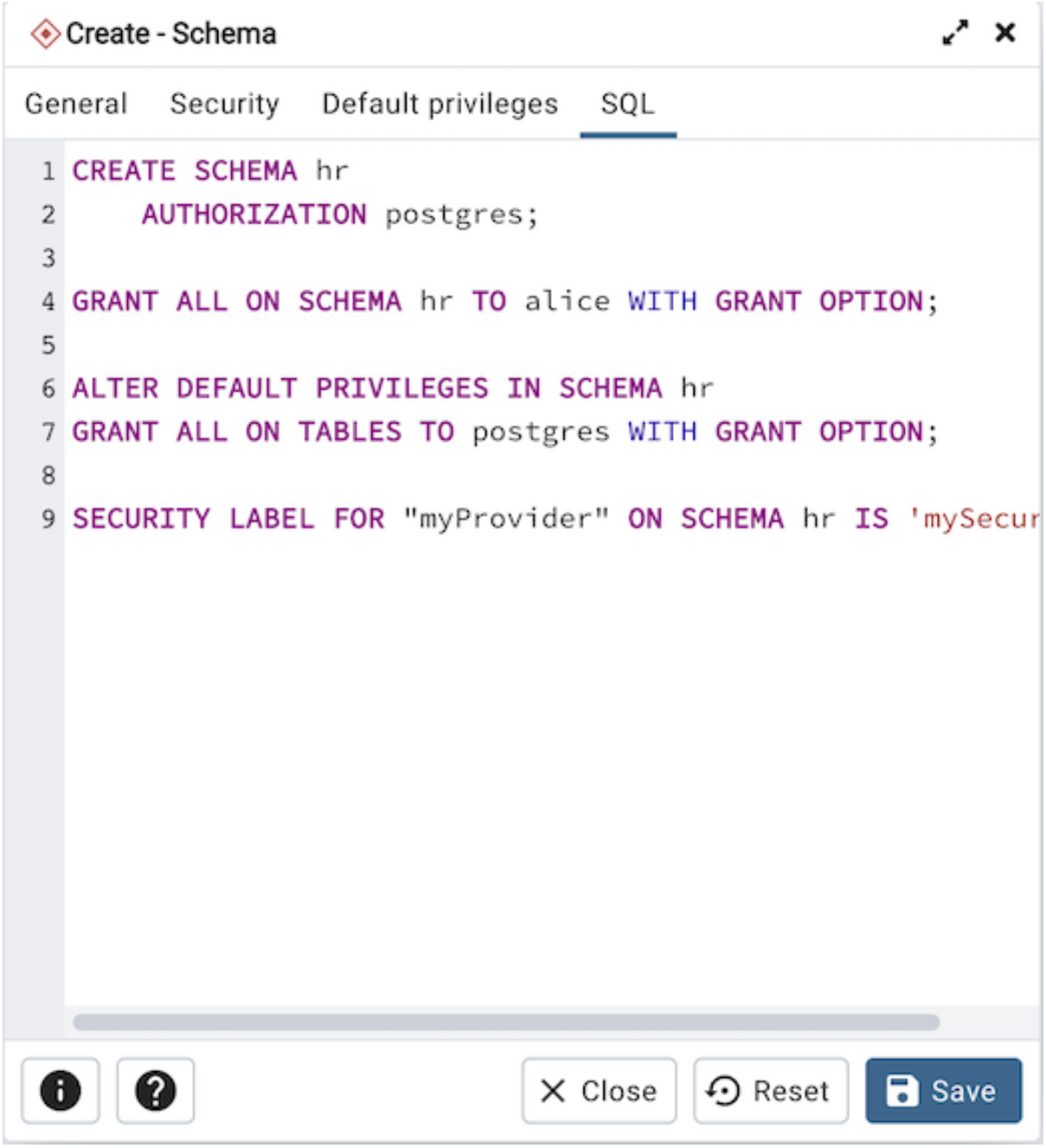
- *Grantee* フィールドのドロップダウンリストボックスから、スキーマで特権が付与されるロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

SQL タブをクリックして続行します。

スキーマ ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.20.1 例

以下は、 **スキーマ** ダイアログでの選択によって生成される sql コマンドの例です。



この例では、hr という名前のスキーマを作成します。このコマンドは、 *USAGE* 特権を *public* に付与し、特権を付与する機能を *alice* に割り当てます。

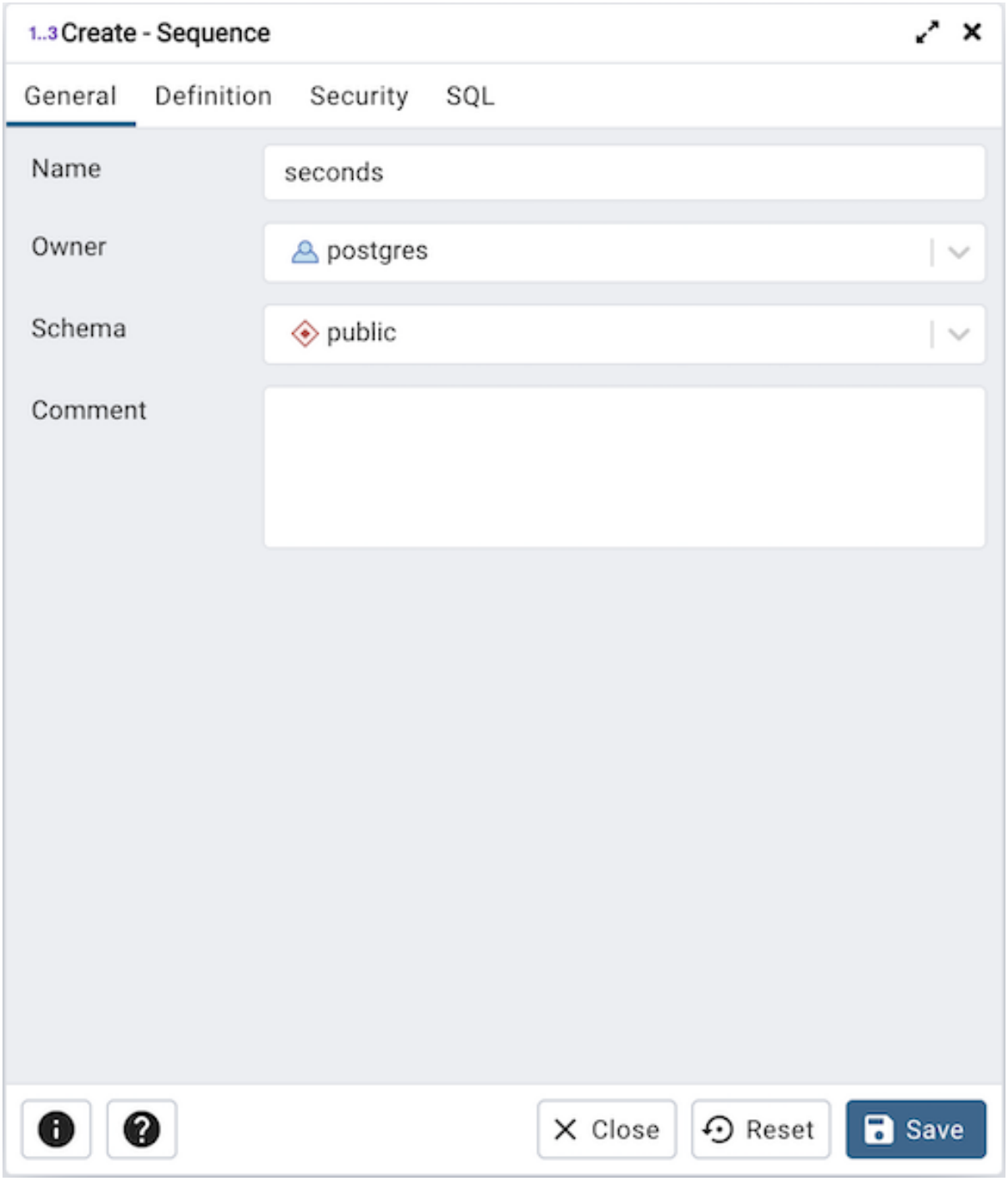
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。

- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.21 Sequence Dialog

シーケンス ダイアログを使用して、シーケンスを作成します。シーケンスは、連続した順序で一意の値を生成します（連続しているとは限りません）。

シーケンス ダイアログは、 **一般** 、 **定義** 、および **セキュリティ** のダイアログタブを使用してシーケンスの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

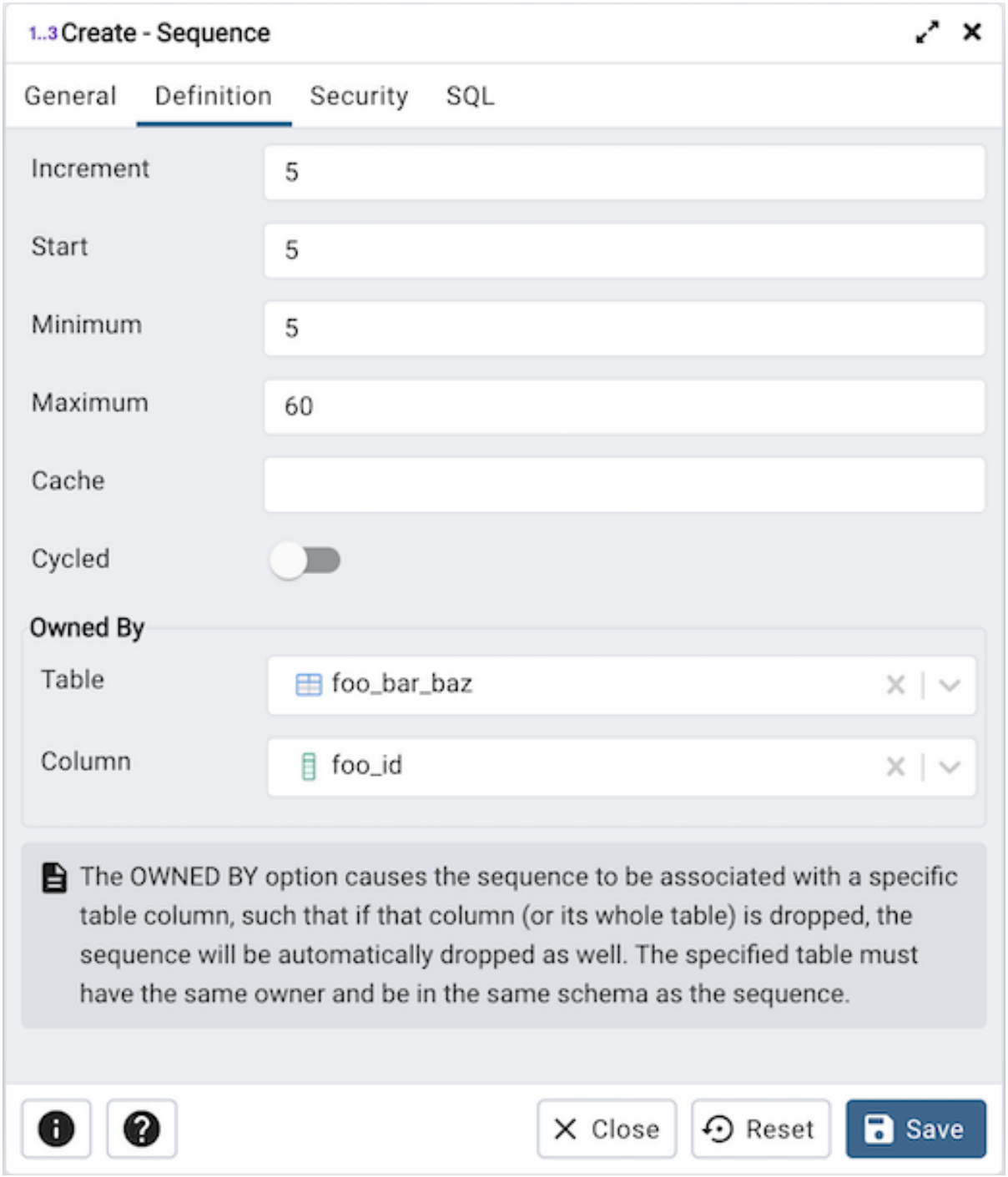


[*General*] タブのフィールドを使用して、シーケンスを識別します。

- *Name* フィールドを使用して、シーケンスに説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。シーケンス名は、同じスキーマ内の他のシーケンス、テーブル、インデックス、ビューまたは外部テーブルの名前とは異なる必要があります。
- **所有者** の横にあるドロップダウンリストボックスを使用して、シーケンスを所有するロールの名前を選択します。
- **スキーマ** の横にあるドロップダウンリストボックスを使用して、シーケンスが存在するスキーマを選択します。

- シーケンスに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



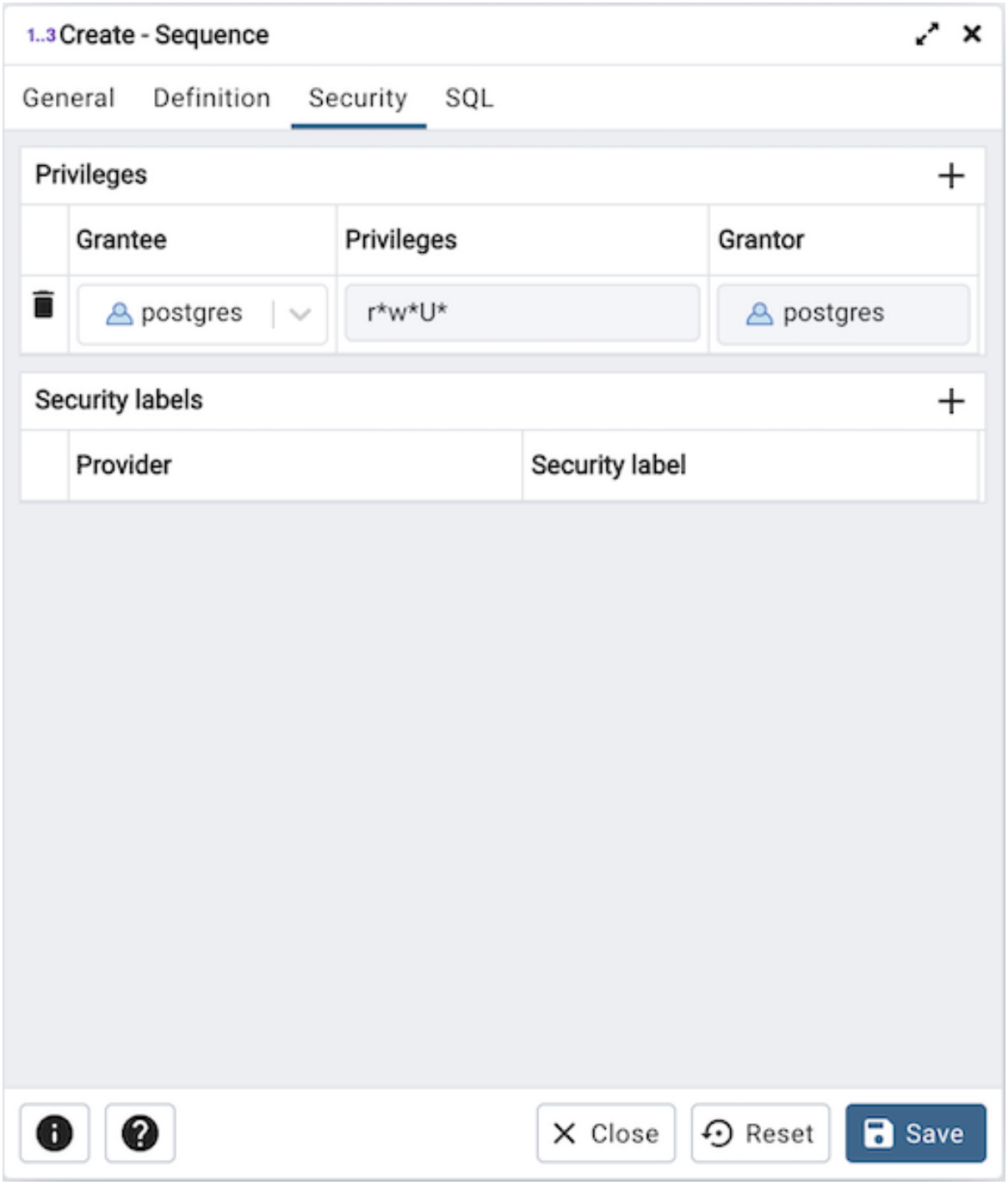
[**定義**] タブのフィールドを使用して、シーケンスを定義します。

- **インクリメント** フィールドを使用して、新しい値を作成するために現在のシーケンス値に追加する値を指定します。
- *Start* フィールドに値を入力して、シーケンスの開始値を指定します。デフォルトの開始値は、昇順の場合は MINVALUE、降順の場合は MAXVALUE です。
- *Minimum* フィールドに値を入力して、シーケンスが生成できる最小値を指定します。この句を指定

しないか、NOMINVALUE を指定すると、デフォルトが使用されます。昇順と降順のデフォルトは、それぞれ 1 と-263-1 です。

- *Maximum* フィールドに値を入力して、シーケンスの最大値を指定します。この句を指定しないか、NOMAXVALUE を指定すると、デフォルト値が使用されます。デフォルトは、昇順と降順でそれぞれ 263-1 と-1 です。
- *Cache* フィールドに値を入力して、アクセスを高速化するために事前に割り当ててメモリに保存するシーケンス番号を指定します。最小値は 1（一度に生成できる値は 1 つだけ、つまりキャッシュ）であり、これもデフォルトです。
- *Cycled* スイッチを *Yes* の位置に移動して、昇順または降順のシーケンスがそれぞれ MAXVALUE または MINVALUE に達したときにシーケンスを折り返せるようにします。制限に達した場合、次に生成される数値はそれぞれ MINVALUE または MAXVALUE です。デフォルトは **いいえ** です。
- *OWNEDBY* オプションを使用すると、シーケンスが特定のテーブル列に関連付けられ、その列（またはそのテーブル全体）が削除されると、シーケンスも自動的に削除されます。指定されたテーブルは、シーケンスと同じ所有者であり、同じスキーマにある必要があります。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、シーケンスのセキュリティラベルを定義します。

特権 パネルを使用して特権を割り当てます。 **追加** アイコン（+）をクリックして権限を設定します。

- *Grantee* フィールドのドロップダウンリストボックスから、特権が付与されるロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、シーケンスに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

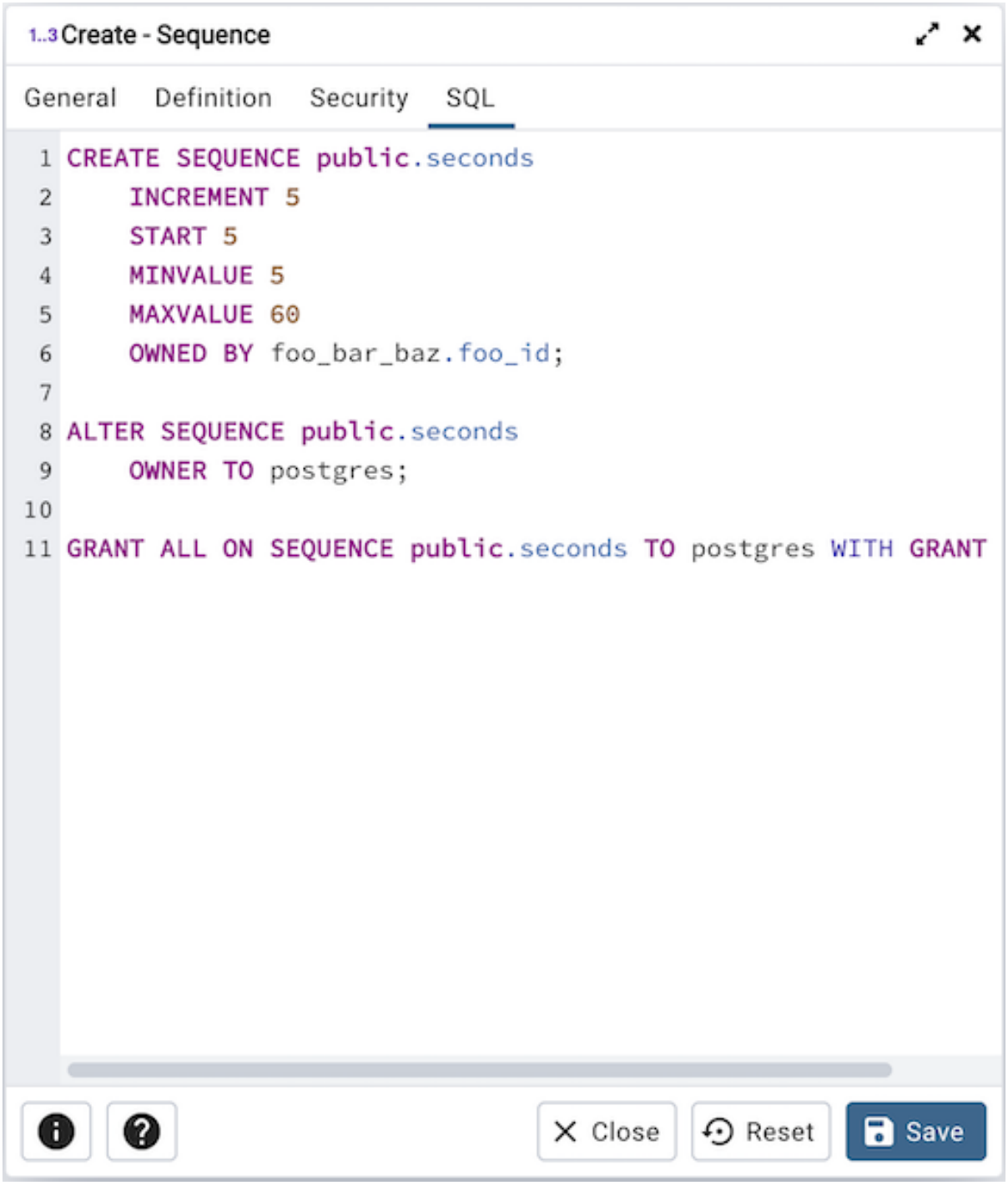
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Sequence ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.21.1 例

以下は、 **シーケンス** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



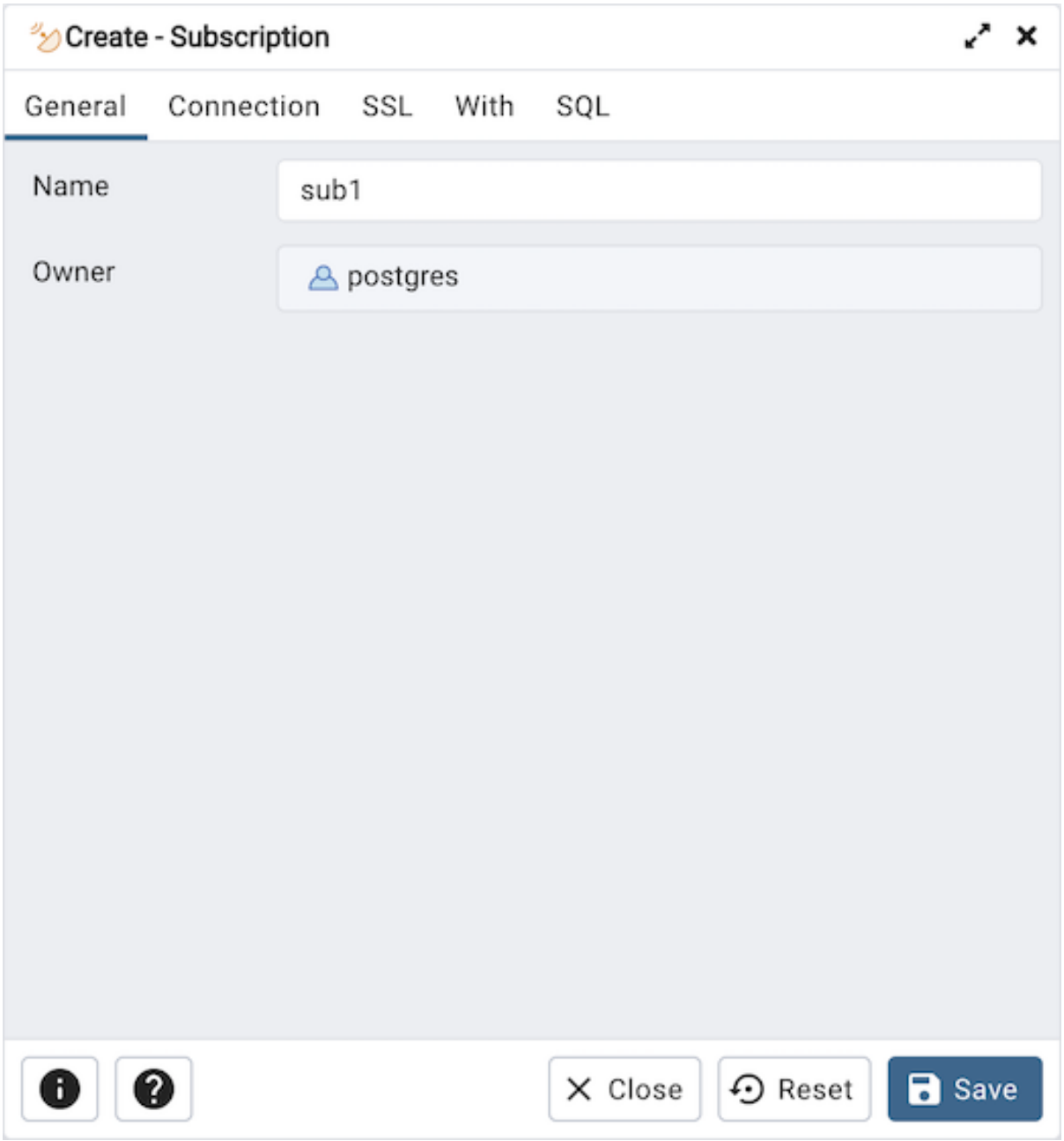
この例は、 *seconds* という名前のシーケンスを示しています。シーケンスは *5* 秒単位で増加し、最大値 *60* に達すると停止します。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.22 Subscription Dialog

サブスクリプション ダイアログを使用して、サブスクリプションを作成します。サブスクリプションは、別のデータベースへの接続と、サブスクライブするパブリケーション (1 つ以上) を定義します。

サブスクリプション ダイアログは、次のダイアログタブを介してサブスクリプションの開発を整理します。**一般** 、 **接続** 、および **ウィズ** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

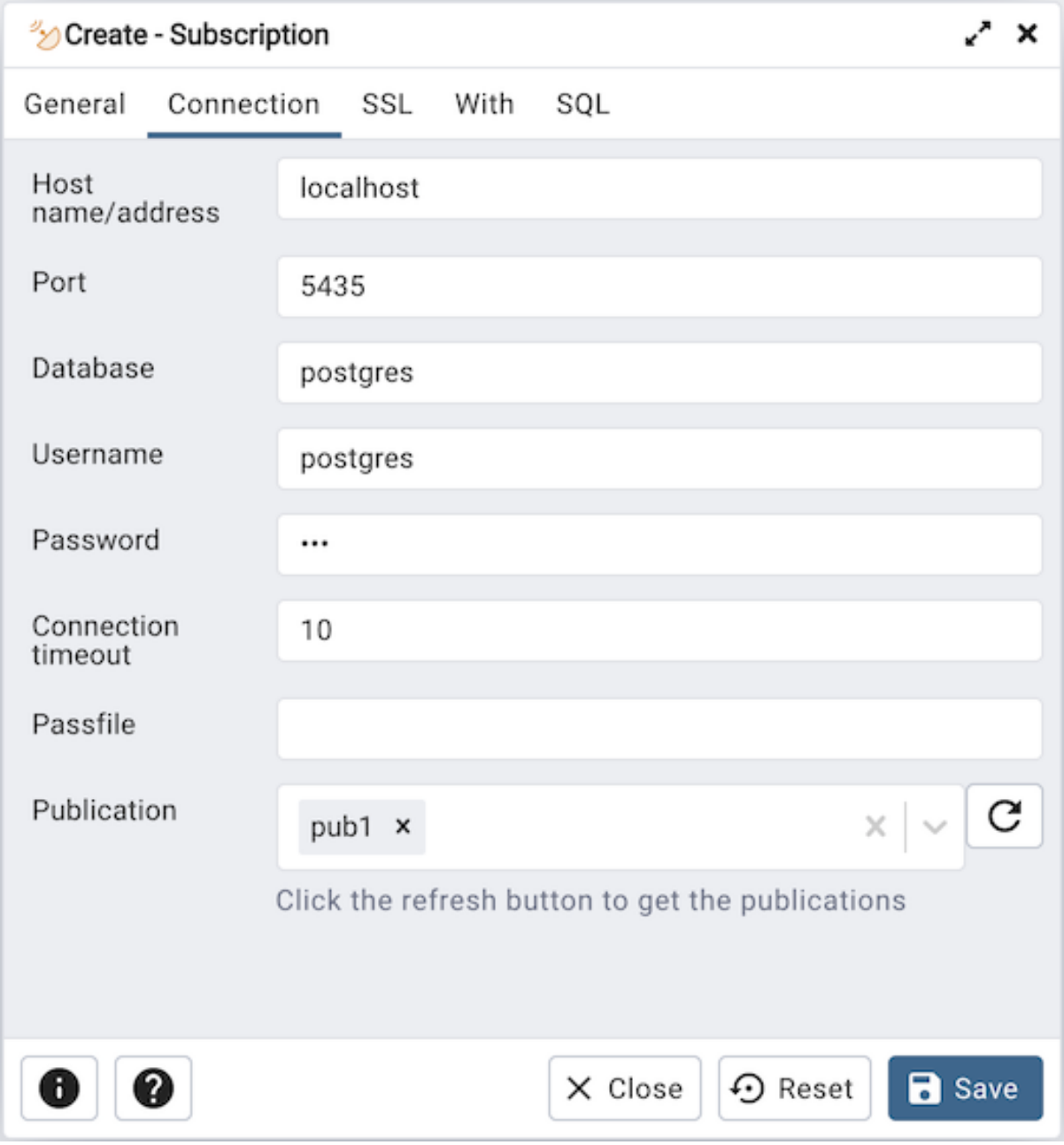


[**全般**] タブのフィールドを使用して、サブスクリプションを識別します。

- *Name* フィールドを使用して、サブスクリプションの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** フィールドは、現在接続しているユーザーとしてユーザーの名前を自動的に取得します。所

有者は、altersubscription を使用してサブスクリプションを作成した後に変更できます。サブスクリプションの所有者にはスーパーユーザー権限があることに注意してください。

[**接続**] タブをクリックして続行します。

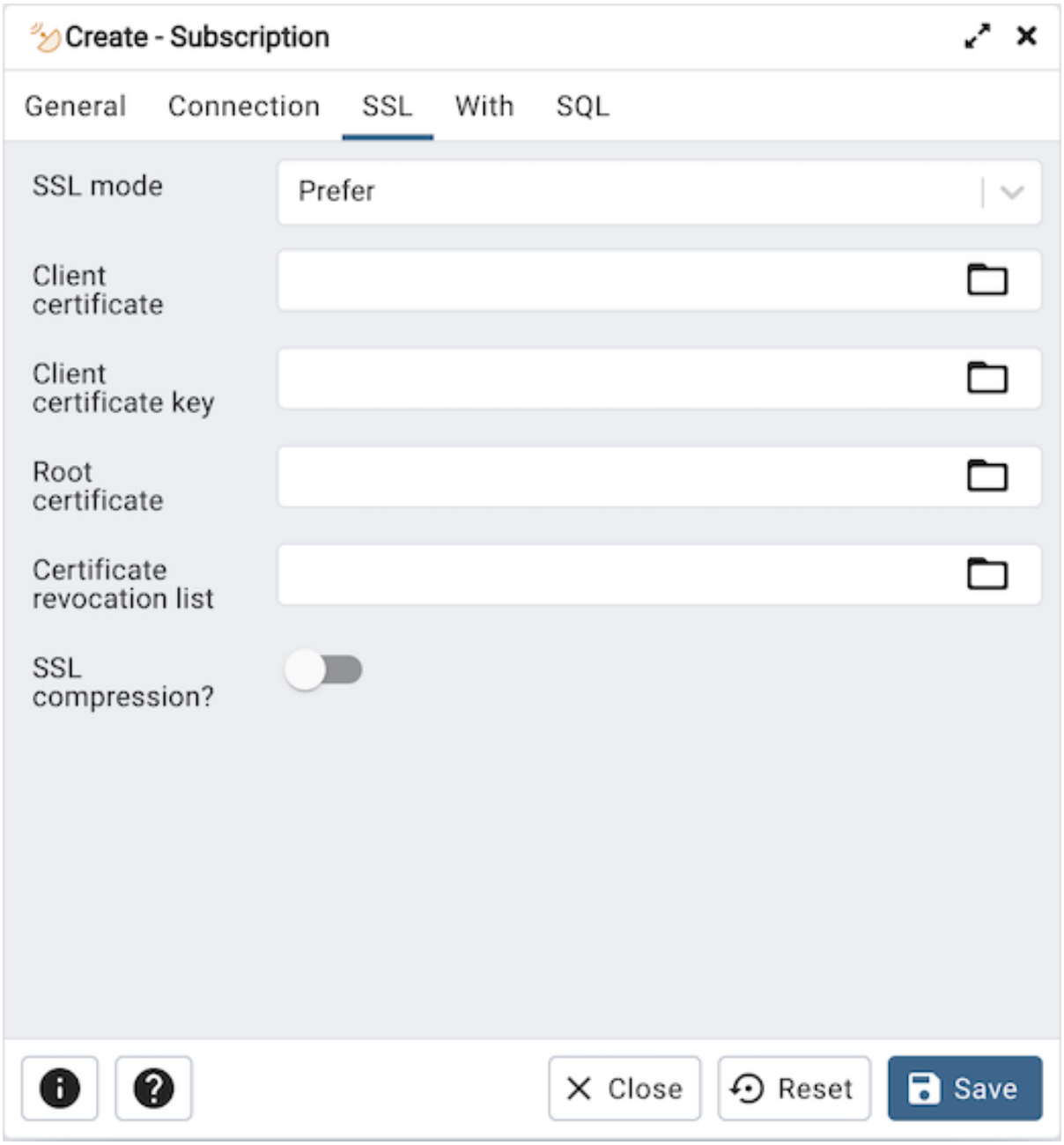


[**接続**] タブを使用して、パブリッシャーへの接続文字列を定義します。

- **ホスト名/アドレス** フィールドを使用して、パブリケーションの有効なホスト名または IP アドレスを入力します。
- *Port* フィールドを使用して、パブリケーションが存在する PostgresServer に接続するポート番号を指定します。
- *Username* フィールドを使用して、パブリケーションに接続するユーザーの名前を入力します。
- *Password* を使用して、ユーザーのパスワードを入力します。

- **データベース** フィールドを使用して、パブリケーションが存在するデータベースに接続します。
- **接続タイムアウト** フィールドを使用して、接続の最大待機時間を秒単位で指定します。ゼロまたは指定されていないことは、無限に待機することを意味します。2 秒未満のタイムアウトを使用することはお勧めしません。
- *Passfile* フィールドを使用して、パスワードファイル（.pgpass）の場所を指定します。.pgpass ファイルを使用すると、ユーザーは接続時にパスワードを入力せずにログインできます。詳細については、Section 33.15 of the Postgres documentation を参照してください。
- *Publication* フィールドを使用して、サブスクライブするパブリッシャーのパブリケーション名を指定します。最後に更新ボタンをクリックして既存のパブリケーションの名前をロードし、リストから選択します。その分野で知られている出版物の名前を書くこともできます。

[*SSL*] タブをクリックして続行します。



[*SSL*] タブのフィールドを使用して、SSL を構成します。

- *SSL* フィールドのドロップダウンリストボックスを使用して、サーバーが使用する SSL 接続のタイプを選択します。SSL 暗号化の使用の詳細については、 Section 33.18 of the Postgres documentation を参照してください。

pgAdmin がサーバーモード（デフォルトモード）でインストールされている場合、プラットフォーム固有のファイルマネージャーダイアログを使用して、SSL 暗号化をサポートするファイルをサーバーにアップロードできます。[ファイルマネージャー] ダイアログにアクセスするには、次の各フィールドの右側にあるアイコンをクリックします。

- **クライアント証明書** フィールドを使用して、クライアント SSL 証明書を含むファイルを指定します。このファイルは、pgAdmin がデスクトップモードでインストールされている場合はデフォルトの *~/.postgresql/postgresql.crt* 、pgAdmin が Web モードでインストールされている場合は *<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.crt* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **クライアント証明書キー** フィールドを使用して、クライアント証明書に使用される秘密鍵を含むファイルを指定します。このファイルは、pgAdmin がデスクトップモードでインストールされている場合はデフォルトの *~/.postgresql/postgresql.key* 、pgAdmin が Web モードでインストールされている場合は *<STORAGE_DIR>/<USERNAME>/.postgresql/postgresql.key* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **ルート証明書** フィールドを使用して、SSL 認証局を含むファイルを指定します。このファイルは、デフォルトの *~/.postgresql/root.crt* を置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- **証明書失効リスト** フィールドを使用して、SSL 証明書失効リストを含むファイルを指定します。このリストは、 *~/.postgresql/root.crl* にあるデフォルトのリストを置き換えます。SSL 接続が確立されていない場合、このパラメーターは無視されます。
- *SSL* **圧縮***?* が *True* に設定されている場合、SSL 接続で送信されるデータは圧縮されます。デフォルト値は *False* です（圧縮は無効です）。SSL 接続が確立されていない場合、このパラメーターは無視されます。

> **警告:** サーバーモードでは、証明書、プライベートキー、および失効リストは、pgAdmin サーバープロセスが実行されるユーザーアカウントが所有するサーバー上のユーザーごとのファイル格納領域に格納されます。これは、サーバーの管理者がこれらのファイルにアクセスできることを意味します。この機能の使用を選択する前に、適切な注意を払う必要があります。

[**とともに**] タブをクリックして続行します。



[*With*] タブを使用して、サブスクリプションのパラメーターを定義します。

- **データをコピーしますか**? スイッチは、サブスクライブされているパブリケーション内の既存のデータをレプリケーションの開始時にコピーするかどうかを指定します。デフォルトでは **はい** に設定されています。
- *Createslot?* スイッチは、コマンドがパブリッシャーにレプリケーションスロットを作成するかどうかを指定します。デフォルトでは **はい** に設定されています。注：パブリッシャーとサブスクライバーが両方とも同じ PostgreSQL サーバー内にある場合、デフォルトで **いいえ** に設定されます。

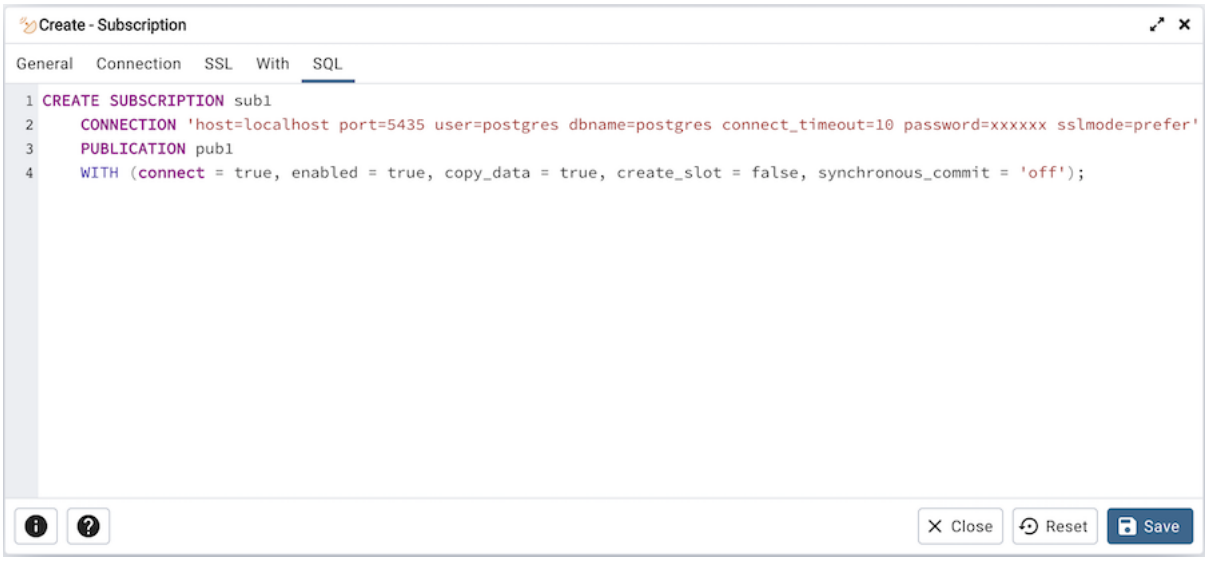
- *Enabled?* スイッチは、サブスクリプションをアクティブにレプリケートするか、セットアップしただけでまだ開始しないかを指定します。デフォルトでは **はい** に設定されています。
- *Connect?* は、CREATESUBSCRIPTION がパブリッシャーに接続するかどうかを指定します。デフォルトでは、**はい** に設定されています。これを *No* に設定すると、enabled、create_slot、および copy_data のデフォルト値が *No* に変更されます。
- *SlotName* フィールドを使用して、使用するレプリケーションスロットの名前を指定します。デフォルトでは、スロット名にサブスクリプションの名前を使用します。
- *Synchronouscommit* フィールドを使用して、synchronous_commit 設定をオーバーライドします。デフォルトでは、**オフ** に設定されています。論理レプリケーションには off を使用しても安全です。同期が失われたためにサブスクライバーがトランザクションを失った場合、データはパブリッシャーから再度送信されます。

SQL タブをクリックして続行します。

サブスクリプション ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

Example

以下は、 **サブスクリプション** ダイアログでのユーザー選択によって生成される sql コマンドの例です。



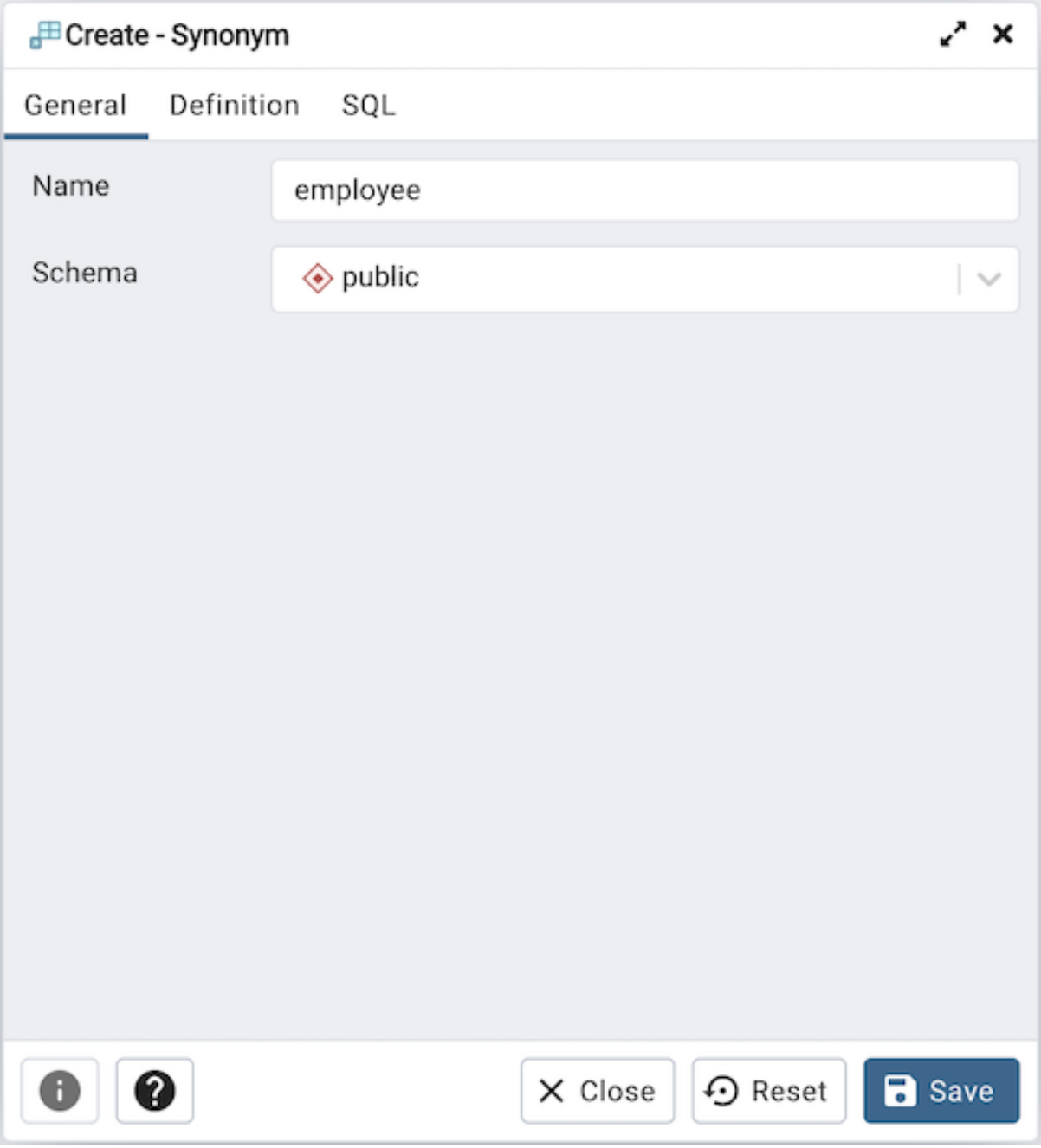
この例では、 *postgres* が所有する *sub1* という名前のサブスクリプションを作成します。パブリケーション *pub1* からデータを複製します。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- **リセット** ボタンをクリックして、すべてのデフォルト設定を復元します。

5.23 Synonym Dialog

シノニム ダイアログを使用して、ターゲットオブジェクトの名前をユーザー定義のシノニムに置き換えます。

シノニム ダイアログは、 *General* タブを介してシノニムの開発を整理します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

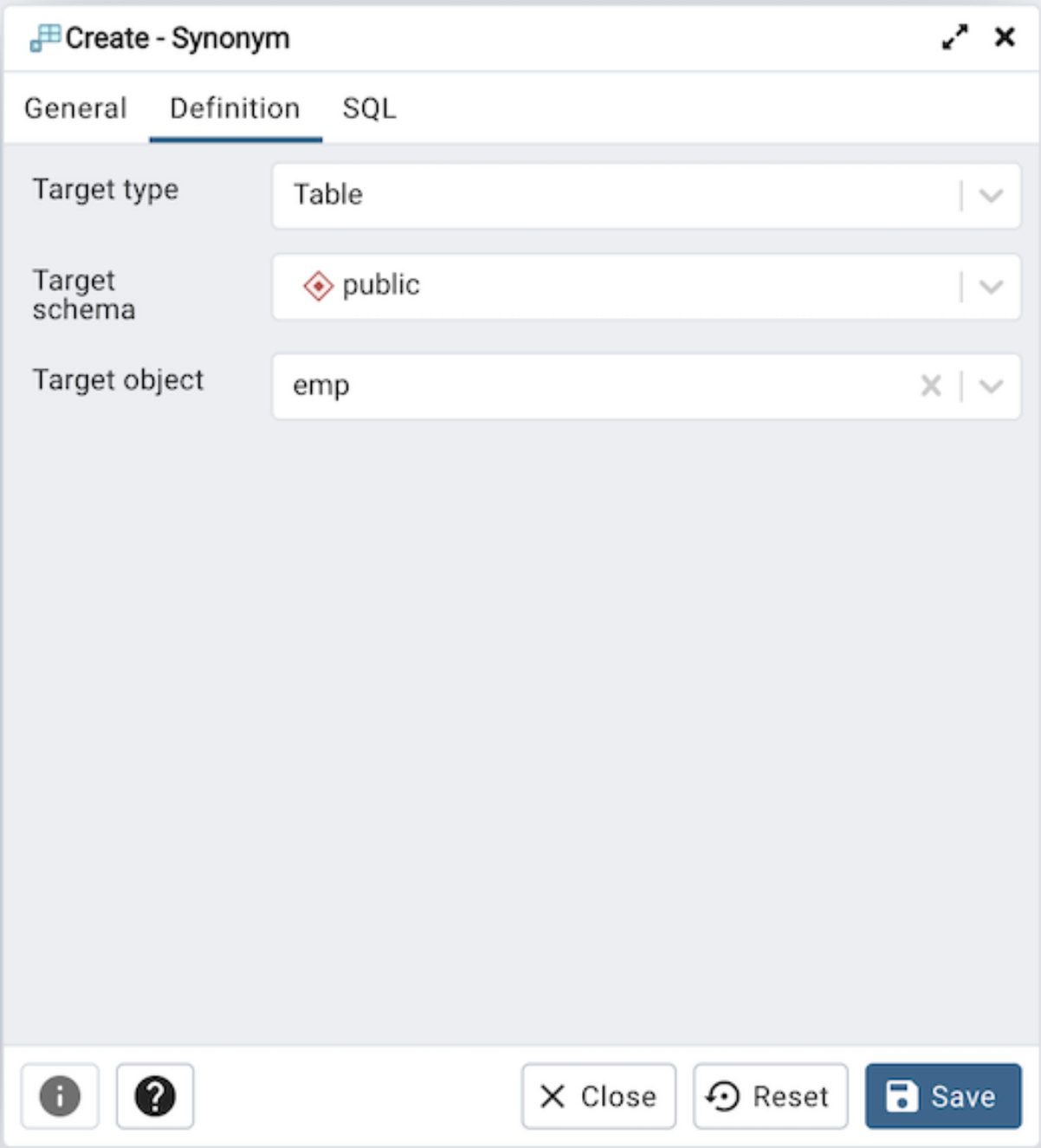


[*General*] タブのフィールドを使用して、同義語を識別します。

- *Name* フィールドを使用して、同義語の名前を指定します。名前は *pgAdmin* ツリーコントロールに表示されます。
- シノニムが存在するスキーマの名前を **スキーマ** フィールドのドロップダウンリストボックスから選

択します。

定義パネルで、ターゲットを特定します。



- **ターゲットタイプ** の横にあるドロップダウンリストボックスを使用して、シノニムによって参照されるオブジェクトのタイプを選択します。
- **ターゲットスキーマ** の横にあるドロップダウンリストボックスを使用して、オブジェクトが存在するスキーマの名前を選択します。
- *TargetObject* の横にあるドロップダウンリストボックスを使用して、シノニムによって参照されるオブジェクトの名前を選択します。

SQL タブをクリックして続行します。

シノニム ダイアログでの選択と入力により、SQL コマンドが生成されます。



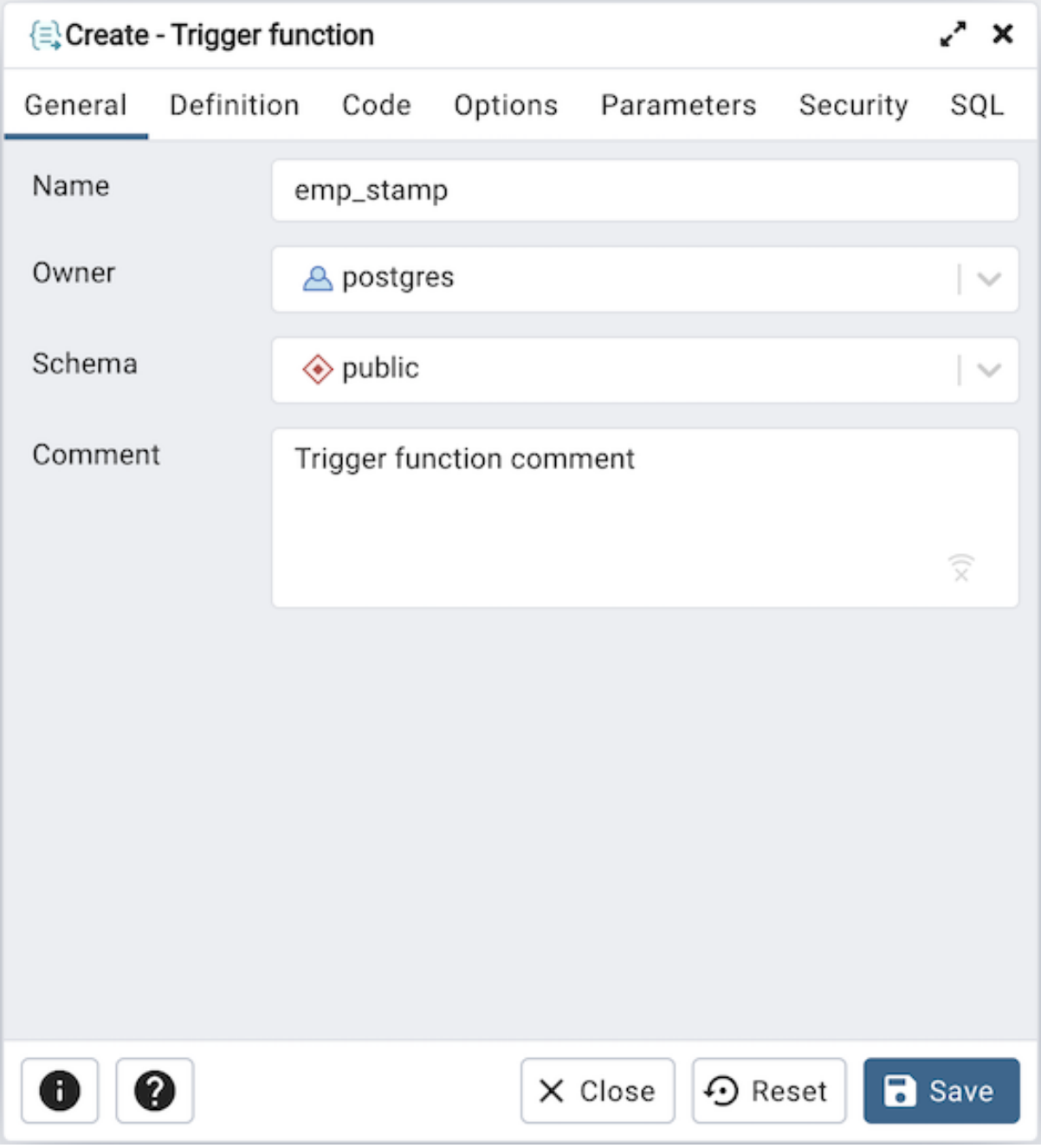
この例では、 *emp_hist* という名前の *emp* テーブルのシノニムを作成します。

- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.24 Trigger Function Dialog

トリガー機能 ダイアログを使用して、trigger_function を作成または管理します。トリガー関数は、トリガーが起動したときに呼び出されるアクションを定義します。

トリガー機能 ダイアログは、 **一般** 、 **定義** 、 **コード** 、 **オプション** 、 **パラメーター** 、 **セキュリティ** のダイアログタブを介してトリガー機能の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

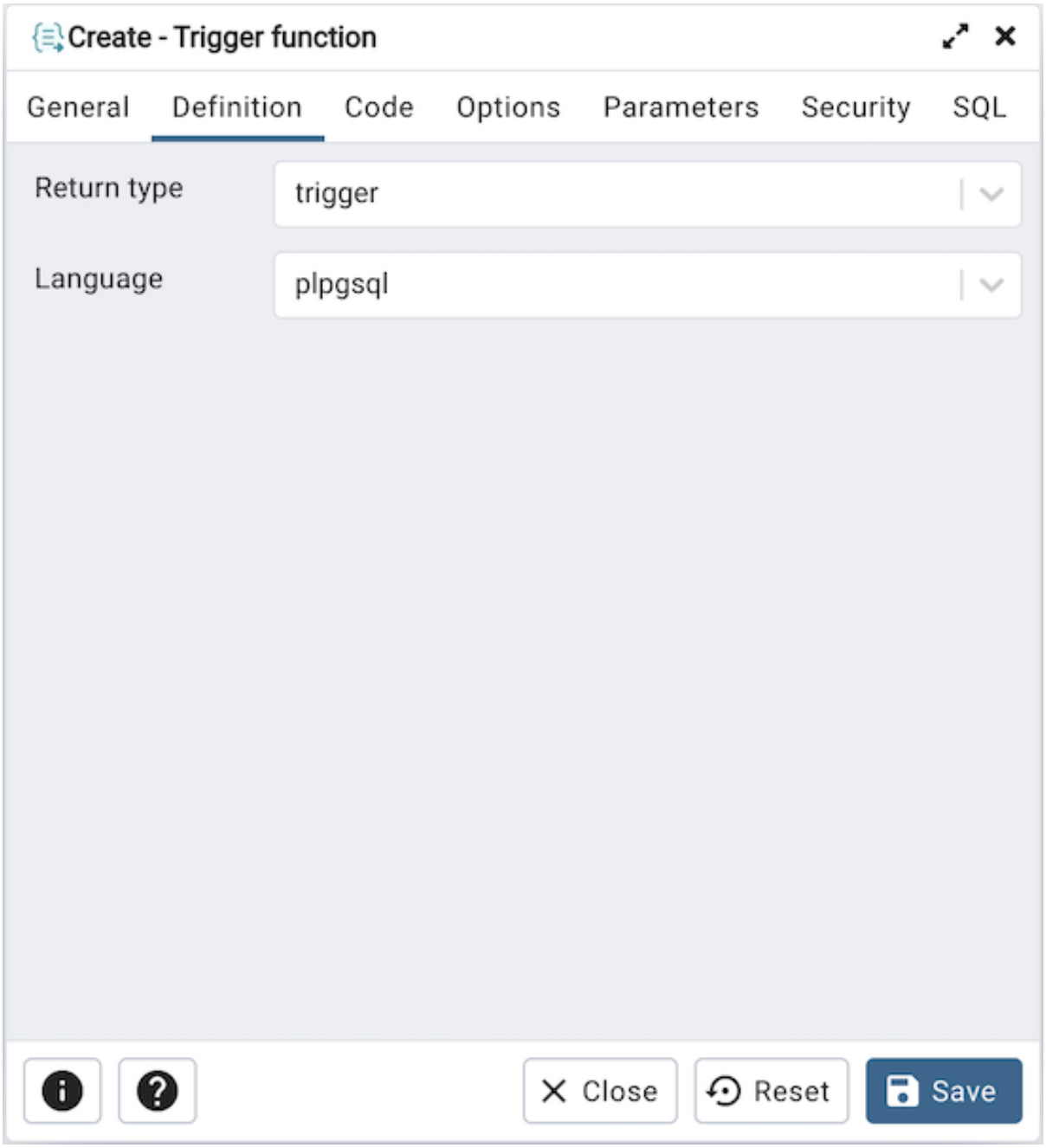


[*General*] タブのフィールドを使用して、トリガー機能を識別します。

- *Name* フィールドを使用して、トリガー関数の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。トリガー関数はアルファベット順に呼び出されることに注意してください。

- **所有者** の横にあるドロップダウンリストボックスを使用して、トリガー関数を所有するロールを選択します。
- **スキーマ** フィールドのドロップダウンリストボックスから、トリガー関数が存在するスキーマの名前を選択します。
- **コメント** フィールドにトリガー機能に関するメモを保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、トリガー機能を定義します。

- *Returntype* の横にあるドロップダウンリストボックスを使用して、トリガー関数に関連付けられている擬似型を指定します。
  - DML トリガーを作成する場合は、 **トリガー** を選択します。

  - DDL トリガーを作成する場合は *event_trigger* を選択します。

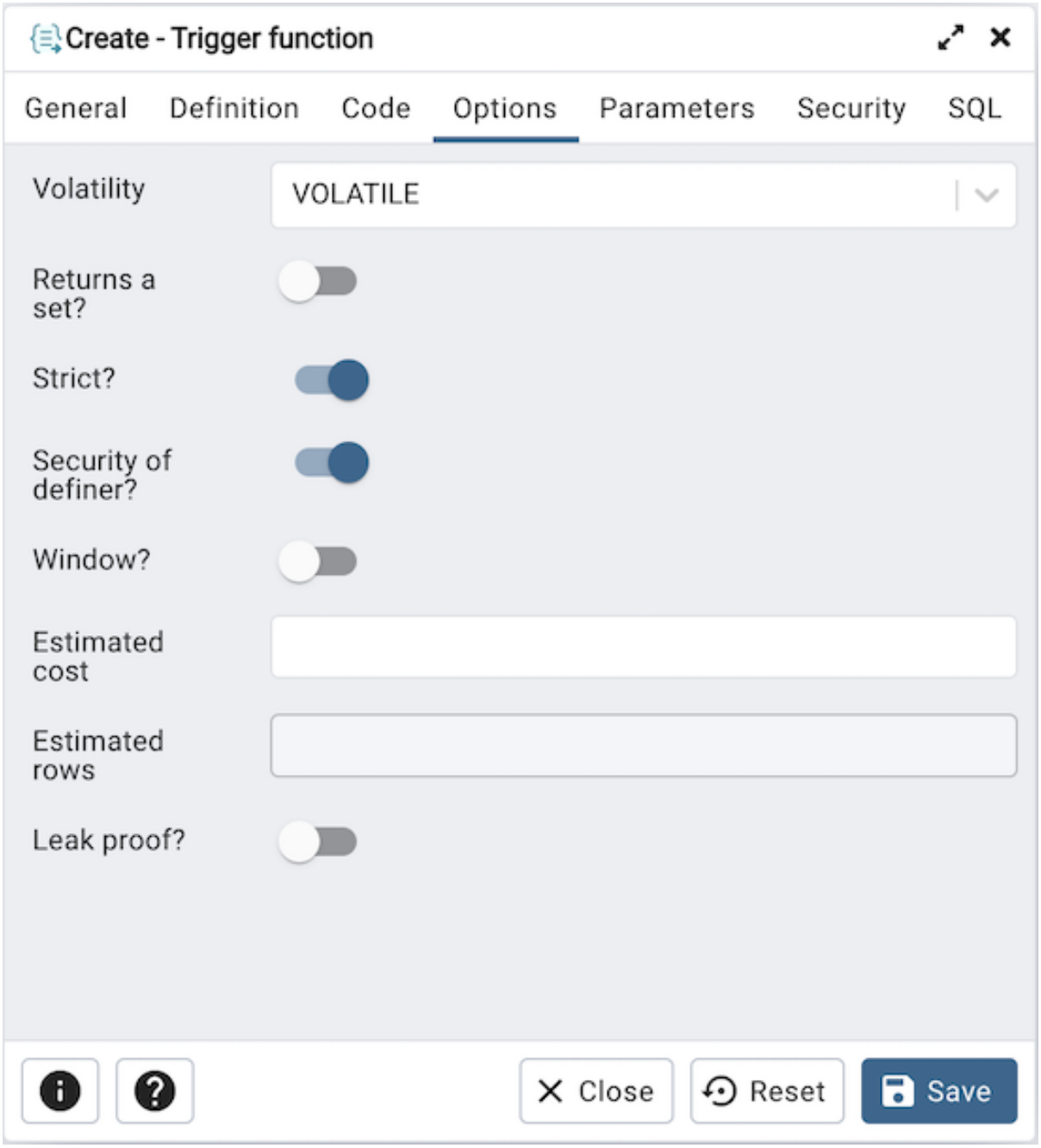
- **言語** の横にあるドロップダウンリストボックスを使用して、実装言語を選択します。デフォルトは *plpgsql* です。

[**コード**] タブをクリックして続行します。



- *Code* フィールドを使用して、トリガー関数が呼び出されたときに実行されるコードを記述します。

[**オプション**] タブをクリックして続行します。

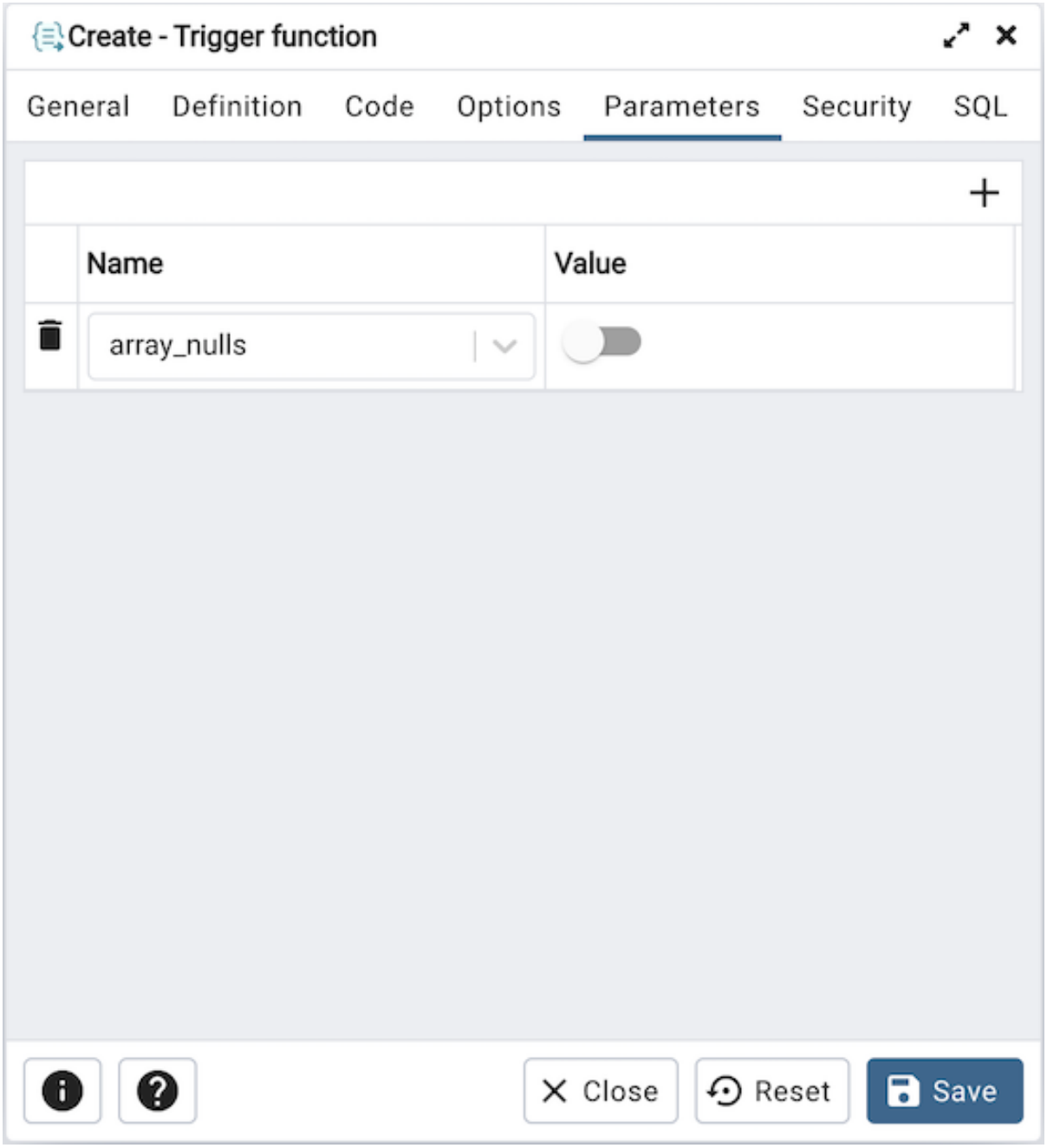


[**オプション**] タブのフィールドを使用して、トリガー機能のアクションを説明または変更します。

- **ボラティリティ** の横にあるドロップダウンリストボックスを使用して、次のいずれかを選択します。
  - *VOLATILE* は、トリガー機能の値が 1 回のテーブルスキャン内でも変更できることを示します。
  - *STABLE* は、トリガー関数がデータベースを変更できないこと、および 1 回のテーブルスキャン内で、同じ引数値に対して一貫して同じ結果を返すことを示します。
  - *IMMUTABLE* は、トリガー関数がデータベースを変更できず、同じ引数値が指定された場合に常に同じ結果を返すことを示します。
- *ReturnsaSet?* スイッチを動かして、トリガー関数が複数の行を含むセットを返すかどうかを示します。デフォルトは **いいえ** です。

- *Strict?* スイッチを移動して、トリガー関数が引数のいずれかが NULL の場合に常に NULL を返すかどうかを示します。 *Yes* の場合、NULL 引数があると関数は実行されません。代わりに NULL 結果が自動的に想定されます。デフォルトは **いいえ** です。
- *Securityofdefiner?* スイッチを移動して、トリガー関数を作成したユーザーの権限で実行することを指定します。デフォルトは **いいえ** です。
- *Window?* スイッチを動かして、トリガー関数がプレーン関数ではなくウィンドウ関数であることを示します。デフォルトは **いいえ** です。これは現在、C で記述されたトリガー関数でのみ役立ちます。
- *Estimatedcost* フィールドを使用して、トリガー関数の推定実行コストを表す正の数を cpu_operator_cost 単位で指定します。関数がセットを返す場合、これは返される行ごとのコストです。
- *Estimatedrows* フィールドを使用して、クエリプランナーがトリガー関数が返すと予想される行の推定数を示す正の数を指定します。これは、関数がセットを返すように宣言されている場合にのみ許可されます。デフォルトの想定は 1000 行です。
- **リーク防止***?* スイッチを動かして、トリガー機能に副作用があるかどうかを示します。デフォルトは **いいえ** です。このオプションはスーパーユーザーのみが設定できます。

[**パラメーター**] タブをクリックして続行します。

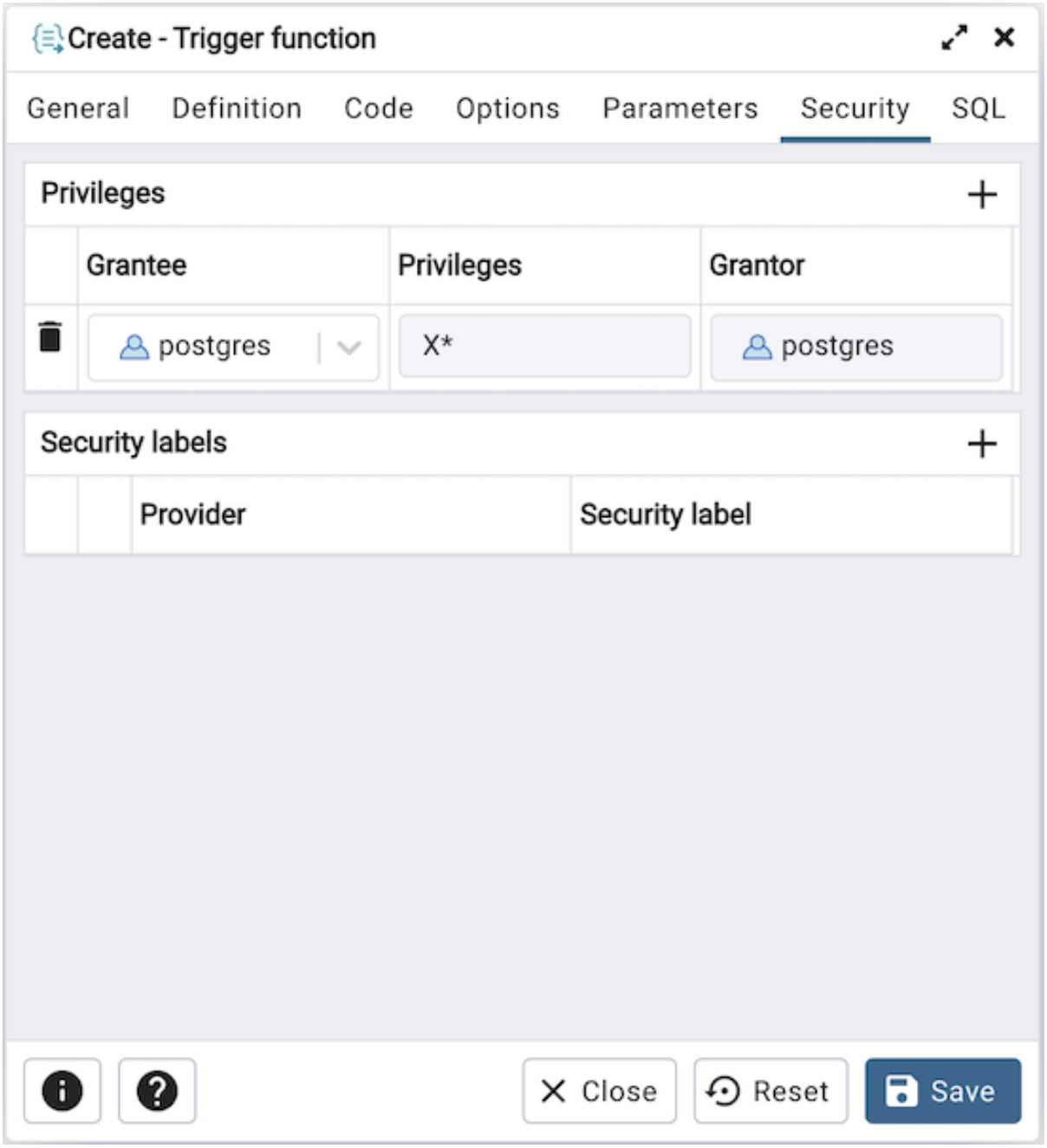


[**パラメーター**] タブのフィールドを使用して、トリガー機能が呼び出されたときに適用される設定を指定します。 **追加** アイコン（+）をクリックして、 **名前 / 値** のペアを下のテーブルに追加します。

- **名前** フィールドのドロップダウンリストボックスを使用して、パラメーターを選択します。
- *Value* フィールドを使用して、選択したパラメーターに関連付けられる値を指定します。このフィールドは状況依存です。

追加 アイコン（+）をクリックして、追加のパラメーターを設定します。パラメーターを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、トリガー機能の使用特権をロールに割り当てます。 **追加** アイコン（+）をクリックして、ロールをテーブルに追加します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、トリガー機能に適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

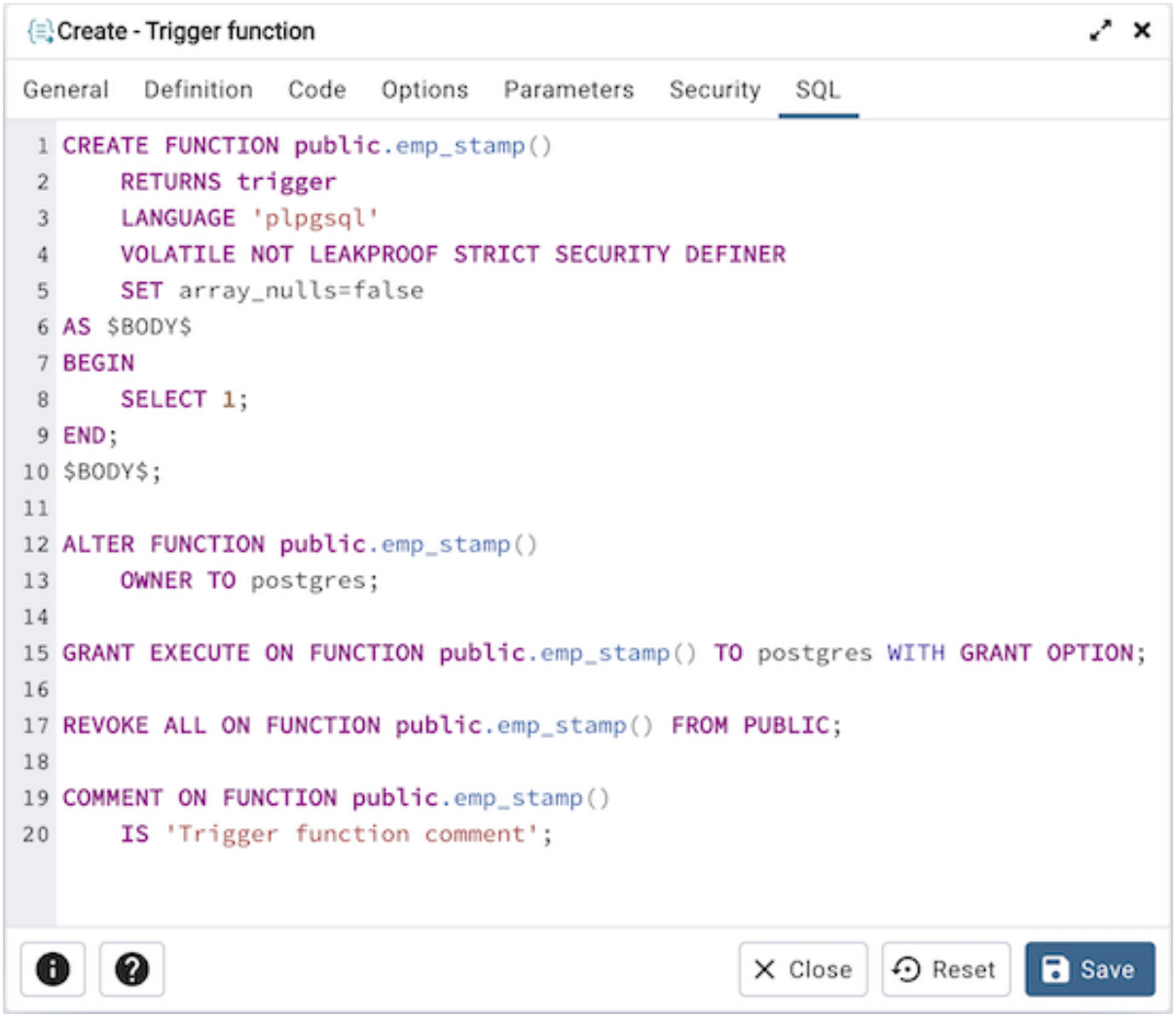
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

トリガー機能 ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。他のタブに再度アクセスして、SQL コマンドを変更します。

5.24.1 例

以下は、 **トリガー機能** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



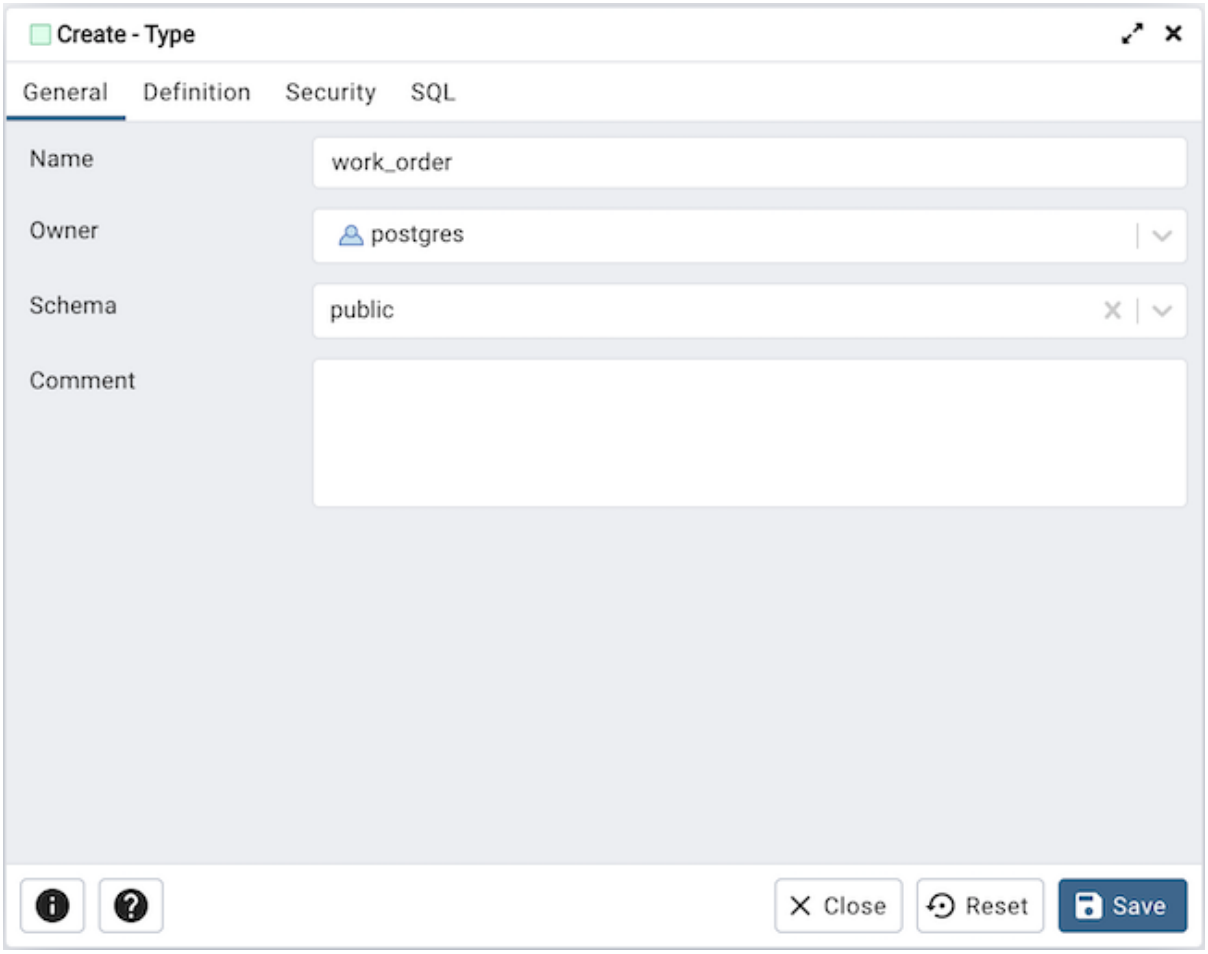
ここに示す例は、 *emp_stamp* という名前のトリガー関数の作成を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.25 Type Dialog

Type ダイアログを使用して、カスタムデータ型を登録します。

タイプ ダイアログは、次のダイアログタブを介してデータ型の開発を編成します。 **一般** 、 **定義** 、および **セキュリティ** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、カスタムデータ型を識別します。

- *Name* フィールドを使用して、タイプを説明する名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。型名は、同じスキーマ内の既存の型、ドメイン、またはテーブルの名前とは異なる必要があります。
- **所有者** の横にあるドロップダウンリストボックスを使用して、タイプを所有するロールを選択します。
- **スキーマ** フィールドのドロップダウンリストボックスから、型が存在するスキーマの名前を選択します。
- タイプに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。

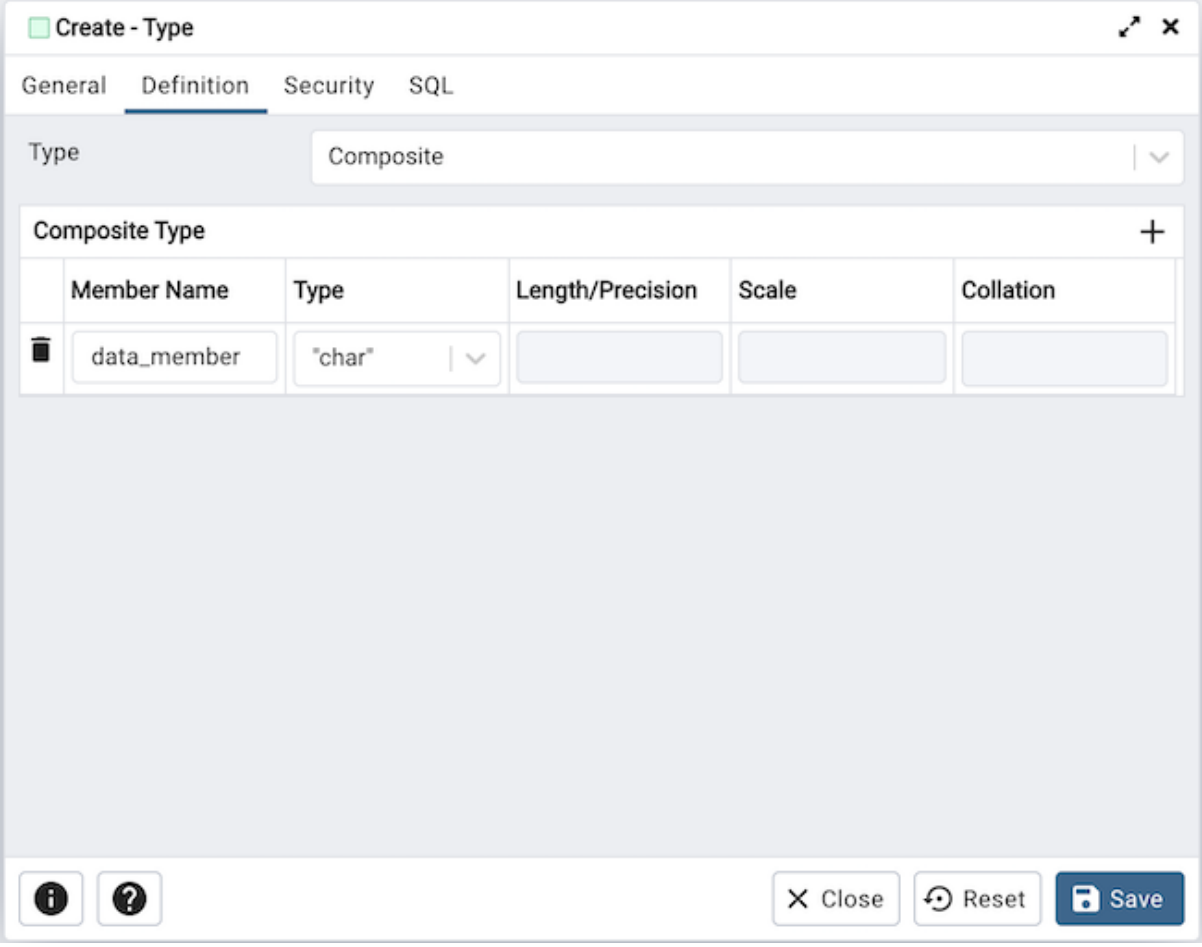
[**定義**] タブの **タイプ** の横にあるドロップダウンリストボックスからデータ型を選択します。下のパネルが変化して、選択したデータ型に適したオプションが表示されます。パネルのフィールドを使用して、データ型を定義します。

5 つのデータ型があります。

- ＊コンポジットタイプ＊
- **列挙型**
- **レンジタイプ**

- **外部タイプ**（または **ベースタイプ** ）
- **シェルタイプ**

[**タイプ**] フィールドで **コンポジット** を選択すると、 **定義** タブに **コンポジットタイプ** パネルが表示されます。

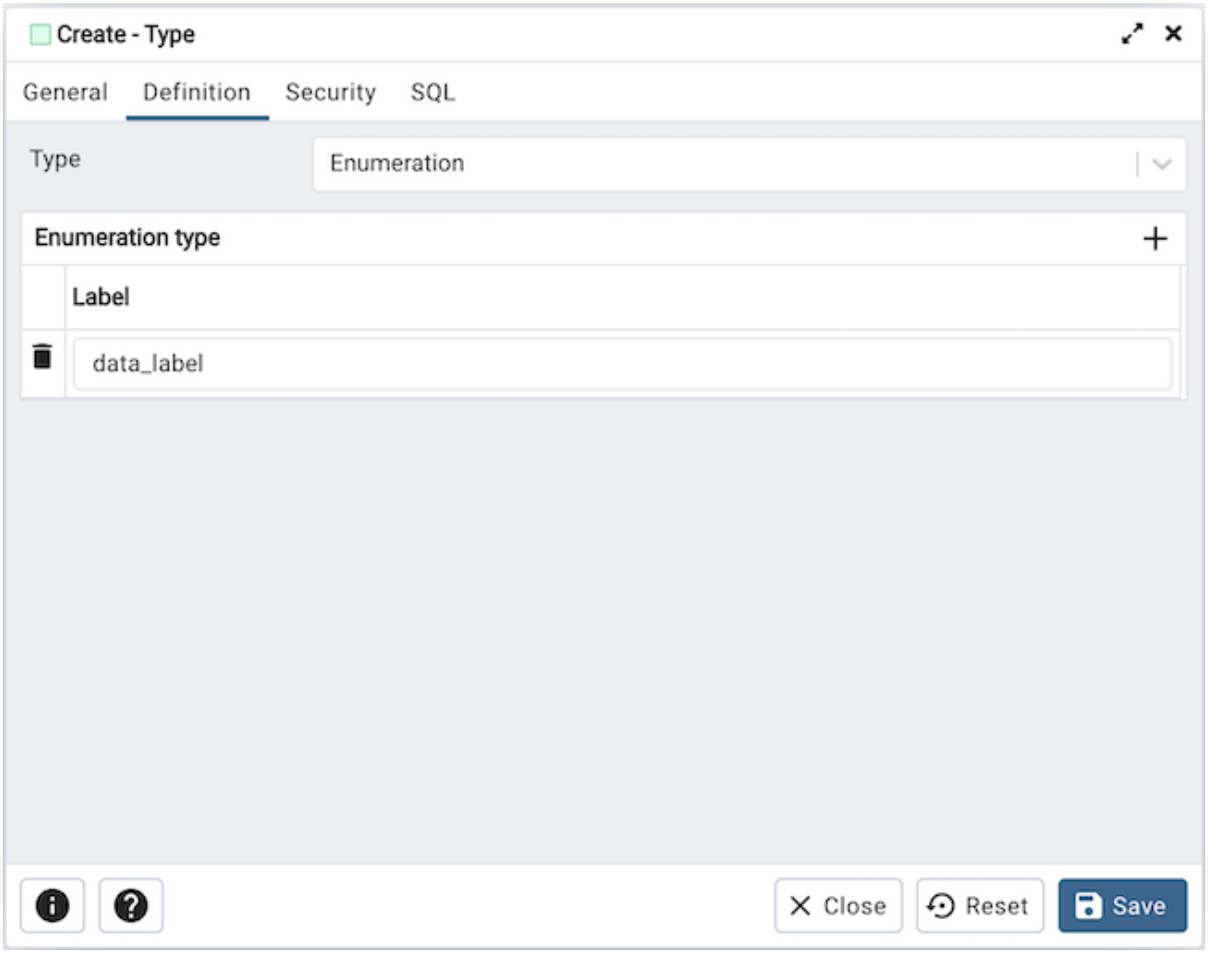


追加 アイコン（+）をクリックして、タイプの属性を指定します。[**一般**] パネルのフィールドは状況依存であり、無効になっている場合があります。

- **メンバー名** フィールドを使用して、属性名を追加します。
- *Type* フィールドのドロップダウンリストボックスを使用して、データ型を選択します。
- **長さ/精度** フィールドを使用して、非数値型の最大長、または数値型の有効桁数の合計を指定します。
- *Scale* フィールドを使用して、小数点の右側の桁数を指定します。
- *Collation* フィールドのドロップダウンリストボックスを使用して、照合を選択します（該当する場合）。

追加 アイコン（+）をクリックして、追加のメンバーを定義します。行を破棄するには、行の左側にあるごみ箱アイコンをクリックします。

[**タイプ**] フィールドで **列挙** を選択すると、 **定義** タブに **列挙タイプ** パネルが表示されます。

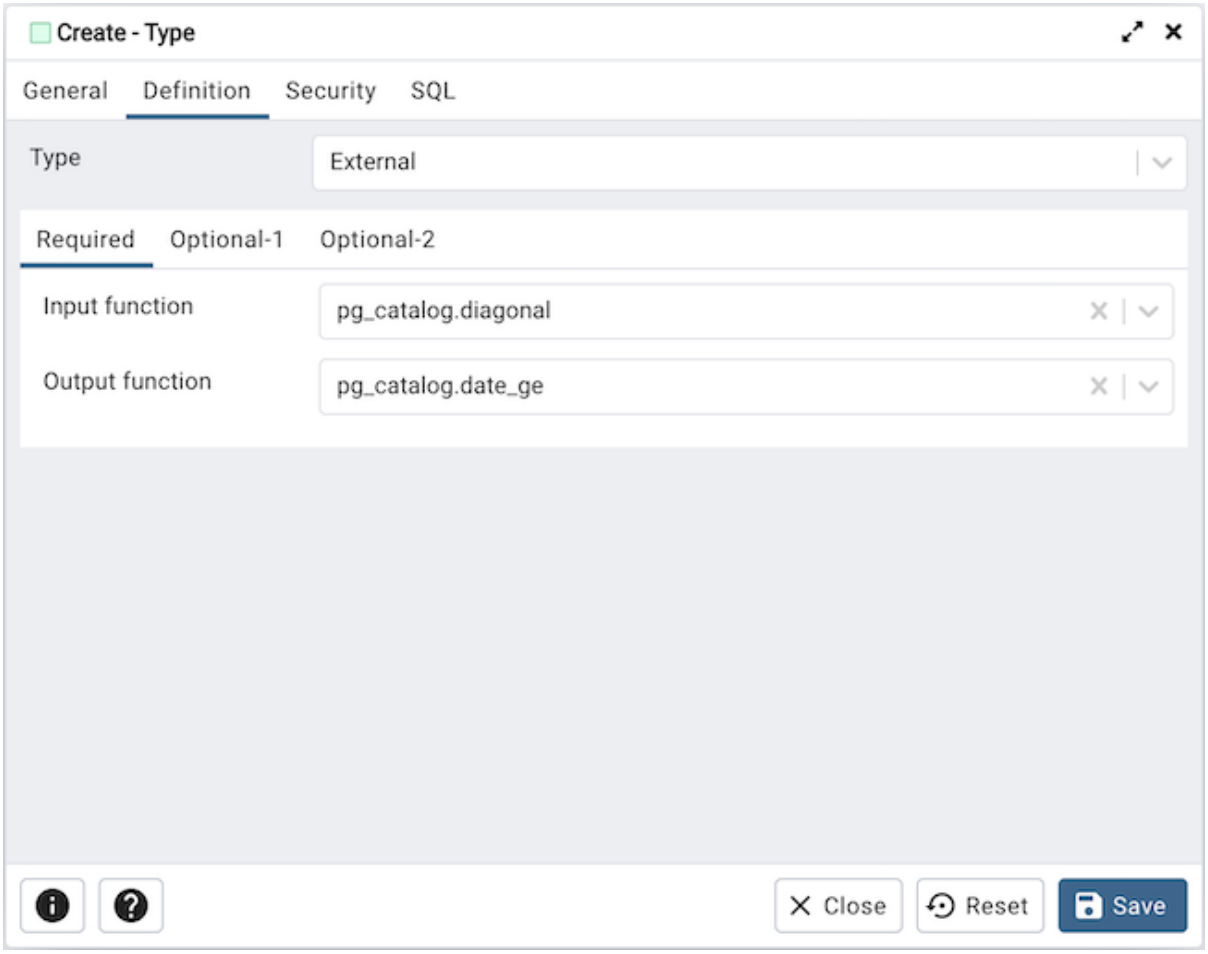


追加 アイコン（+）をクリックして、タイプのラベルを指定します。

- *Label* フィールドを使用して、64 バイト未満のラベルを追加します。

各選択の後に **追加** アイコン（+）をクリックして、追加のラベルを作成します。ラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックします。

External を選択すると、 *Definition* タブに *ExternalType* パネルが表示されます。



[**必須**] タブで：

- **入力機能** フィールドの横にあるドロップダウンリストボックスを使用して、input_function を追加します。input_function は、型の外部テキスト表現を、型に定義された演算子と関数で使用される内部表現に変換します。
- *Outputfunction* フィールドの横にあるドロップダウンリストボックスを使用して、output_function を追加します。output_function は、型に定義された演算子と関数で使用される型の内部表現を型の外部テキスト表現に変換します。

[*Optional-1*] タブで：

- オプションの *ReceiveFunction* フィールドの横にあるドロップダウンリストボックスを使用して、receive_function を選択します。オプションの receive_function は、型の外部バイナリ表現を内部表現に変換します。この関数が指定されていない場合、型はバイナリ入力に参加できません。
- オプションの *Sendfunction* フィールドの横にあるドロップダウンリストボックスを使用して、send_function を選択します。オプションの send_function は、内部表現から外部バイナリ表現に変換します。この関数が指定されていない場合、型はバイナリ出力に参加できません。
- オプションの **関数内の** *Typmod* フィールドタブの横にあるドロップダウンリストボックスを使用して、type_modifier_input_function を選択します。
- オプションの *Typmodoutfunction* フィールドタブの横にあるドロップダウンリストボックスを使用して、type_modifier_output_function を選択します。type_modifier_output_function を省略することがで

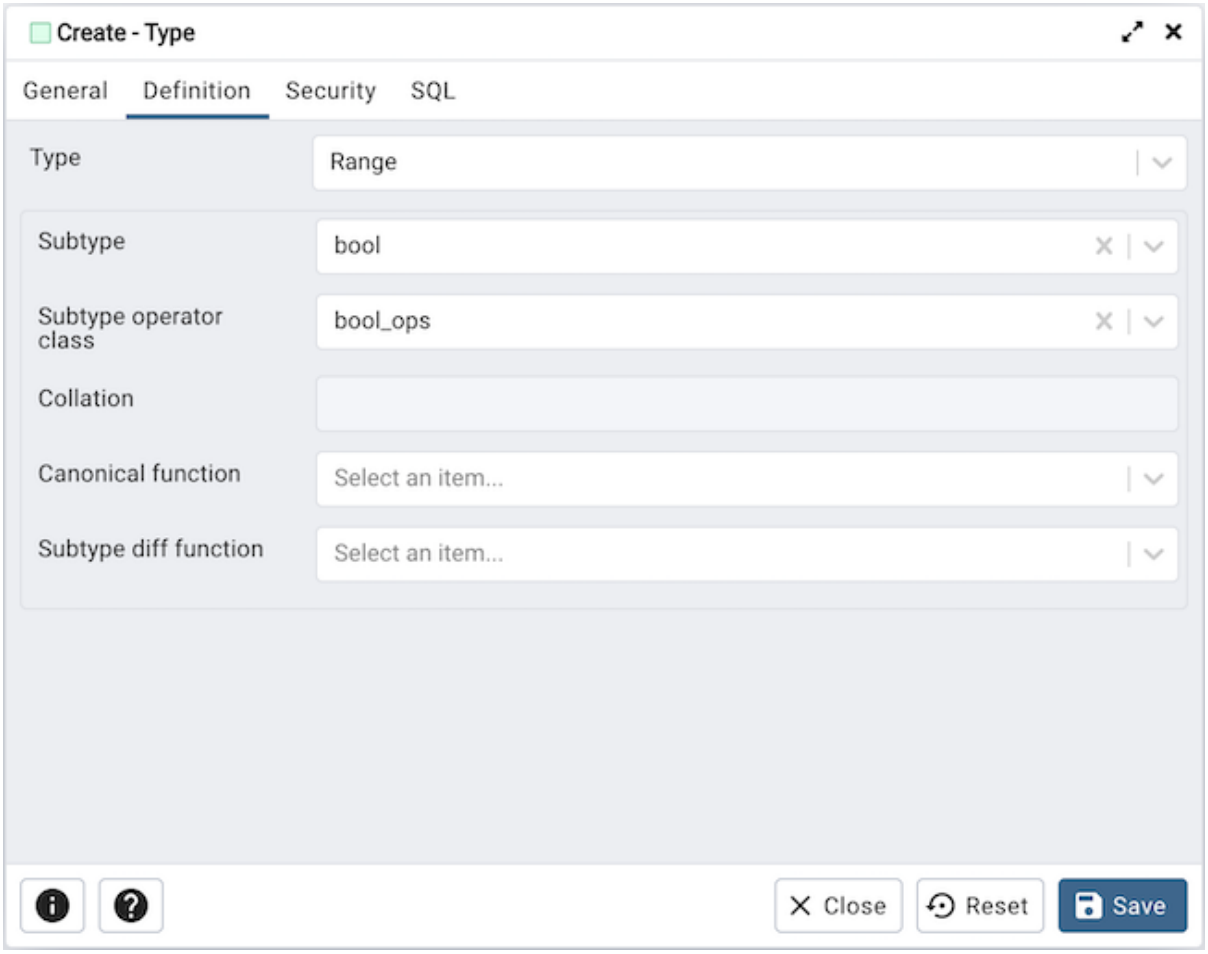
きます。その場合、デフォルトの表示形式は、括弧で囲まれた格納された typmod 整数値です。

- オプションの **内部長** を使用して、内部表現の値を指定します。
- *Variable?* スイッチを移動して、内部表現が可変長（VARIABLE）であることを指定します。デフォルトは固定長の正の整数です。
- データ型の列のデフォルトが null 値以外の場合、オプションの *Default* フィールドにデフォルト値を指定します。デフォルトは DEFAULT キーワードで指定します。(デフォルトは、特定の列にアタッチされた明示的な DEFAULT 句によってオーバーライドできます。)
- オプションの **分析関数** フィールドの横にあるドロップダウンリストボックスを使用して、データ型の列の型固有の統計収集を実行する関数を選択します。
- オプションの **カテゴリタイプ** フィールドの横にあるドロップダウンリストボックスを使用して、あいまいな状況で適用される暗黙的なキャストを制御します。
- **優先***?* スイッチを **はい** に移動して、選択したカテゴリタイプが優先されることを指定します。デフォルトは **いいえ** です。

[*Optional-2*] タブで：

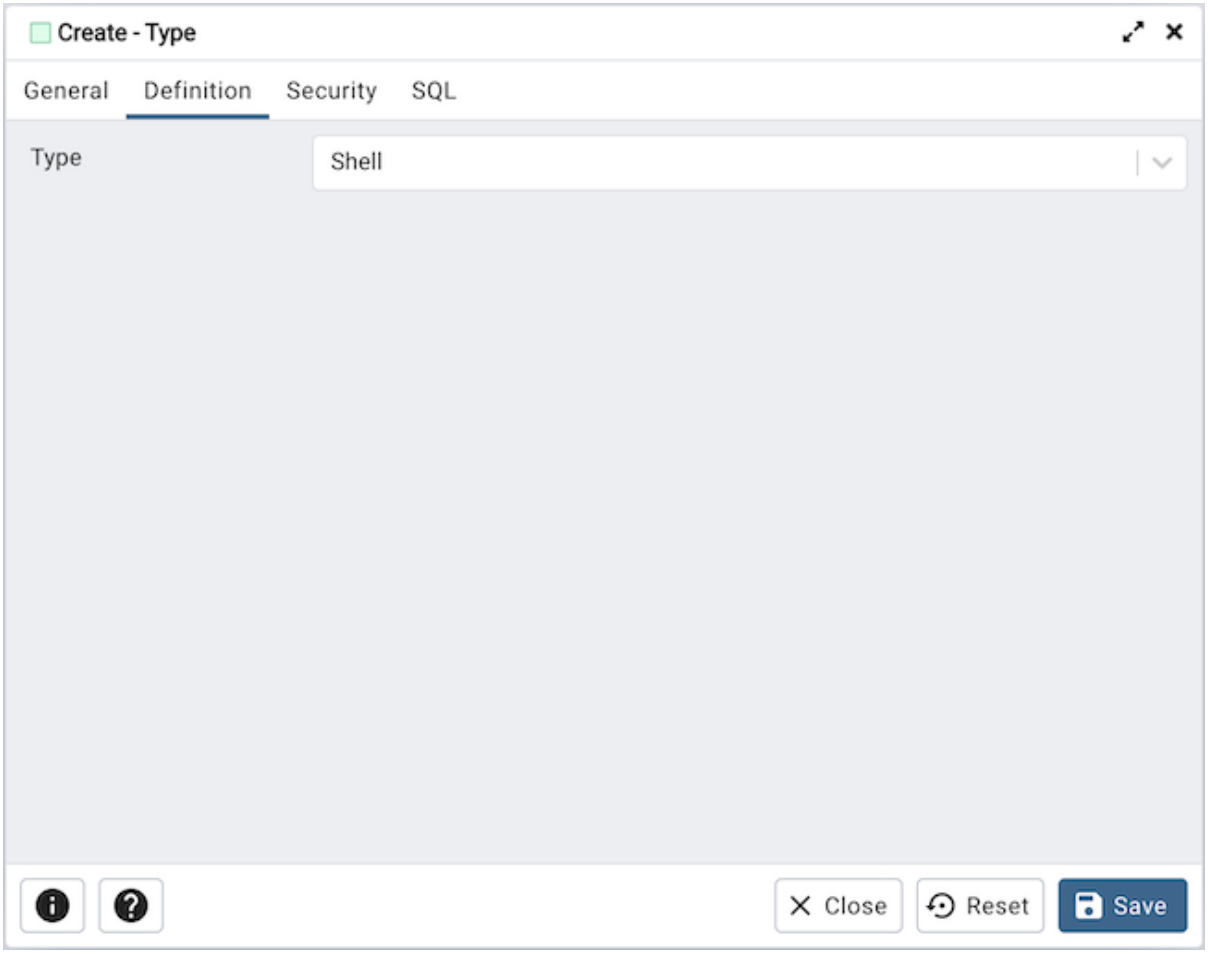
- オプションの **要素タイプ** フィールドの横にあるドロップダウンリストボックスを使用して、データ型を指定します。
- オプションの *Delimiter* フィールドを使用して、このデータ型の配列の外部表現の値の間で使用される区切り文字を示します。デフォルトの区切り文字はコンマ（,）です。区切り文字は、配列型自分自身ではなく、配列要素型に関連付けられていることに注意してください。
- **アライメントタイプ** の横にあるドロップダウンリストボックスを使用して、データ型に必要なストレージアライメントを指定します。許可される値（char、int2、int4、および double）は、1、2、4、または 8 バイト境界のアライメントに対応します。
- オプションの **ストレージタイプ** の横にあるドロップダウンリストボックスを使用して、データを保存する戦略を選択します。
- *Passedbyvalue?* スイッチを *Yes* に移動して、既存のデータ型の値をオーバーライドします。デフォルトは **いいえ** です。
- *Collatable?* スイッチを *Yes* に移動して、列定義を指定します。タイプの式は、COLLATE 句を使用して照合情報を伝送できます。デフォルトは **いいえ** です。

Type フィールドで *Range* を選択すると、 *Definition* タブに *Range* パネルが表示されます。 **範囲** パネルのフィールドは状況依存であり、無効になっている場合があります。



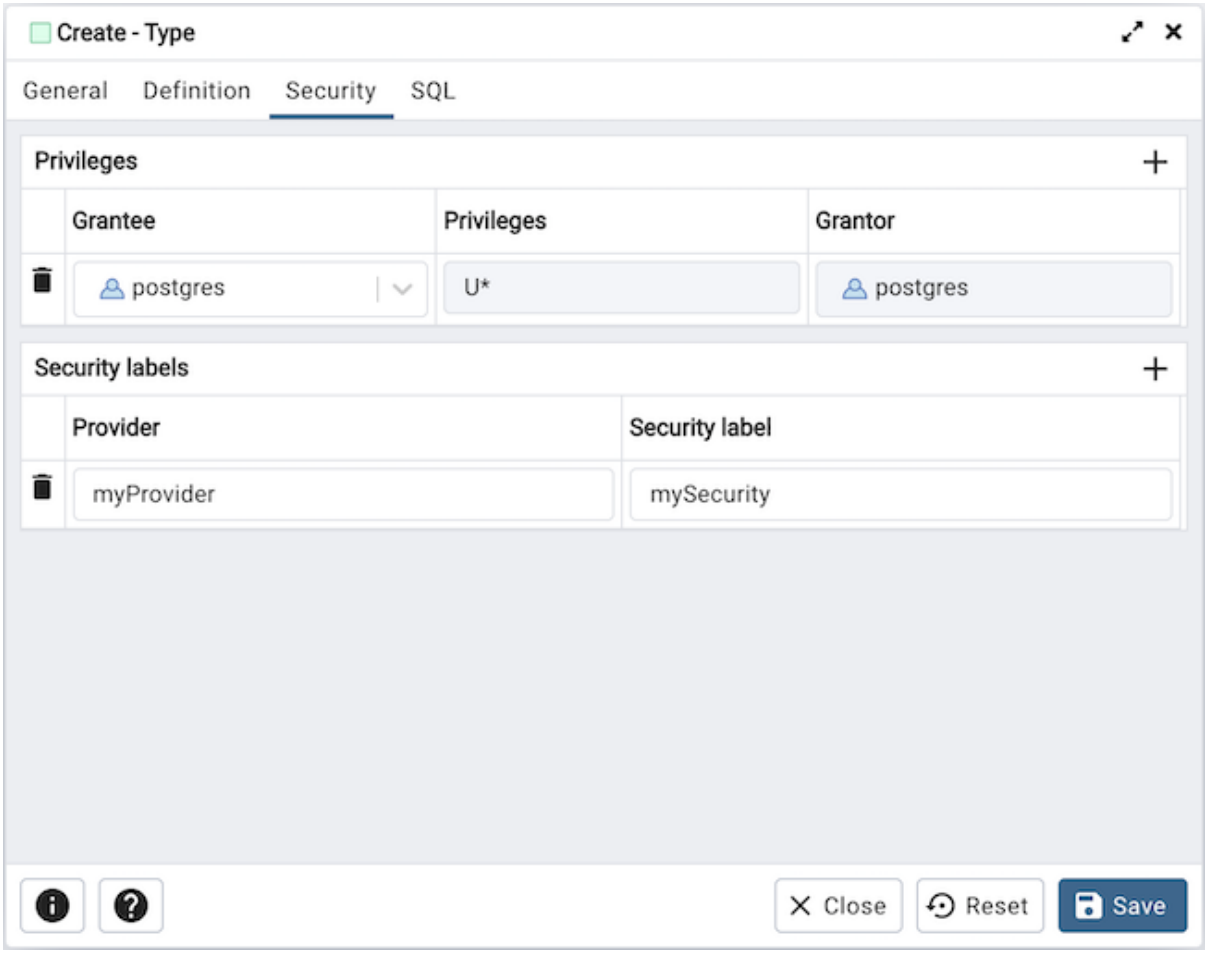
- **サブタイプ** の横にあるドロップダウンリストボックスを使用して、関連する b ツリー演算子クラスを選択します（範囲タイプの値の順序を決定します）。
- **サブタイプのオペレータークラス** の横にあるドロップダウンリストボックスを使用して、デフォルト以外のオペレータークラスを使用します。
- サブタイプが照合可能な場合、 *Collation* の横にあるドロップダウンリストボックスを使用して、範囲の順序付けでデフォルト以外の照合を使用します。
- **標準関数** の横にあるドロップダウンリストボックスを使用して、範囲値を標準形式に変換します。
- *Sub-typediffffunction* の横にあるドロップダウンリストボックスを使用して、ユーザー定義の subtype_diff 関数を選択します。

Type フィールドで *Shell* を選択すると、 *Definition* タブに *Shell* パネルが表示されます。



シェル型は型のプレースホルダーであり、パラメーターはありません。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

Privileges パネルを使用して、タイプに特権を割り当てます。 **追加** アイコン（+）をクリックして、権限を付与します。

- *Grantee* フィールドのドロップダウンリストボックスから、タイプに対する特権を付与するロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

セキュリティラベル パネルを使用して、タイプに適用するセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを

格納するメカニズムを提供するだけです。

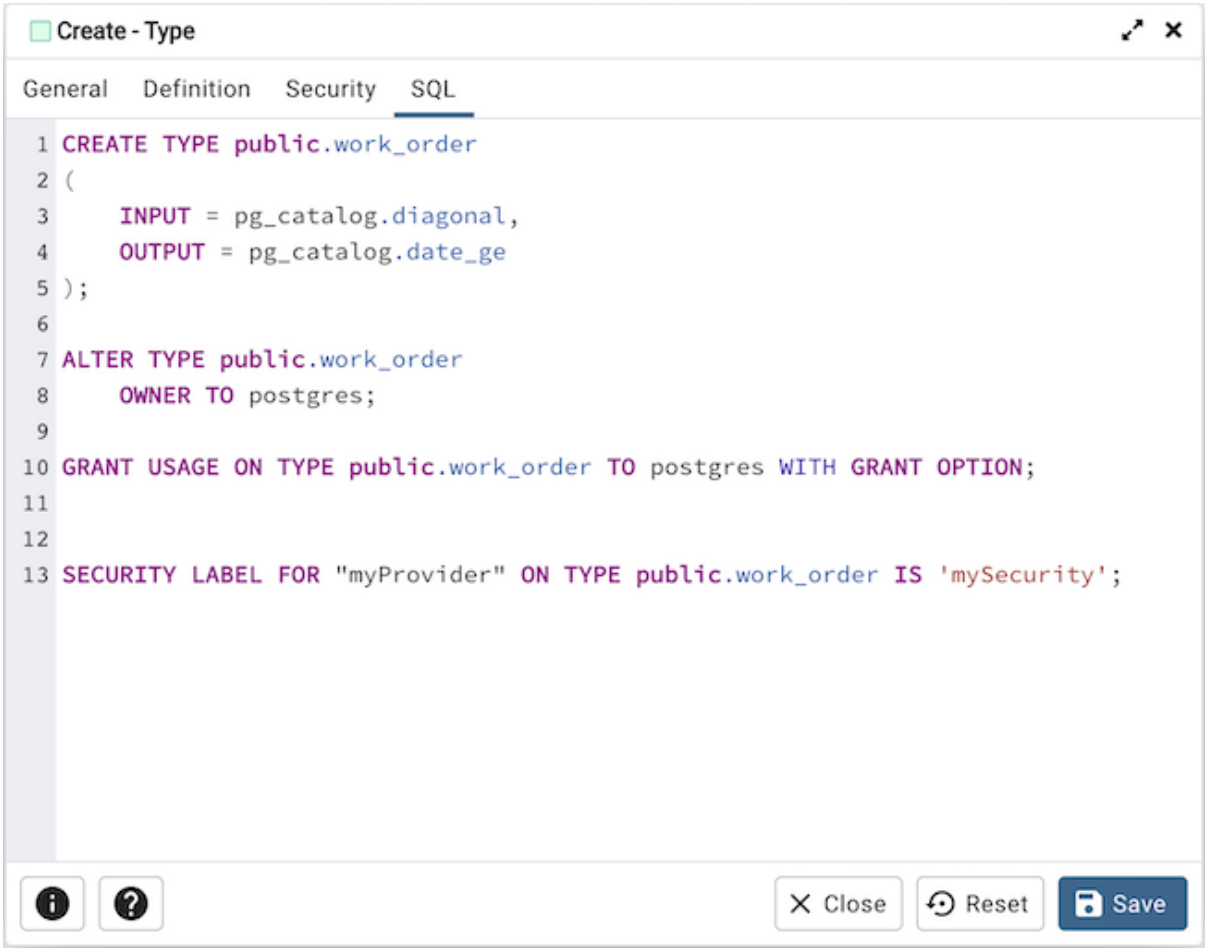
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Type ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.25.1 例

以下は、[**タイプ**] ダイアログでのユーザー選択によって生成された sql コマンドの例です。



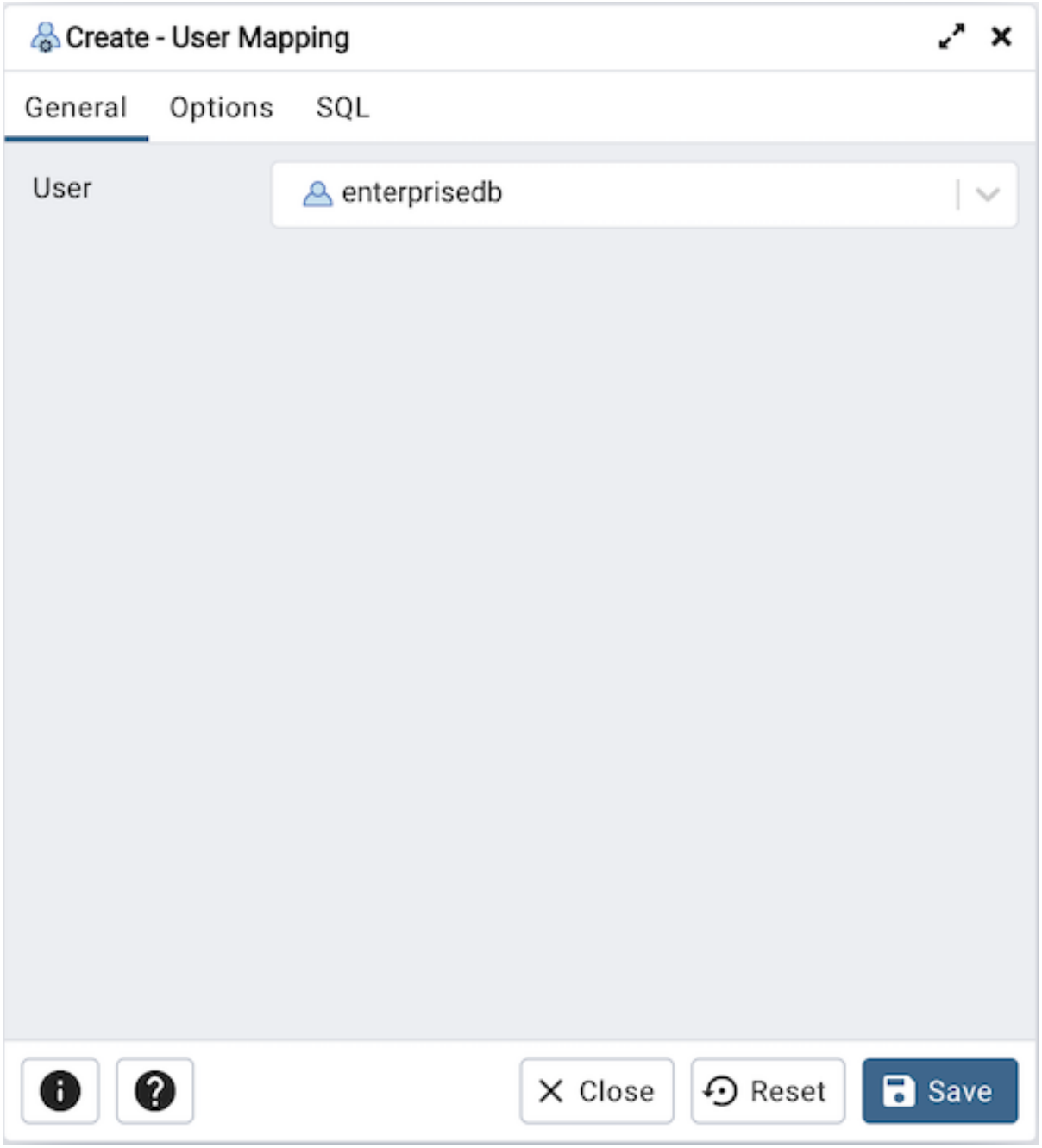
この例は、*work_order* という名前のデータ型の作成を示しています。データ型は、新規、オープン、クローズの 3 つのラベルを持つ列挙型です。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.26 User Mapping Dialog

ユーザーマッピング ダイアログを使用して、ユーザーから外部サーバーへの新しいマッピングを定義します。

[**ユーザーマッピング**] ダイアログでは、 **一般** と **オプション** のダイアログタブを使用してユーザーマッピングの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

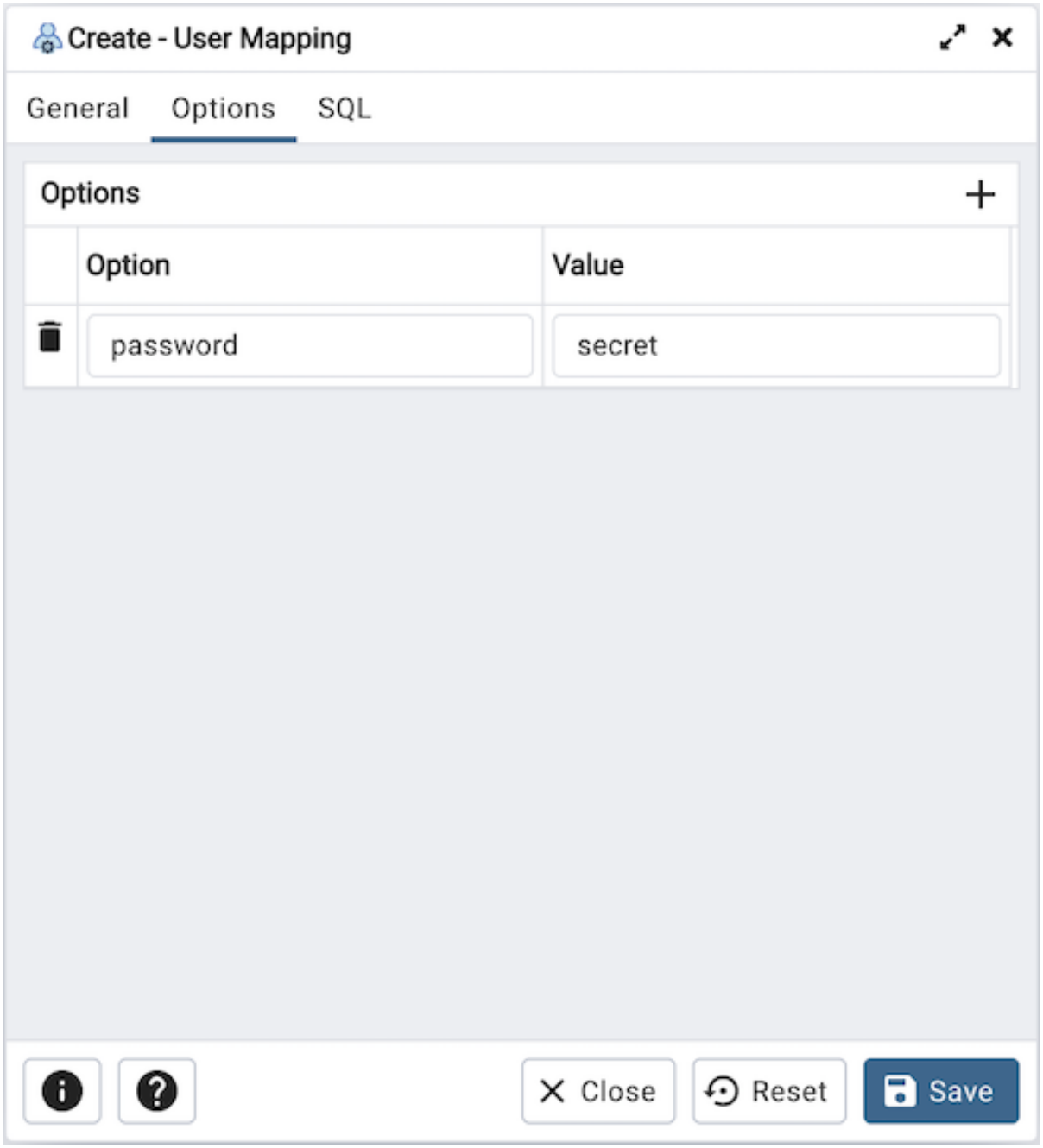


[**全般**] タブの **ユーザー** フィールドのドロップダウンリストボックスを使用して、接続ロールを識別します。

- 現在のロールの名前を使用するには、 *CURRENT_USER* を選択します。
- 他のユーザー固有のマッピングが適用できない場合は、 *PUBLIC* を選択します。

- 事前定義されたロール名を選択して、既存のユーザーの名前を指定します。

[**オプション**] タブをクリックして続行します。



[**オプション**] タブのフィールドを使用して、接続オプションを指定します。受け入れられるオプションの名前と値は、ユーザーマッピングで指定されたサーバーに関連付けられた外部データラッパーに固有です。[**追加**] ボタンをクリックして、オプション/値のペアを追加します。

- *Option* フィールドにオプション名を指定します。
- *Value* フィールドに対応する値を入力します。

Add をクリックして、追加のオプション/値のペアをそれぞれ指定します。オプションを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

ユーザーマッピング ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.26.1 例

以下は、 **ユーザーマッピング** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、 *hdfs_server* のユーザーマッピングを示しています。ユーザーは *CURRENT_USER* で、パスワードは *secret* です。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。

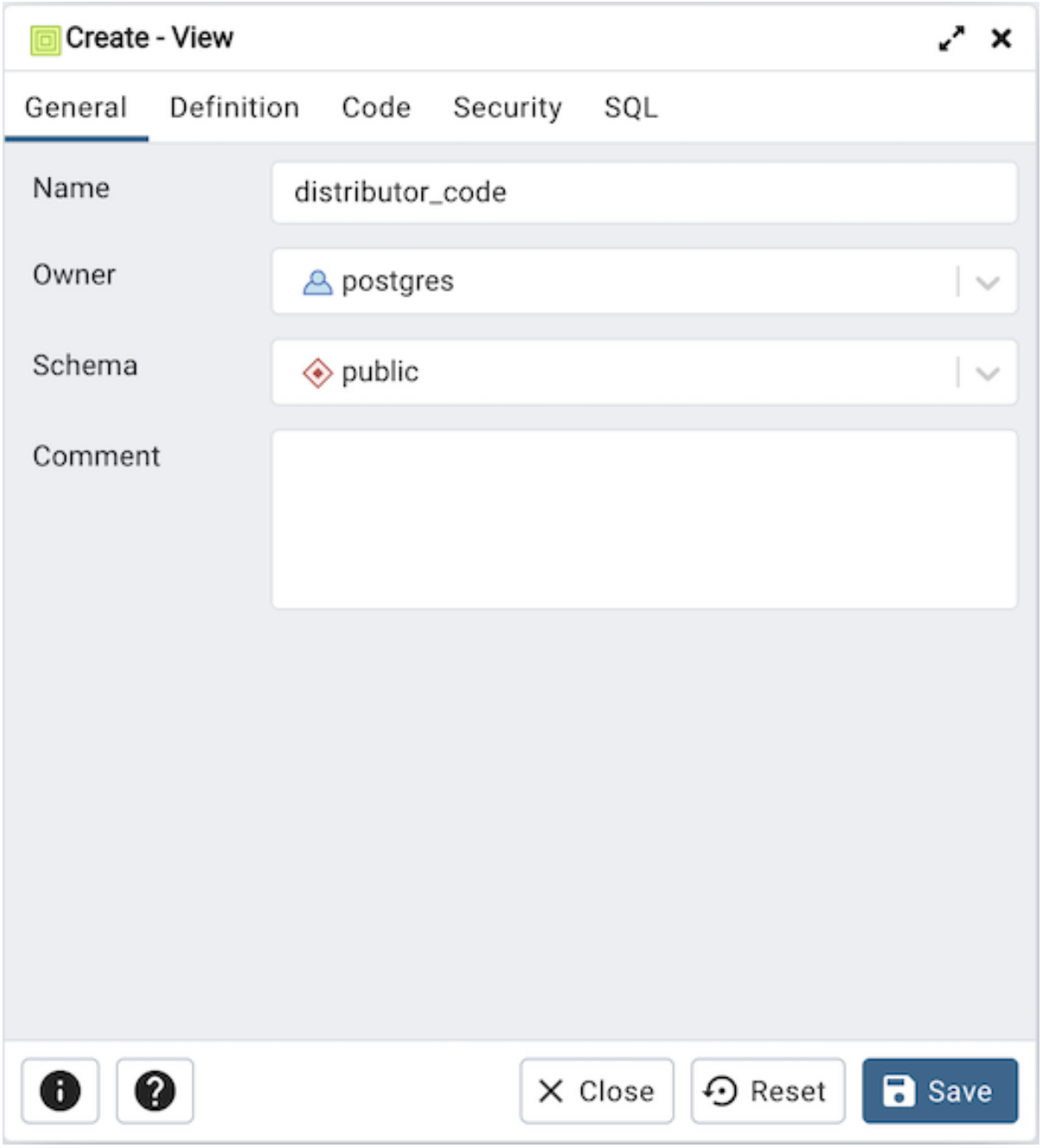
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

5.27 View Dialog

ビューを定義するには、 *View* ダイアログを使用します。ビューは物理的に具体化されません。クエリは、ビューがクエリで参照されるたびに実行されます。

ビュー ダイアログは、次のダイアログタブを介してビューの開発を編成します。 *SQL* タブには、ダイアログの選択によって生成された SQL コードが表示されます。

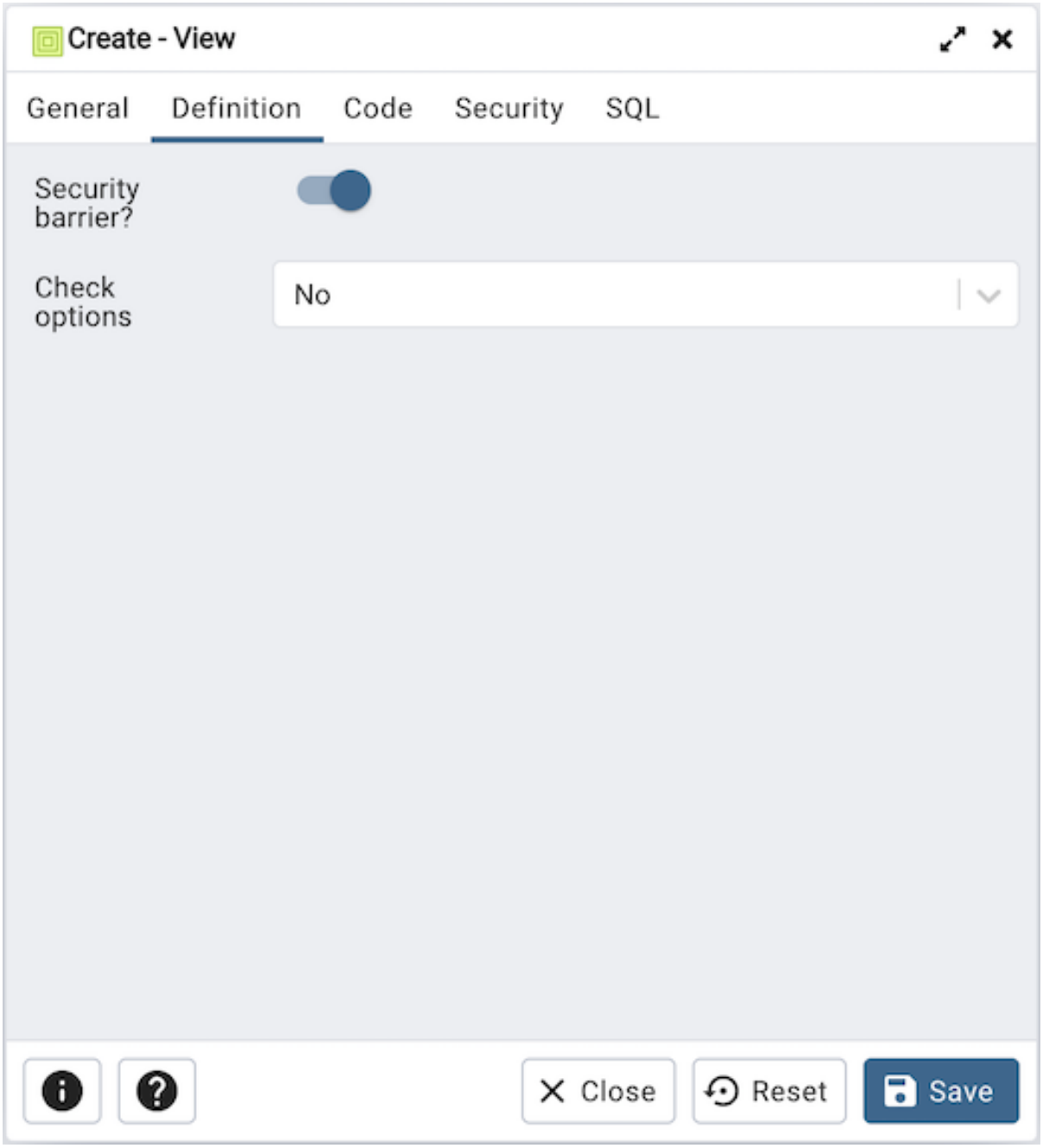
[**全般**] タブをクリックして開始します。



[**全般**] タブのフィールドを使用して、ビューを識別します。

- *Name* フィールドを使用して、ビューにわかりやすい名前を追加します。ビューの名前は、同じスキーマ内の他のビュー、テーブル、シーケンス、インデックス、または外部テーブルの名前とは異なる必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。
- **所有者** の横にあるドロップダウンリストボックスを使用して、ビューを所有するロールを選択します。
- 必要に応じて、ビューが存在するスキーマの名前を **スキーマ** フィールドのドロップダウンリストボックスから選択します。
- ビューに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、ビューのプロパティを定義します。

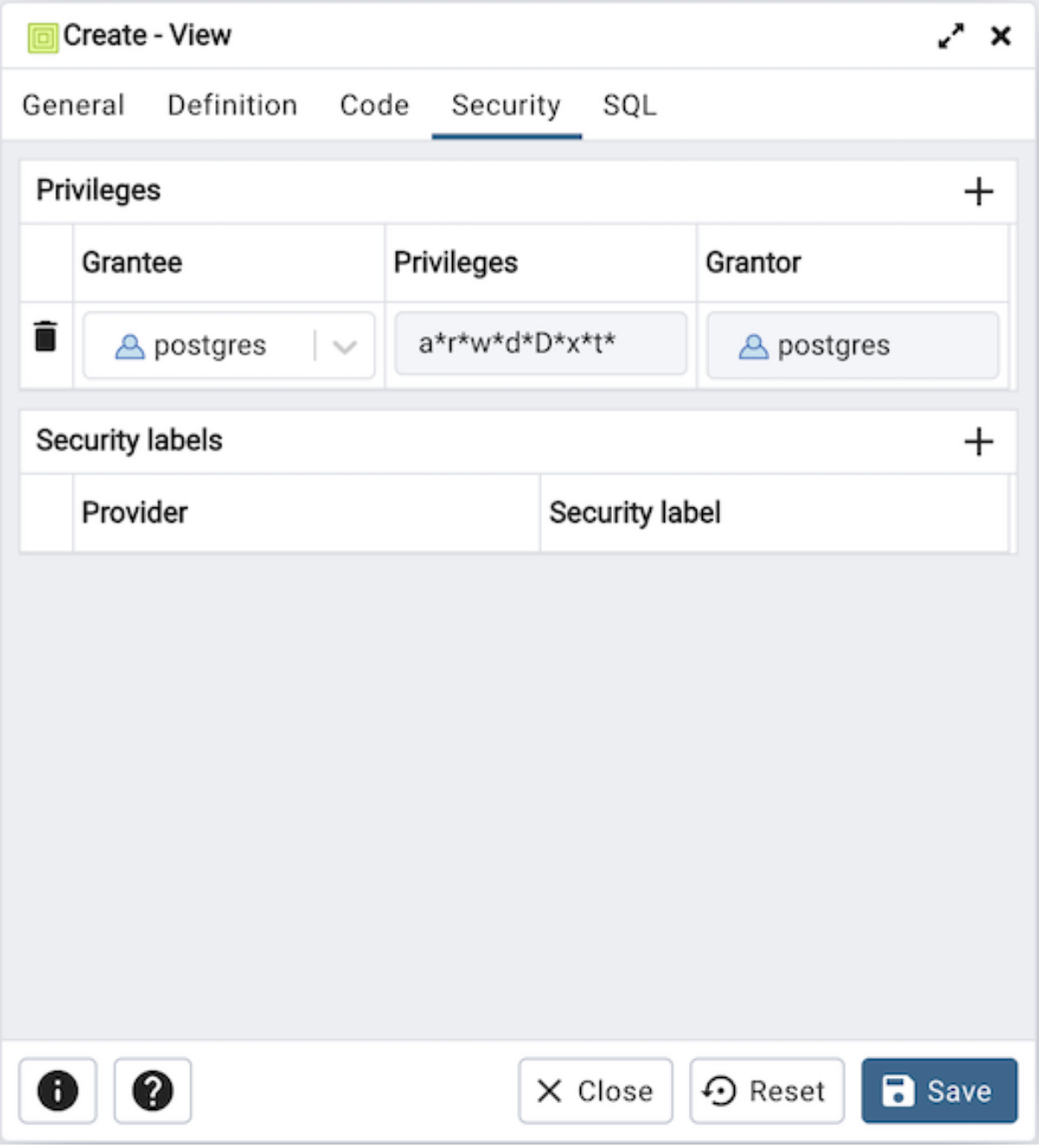
- ビューがセキュリティバリアとして機能することを示すには、**セキュリティバリア** スイッチを **はい** に設定します。セキュリティバリアルールの定義と使用の詳細については、PostgreSQL ドキュメントのセクション 38.5 を参照してください。
- **チェックオプション** の横にあるドロップダウンリストボックスを使用して、 **いいえ** 、 **ローカル** または **カスケード** から選択します。
  - *Local* オプションは、新しい行がビューで定義された条件に対してのみチェックされることを指定します。基になるベースビューで定義された条件はチェックされません（CHECKOPTION を指定しない限り）。
  - **カスケード** オプションは、新しい行がビューおよび基になるすべてのベースビューの条件に対してチェックされることを指定します。

[**コード**] タブをクリックして続行します。



[コード] タブのワークスペースを使用して、ビューを作成するクエリを作成します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、ビューに特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスから、特権を付与するロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示され

ます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

セキュリティラベル パネルを使用して、ビューに適用するセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

View ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

5.27.1 例

以下は、[**表示**] ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、 *distributor_code* という名前のビューを作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

第6章　Creating or Modifying a Table

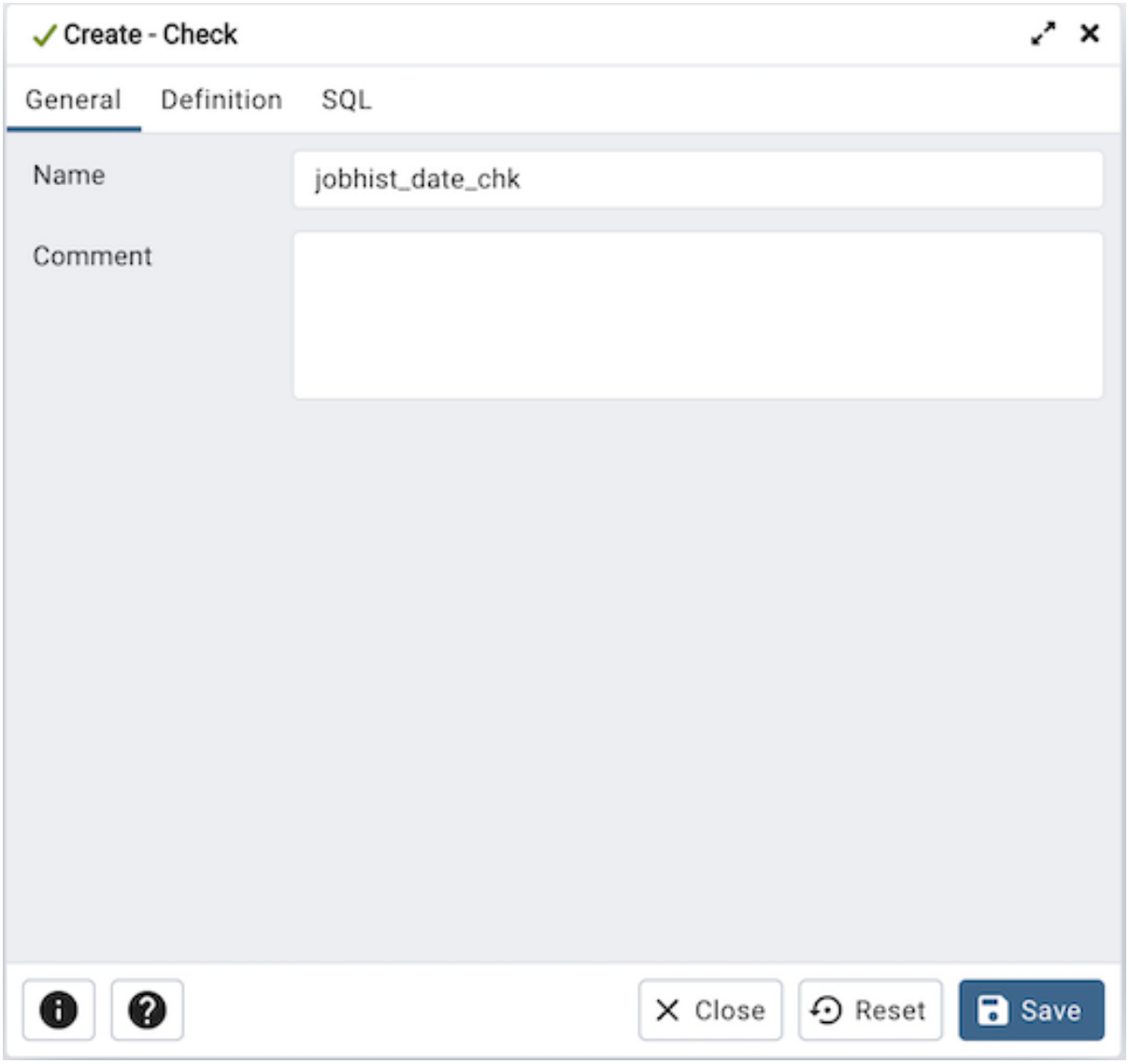
pgAdmin4 は、すべてのテーブルのプロパティと属性を変更できるダイアログを提供します。

データベースオブジェクトを作成できるダイアログにアクセスするには、pgAdmin ツリーコントロールでオブジェクトタイプを右クリックし、そのオブジェクトの *Create* オプションを選択します。たとえば、新しいテーブルを作成するには、ツリーコントロールからデータベースを選択し、データベースの下のスキーマを選択し、 **テーブル** ノードを右クリックして、 **テーブルの作成...** を選択します。

6.1 Check Dialog

チェック制約を定義または変更するには、 *Check* ダイアログを使用します。チェック制約は、挿入または更新操作を成功させるために新しい行または更新された行が満たす必要があるブール値を生成する式を指定します。

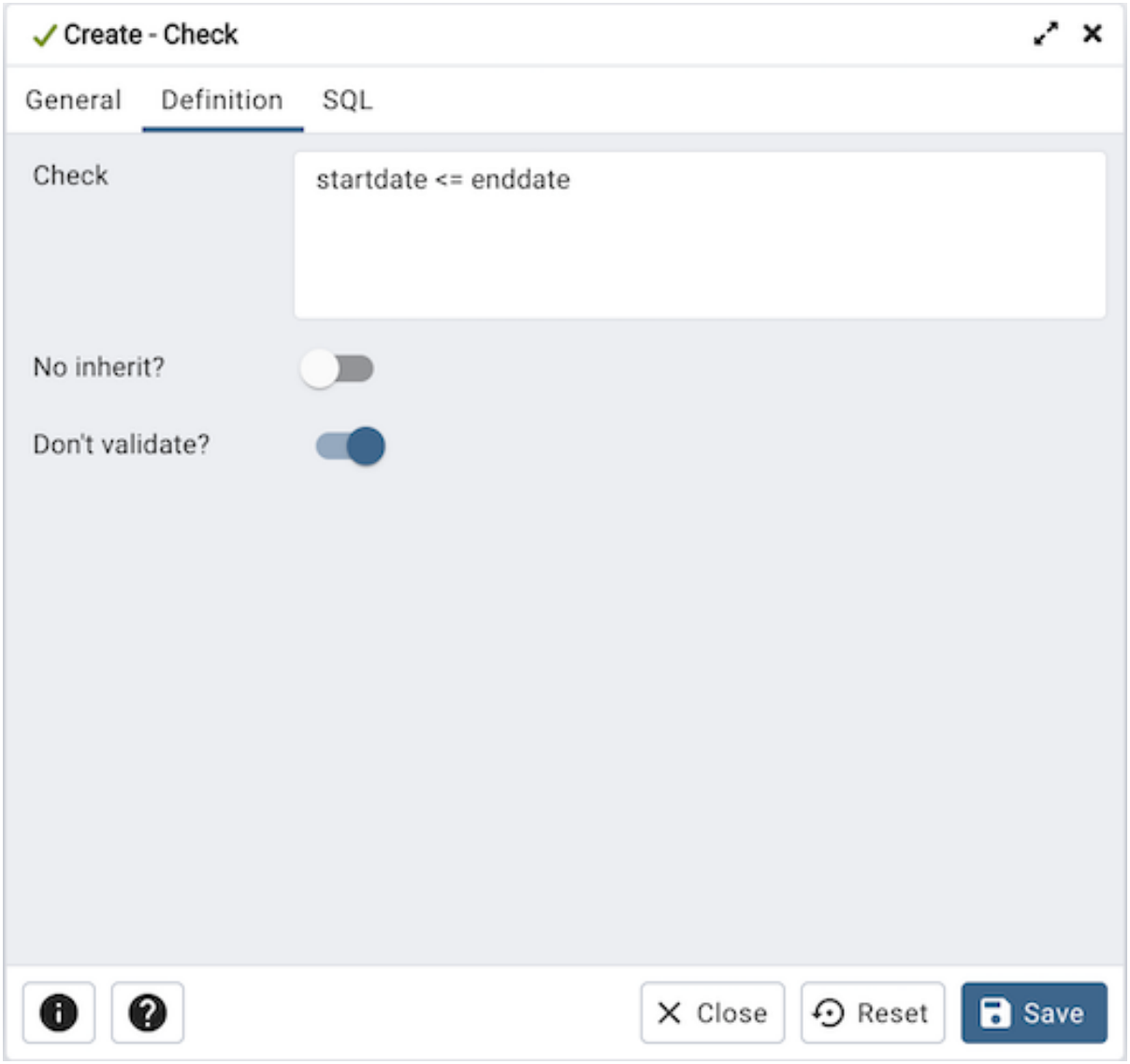
Check ダイアログは、 *General* および *Definition* タブを介してチェック制約の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、チェック制約を識別します。

- *Name* フィールドを使用して、*pgAdmin* ツリーコントロールに表示されるチェック制約の説明的な名前を指定します。PostgreSQL9.5 以降では、テーブルに複数のチェック制約がある場合、行ごとに名前のアルファベット順で NOTNULL 制約の後にテストされます。
- チェック制約に関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、チェック制約を定義します。

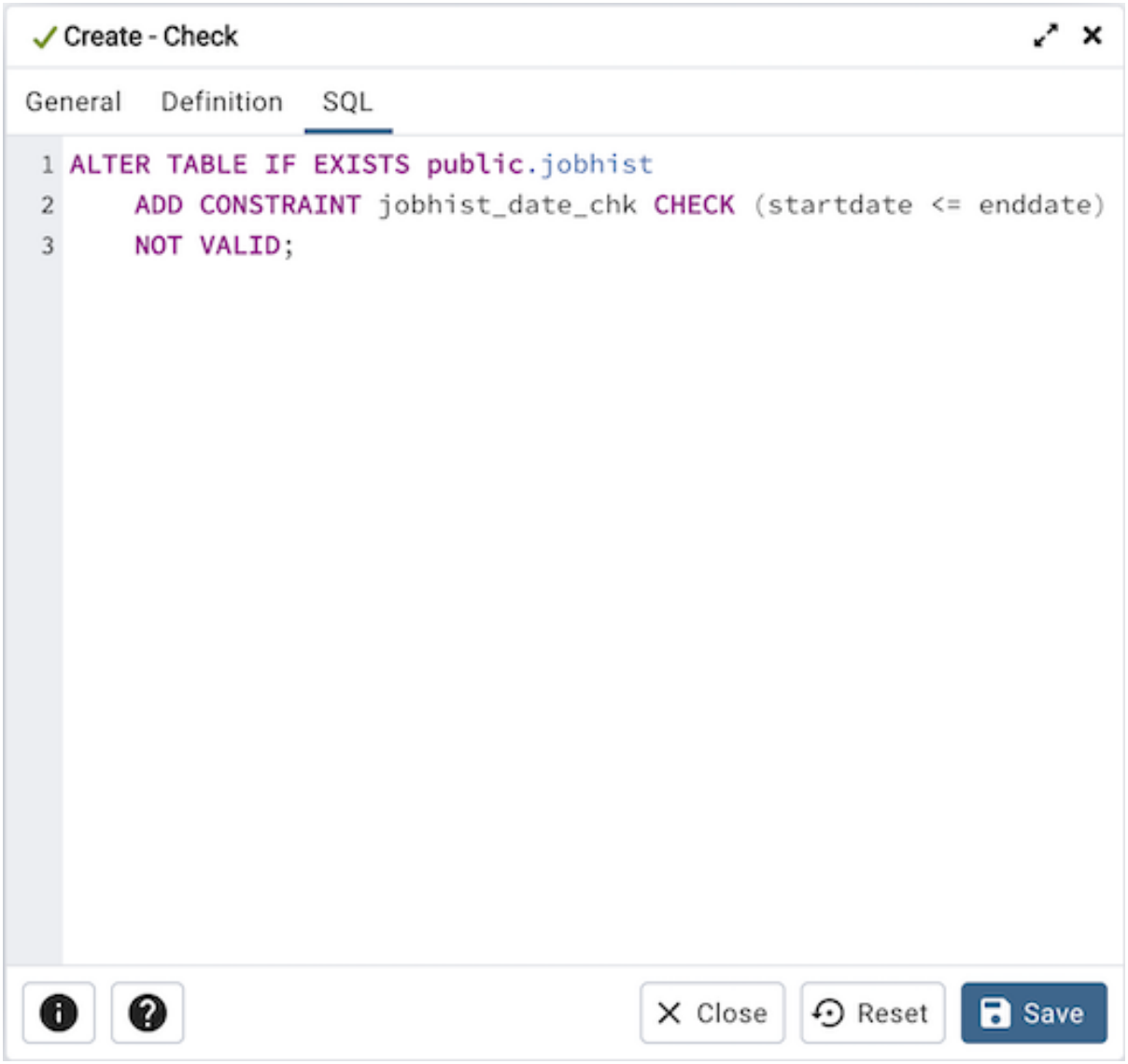
- *Check* フィールドに、行が満たさなければならない式を入力します。
- *NoInherit?* スイッチを *Yes* の位置に移動して、この制約がテーブルの子に自動的に継承されないように指定します。デフォルトは *No* で、制約はすべての子に継承されます。
- *Don'tvalidation?* スイッチを *No* の位置に移動して、既存のデータの検証をスキップします。制約は、テーブル内のすべての行に適用されるとは限りません。デフォルトは **はい** です。

SQL タブをクリックして続行します。

Check ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

Example

以下は、 *Check* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



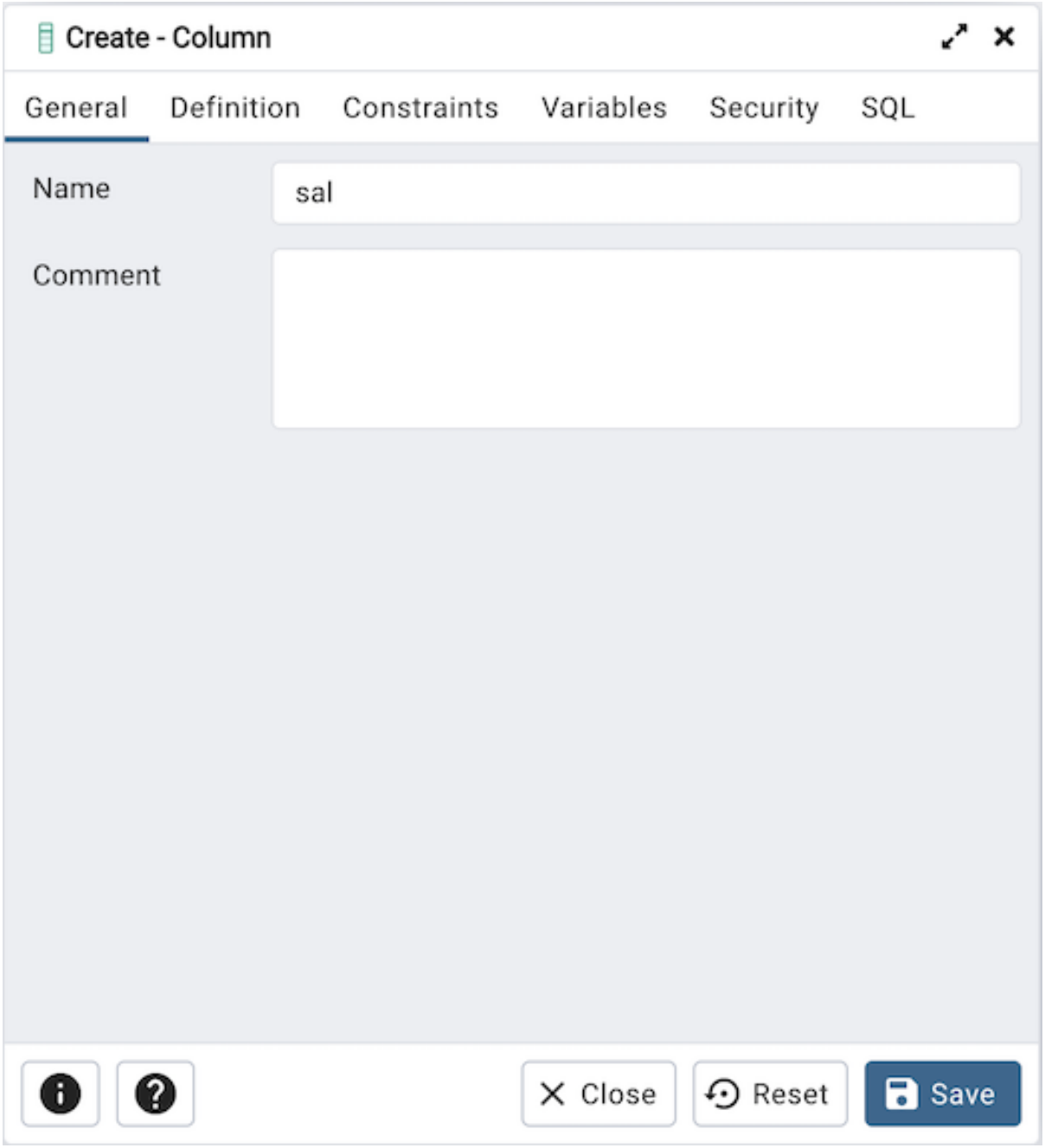
示されている例は、*jobhist* テーブルの *startdate* 列に *jobhist_date_chk* という名前のチェック制約を作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.2 Column Dialog

Column ダイアログを使用して、既存のテーブルに列を追加するか、列定義を変更します。

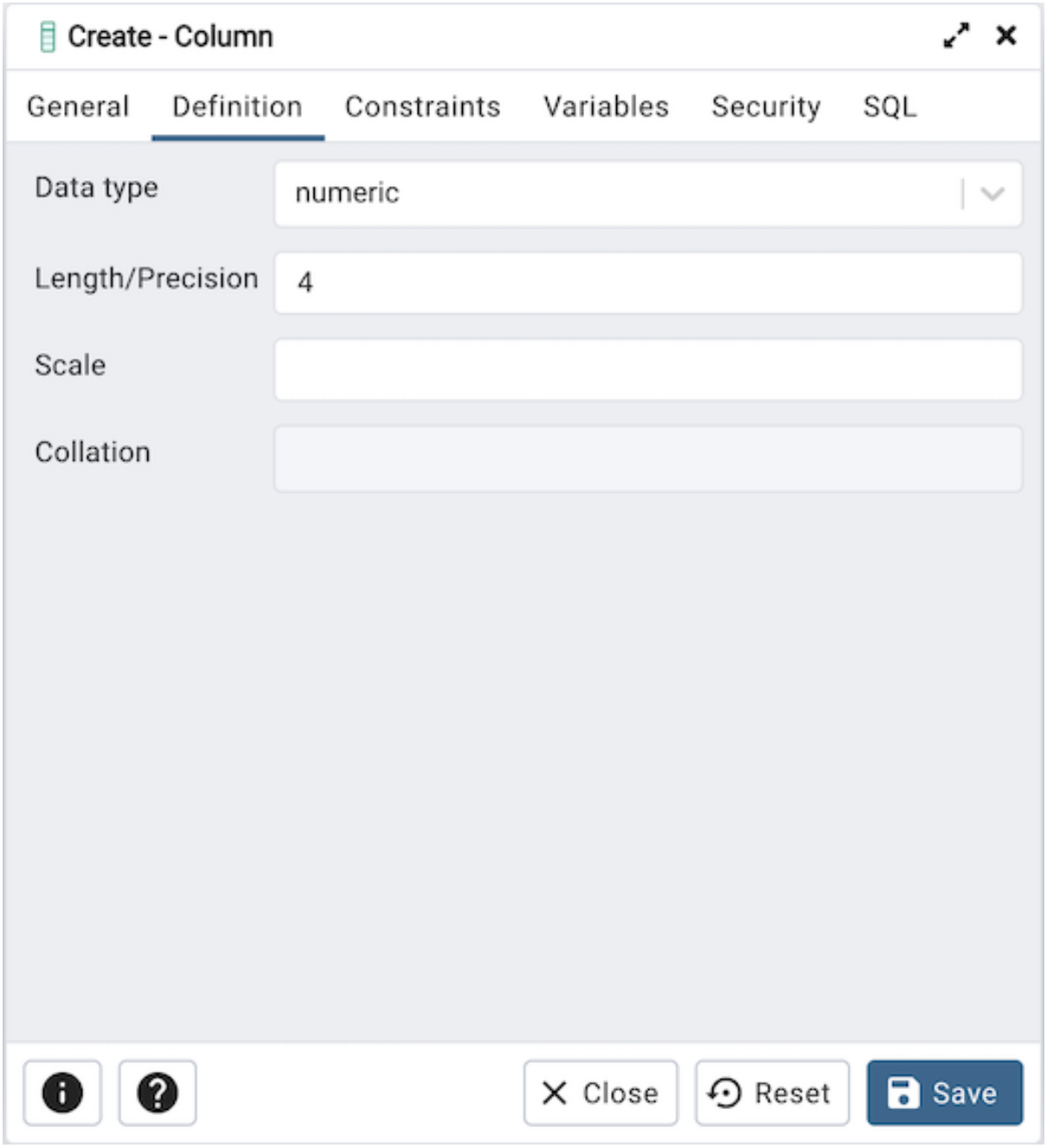
列 ダイアログは、 **一般** 、 **定義** 、および **セキュリティ** のダイアログタブを介して列の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[*General*] タブのフィールドを使用して、列を識別します。

- *Name* フィールドを使用して、列にわかりやすい名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。このフィールドは必須です。
- 列に関するメモを **コメント** フィールドに保存します。

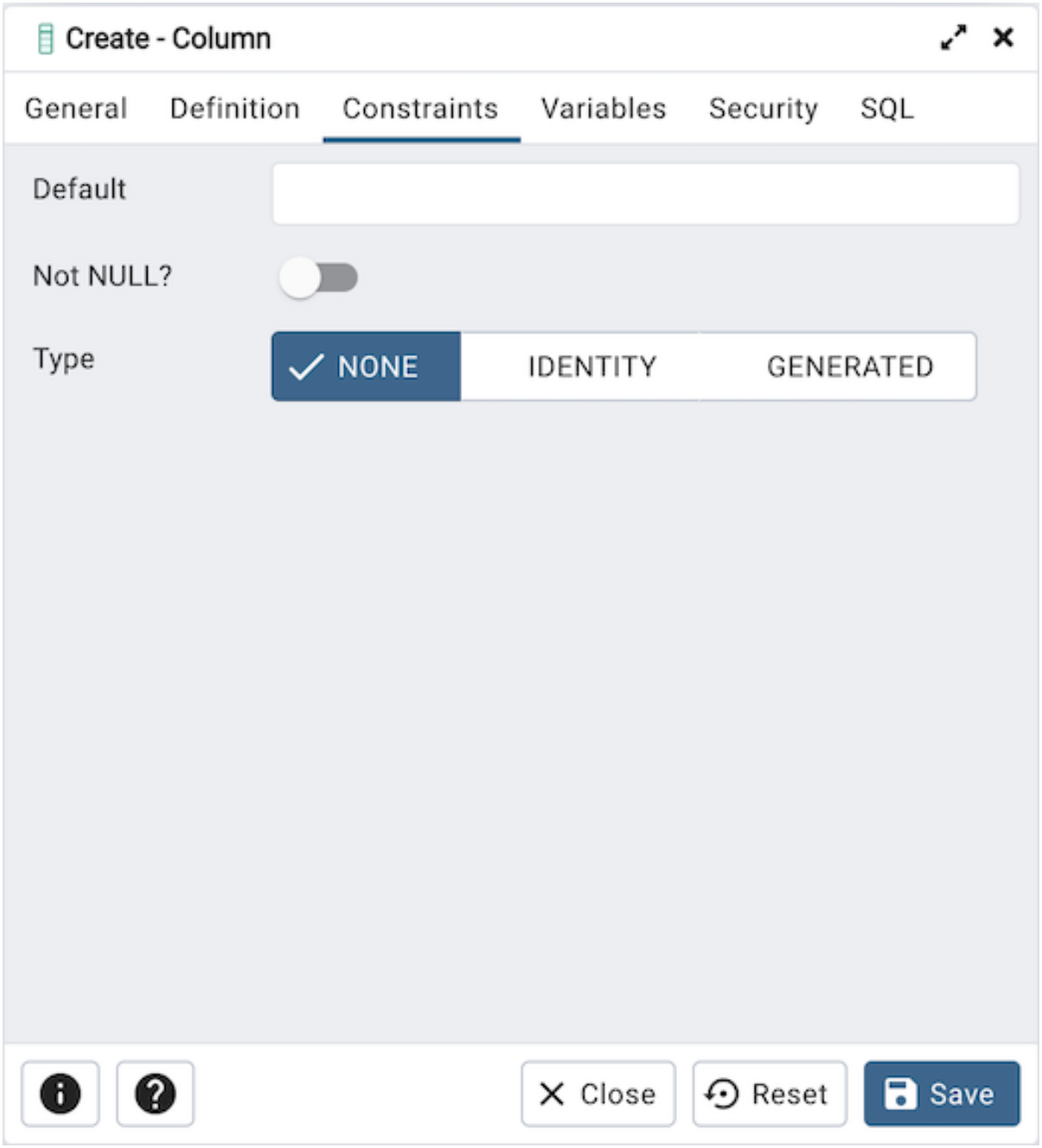
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、列のパラメーターを追加します。(該当しない場合、フィールドは無効になります。)

- *DataType* の横にあるドロップダウンリストボックスを使用して、列のデータ型を選択します。PostgreSQL でサポートされているデータ型の詳細については、Postgres コアドキュメントの 8 章を参照してください。このフィールドは必須です。
- **長さ/精度** および **スケール** フィールドを使用して、数値の最大有効桁数、またはテキスト値の最大文字数を指定します。
- *Collation* の横にあるドロップダウンリストボックスを使用して、列に照合設定を適用します。

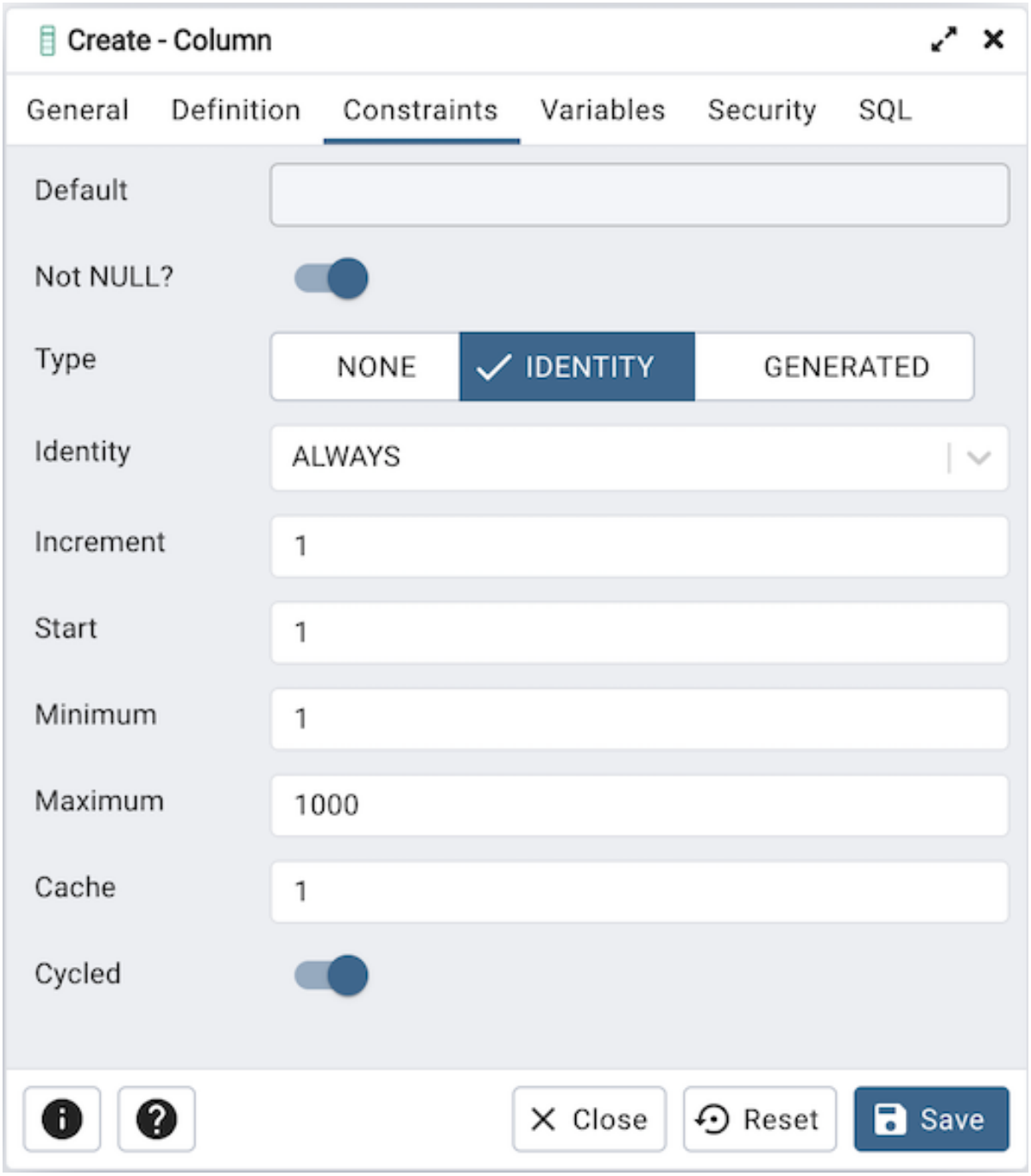
[*Constraints*] タブをクリックして続行します。



[*Constraints*] タブのフィールドを使用して、列の制約を指定します。（該当しない場合、フィールドは無効になります。）

- *DefaultValue* フィールドを使用して、デフォルトのデータ値を指定します。
- *NotNull* スイッチを *Yes* の位置に移動して、列に null 値を含めないことを指定します。デフォルトは **いいえ** です。
- *Type* フィールドを使用して、列のタイプ（NONE/IDENTITY/GENERATED）を指定します。デフォルトは *NONE* です。

IDENTITY タイプをクリックして、Identity 列を作成します。



次のフィールドを使用して、*IDENTITY* 列を作成します。ID 列は、PG/EPAS バージョン 10 以降に適用されます。

- *Identity* フィールドを使用して、ALWAYS または BYDEFAULT を指定します。この句は、INSERT ステートメントでユーザー指定の値よりもシーケンス値を優先する方法を決定するために使用されます。
- **インクリメント** フィールドを使用して、新しい値を作成するために現在のシーケンス値に追加する値を指定します。
- *Start* フィールドに値を入力して、シーケンスの開始値を指定します。デフォルトの開始値は、昇順の場合は MINVALUE、降順の場合は MAXVALUE です。
- *Minimum* フィールドに値を入力して、シーケンスが生成できる最小値を指定します。この句を指定

しないか、NOMINVALUE を指定すると、デフォルトが使用されます。昇順と降順のデフォルトは、それぞれ 1 と-263-1 です。

- *Maximum* フィールドに値を入力して、シーケンスの最大値を指定します。この句を指定しないか、NOMAXVALUE を指定すると、デフォルト値が使用されます。デフォルトは、昇順と降順でそれぞれ 263-1 と-1 です。
- *Cache* フィールドに値を入力して、アクセスを高速化するために事前に割り当ててメモリに保存するシーケンス番号を指定します。最小値は 1（一度に生成できる値は 1 つだけ、つまりキャッシュ）であり、これもデフォルトです。
- *Cycled* スイッチを *Yes* の位置に移動して、昇順または降順のシーケンスがそれぞれ MAXVALUE または MINVALUE に達したときにシーケンスを折り返せるようにします。制限に達した場合、次に生成される数値はそれぞれ MINVALUE または MAXVALUE です。デフォルトは **いいえ** です。

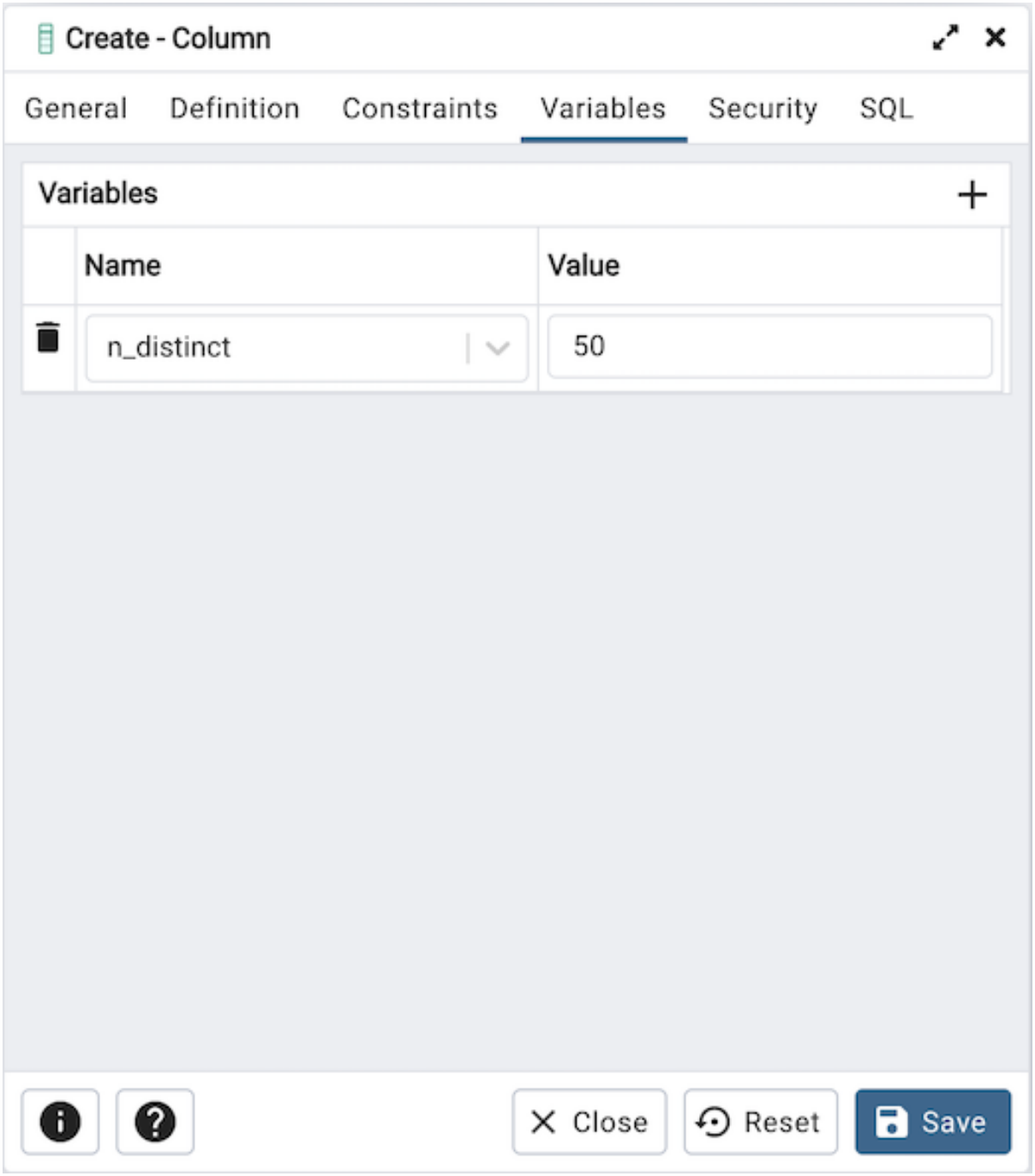
GENERATED タイプをクリックして、生成列を作成します。



次のフィールドを使用して、 *GENERATED* 列を作成します。生成された列は、PG/EPAS バージョン 12 以降に適用されます。

- *Expression* フィールドを使用して、生成式を指定します。テーブル内の他の列を参照できますが、他の生成された列は参照できません。使用される関数と演算子は不変である必要があります。他のテーブルへの参照は許可されません。

[**変数**] タブをクリックして続行します。



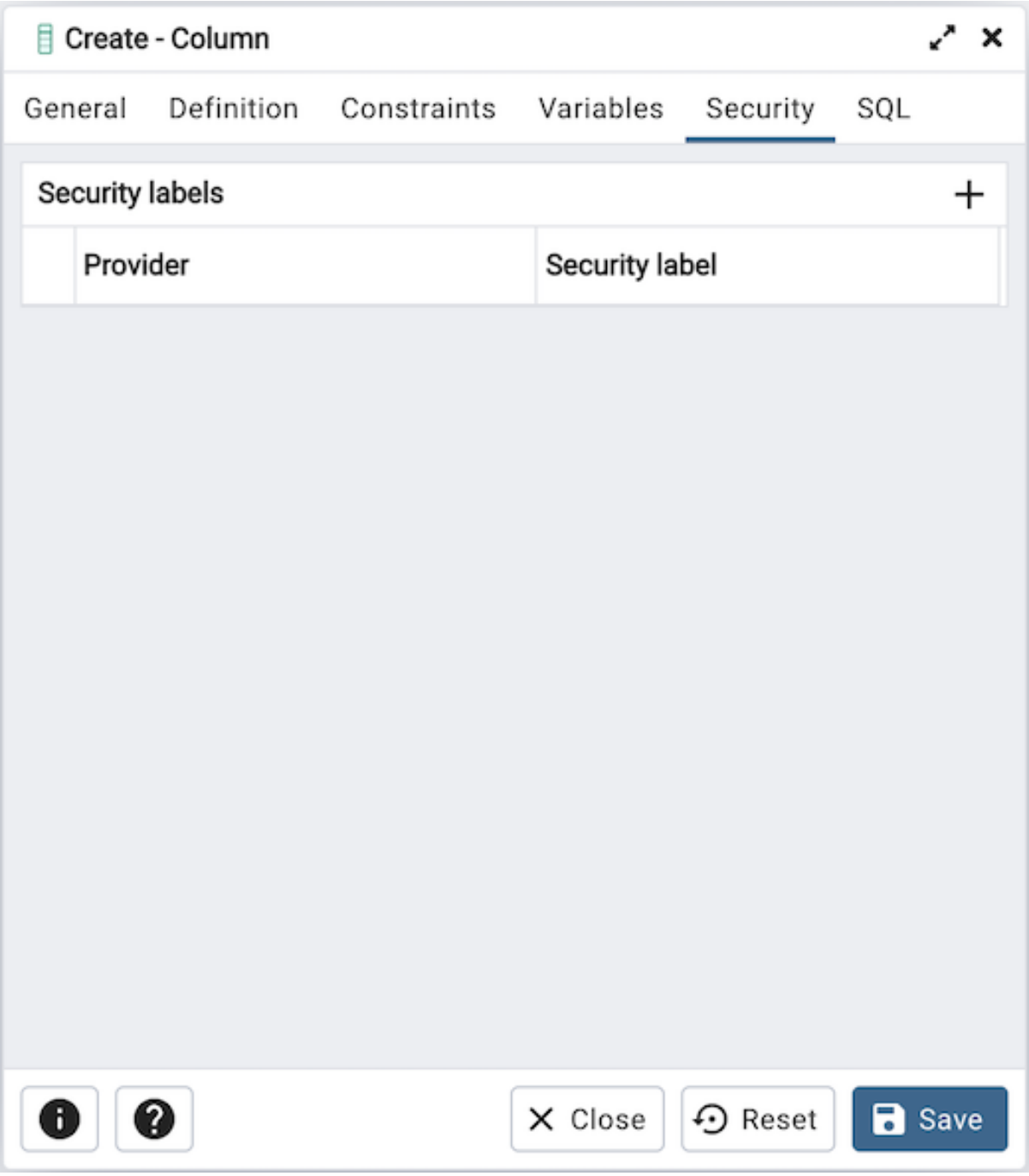
[変数] タブを使用して、列に存在する可能性のある個別の値の数を指定します。この値は、ANALYZE コマンドによる推定をオーバーライドします。 **追加** アイコン（+）をクリックして、 **名前 / 値** のペアを追加します。

- **名前** フィールドのドロップダウンリストボックスから変数の名前を選択します。
  - *n_distinct* を選択して、列の個別の値の数を指定します。
  - *n_distinct_inherited* を選択して、テーブルとその子の個別の値の数を指定します。
- *Value* フィールドに個別の値の数を指定します。詳細については、 ALTER TABLE のドキュメントを参照してください。

追加 アイコン（+）をクリックして、追加の **名前 / 値** のペアを指定します。変数を破棄するには、行の左

側にあるごみ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、属性を割り当て、セキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

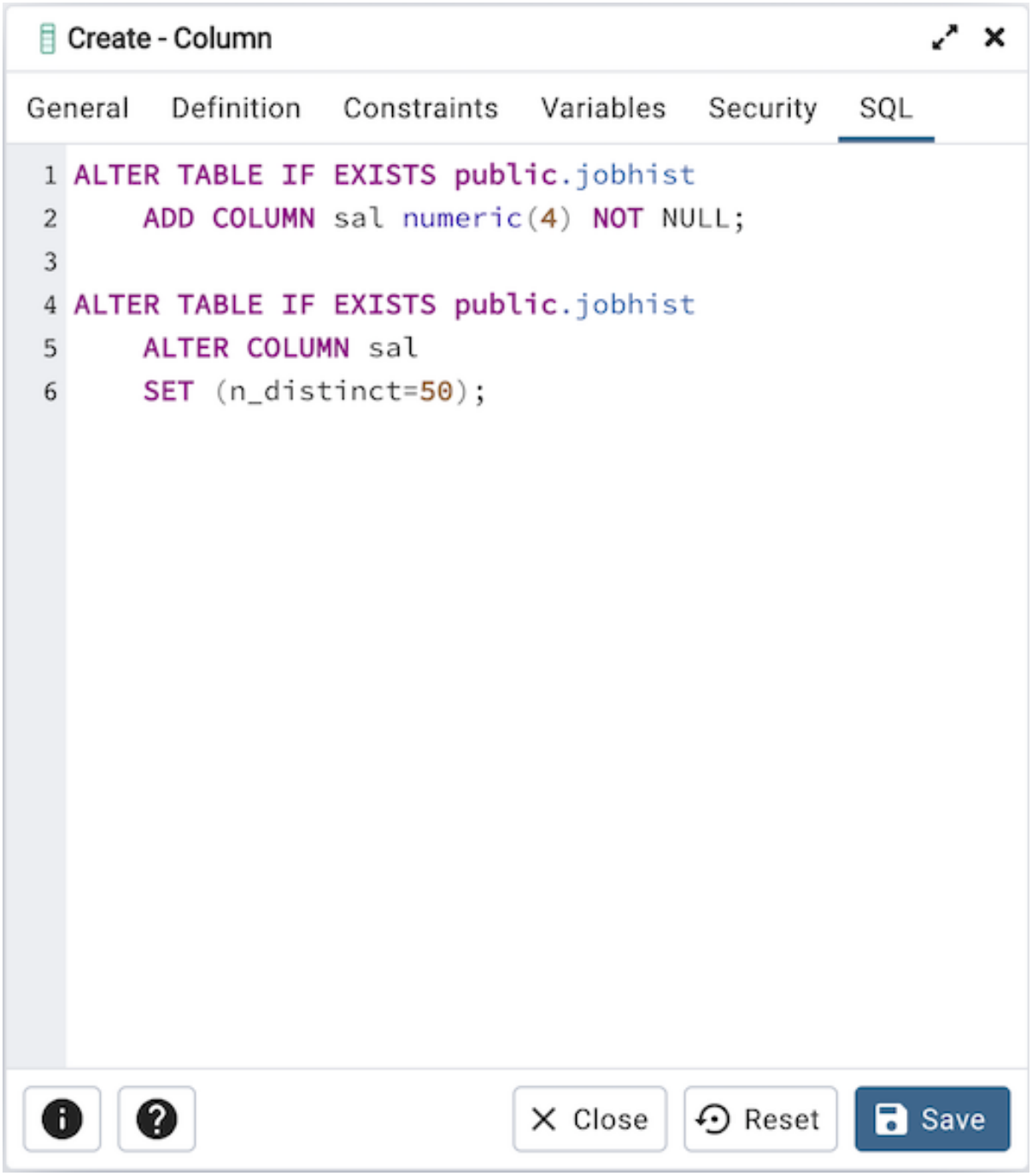
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、*DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

Column ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.2.1 例

以下は、 *Column* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



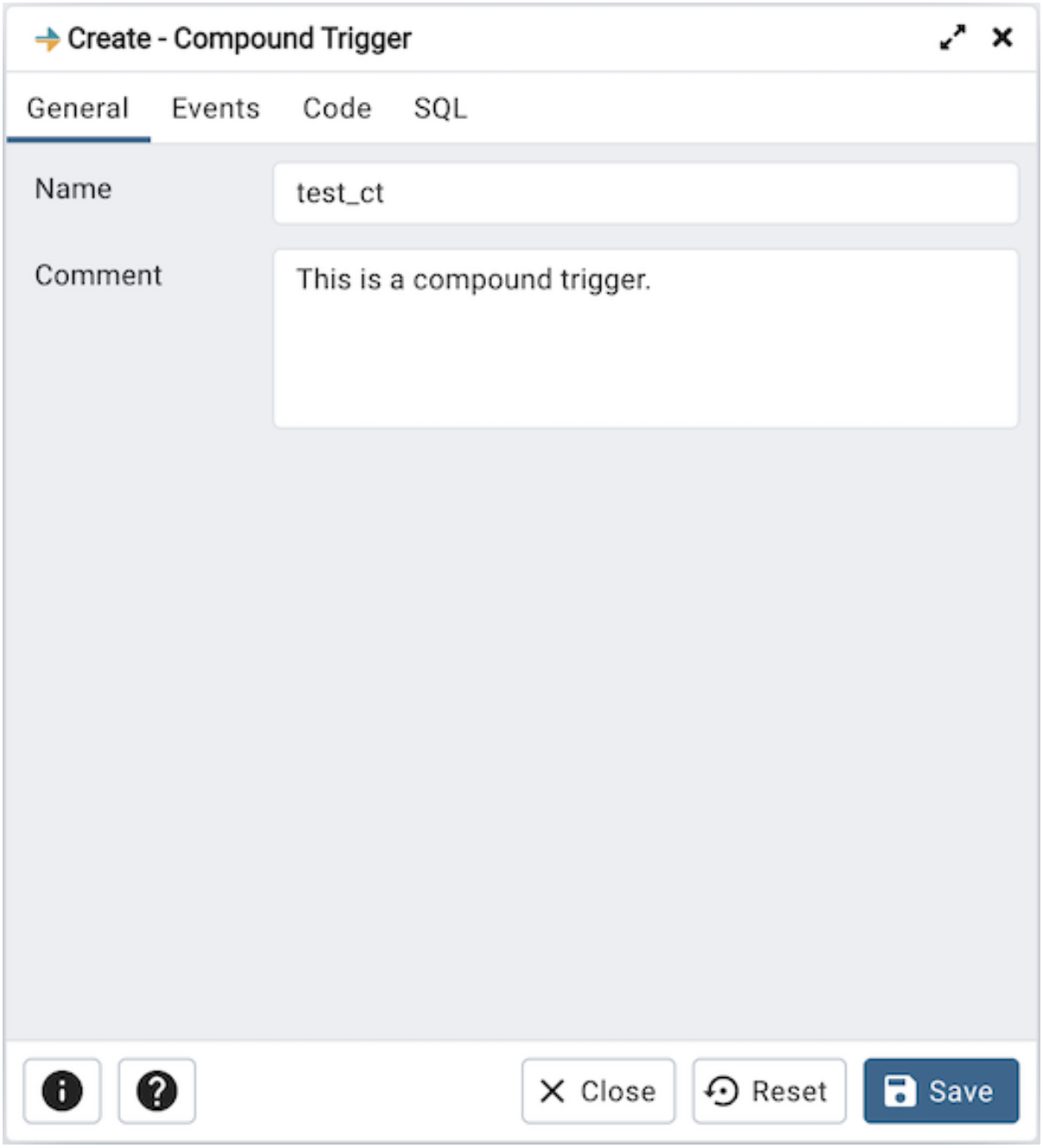
ここに示す例は、 *jobhist* という名前のテーブルに *sal* という名前の列を作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.3 Compound Trigger Dialog

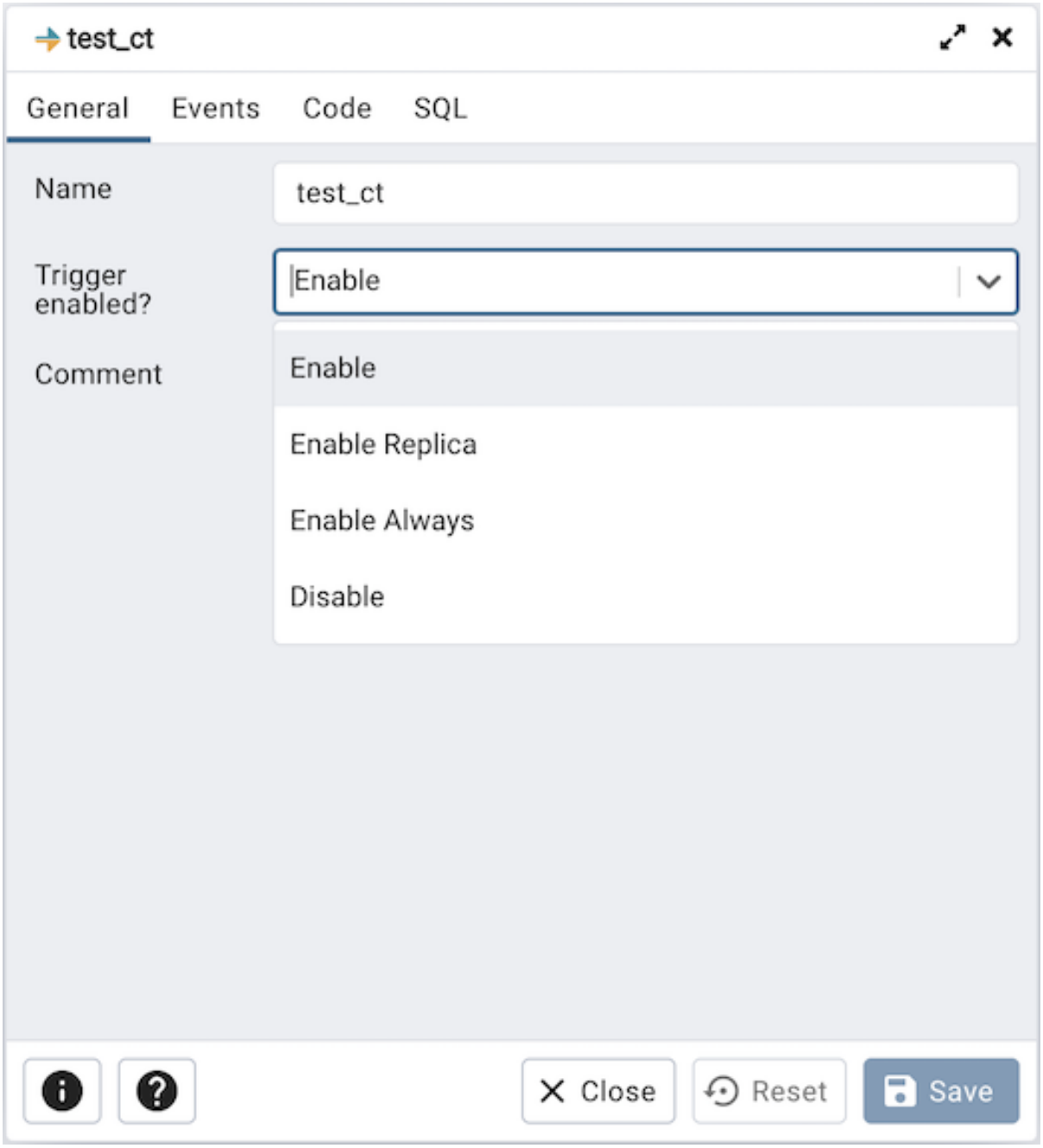
複合トリガー ダイアログを使用して、複合トリガーを作成するか、既存の複合トリガーを変更します。 **複合トリガー** は、EPAS サーバー 12 以降でのみサポートされています。複合トリガーは、特定のイベントが発生したときに指定されたコードを実行します。

複合トリガー ダイアログは、 **一般** 、 **イベント** 、および **コード** のダイアログタブを介して複合トリガーの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、複合トリガーを識別します。

- *Name* フィールドを使用して、複合トリガーに説明的な名前を追加します。これは、同じテーブルの他の複合トリガーの名前とは異なる必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。
- 複合トリガーに関するメモを **コメント** フィールドに保存します。



- **トリガー有効** フィールドは、トリガーが作成されると、複合トリガーダイアログで使用できます。4 つのオプションから 1 つを選択できます。

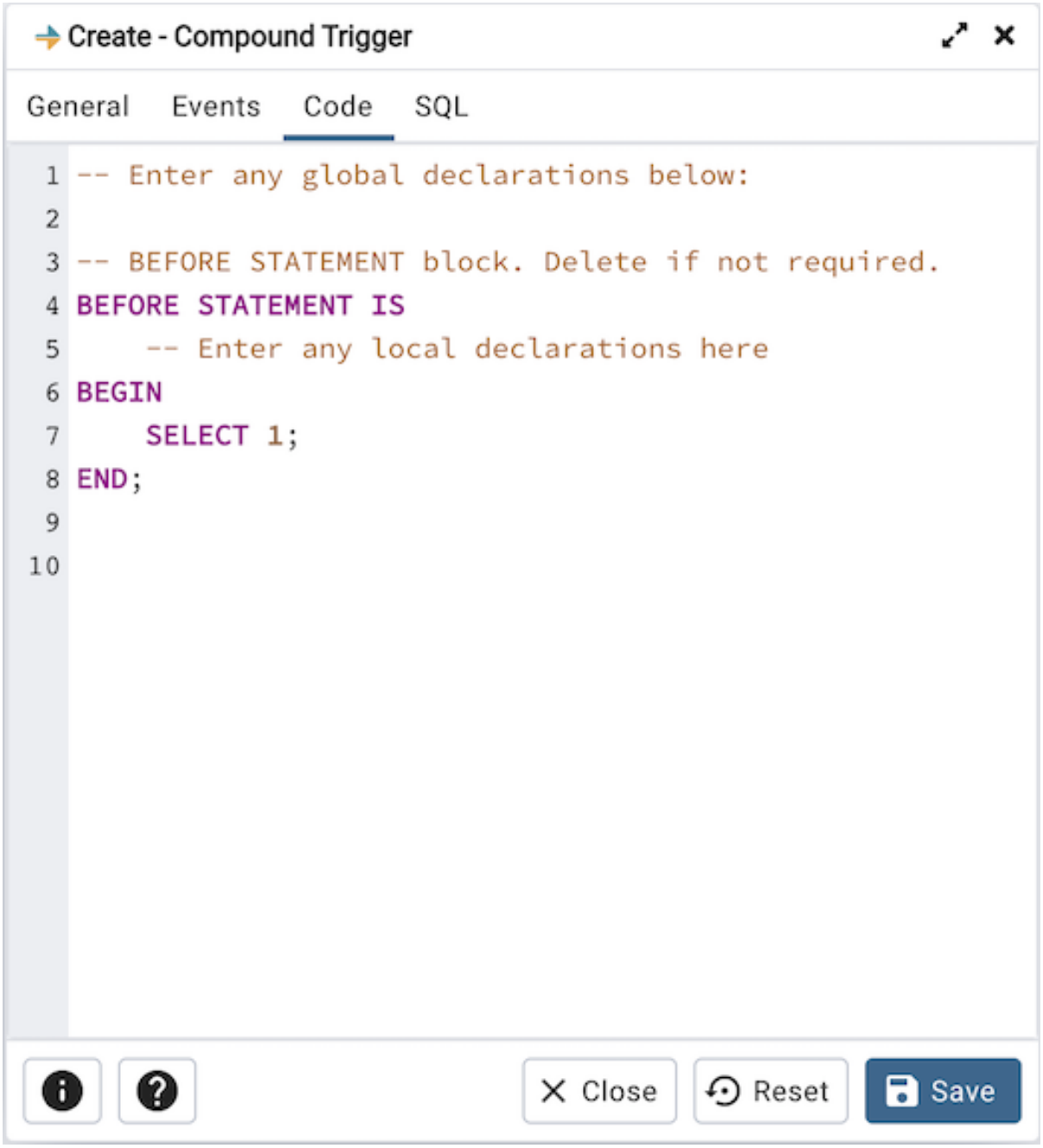
[**イベント**] タブをクリックして続行します。



[**イベント**] タブのフィールドを使用して、複合トリガーを起動する方法とタイミングを指定します。

- 複合トリガーを呼び出すイベントのタイプを選択します。イベントタイプを選択するには、イベントの横にあるスイッチを *YES* の位置に動かします。サポートされているイベントタイプは *INSERT* 、 *UPDATE* 、 *DELETE* 、および *TRUNCATE* です。ビューは TRUNCATE トリガーを持つことはできません。
- *When* フィールドを使用して、複合トリガーを呼び出すブール条件を提供します。
- 列固有の複合トリガーを定義する場合は、 *Columns* フィールドを使用して、複合トリガーのターゲットとなる列を指定します。

[**コード**] タブをクリックして続行します。



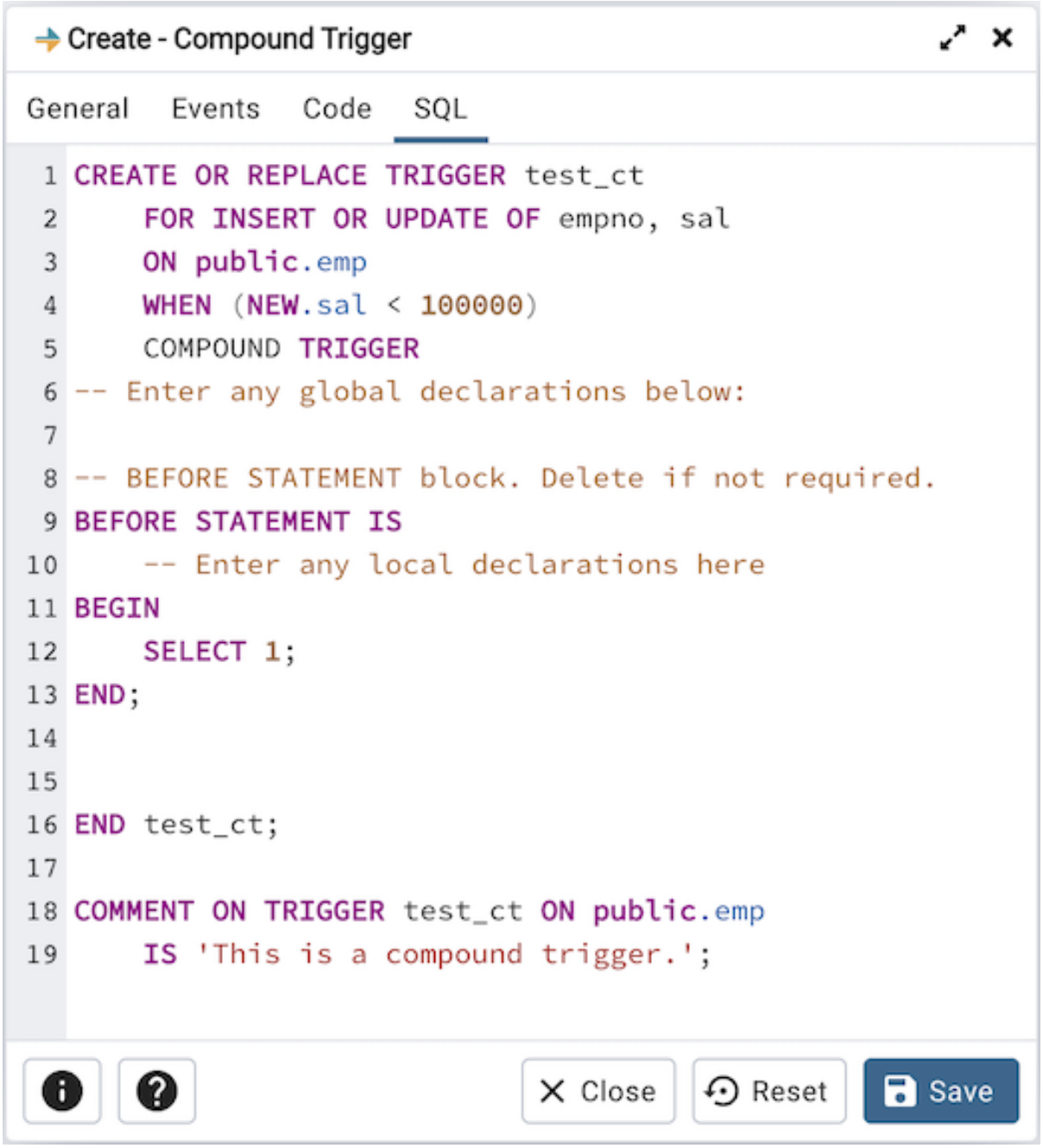
Code フィールドを使用して、コンパウンドがトリガーが発火します。基本テンプレートにはプレースホルダーが用意されています。

SQL タブをクリックして続行します。

複合トリガー ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.3.1 例

以下は、 **複合トリガー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



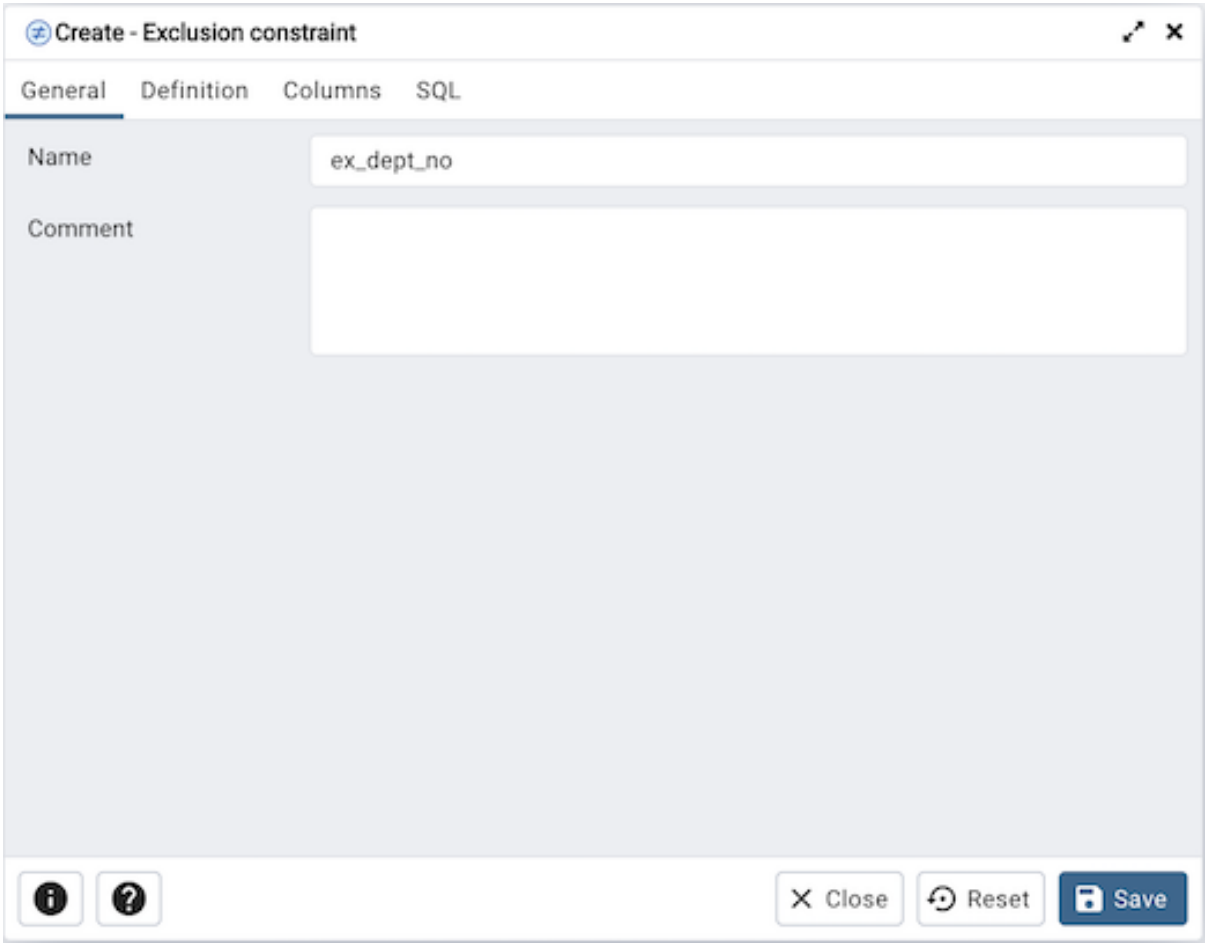
この例では、 *test_ct* という名前の複合トリガーの作成を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.4 Exclusion Constraint Dialog

除外制約の動作を定義または変更するには、 **除外制約** ダイアログを使用します。除外制約は、指定された列または式で（指定された演算子を使用して）2 つの行を比較した場合、少なくとも 1 つの演算子比較が false または null を返すことを保証します。

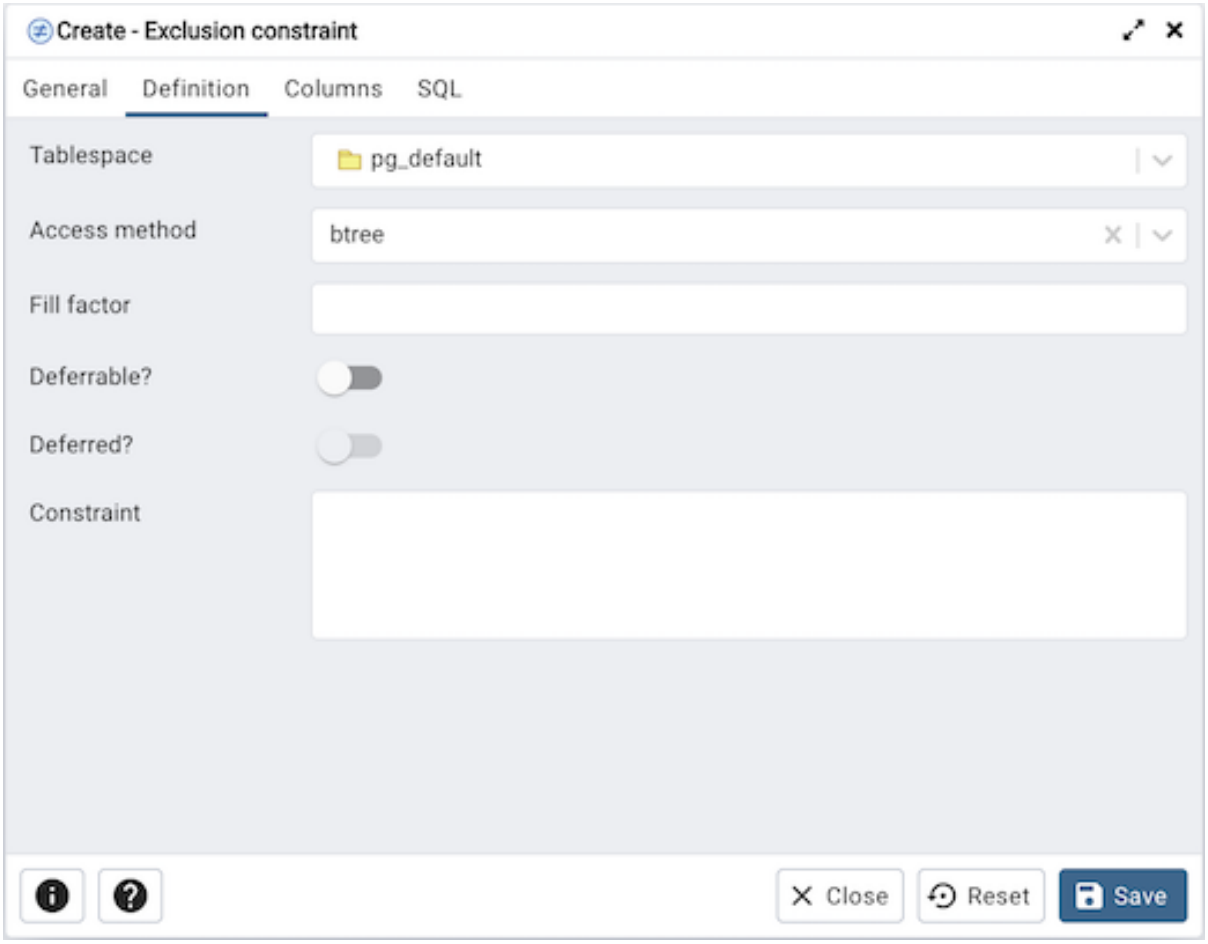
除外制約 ダイアログは、 **一般** 、 **定義** 、および **列** のダイアログタブを使用して除外制約の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、除外制約を識別します。

- *Name* フィールドを使用して、除外制約に説明的な名前を付けます。名前は *pgAdmin* ツリーコントロールに表示されます。

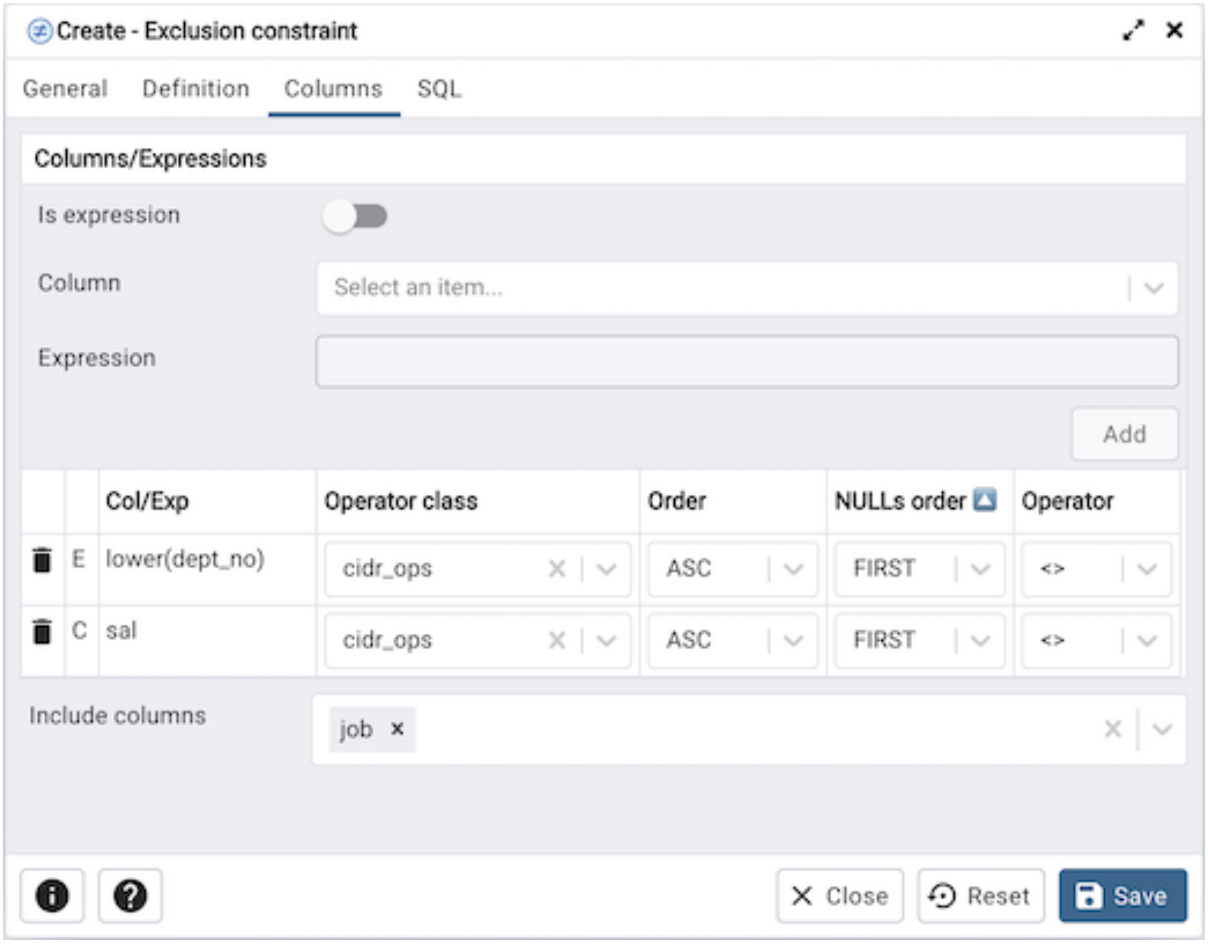
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、除外制約を定義します。

- *Tablespace* の横にあるドロップダウンリストボックスを使用して、除外制約に関連付けられたインデックスが存在するテーブルスペースを選択します。
- **アクセス方法** の横にあるドロップダウンリストボックスを使用して、除外制約を実装するときに使用されるインデックスの種類を指定します。
  - *gist* を選択して GiST インデックスを指定します。
  - スペースパーティションの GiST インデックスを指定するには、 *spgist* を選択します。
  - *btree* を選択して B ツリーインデックスを指定します。
  - *hash* を選択して、ハッシュインデックスを指定します。
- *FillFactor* フィールドを使用して、テーブルと関連するインデックスの FILLFACTOR を指定します。FILLFACTOR は 10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- *Deferrable?* スイッチを *Yes* の位置に移動して、制約のタイミングが遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。
- *Constraint* フィールドを使用して、行をテーブルに含める必要がある条件を指定します。

[**列**] タブをクリックして続行します。



[**列**] タブのフィールドを使用して、制約を適用する列または式を指定します。 **式は***?* スイッチを使用して、式のテキスト入力を有効にします。 *Column* の横にあるドロップダウンリストボックスを使用して、列を選択します。 **列** を選択するか **式** を入力したら、 **追加** アイコン（+）をクリックして、列/式のアクションの詳細を提供します。

- *Col/Exp* フィールドには、*Column* ドロップダウンリストボックスでの選択または入力した *Expression* が入力されます。
- 必要に応じて、 **演算子クラス** のドロップダウンリストボックスを使用して、列のインデックスで使用される演算子クラスを指定します。
- *DESC* スイッチを *DESC* に移動して、降順の並べ替えを指定します。デフォルトは *ASC* で、昇順の並べ替えを指定します。
- *NULL* **オーダー** 列を使用して、NULL 値の配置を指定します（並べ替えの場合）。 *FIRST* または *LAST* を指定します。
- **演算子** の横にあるドロップダウンリストを使用して、比較または条件演算子を指定します。

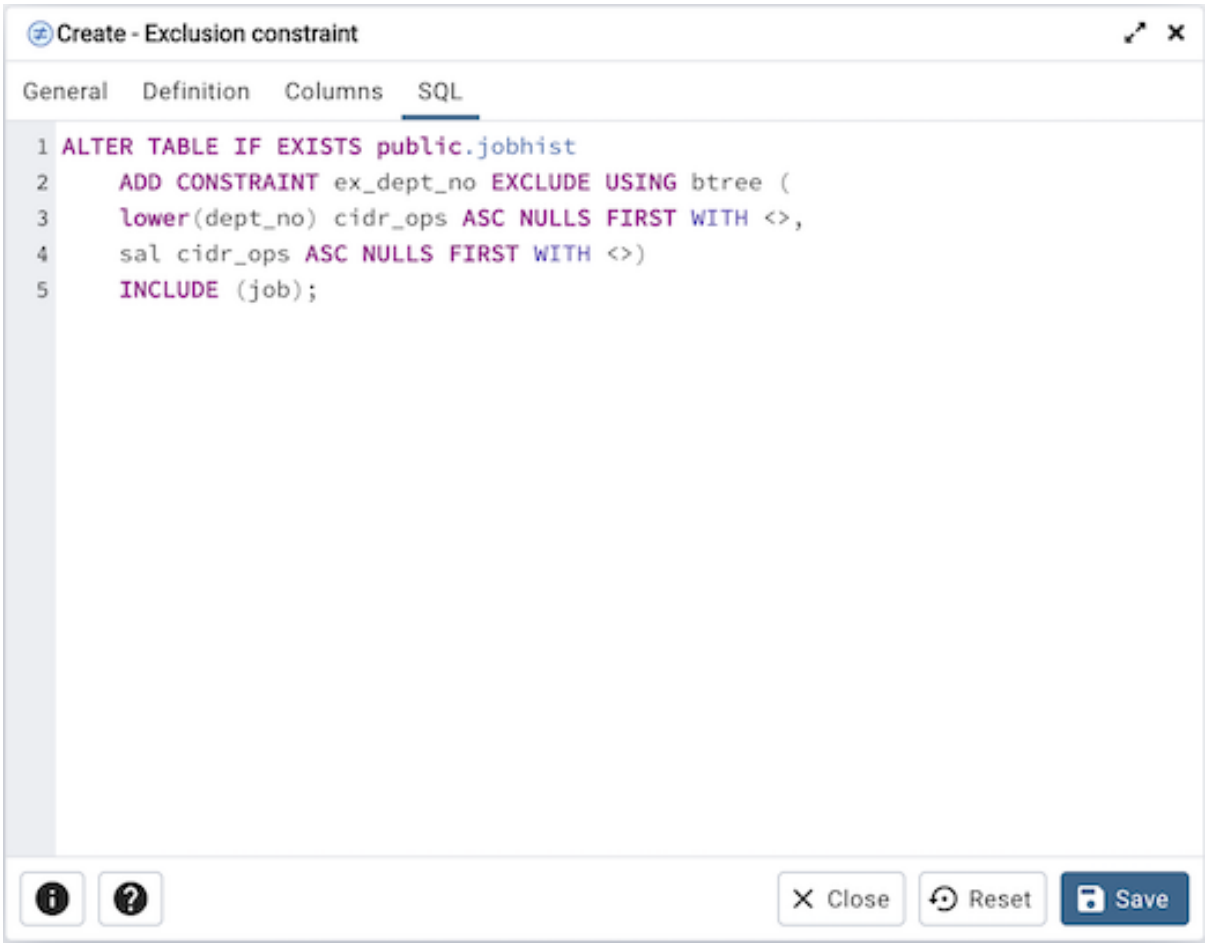
Includecolumns フィールドを使用して、制約の *INCLUDE* 句の列を指定します。このオプションは Postgres11 以降で使用できます。

SQL タブをクリックして続行します。

除外制約 ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.4.1 例

以下は、 **除外制約** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



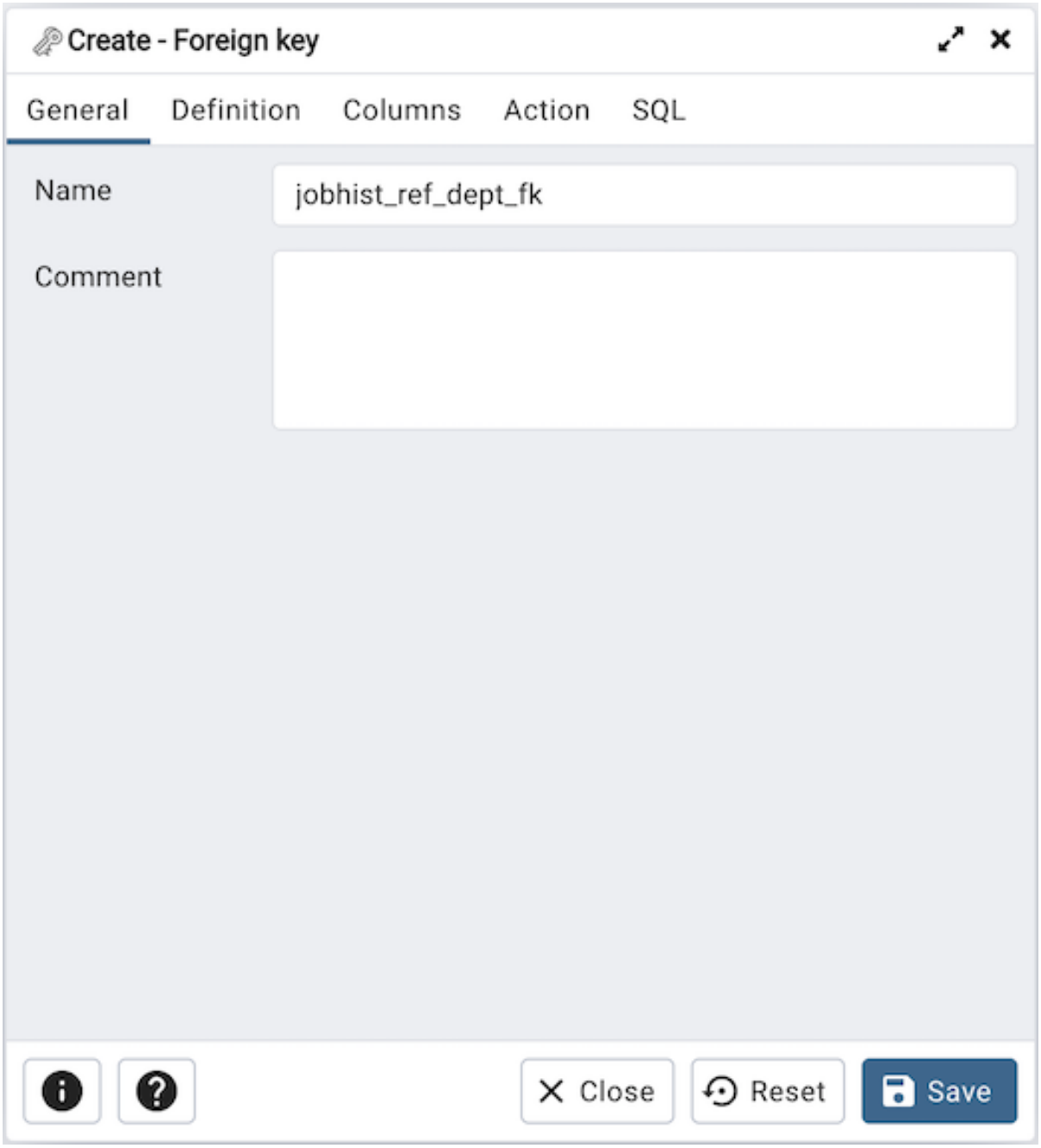
この例は、 *ex_dept_no* という名前の除外制約の作成を示しています。制約は btree インデックスを使用します。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.5 Foreign key Dialog

外部キー ダイアログを使用して、外部キー制約の動作を指定します。外部キー制約は、2 つのテーブル間の参照整合性を維持します。一時表と永続表の間に外部キー制約を定義することはできません。

外部キー ダイアログは、次のダイアログタブを介して外部キー制約の開発を編成します。 **一般** 、 **定義** 、 **列** 、および **アクション** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、外部キー制約を識別します。

- *Name* フィールドを使用して、外部キーの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **コメント** フィールドに外部キー制約に関するメモを保存します。

[**定義**] タブをクリックして続行します。



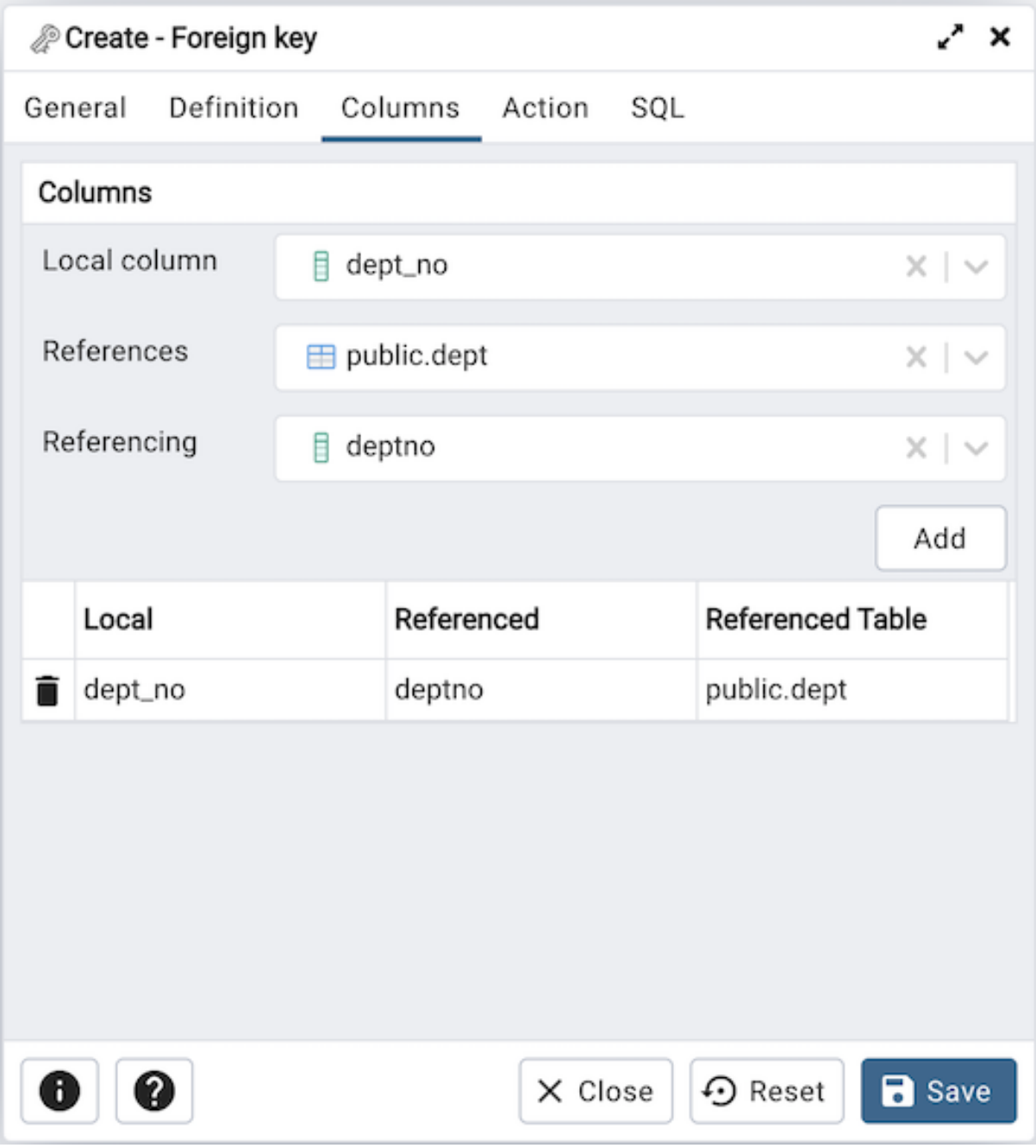
[**定義**] タブのフィールドを使用して、外部キー制約を定義します。

- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。
- *Matchtype* スイッチを移動して、制約によって適用されるマッチングのタイプを指定します。
  - いずれかの列が NULL の場合、マルチ列外部キーのすべての列を NULL にする必要があることを示すには、 *Full* を選択します。すべての列が null の場合、参照先のテーブルで行が一致する必要はありません。
  - 単一の外部キー列を null にできるように指定するには、[**シンプル**] を選択します。いずれかの

列が null の場合、参照先のテーブルで行が一致する必要はありません。

- *Validated* スイッチを *Yes* の位置に移動して、このダイアログへの変更を保存するときに既存のテーブルのコンテンツを（外部キーまたはチェック制約に対して）検証するようにサーバーに指示します。
- **自動 *FK* インデックス** スイッチを **いいえ** の位置に移動して、自動インデックス機能を無効にします。
- *AutoFKIndex* スイッチが *Yes* の位置にある場合、 *CoveringIndex* の横にあるフィールドはインデックスの名前を生成します。または、このフィールドは無効になっています。

[**列**] タブをクリックして続行します。



[**列**] タブのフィールドを使用して、1 つ以上の参照列を指定します。外部キー制約では、テーブルの 1 つ以上の列に、参照されるテーブルの行の参照される列の値と一致する値のみが含まれている必要があります。

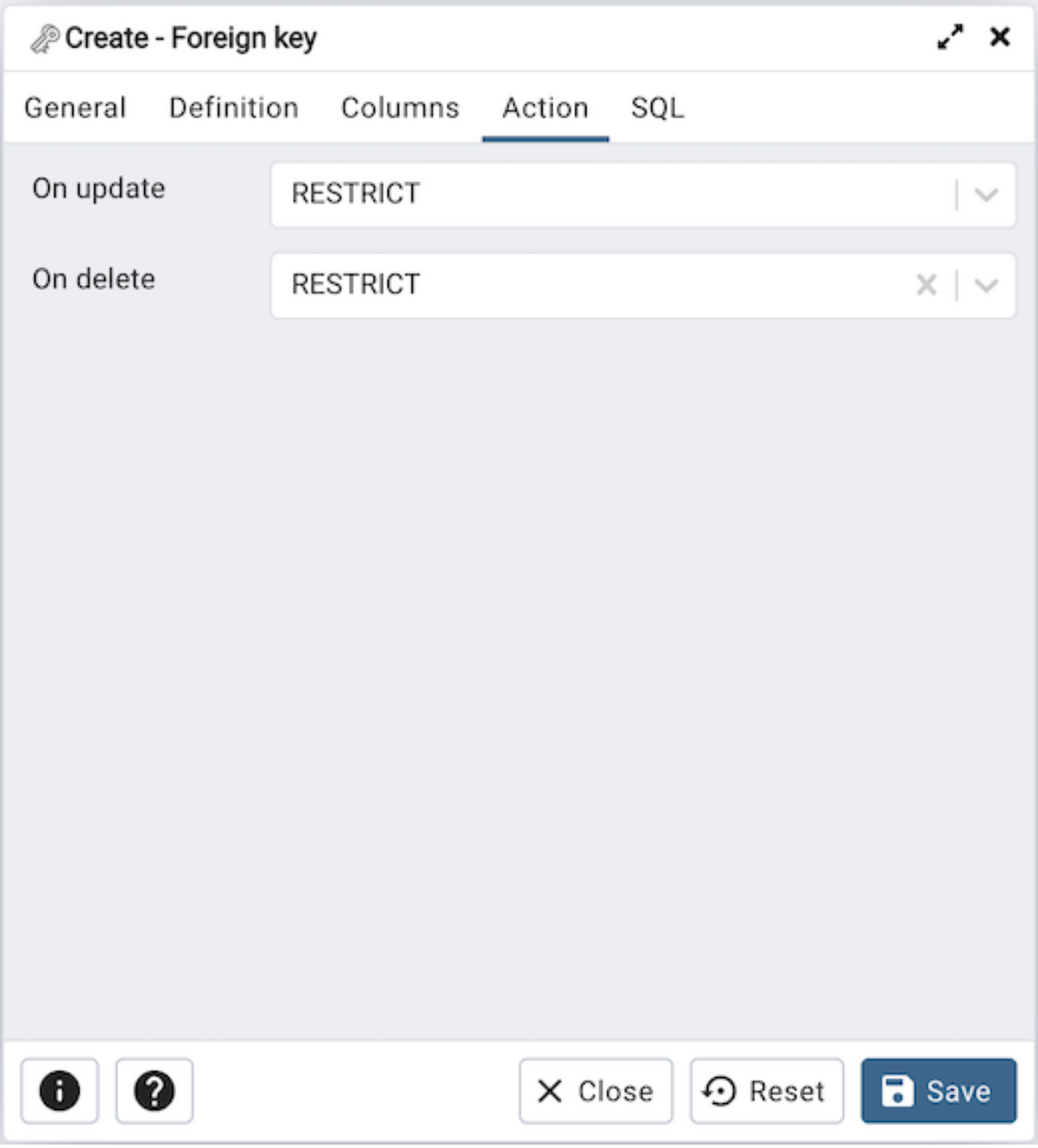
- **ローカル列** の横にあるドロップダウンリストボックスを使用して、外部テーブルと比較する現在の

テーブルの列を指定します。

- *References* の横にあるドロップダウンリストボックスを使用して、比較列が存在するテーブルの名前を指定します。
- *Referencing* の横にあるドロップダウンリストボックスを使用して、外部テーブルの列を指定します。

追加 アイコン（+）をクリックして、リストに列を追加します。上記の手順を繰り返し、 **追加** アイコン（+）をクリックして列を追加します。エントリを破棄するには、エントリの左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**アクション**] タブをクリックして続行します。



[**アクション**] タブのドロップダウンリストボックスを使用して、テーブル内のデータが更新または削除されたときに実行される外部キー制約に関連する動作を指定します。

- *Onupdate* の横にあるドロップダウンリストボックスを使用して、テーブル内のデータが更新されたときに実行されるアクションを選択します。
- **削除時** の横にあるドロップダウンリストボックスを使用して、テーブル内のデータが削除されたときに実行されるアクションを選択します。

サポートされているアクションは次のとおりです。

アクションなし	削除または更新により外部キー制約違反が作成されることを示すエラーを生成します。制約が据え置かれている場合、参照する行がまだ存在すると、このエラーが制約チェック時に生成されます。これがデフォルトです。
制限	削除または更新によって外部キー制約違反が発生することを示すエラーをスローします。これは、チェックが遅延可能でないことを除いて NOACTION と同じです。
カスケード	削除した行を参照している行を削除するか、参照している列の値を参照されている列の新しい値にそれぞれ更新します。
NULL に設定	参照する列を null に設定します。
デフォルトに設定	参照する列をデフォルト値に設定します。参照されるテーブルには、デフォルト値と一致する行が必要です（null でない場合）。そうでない場合、操作は失敗します。

SQL タブをクリックして続行します。

外部キー ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.5.1 例

以下は、 **外部キー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



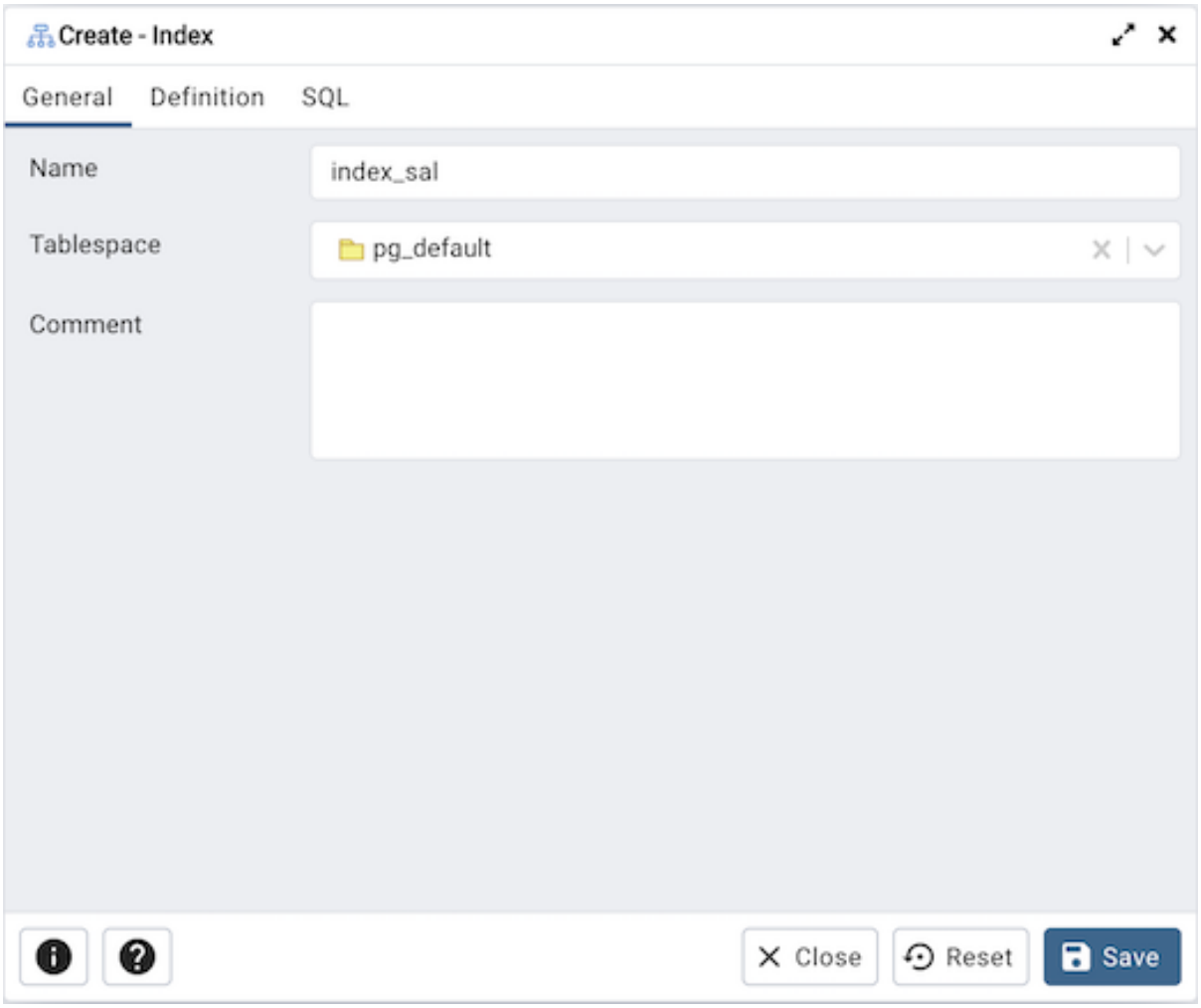
ここに示す例は、 *jobhist* テーブルの *dept_no* 列の値を *dept* テーブルの *deptno* 列の値と一致させる *jobhist_ref_dept_fk* という名前の外部キー制約を作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.6 Index Dialog

Index ダイアログを使用して、指定したテーブルまたはマテリアライズドビューにインデックスを作成します。

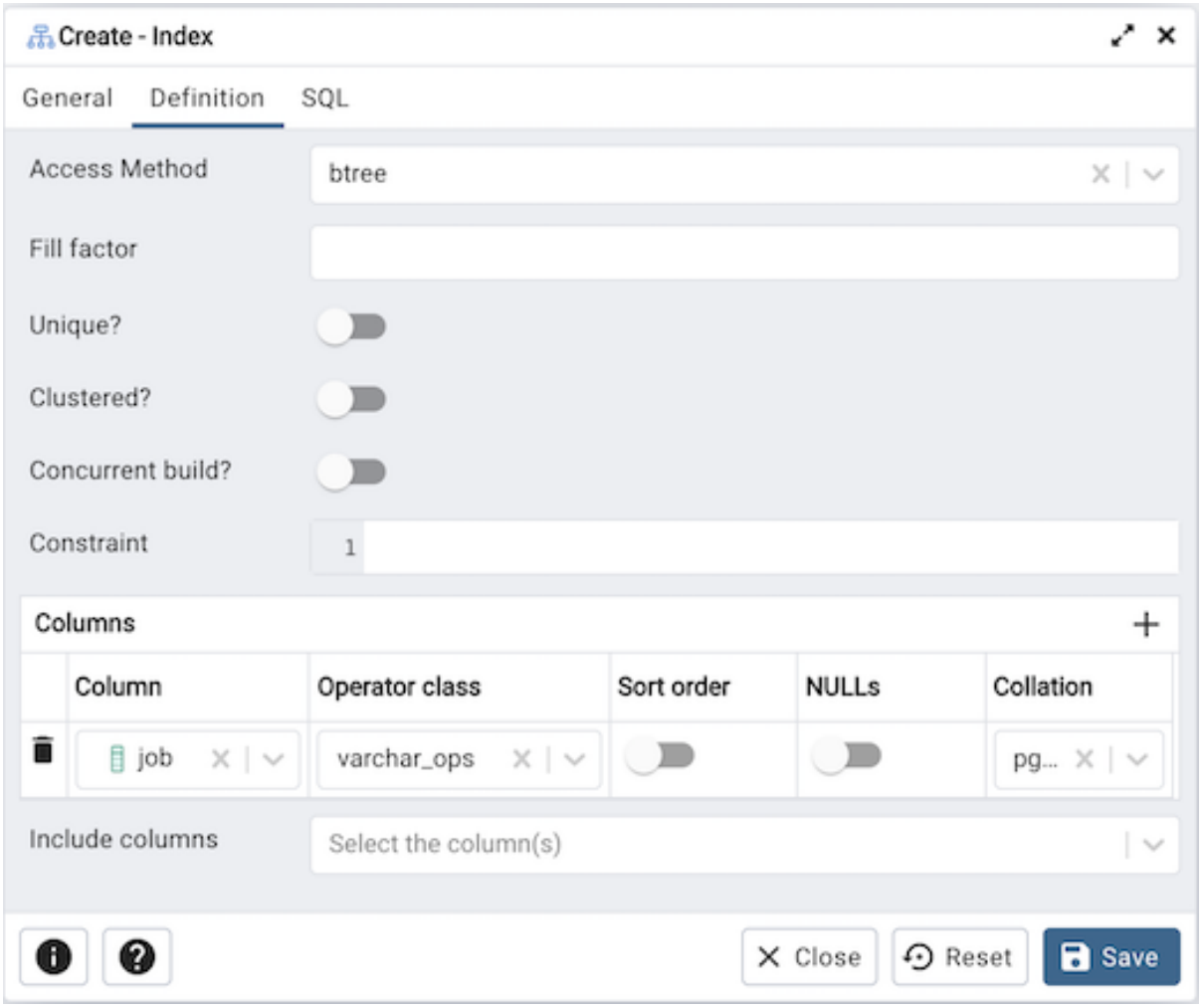
インデックス ダイアログは、次のダイアログタブを介してインデックスの開発を編成します。 **一般** と **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、インデックスを識別します。

- *Name* フィールドを使用して、インデックスの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- *Tablespace* の横にあるドロップダウンリストボックスを使用して、インデックスが存在するテーブルスペースを選択します。
- インデックスに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、インデックスを定義します。

- **アクセス方法** の横にあるドロップダウンリストボックスを使用して、インデックスの種類を選択します。
  - *btree* を選択して B ツリーインデックスを作成します。B ツリーインデックスを使用すると、何らかの順序で並べ替えることができるデータの等値クエリと範囲クエリを管理するときのパフォーマンスが向上する場合があります。
  - *hash* を選択して、ハッシュインデックスを作成します。ハッシュインデックスを使用すると、単純な等値比較を管理する際のパフォーマンスが向上する場合があります。
  - *gist* を選択して GiST インデックスを作成します。GiST インデックスは、2 次元の幾何学的データ型と最近傍検索を管理する際のパフォーマンスを向上させる場合があります。
  - *gin* を選択して GIN インデックスを作成します。GIN インデックスは、複数のキーを持つ値を管理する際のパフォーマンスを向上させる場合があります。
  - *spgist* を選択して、領域分割された GiST インデックスを作成します。SP-GiST インデックスは、バランスの取れていないデータ構造を管理する際のパフォーマンスを向上させる場合があります。
  - *brin* を選択して、BRIN インデックスを作成します。BRIN インデックスを使用すると、最小値と最大値と範囲を管理する際のパフォーマンスが向上する場合があります。

- *FillFactor* フィールドを使用して、インデックスの FILLFACTOR を指定します。FILLFACTOR は、選択したメソッドが各インデックスページをどの程度埋めようとするかを指定します。
- *Unique?* スイッチを *Yes* の位置に移動して、インデックスの作成時とデータの追加時にテーブル内の値が重複していないかどうかを確認します。デフォルトは **いいえ** です。
- *Clustered?* スイッチを *Yes* の位置に移動して、サーバーにテーブルをクラスター化するように指示します。
- **コンカレントビルド***?* スイッチを *Yes* の位置に移動して、テーブルでの同時挿入、更新、または削除を妨げるロックを取得せずにインデックスを構築します。
- *Constraint* フィールドを使用して、制約式を提供します。制約式は、インデックス内のエントリを制約を満たす行に制限します。

[列] パネルの状況依存フィールドを使用して、インデックスが照会する列を指定します。 **追加** アイコン（+）をクリックして列を追加します。

- *Column* フィールドのドロップダウンリストボックスを使用して、テーブルの列の名前を選択します。
- 有効になっている場合、ドロップダウンリストボックスを使用して使用可能な **演算子クラス** を選択し、列で実行されるアクションのタイプを指定します。
- 有効になっている場合は、 **並べ替え** スイッチを動かして並べ替え順序を指定します。
  - *ASC* を選択して、昇順の並べ替えを指定します（デフォルト）。
  - *DESC* を選択して、降順の並べ替えを指定します。
- 有効になっている場合は、 *Nulls* スイッチを動かして null の並べ替え順序を指定します。
  - null 以外の前に null を並べ替えるには、 *First* を選択します。
  - [**最後**] を選択して、ヌル以外の後でヌルをソートするように指定します（デフォルト）。
- *Collation* フィールドのドロップダウンリストボックスを使用して、インデックスに使用する照合を選択します。

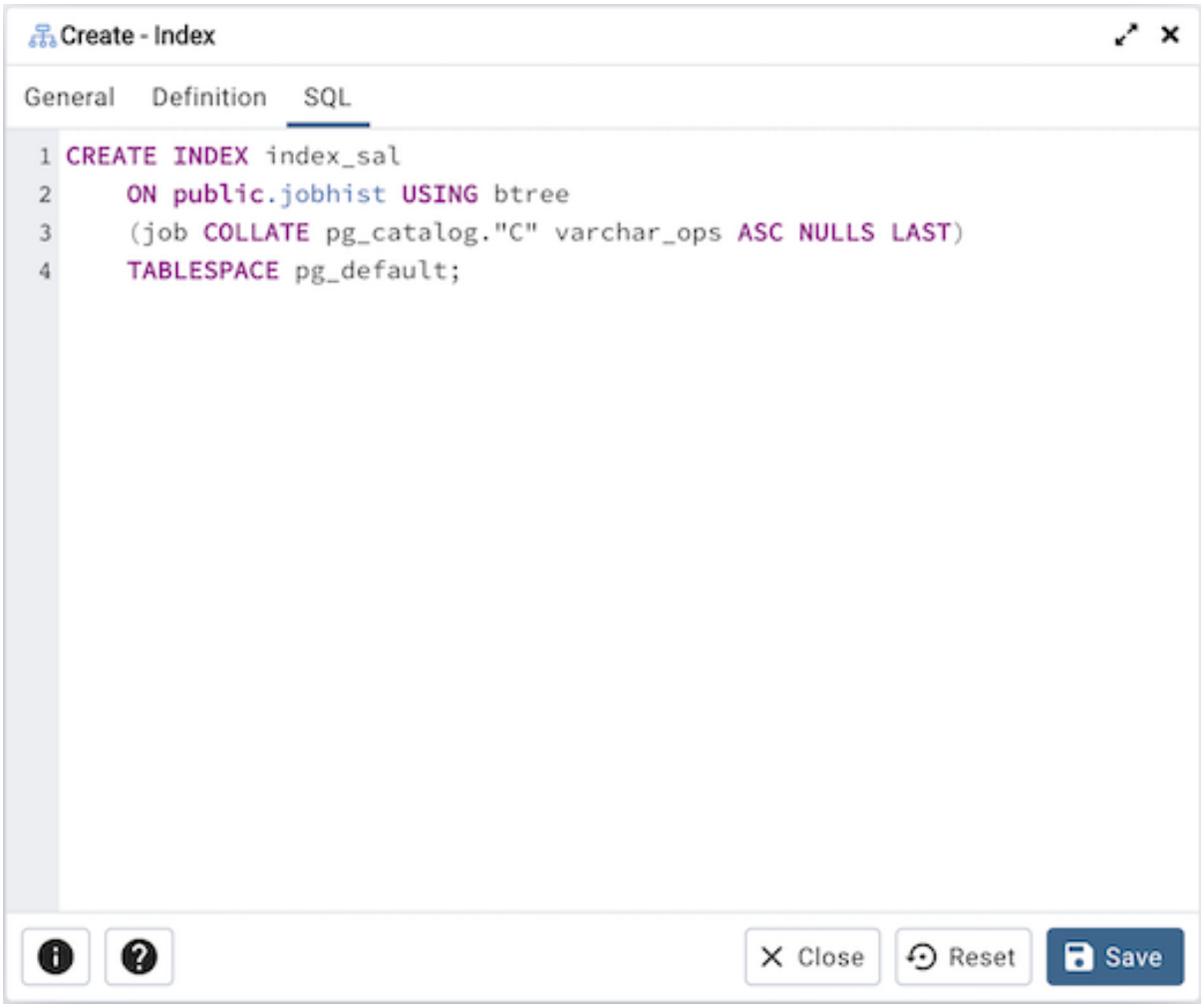
Includecolumns フィールドを使用して、インデックスの *INCLUDE* 句の列を指定します。このオプションは Postgres11 以降で使用できます。

SQL タブをクリックして続行します。

Index ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.6.1 例

以下は、 *Index* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、 *jobhist* テーブルの *job* 列の値をインデックス化する *index_sal* という名前のインデックスを作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.7 Primary key Dialog

主キー ダイアログを使用して、主キー制約を作成または変更します。主キー制約は、列または列のグループがテーブル内の行を一意に識別することを示します。これには、選択した列の値が一意であり、null でない必要があります。

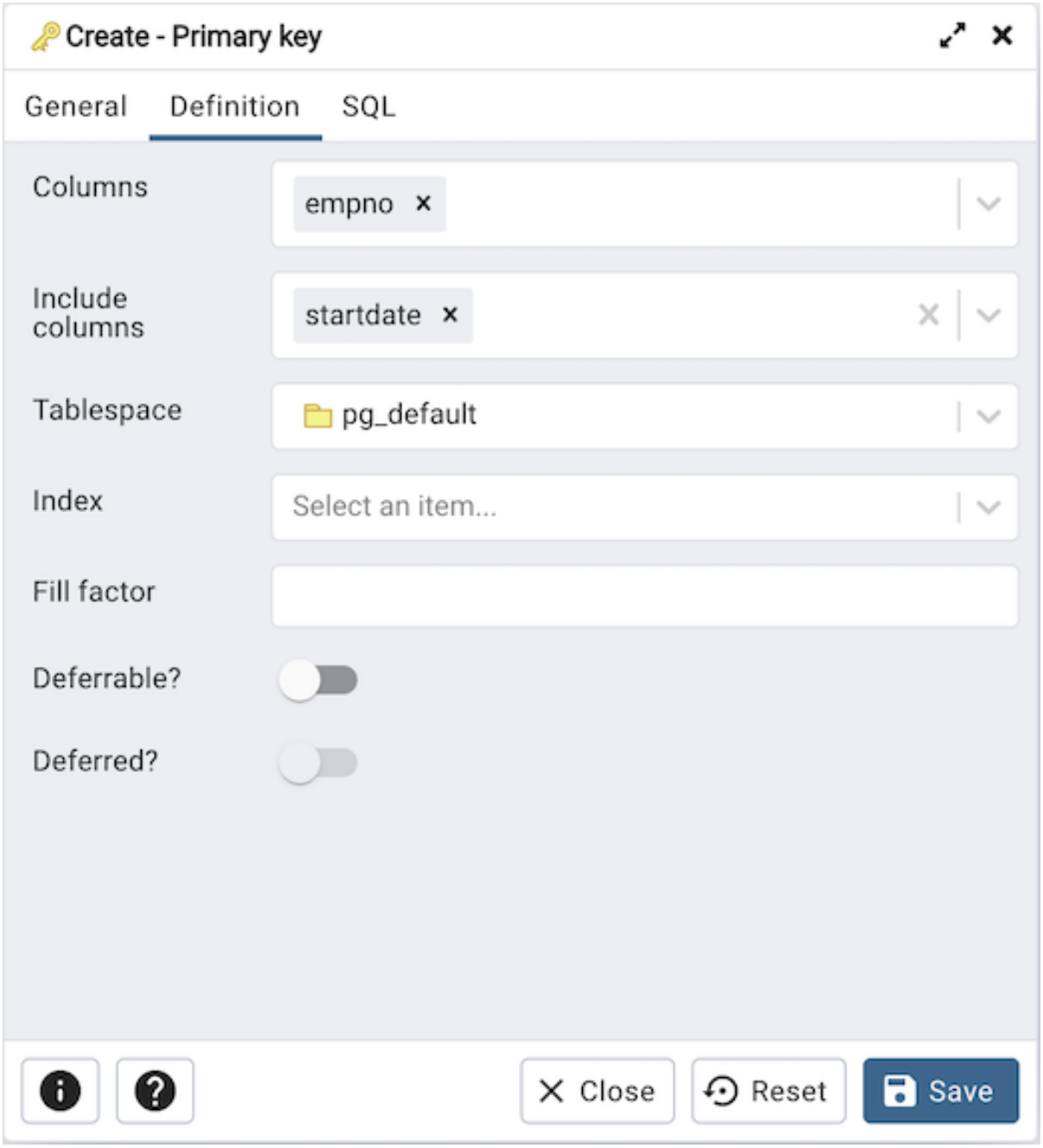
主キー ダイアログは、**一般** および **定義** タブを介して主キー制約の開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、主キーを識別します。

- *Name* フィールドを使用して、主キー制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、主キー制約を定義します。

- *Columns* フィールド内をクリックし、ドロップダウンリストボックスから 1 つ以上の列名を選択します。選択を削除するには、列名の左側にある *x* をクリックします。主キー制約は、同じテーブルに対して定義された一意の制約とは異なる必要があります。制約用に選択された列は個別である必要があります。
- *Includecolumns* フィールドを使用して、インデックスの *INCLUDE* 句の列を指定します。このオプションは Postgres11 以降で使用できます。
- *Tablespace* フィールドのドロップダウンリストボックスから、主キー制約が存在するテーブルスペースの名前を選択します。

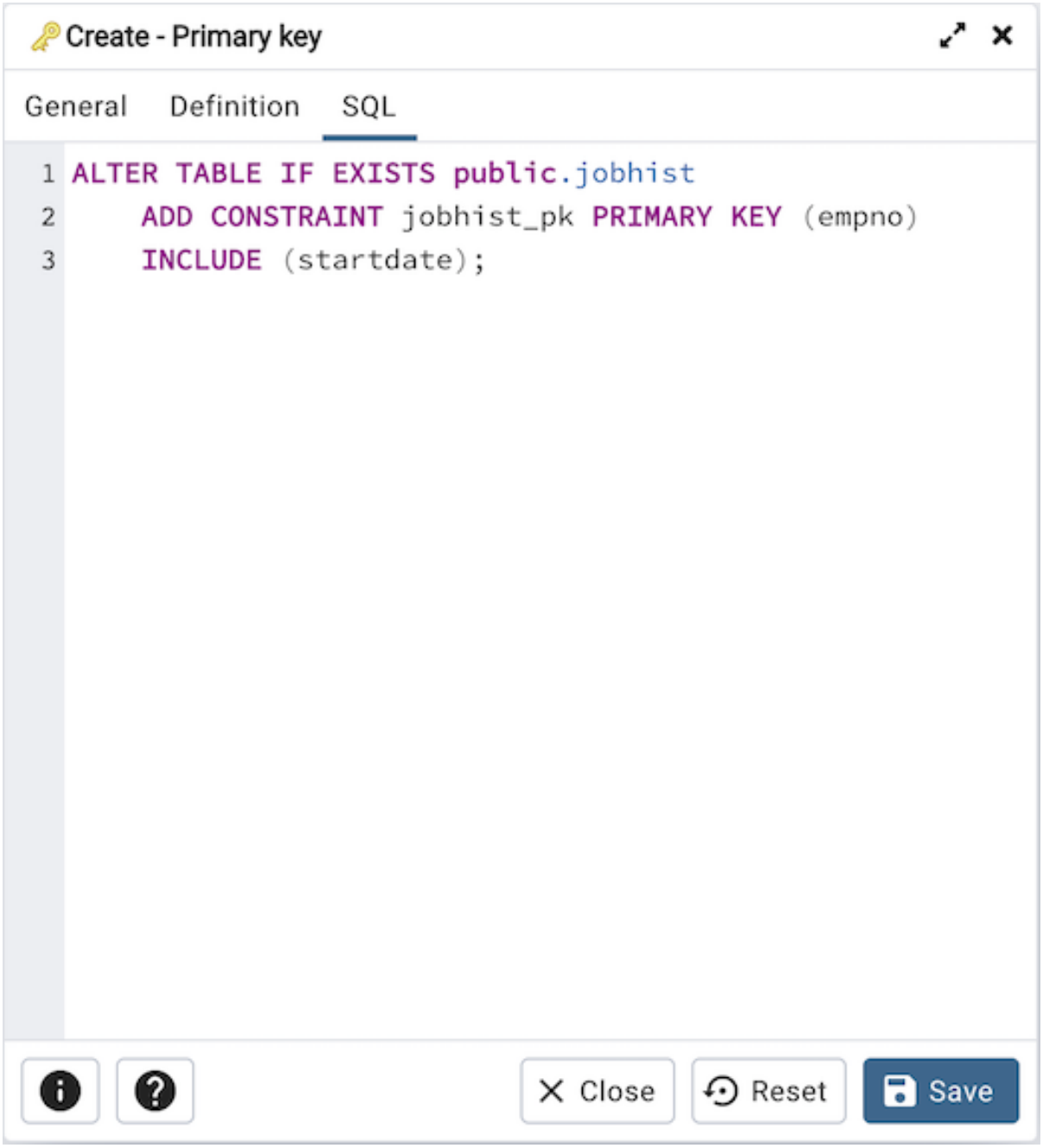
- **インデックス** フィールドのドロップダウンリストボックスからインデックスの名前を選択します。このフィールドはオプションです。主キーを追加すると、主キーにリストされている列または列のグループに一意の B ツリーインデックスが自動的に作成され、列に NOTNULL のマークが付けられます。
- *FillFactor* フィールドを使用して、テーブルとインデックスの FILLFACTOR を指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。

SQL タブをクリックして続行します。

プライマリキー ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.7.1 例

以下は、 **プライマリキー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、*jobhist* テーブルの *empno* 列に *jobhist_pk* という名前の主キー制約を作成する方法を示しています。

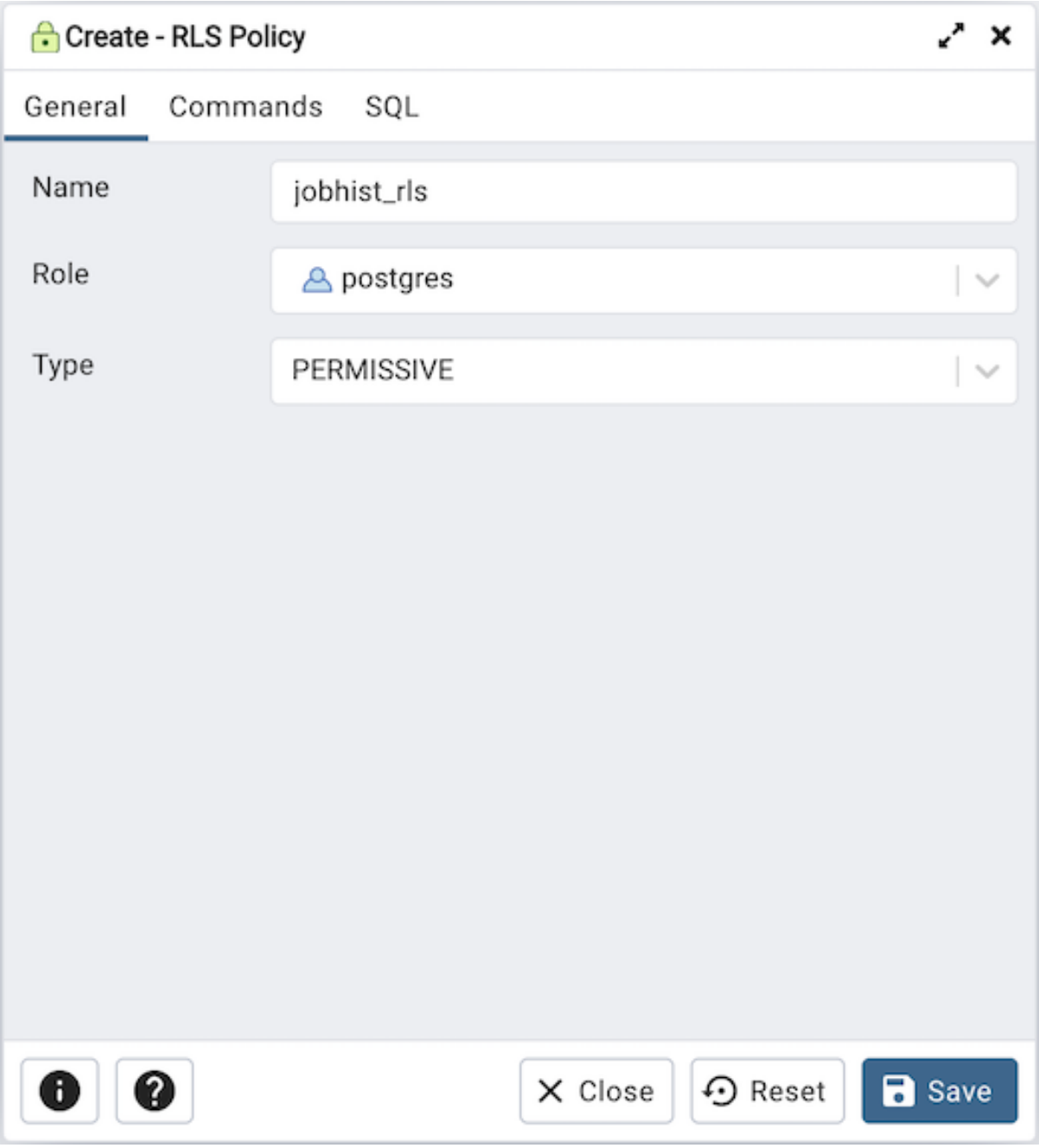
- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.8 RLS Policy Dialog

RLS **ポリシー** ダイアログを使用して、行レベルのセキュリティポリシーを作成します。

注釈: 行レベルセキュリティがテーブルレベルで有効になっていて、ポリシーが作成されていない場合、デフォルトで **拒否ポリシー** が適用されます。つまり、そのテーブルの行は表示されず、変更できません。

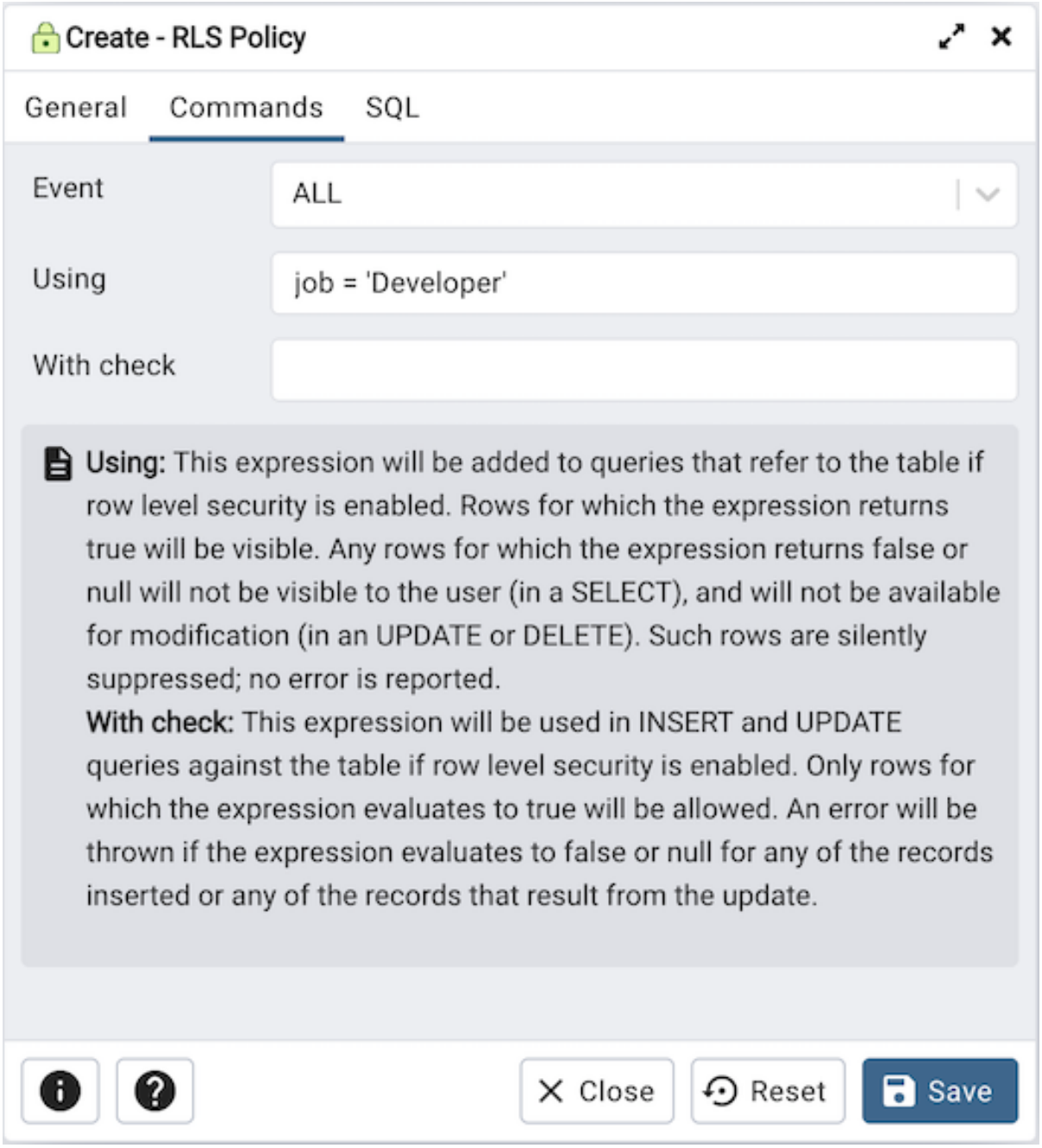
RLS **ポリシー** ダイアログは、次のダイアログタブを使用して行レベルセキュリティポリシーを作成します：**全般** および **コマンド** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、RLS ポリシーを定義します。

- *Name* フィールドを使用して、RLS ポリシーの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **ロール** の横にあるドロップダウンリストボックスを使用して、RLS ポリシーを適用するロールを選択します。
- *Type* の横にあるドロップダウンリストボックスを使用して、ポリシーの種類を選択します。

[**コマンド**] タブをクリックして続行します。



[**コマンド**] タブのフィールドを使用して、RLS ポリシーを定義します。

- **イベント** の横にあるドロップダウンリストボックスを使用して、ポリシーを適用するコマンドを選択します。有効なオプションは、ALL、SELECT、INSERT、UPDATE、および DELETE です。デフォルトは ALL です。

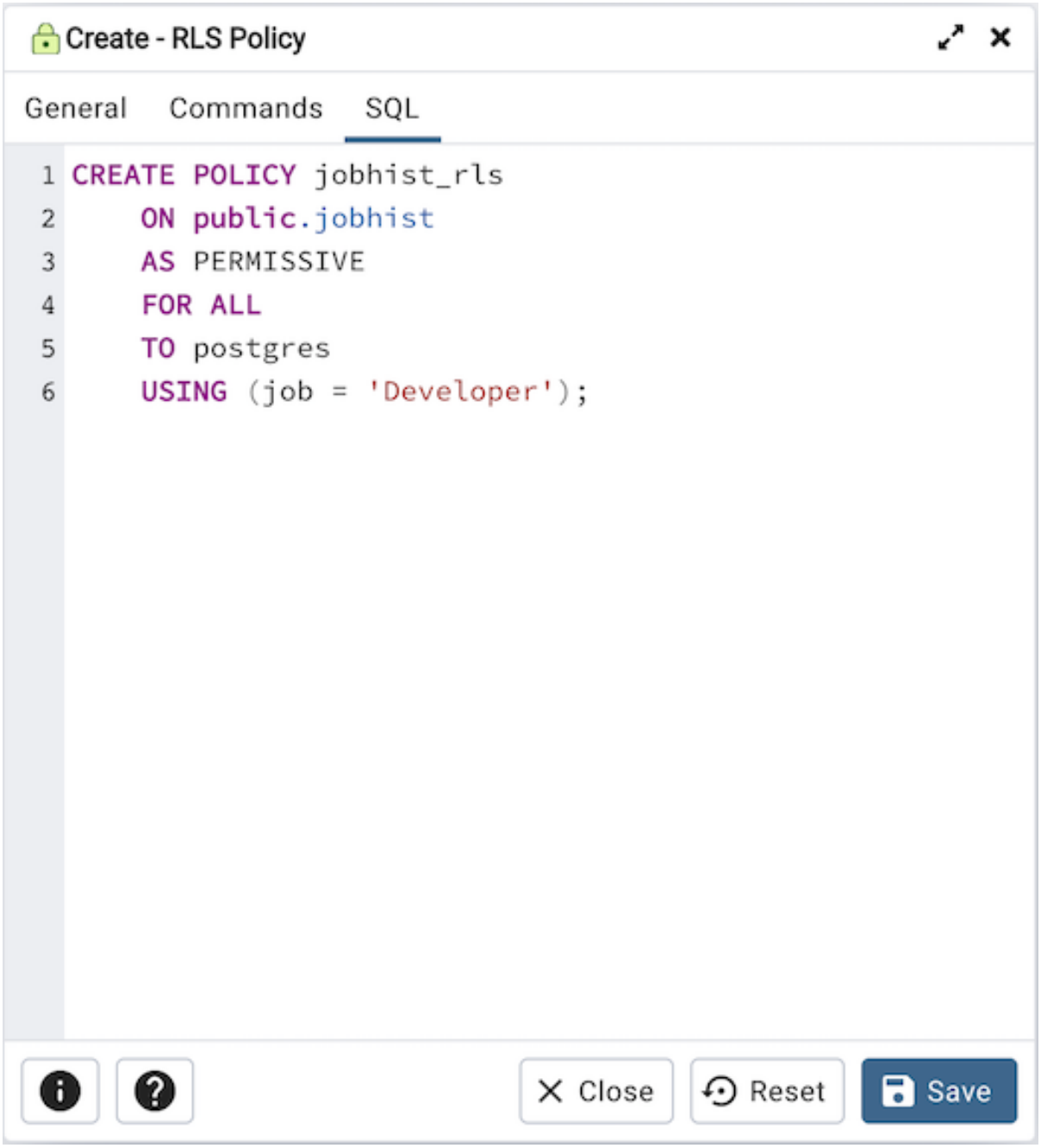
- *Using* フィールドを使用して、ブール値を返す SQL 条件式を追加します。行レベルのセキュリティが有効になっている場合、この式はテーブルを参照するクエリに追加されます。
- **チェックあり** フィールドを使用して、ブール値を返す SQL 条件式を追加します。この式は、行レベルのセキュリティが有効になっている場合、テーブルに対する INSERT および UPDATE クエリで使用されます。

SQL タブをクリックして続行します。

RLSPolicy ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.8.1 例

以下は、 *RLSPolicy* ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、 *jobhist* テーブルに行レベルセキュリティを適用する *jobhist_rls* という名前の RLS ポリシーを作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.9 Rule Dialog

Rule ダイアログを使用して、指定したテーブルまたはビューのルールを定義または変更します。PostgreSQL ルールを使用すると、テーブルに対して SELECT、INSERT、UPDATE、または DELETE が実行されたときに実行される追加のアクションを定義できます。

ルール ダイアログは、 **一般** 、 **定義** 、 **条件** 、 **コマンド** タブを介してルールの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、ルールを識別します。

- *Name* フィールドを使用して、ルールの説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。同じテーブルに対する複数のルールは、名前のアルファベット順に適用されます。

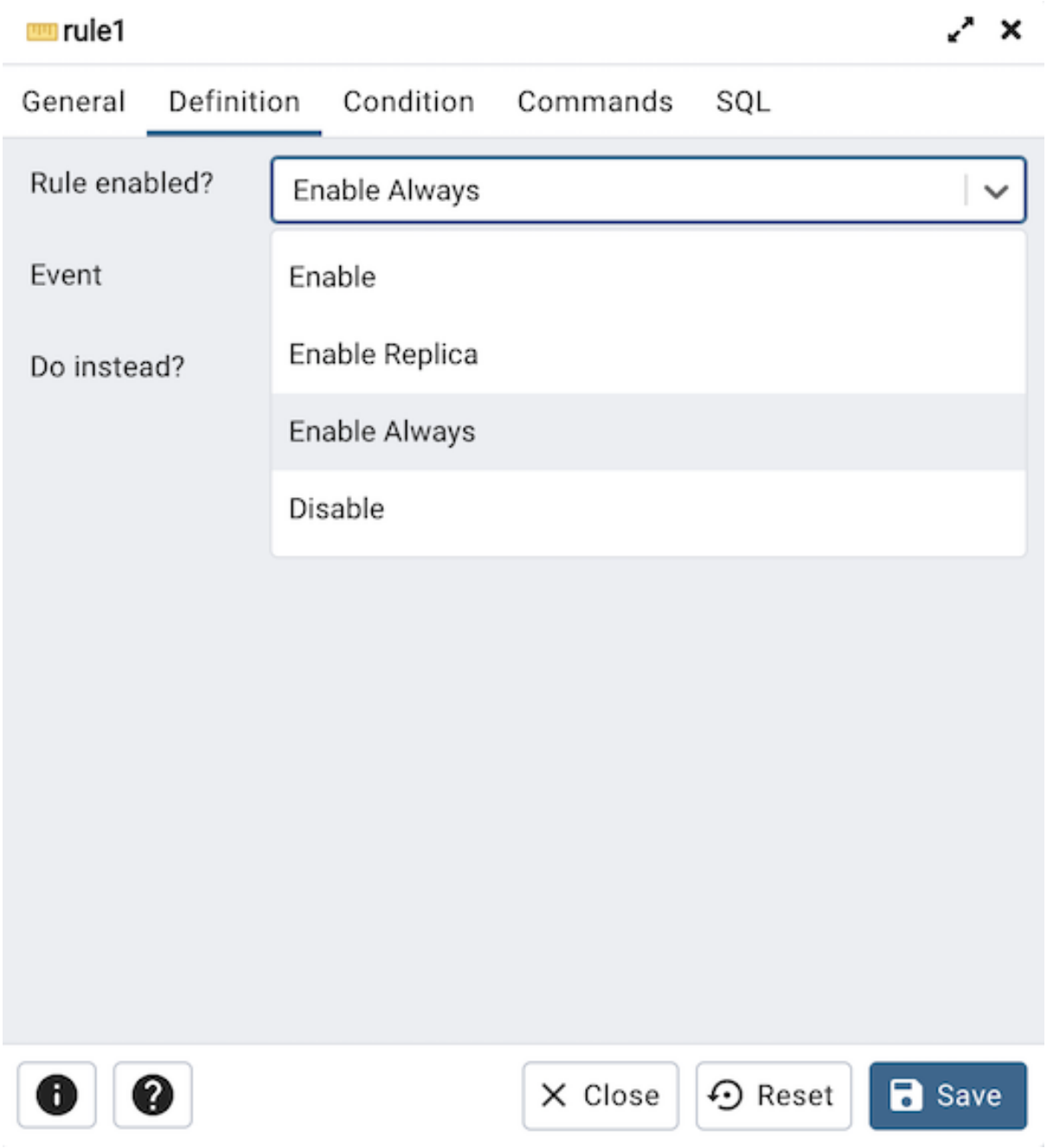
- ルールに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



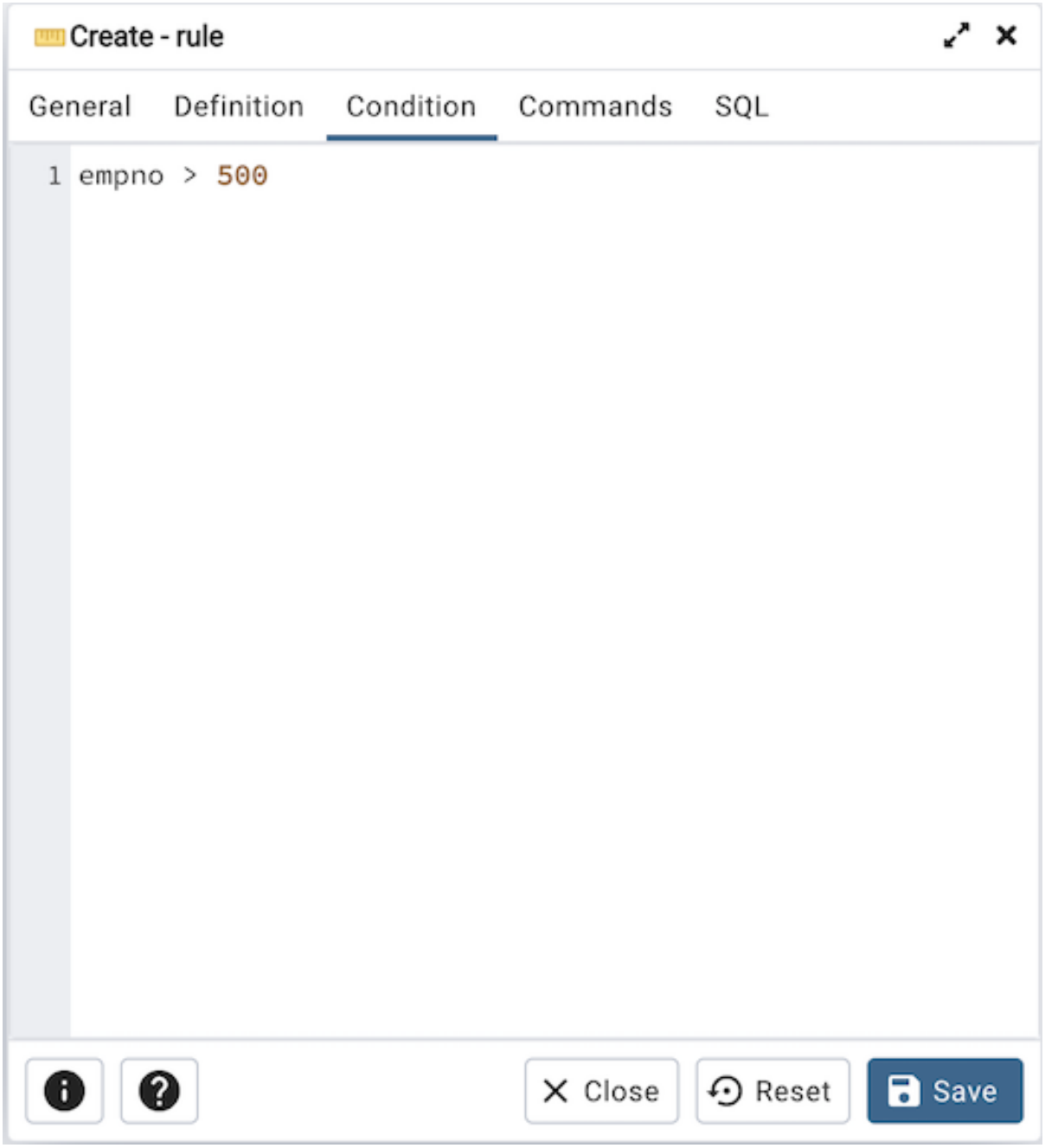
[**定義**] タブのフィールドを使用して、パラメーターを書き込みます。

- *Event* フィールド内をクリックして、ルールを呼び出すイベントのタイプを選択します。イベントは *Select* 、 *Insert* 、 *Update* 、または *Delete* です。
- **代わりに** スイッチを **はい** に移動し、元のコマンドの代わりにコマンドを実行することを示します。Do 代わりに *No* を指定すると、元のコマンドに加えてルールが呼び出されます。



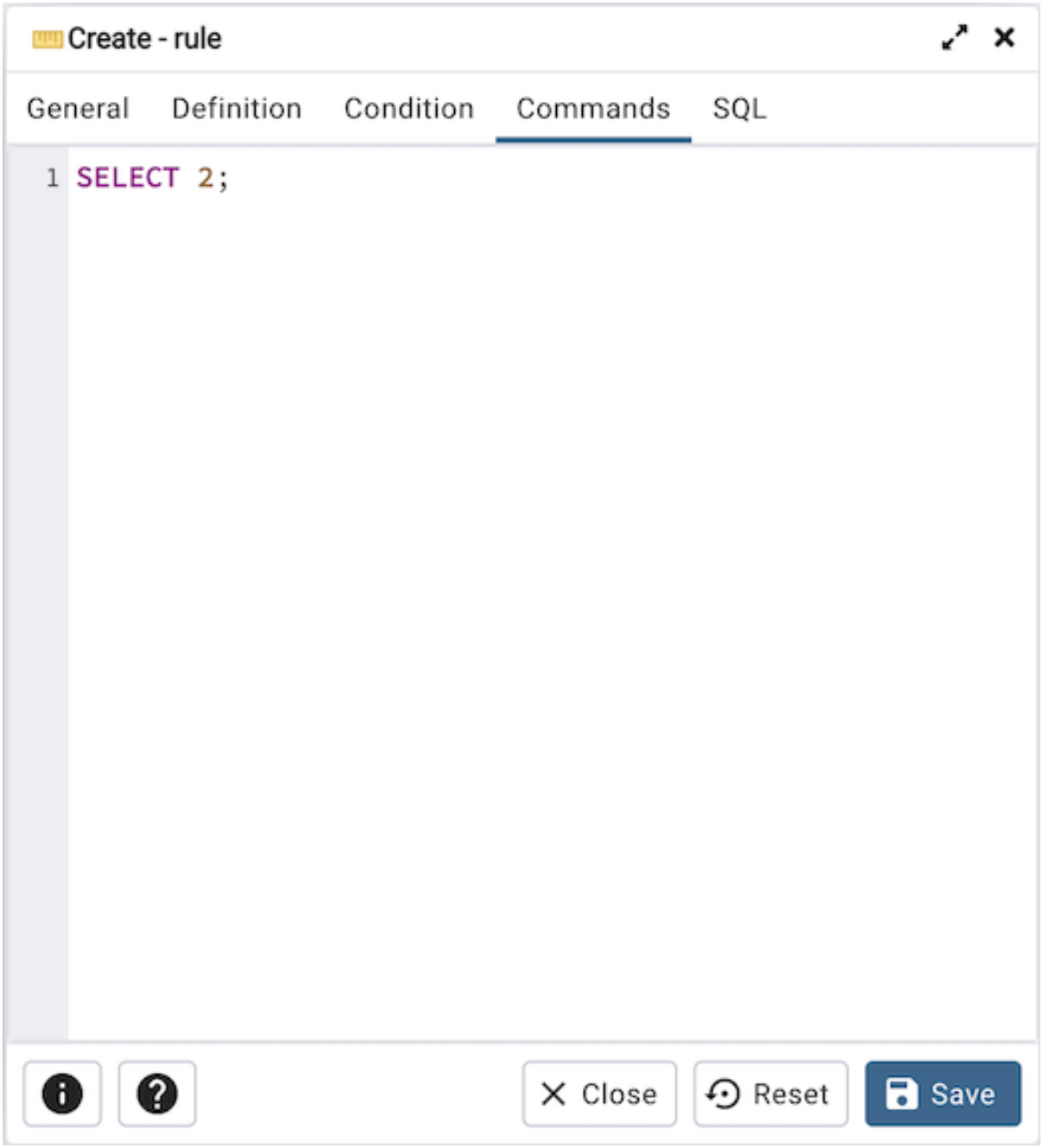
- ルールが作成されると、ルールダイアログで **ルール有効** フィールドが使用可能になります。4 つのオプションから 1 つを選択できます。

[**条件**] タブをクリックして続行します。



エディターでブール値を返す SQL 条件式を指定します。

[**コマンド**] タブをクリックして続行します。



ルールによって実行されるアクションを定義するコマンドをエディターで提供します。

SQL タブをクリックして続行します。

ルール ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.9.1 例

以下は、[**ルール**] ダイアログでのユーザー選択によって生成される sql コマンドの例です。

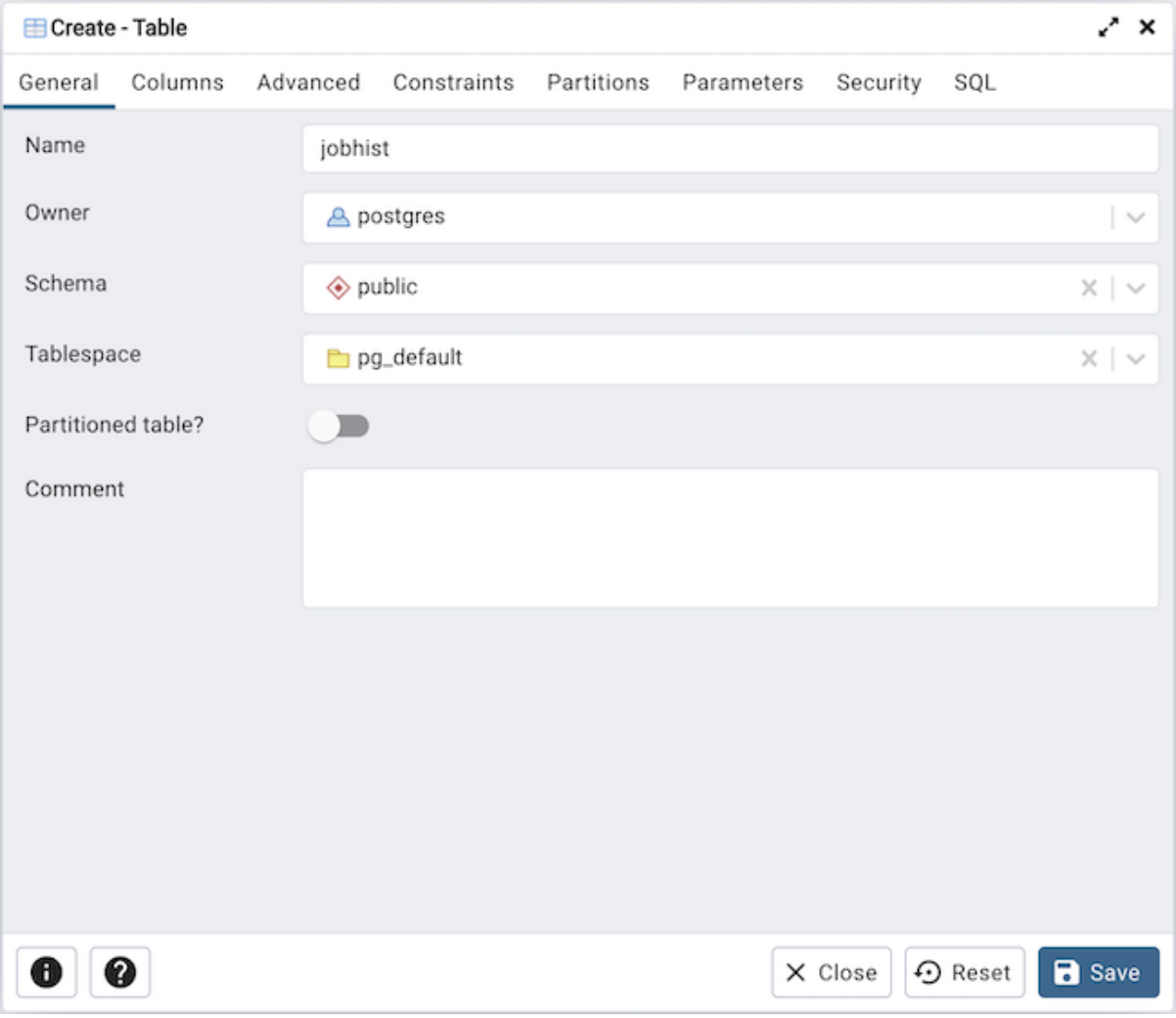


- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.10 Table Dialog

テーブル ダイアログを使用して、テーブルを作成または変更します。

テーブル ダイアログは、 **一般** 、 **列** 、 **制約** 、 **詳細設定** 、 **パーティション** 、 **パラメーター** 、および **セキュリティ** のダイアログタブを使用してテーブルの開発を編成します。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。

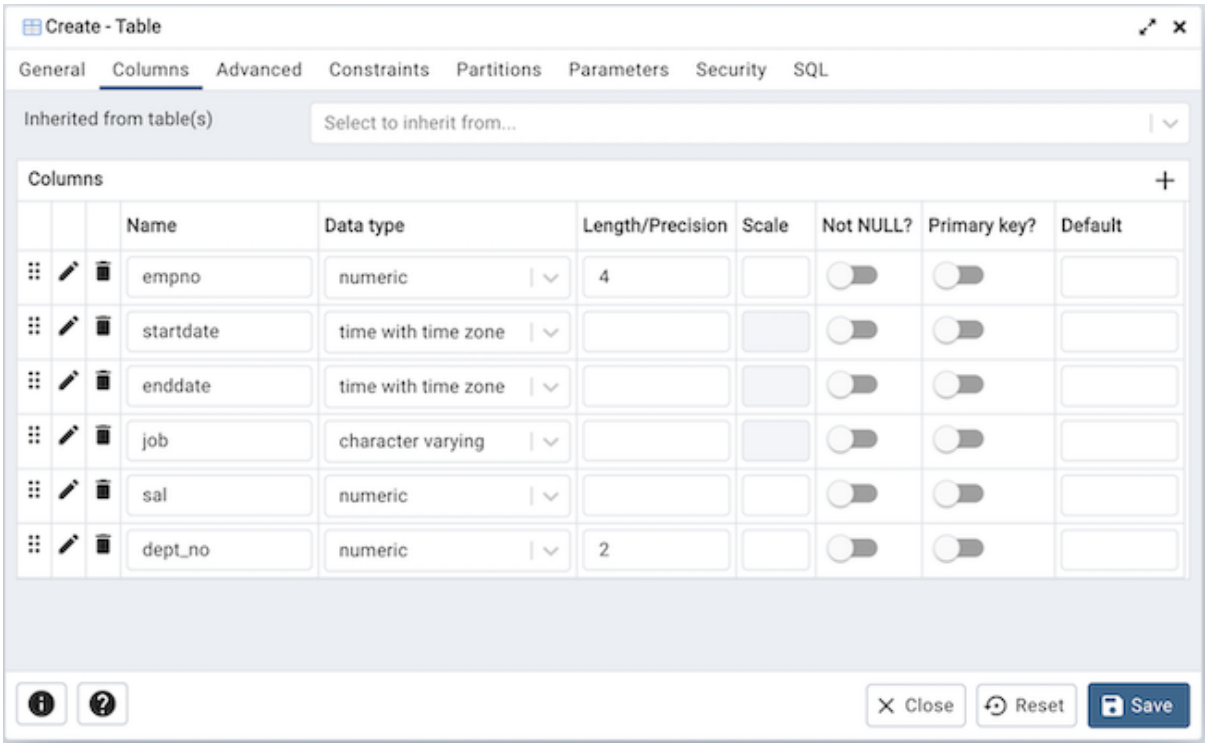


[**全般**] タブのフィールドを使用して、テーブルを識別します。

- *Name* フィールドを使用して、テーブルに説明的な名前を追加します。テーブルは、同じスキーマ内の既存のテーブル、シーケンス、インデックス、ビュー、外部テーブル、またはデータ型と同じにすることはできません。指定した名前は *pgAdmin* ツリーコントロールに表示されます。このフィールドは必須です。
- **所有者** フィールドのドロップダウンリストボックスからテーブルの所有者を選択します。デフォルトでは、テーブルの所有者はテーブルを作成するロールです。
- **スキーマ** フィールドのドロップダウンリストボックスから、テーブルを配置するスキーマの名前を選択します。
- *Tablespace* フィールドのドロップダウンリストボックスを使用して、テーブルが保存されるテーブルスペースを指定します。

- パーティションテーブルを作成する場合は、 **パーティションテーブル***?* スイッチを **はい** に移動します。オプションは PostgreSQL10 以降で利用できます。
- テーブルに関するメモを **コメント** フィールドに保存します。

[**列**] タブをクリックして続行します。



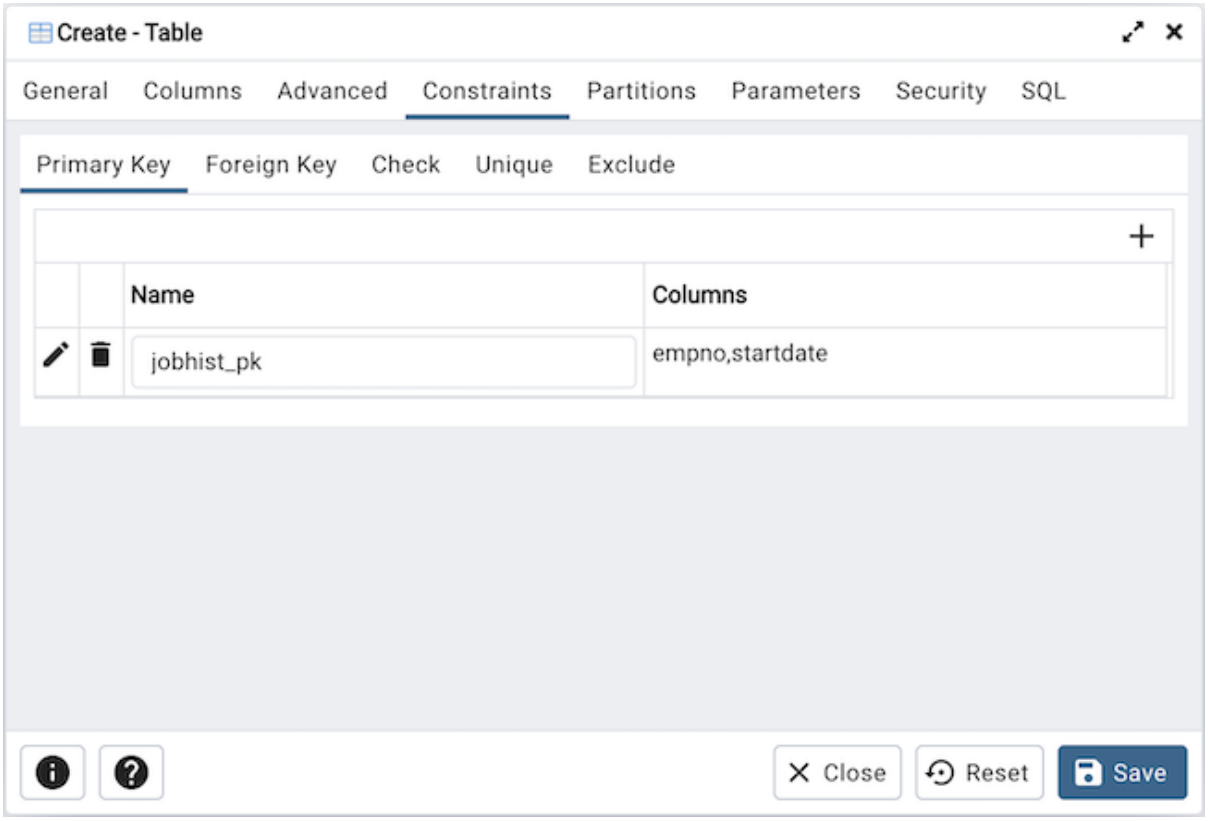
テーブルから継承 の横にあるドロップダウンリストボックスを使用して、親テーブルを指定します。テーブルは選択した親テーブルから列を継承します。 *Inheritedfromtable(s)* フィールド内をクリックして、ドロップダウンリストからテーブル名を選択します。を繰り返して、他の親テーブルを追加します。親名の左側にある *x* をクリックして、選択したテーブルを削除します。継承された列名とデータ型は現在のダイアログでは編集できないことに注意してください。これらは親レベルで変更する必要があります。

追加 アイコン（+）をクリックして、 **列** テーブルの列の名前とそのデータ型を指定します。

- *Name* フィールドを使用して、列にわかりやすい名前を追加します。
- [**データ型**] フィールドのドロップダウンリストボックスを使用して、列のデータ型を選択します。これには配列指定子を含めることができます。PostgreSQL でサポートされているデータ型の詳細については、コアドキュメントの 8 章を参照してください。
- 有効になっている場合、 **長さ/精度** および **スケール** フィールドを使用して、数値の最大有効桁数、またはテキスト値の最大文字数を指定します。
- 列フィールドに値を要求するには、 *NotNULL?* スイッチを *Yes* の位置に移動します。
- **主キー***?* スイッチを **はい** の位置に移動して、列が主キー制約であることを指定します。

追加 アイコン（+）をクリックして、列を追加します。列を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[*Constraints*] タブをクリックして続行します。



[**制約**] タブのフィールドを使用して、テーブルまたは列の制約を指定します。オプションの制約句は、*INSERT* または *UPDATE* 操作を成功させるために新しい行または更新された行が満たす必要がある制約（テスト）を指定します。[*Constraints*] パネルで次のいずれかのタブを選択して、適切なコンストレインタイプを選択します。

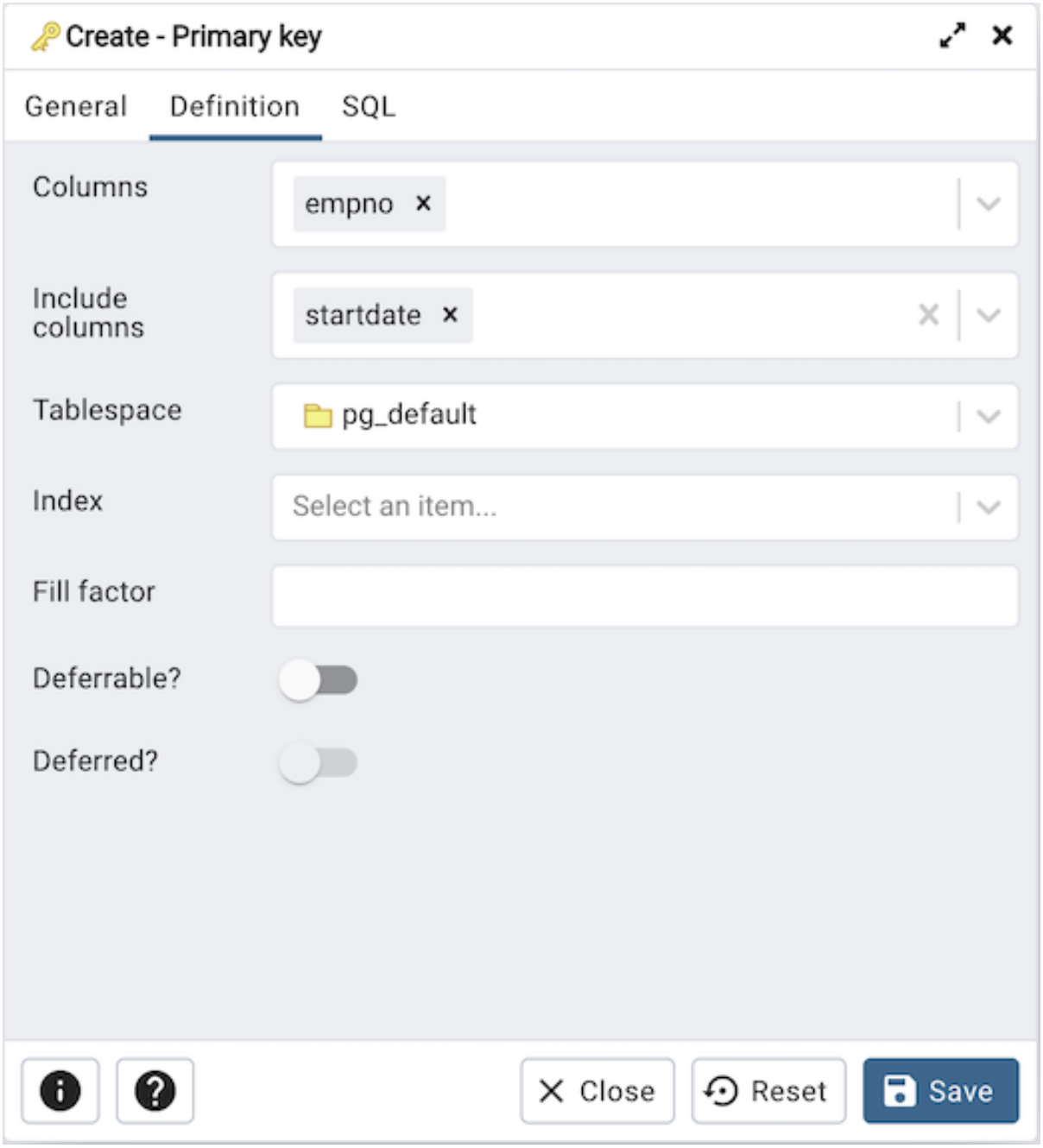
タブ名	制約
プライマリキー	テーブルの各行に一意の識別子を提供します。
外部キー	2 つのテーブル間の参照整合性を維持します。
チェック	挿入または変更する前に、データが式または条件を満たす必要があります。
ユニーク	列または列のグループに含まれるデータが、テーブル内のすべての行で一意であることを保証します。
除外	指定された列または式で（指定された演算子を使用して）2 つの行を比較した場合、少なくとも 1 つの演算子の比較が false または null を返すことを保証します。

テーブルの主キーを追加するには、[**主キー**] タブを選択し、 **追加** アイコン（+）をクリックします。プライマリキーを定義するには、 **ゴミ箱** アイコンの左側にある **編集** アイコンをクリックします。 **プライマリキー** ダイアログ（*pgAdmin* ツリーコントロールの *Constraints* を右クリックしてアクセス）に似たダイアログが開きます。

[**全般**] タブのフィールドを使用して、主キーを識別します。

- *Name* フィールドを使用して、主キー制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **コメント** フィールドに主キーに関するメモを入力します。

[**定義**] タブをクリックして続行します。

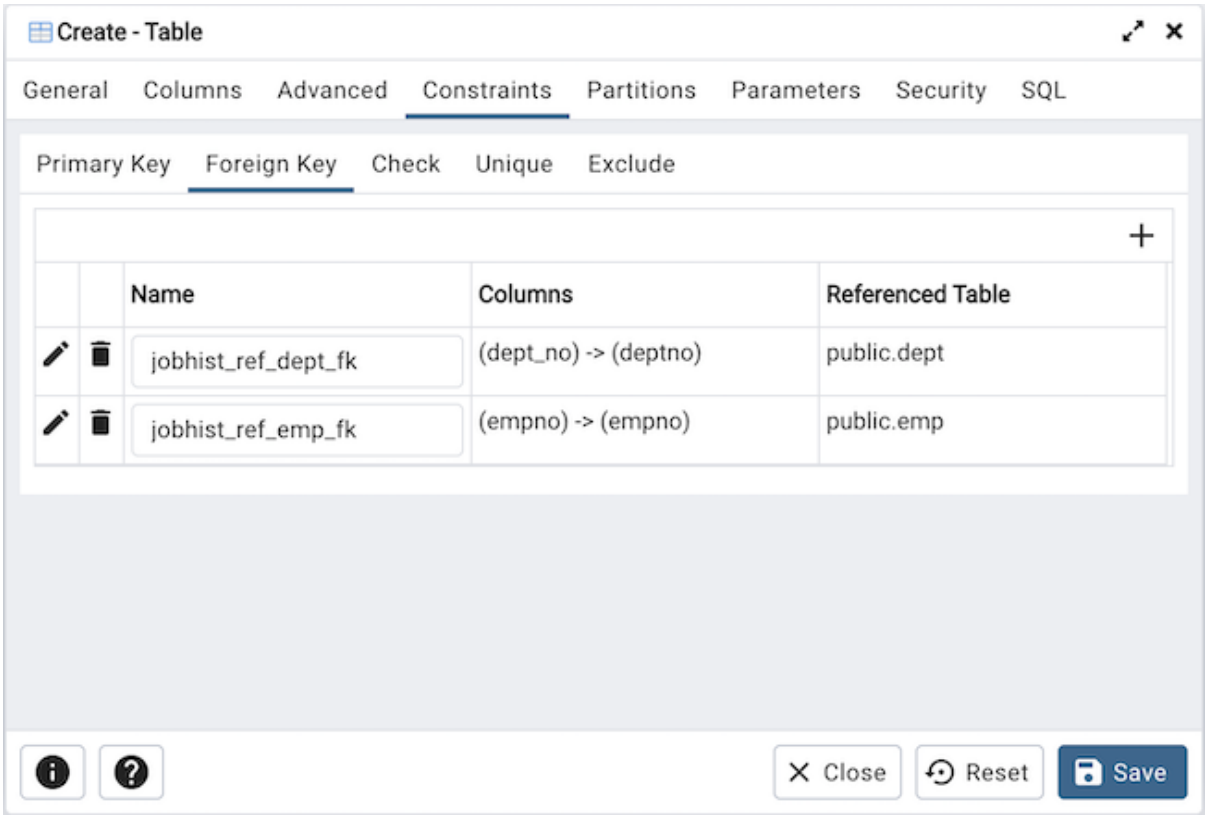


[**定義**] タブのフィールドを使用して、主キー制約を定義します。

- *Columns* フィールド内をクリックし、ドロップダウンリストボックスから 1 つ以上の列名を選択します。選択を削除するには、列名の左側にある *x* をクリックします。主キー制約は、同じテーブルに対して定義された一意の制約とは異なる必要があります。制約用に選択された列は個別である必要があります。
- *Tablespace* フィールドのドロップダウンリストボックスから、主キー制約が存在するテーブルスペースの名前を選択します。
- *FillFactor* フィールドを使用して、テーブルとインデックスの FILLFACTOR を指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで

延期できることを指定します。デフォルトは **いいえ** です。

- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。



外部キー制約を追加するには、**外部キー** タブを選択し、**追加** アイコン（+）をクリックします。制約を定義するには、**ゴミ箱** アイコンの左側にある **編集** アイコンをクリックします。 **外部キー** ダイアログ（*pgAdmin* ツリーコントロールの *Constraints* を右クリックしてアクセス）に似たダイアログが開きます。

[**全般**] タブのフィールドを使用して、外部キー制約を識別します。

- *Name* フィールドを使用して、外部キー制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **コメント** フィールドに外部キーに関するメモを入力します。

[**定義**] タブをクリックして続行します。



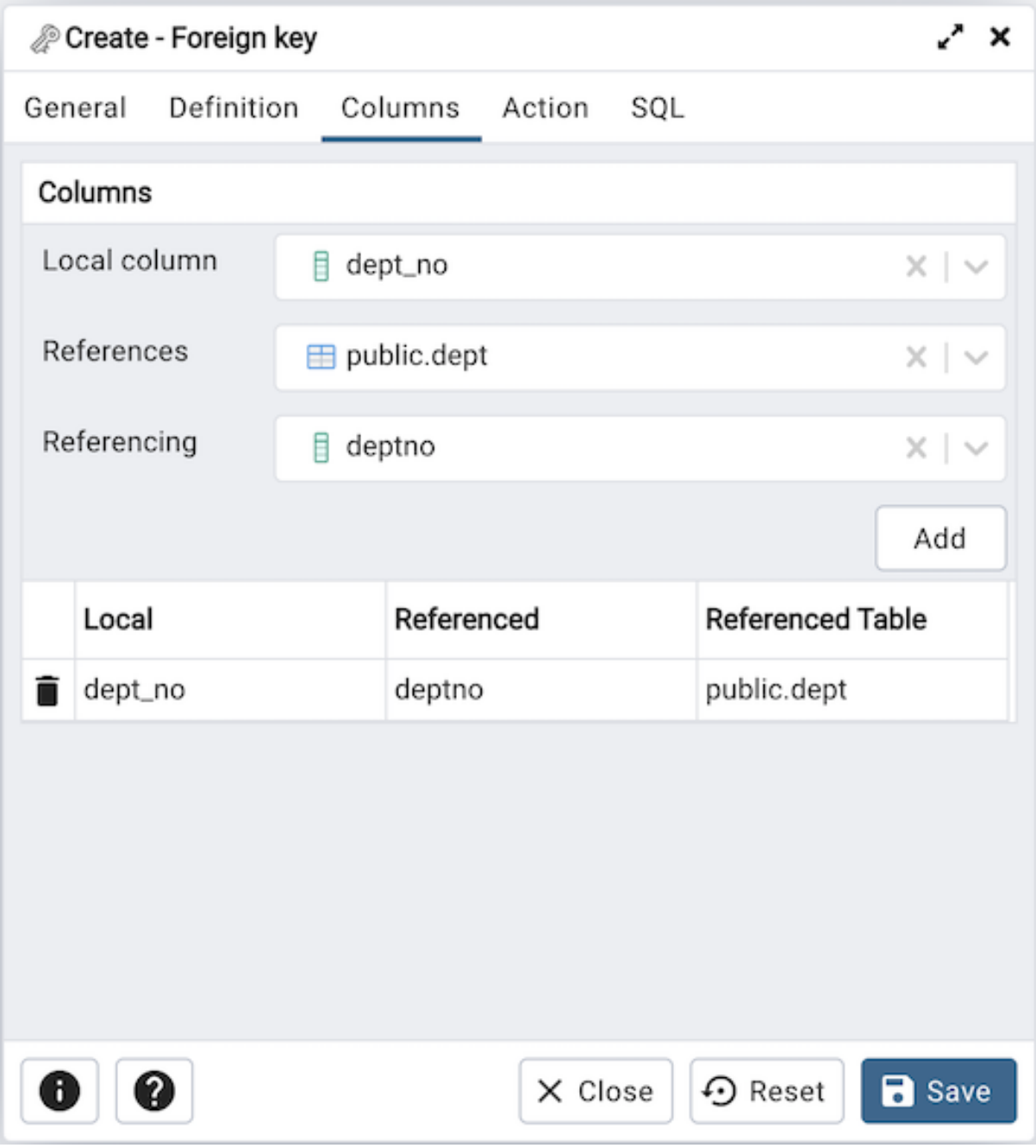
[**定義**] タブのフィールドを使用して、外部キー制約を定義します。

- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。
- *Matchtype* スイッチを移動して、制約によって適用されるマッチングのタイプを指定します。
  - いずれかの列が NULL の場合、マルチ列外部キーのすべての列を NULL にする必要があることを示すには、 *Full* を選択します。すべての列が null の場合、参照先のテーブルで行が一致する必要はありません。
  - 単一の外部キー列を null にできるように指定するには、[**シンプル**] を選択します。いずれかの

列が null の場合、参照先のテーブルで行が一致する必要はありません。

- *Validated* スイッチを *Yes* の位置に移動して、このダイアログへの変更を保存するときに既存のテーブルのコンテンツを（外部キーまたはチェック制約に対して）検証するようにサーバーに指示します。
- **自動 *FK* インデックス** スイッチを **いいえ** の位置に移動して、自動インデックス機能を無効にします。
- *AutoFKIndex* スイッチが *Yes* の位置にある場合、 *CoveringIndex* の横にあるフィールドはインデックスの名前を生成します。または、このフィールドは無効になっています。

[**列**] タブをクリックして続行します。



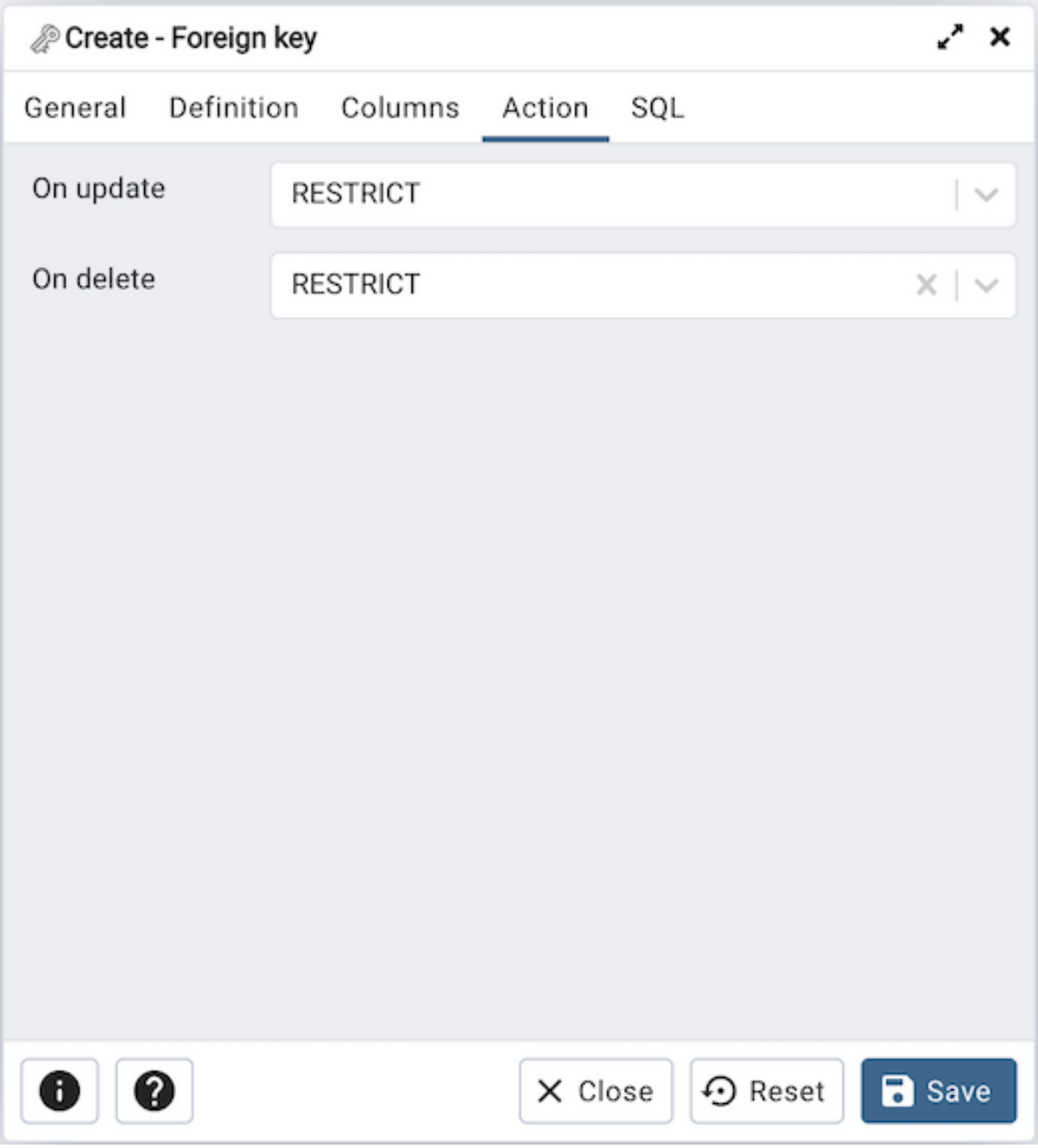
[列] タブのフィールドを使用して、1 つ以上の参照列を指定します。

外部キー制約では、テーブルの 1 つ以上の列に、参照されるテーブルの行の参照される列の値と一致する値のみが含まれている必要があります。

- **ローカル列** の横にあるドロップダウンリストボックスを使用して、外部テーブルと比較する現在のテーブルの列を指定します。
- *References* の横にあるドロップダウンリストボックスを使用して、比較列が存在するテーブルの名前を指定します。
- *Referencing* の横にあるドロップダウンリストボックスを使用して、外部テーブルの列を指定します。

追加 アイコン（+）をクリックして、リストに列を追加します。上記の手順を繰り返し、 **追加** アイコン（+）をクリックして列を追加します。エントリを破棄するには、エントリの左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

[**アクション**] タブをクリックして続行します。



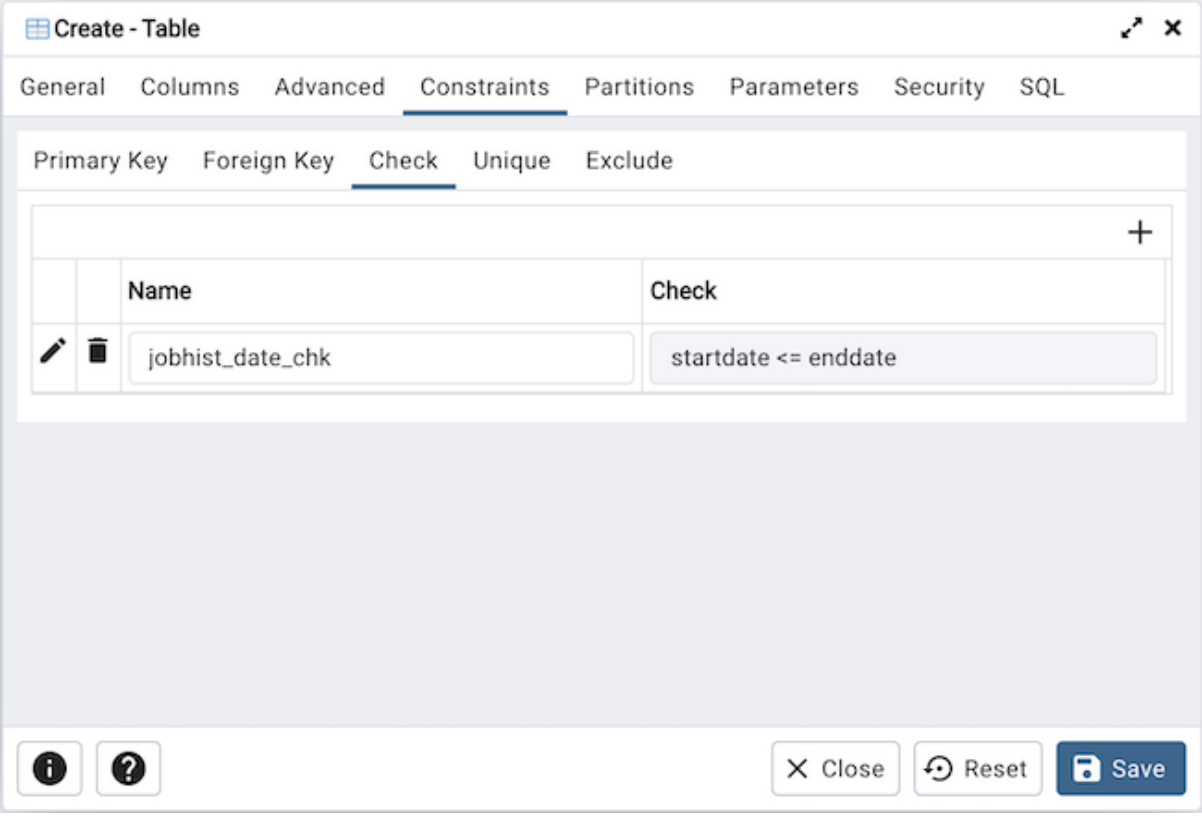
[**アクション**] タブのドロップダウンリストボックスを使用して、テーブル内のデータが更新または削除さ

れたときに実行される外部キー制約に関連する動作を指定します。

- *Onupdate* の横にあるドロップダウンリストボックスを使用して、テーブル内のデータが更新されたときに実行されるアクションを選択します。
- **削除時** の横にあるドロップダウンリストボックスを使用して、テーブル内のデータが削除されたときに実行されるアクションを選択します。

サポートされているアクションは次のとおりです。

アクションなし	削除または更新により外部キー制約違反が作成されることを示すエラーを生成します。制約が据え置かれている場合、参照する行がまだ存在すると、このエラーが制約チェック時に生成されます。これがデフォルトです。
制限	削除または更新によって外部キー制約違反が発生することを示すエラーをスローします。これは、チェックが遅延可能でないことを除いて NOACTION と同じです。
カスケード	削除した行を参照している行を削除するか、参照している列の値を参照されている列の新しい値にそれぞれ更新します。
NULL に設定	参照する列を null に設定します。
デフォルトに設定	参照する列をデフォルト値に設定します。参照されるテーブルには、デフォルト値と一致する行が必要です（null でない場合）。そうでない場合、操作は失敗します。

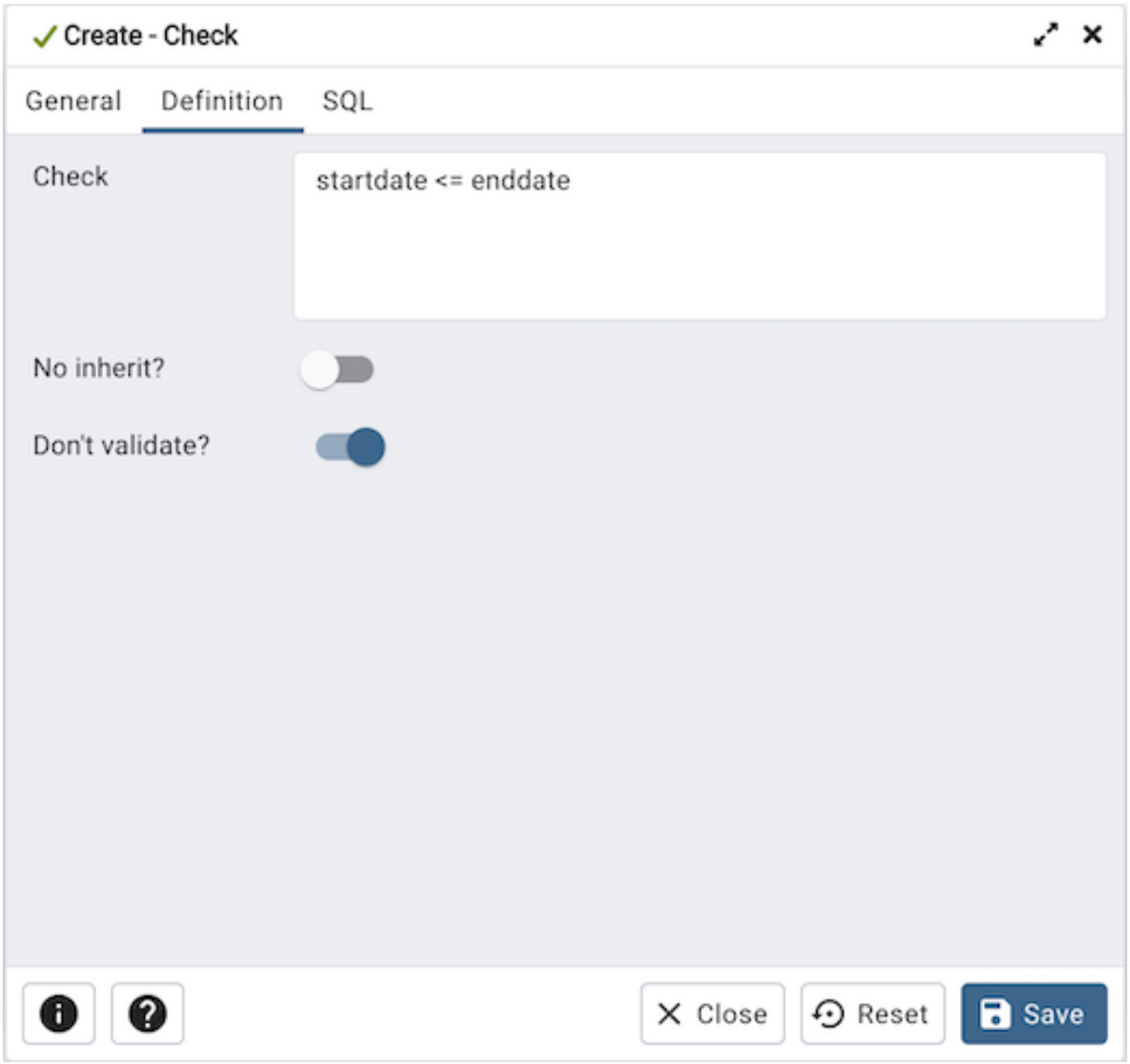


チェック制約を追加するには、パネルの *Check* タブを選択し、*Add* アイコン（+）をクリックします。チェック制約を定義するには、 **ゴミ箱** アイコンの左側にある **編集** アイコンをクリックします。 *Check* ダイアログ（ *pgAdmin* ツリーコントロールの *Constraints* を右クリックしてアクセス）に似たダイアログが開きます。

[**全般**] タブのフィールドを使用して、チェック制約を識別します。

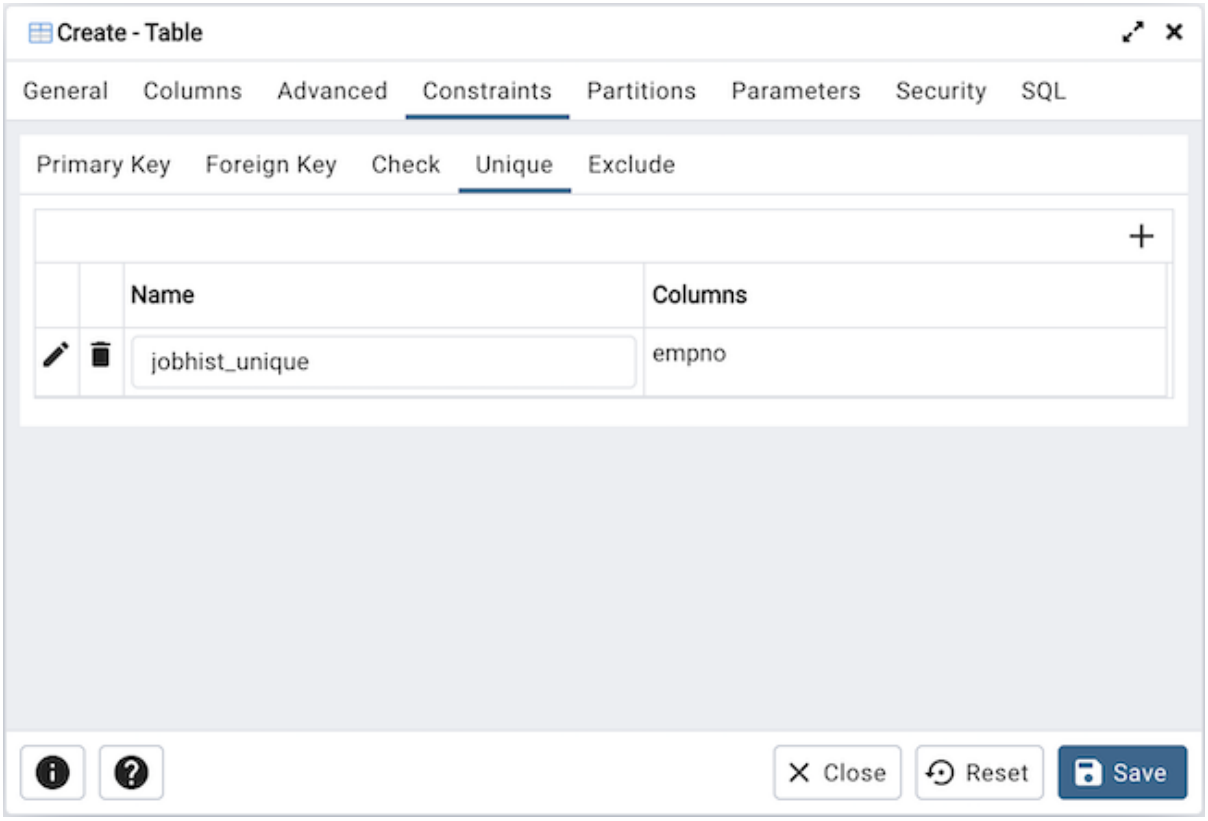
- *Name* フィールドを使用して、チェック制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。PostgreSQL9.5 以降では、テーブルに複数のチェック制約がある場合、行ごとに名前のアルファベット順で NOTNULL 制約の後にテストされます。
- **コメント** フィールドにチェック制約に関するメモを入力します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、チェック制約を定義します。

- *Check* フィールドに、行が満たさなければならない式を入力します。このフィールドは必須です。
- *NoInherit?* スイッチを *Yes* の位置に移動して、この制約がテーブルの子に自動的に継承されないように指定します。デフォルトは *No* で、制約はすべての子に継承されます。
- *Don'tvalidation?* スイッチを *No* の位置に移動して、既存のデータの検証をスキップします。制約は、テーブル内のすべての行に適用されるとは限りません。デフォルトは **はい** です。

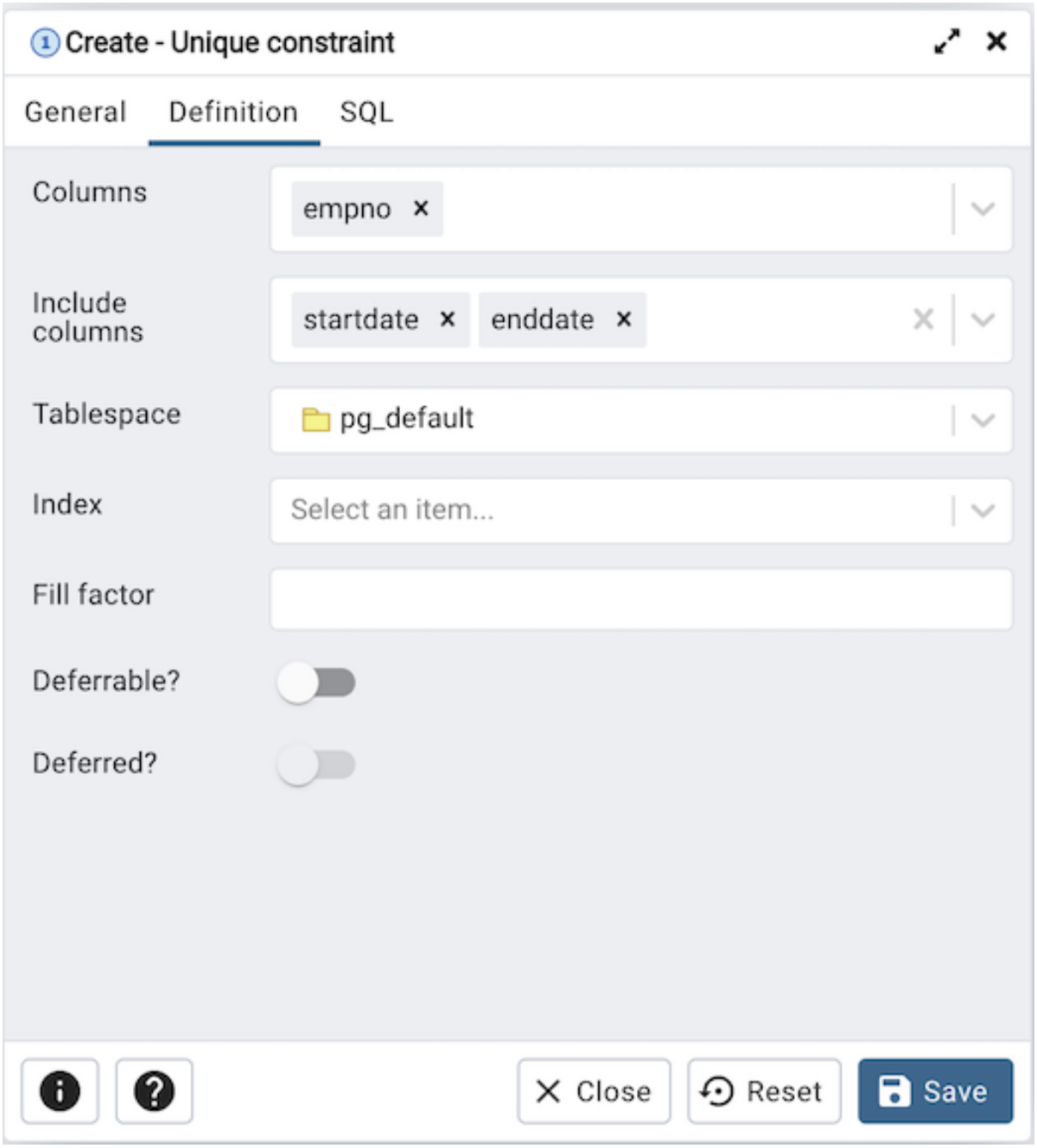


ユニーク制約を追加するには、パネルの *Unique* タブを選択し、 *Add* アイコン（+）をクリックします。制約を定義するには、 **ゴミ箱** アイコンの左側にある **編集** アイコンをクリックします。 **ユニークな制約** ダイアログ（ *pgAdmin* ツリーコントロールの *Constraints* を右クリックしてアクセス）に似たダイアログが開きます。

[**全般**] タブのフィールドを使用して、一意の制約を識別します。

- *Name* フィールドを使用して、一意性制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。
- **コメント** フィールドに一意の制約に関するメモを入力します。

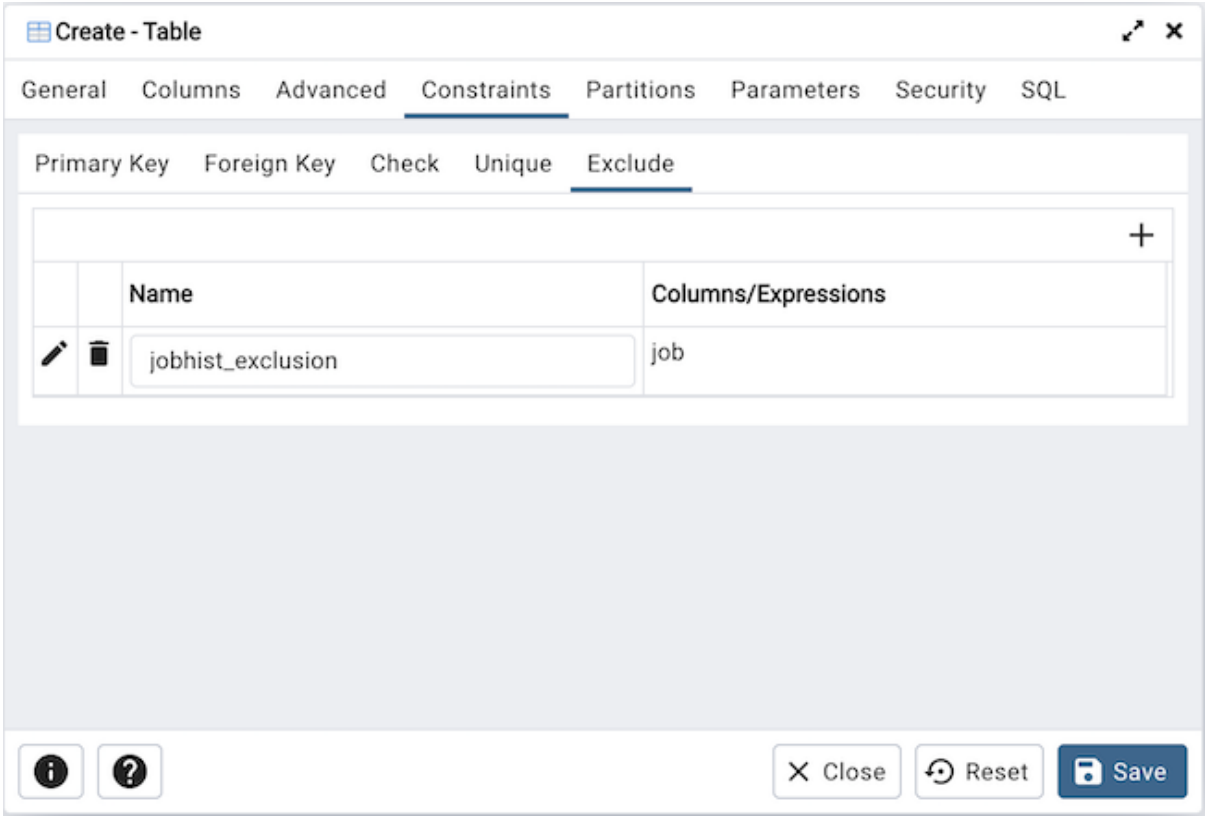
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、一意の制約を定義します。

- *Columns* フィールド内をクリックし、ドロップダウンリストボックスから 1 つ以上の列名を選択します。選択を削除するには、列名の左側にある *x* をクリックします。ユニーク制約は、同じテーブルに対して定義された主キー制約とは異なる必要があります。制約用に選択された列は個別である必要があります。
- *Tablespace* フィールドのドロップダウンリストボックスから、一意性制約が存在するテーブルスペースの名前を選択します。
- *FillFactor* フィールドを使用して、テーブルとインデックスの FILLFACTOR を指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。

- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。

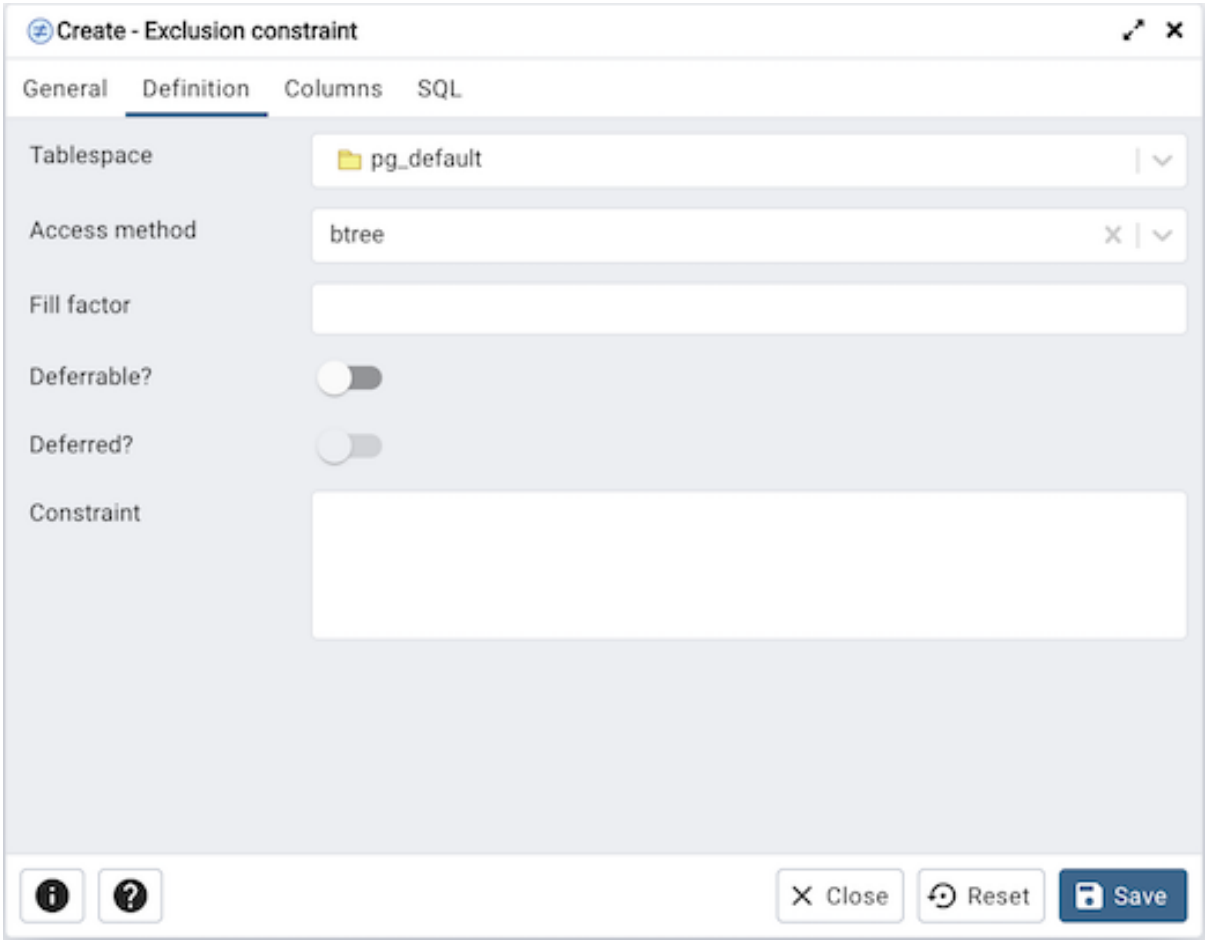


除外制約を追加するには、パネルの **除外** タブを選択し、**追加** アイコン（+）をクリックします。制約を定義するには、**ゴミ箱** アイコンの左側にある **編集** アイコンをクリックします。 **除外制約** ダイアログ（*pgAdmin* ツリーコントロールの *Constraints* を右クリックしてアクセス）に似たダイアログが開きます。

[**全般**] タブのフィールドを使用して、除外制約を識別します。

- *Name* フィールドを使用して、除外制約に説明的な名前を付けます。名前は *pgAdmin* ツリーコントロールに表示されます。
- **コメント** フィールドに除外制約に関するメモを入力します。

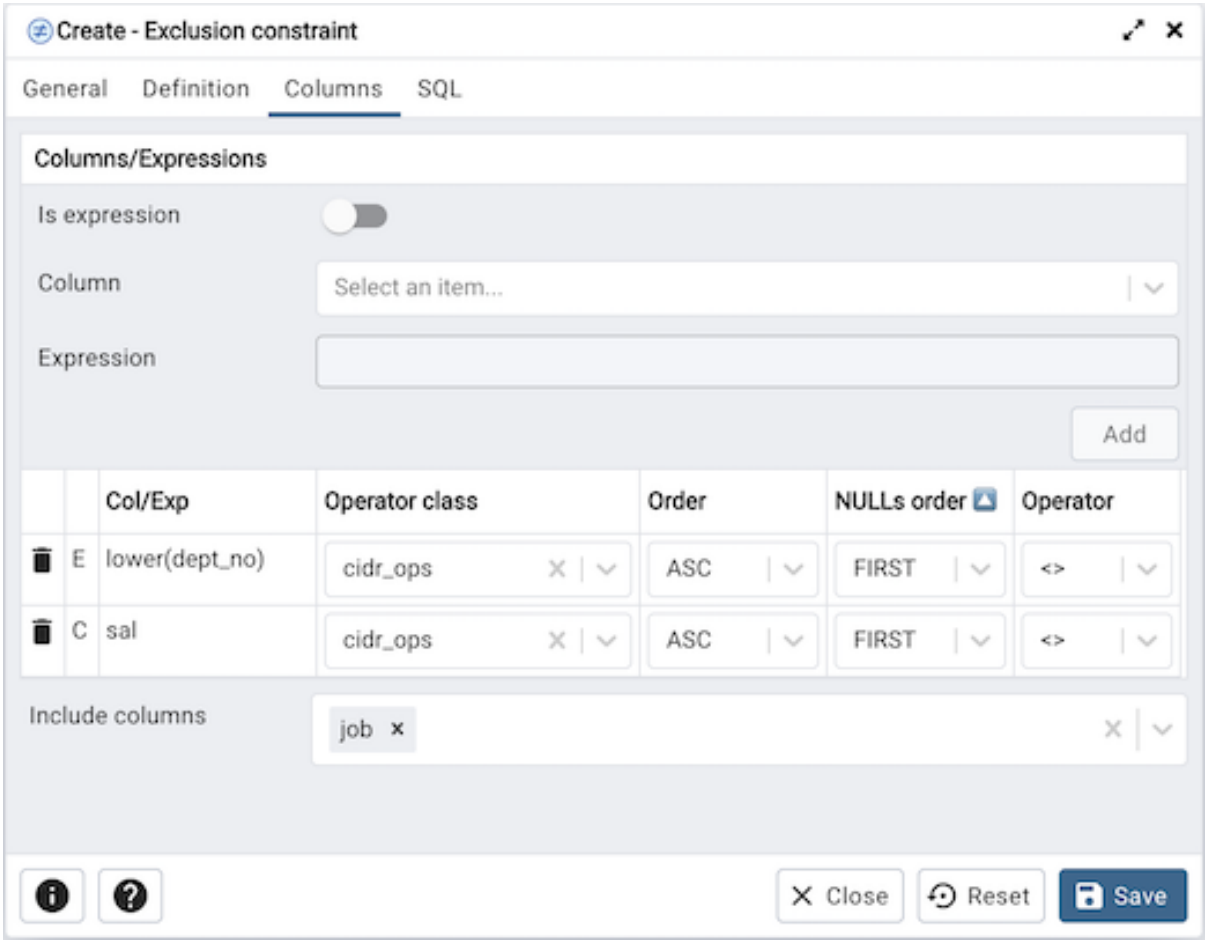
[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、除外制約を定義します。

- *Tablespace* の横にあるドロップダウンリストボックスを使用して、除外制約に関連付けられたインデックスが存在するテーブルスペースを選択します。
- **アクセス方法** の横にあるドロップダウンリストボックスを使用して、除外制約を実装するときに使用されるインデックスの種類を指定します。
  - *gist* を選択して、GiST インデックスを指定します（デフォルト）。
  - スペースパーティションの GiST インデックスを指定するには、 *spgist* を選択します。
  - *btree* を選択して B ツリーインデックスを指定します。
  - *hash* を選択して、ハッシュインデックスを指定します。
- *FillFactor* フィールドを使用して、テーブルと関連するインデックスの FILLFACTOR を指定します。FILLFACTOR は 10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- *Deferrable?* スイッチを *Yes* の位置に移動して、制約のタイミングが遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。
- *Constraint* フィールドを使用して、行をテーブルに含める必要がある条件を指定します。

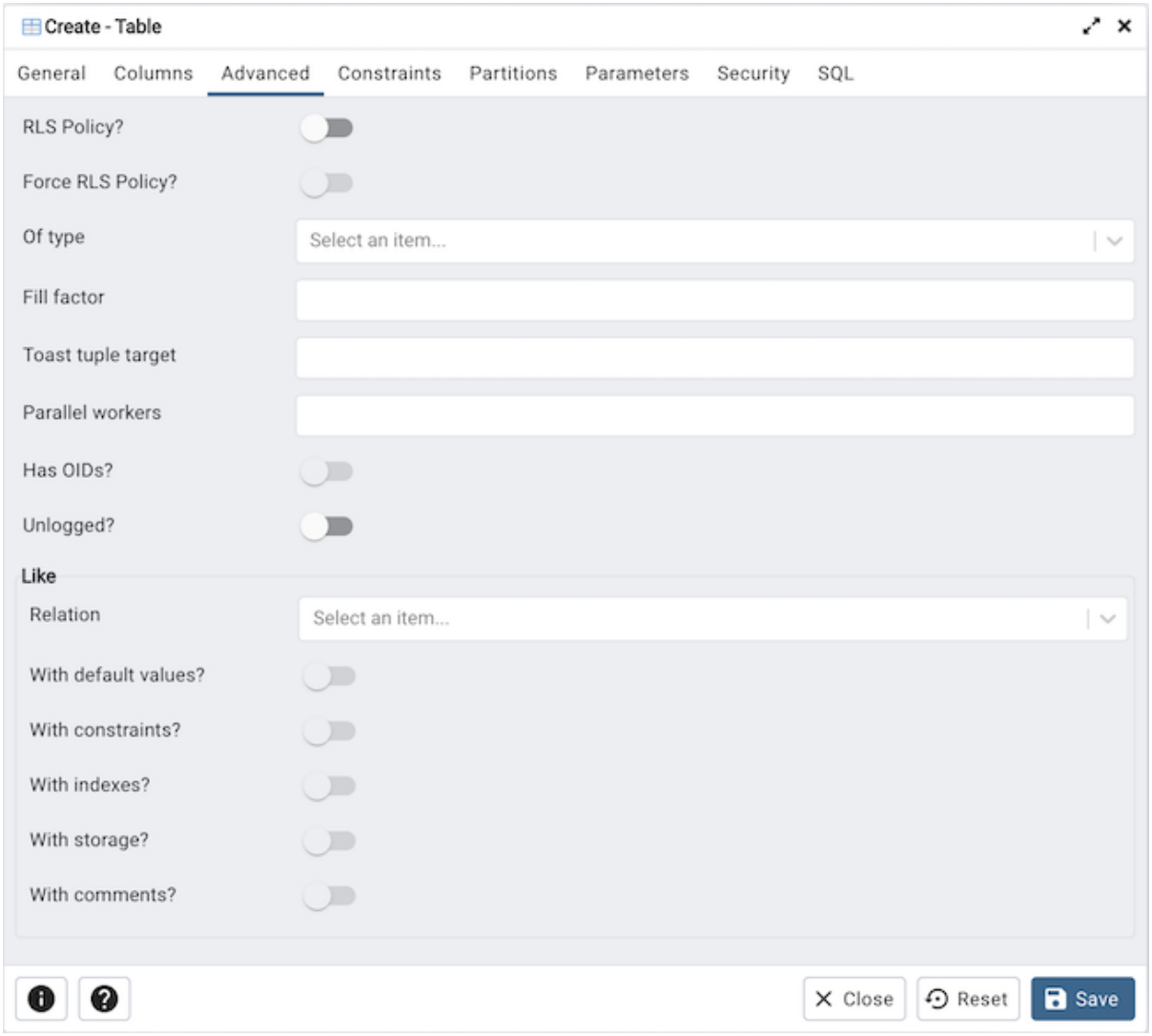
[**列**] タブをクリックして続行します。



[**列**] タブのフィールドを使用して、制約を適用する列を指定します。 **列** の横にあるドロップダウンリストボックスを使用して列を選択し、 **追加** アイコン（+）をクリックして列のアクションの詳細を指定します。

- *Column* フィールドには、 *Column* ドロップダウンリストボックスでの選択が入力されます。
- 必要に応じて、 **演算子クラス** のドロップダウンリストボックスを使用して、列のインデックスで使用される演算子クラスを指定します。
- *DESC* スイッチを *DESC* に移動して、降順の並べ替えを指定します。デフォルトは *ASC* で、昇順の並べ替えを指定します。
- *NULL* **オーダー** スイッチを *LAST* に移動して、NULL の昇順の並べ替えを定義します。デフォルトは *FIRST* で、降順を指定します。
- **演算子** の横にあるドロップダウンリストを使用して、比較または条件演算子を指定します。

[**詳細**] タブをクリックして続行します。



[*Advanced*] タブのフィールドを使用して、テーブルの高度な機能を定義します。

- *RLSPolicy?* スイッチを *Yes* の位置に移動して、行レベルのセキュリティを有効にします。
- *ForceRLSPolicy?* を *Yes* の位置に移動して、テーブルの所有者にポリシーを強制します。
- *Oftype* の横にあるドロップダウンリストボックスを使用して、指定した複合型からテーブル構造をコピーします。型が削除されると（DROPTYPE...CASCADE を使用して）、型付きテーブルが削除されることに注意してください。
- *FillFactor* フィールドを使用して、テーブルの FillFactor を指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- **トーストタプルターゲット** フィールドを使用して、テーブルの toast_tuple_target ストレージパラメーターを設定します。toast_tuple_target 値はバイト単位で、最小値は 128 です。このフィールドは PostgreSQL バージョン>=11 でのみ有効になります
- **パラレルワーカー** フィールドを使用して、テーブルの parallel_workers ストレージパラメーターを設定します。parallel_workers は、テーブルの並列スキャンを支援するために使用されるワーカー数を設定します。このフィールドは PostgreSQL バージョン>=9.6 でのみ有効になります
- *OID* **はありますか***?* スイッチを *Yes* の位置に移動して、テーブル内の各行にシステム割り当てのオブ

ジェクト識別子があることを指定します。デフォルトは **いいえ** です。

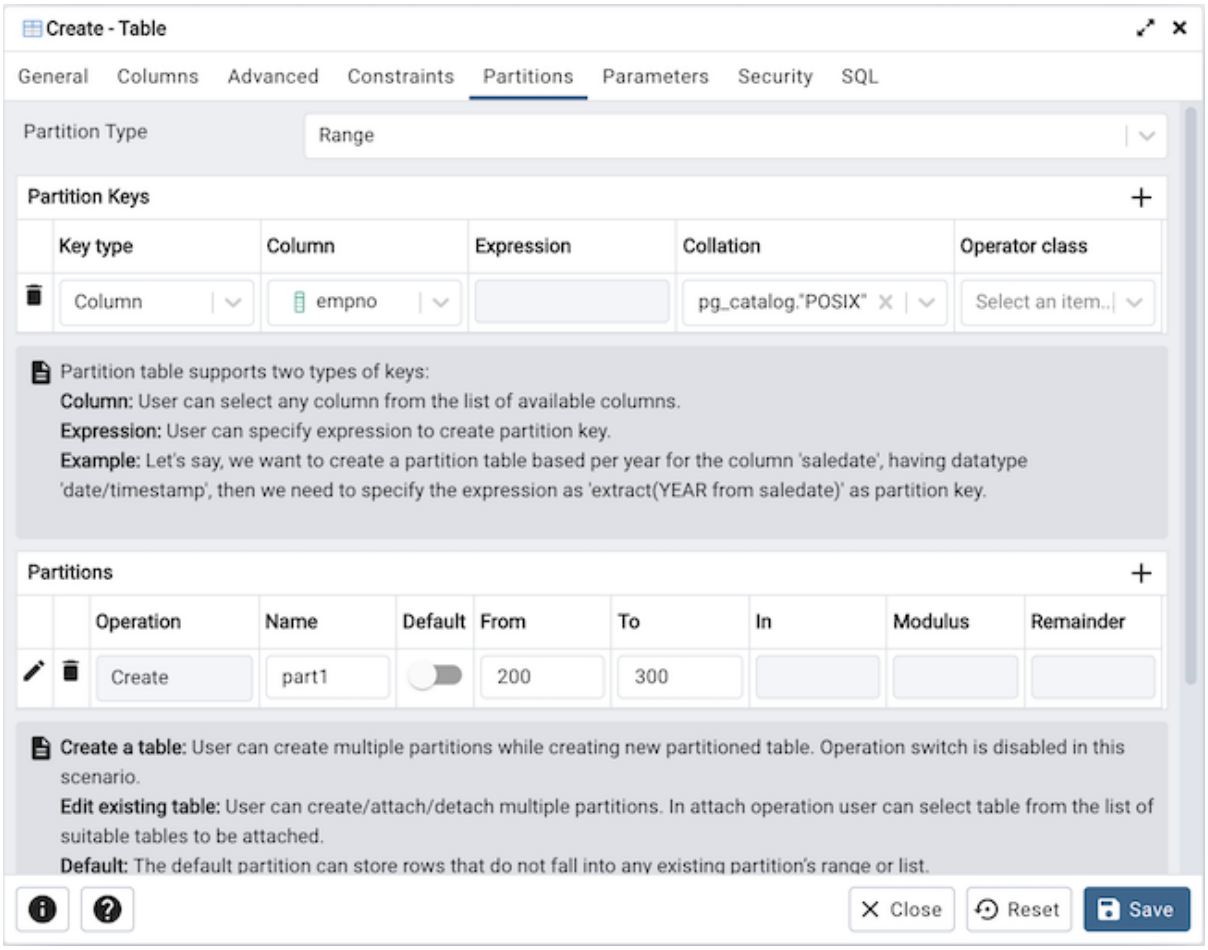
- *Unlogged?* スイッチを *Yes* の位置に移動して、テーブルのログを無効にします。ログに記録されないテーブルに書き込まれたデータは、ログ先行書き込みに書き込まれません。ログに記録されないテーブルで作成されたインデックスも自動的にログに記録されます。デフォルトは **いいえ** です。

Like ボックスのフィールドを使用して、テーブルが列名、データ型、および not-null 制約を自動的にコピーする既存のテーブルの属性を指定します。新しいテーブルまたは変更したテーブルを保存した後、元のテーブルへの変更は新しいテーブルに適用されません。

- *Relation* の横にあるドロップダウンリストボックスを使用して、参照テーブルを選択します。
- **デフォルト値を使用しますか***?* スイッチを **はい** の位置に移動して、デフォルト値をコピーします。
- **制約あり***?* スイッチを **はい** の位置に移動して、テーブルと列の制約をコピーします。
- **インデックス付き***?* スイッチを **はい** の位置に移動して、インデックスをコピーします。
- **ストレージあり***?* スイッチを **はい** の位置に移動して、ストレージ設定をコピーします。
- **コメント付き***?* スイッチを **はい** の位置に移動して、コメントをコピーします。

PostgreSQL10 以降では、[**パーティション**] タブが表示されます。

[**パーティション**] タブをクリックして続行します。



[**パーティション**] タブのフィールドを使用して、テーブルのパーティションを作成します。

- *PartitionType* 選択ボックスからパーティションの種類を選択します。3 つのオプションがあります。範囲、リスト、ハッシュ。ハッシュオプションは、PostgreSQL バージョン>=11 でのみ有効になります。

PartitionKeys パネルを使用して、パーティションキーを定義します。 **追加** アイコン（+）をクリックして、各パーティションキーの選択を追加します。

- *Keytype* フィールドでパーティションキーの種類を選択します。
- *Keytype* フィールドで列オプションが選択されている場合、 *Column* フィールドでパーティション列を選択します。
- *Keytype* フィールドで Expression オプションを選択した場合、 *Expression* フィールドに式を指定します。

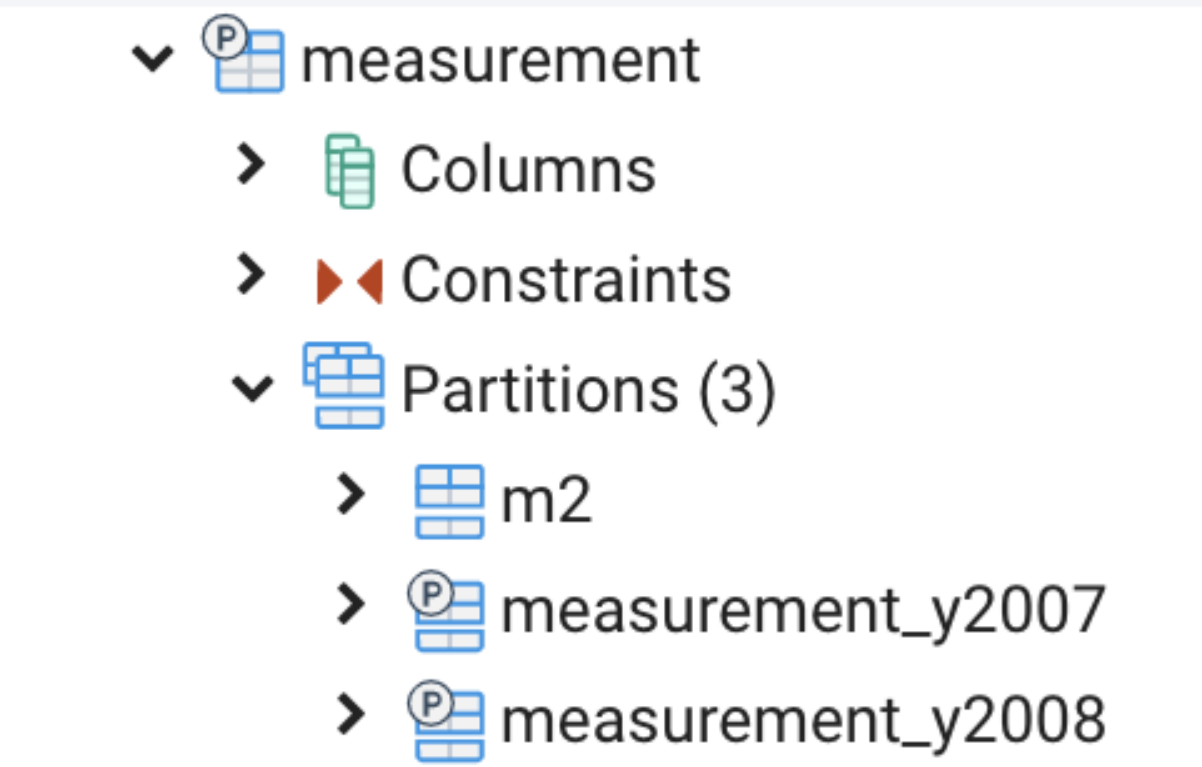
Partitions パネルを使用して、テーブルのパーティションを定義します。 **追加** アイコン（+）をクリックして、各パーティションを追加します。

- **操作** スイッチを **アタッチ** に移動して、パーティションをアタッチします。デフォルトでは **作成** です。
- *Name* フィールドを使用して、パーティションの名前を追加します。
- パーティションタイプがレンジまたはリストの場合、 **デフォルト** フィールドが有効になります。
- パーティションタイプが範囲の場合、 *From* および *To* フィールドが有効になります。
- パーティションタイプがリストの場合、 *In* フィールドが有効になります。
- パーティションタイプがハッシュの場合、 *Modulus* および *Remainder* フィールドが有効になります。

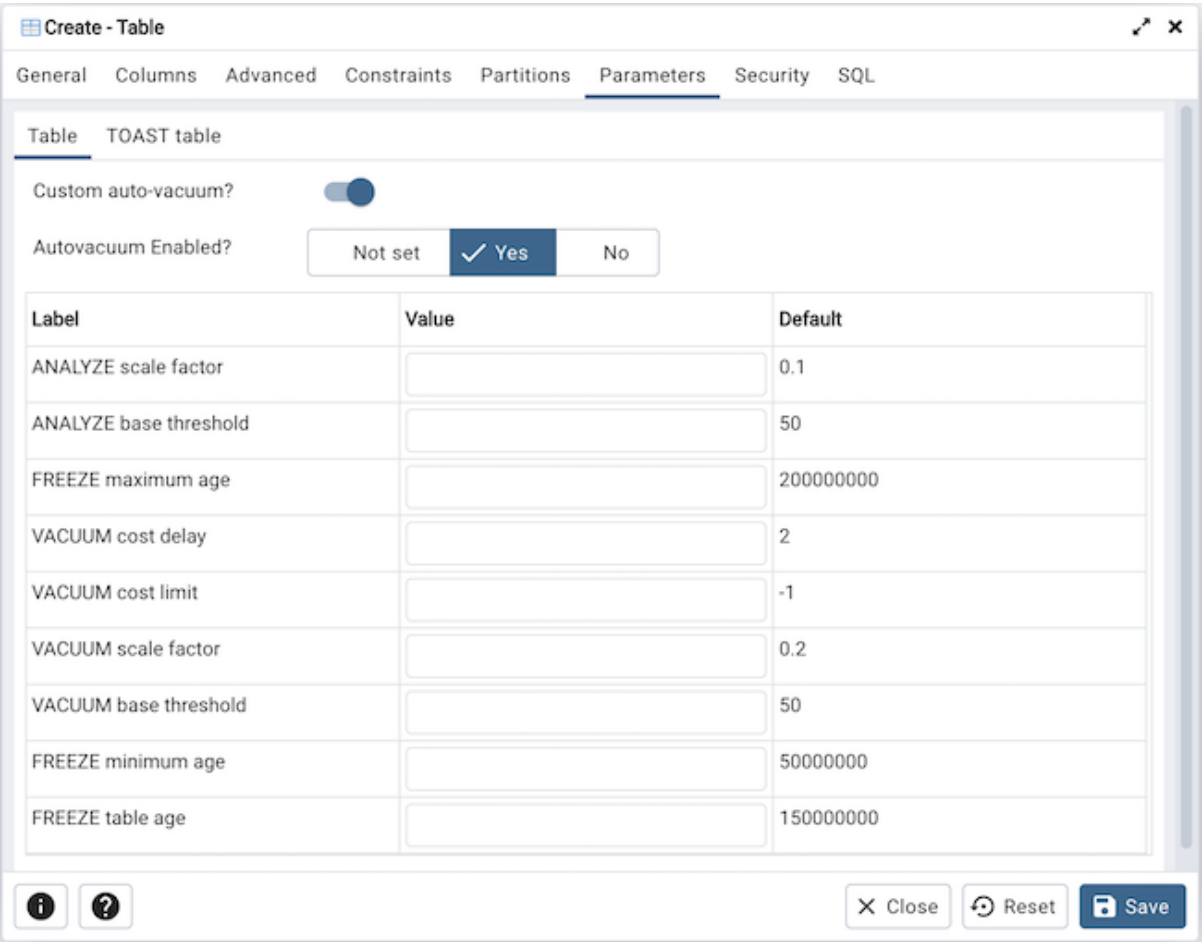
ユーザーはパーティションを作成し、パーティションテーブルとして定義できます。 **編集** アイコンをクリックして、パーティションのプロパティを展開します。[**パーティション**] タブを使用して、そのパーティションをパーティションテーブルとして作成します。

- パーティションテーブルを作成する場合は、 **パーティションテーブル***?* スイッチを **はい** に移動します。
- *PartitionType* 選択ボックスからパーティションの種類を選択します。
- *PartitionKeys* パネルを使用して、パーティションキーを定義します。

ブラウザーツリーでのマルチレベルパーティションテーブルのビュー：



[**パラメーター**] タブをクリックして続行します。

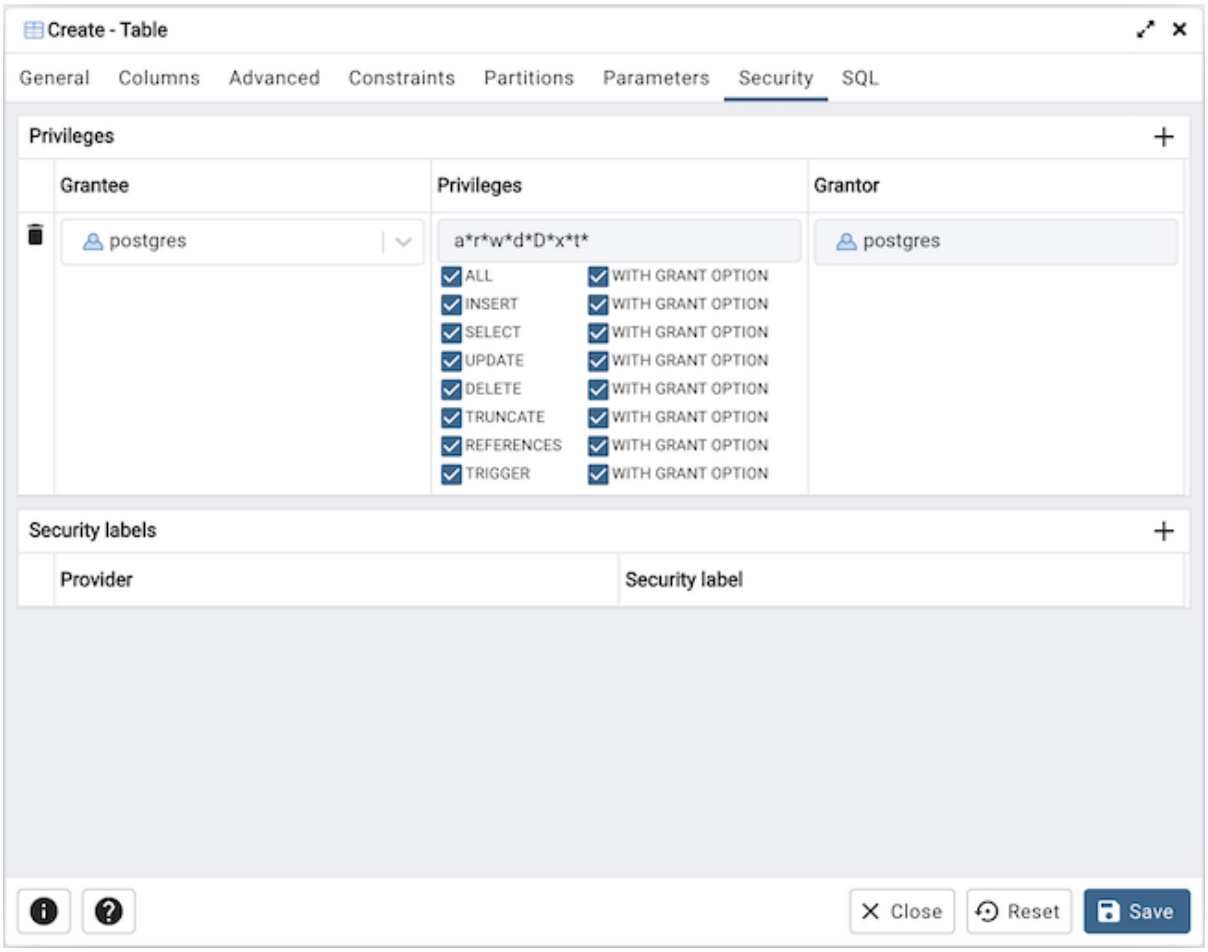


パラメーター タブの中にネストされたタブを使用して、VACUUM および ANALYZE のしきい値を指定します。[**テーブル**] タブと **トーストテーブル** タブを使用して、テーブルと関連するトーストテーブルの値をカスタマイズします。

- **カスタム自動バキューム**? スイッチを **はい** の位置に移動して、テーブルでカスタムメンテナンスを実行し、 **バキュームテーブル** で値を選択します。 **バキュームテーブル** は、メンテナンス操作のデフォルト値を提供します。
- *Autovacuumenabled?* を *Notset* に変更すると、autovacuum_enabled がリセットされます。

ラベル 列にリストされている各メトリックの **値** 列にカスタム値を入力します。

[**セキュリティ**] タブをクリックして続行します。



[**セキュリティ**] タブを使用して、特権を割り当て、セキュリティラベルを定義します。

[*Privileges*] パネルを使用して、ロールに特権を割り当てます。 **追加** アイコン（+）をクリックして、データベースオブジェクトの特権を設定します。

- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。

追加 アイコン（+）をクリックして、追加の特権を割り当てます。特権を破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SecurityLabels パネルを使用して、ファンクションに適用されるセキュリティラベルを定義します。 **追加** アイコン（+）をクリックして、各セキュリティラベルの選択を追加します。

- *Provider* フィールドにセキュリティラベルプロバイダーを指定します。名前付きプロバイダーをロードし、提案されたラベル付け操作に同意する必要があります。
- **セキュリティラベル** フィールドにセキュリティラベルを指定します。特定のラベルの意味は、ラベルプロバイダーの裁量にあります。PostgreSQL は、ラベルプロバイダーがセキュリティラベルを解釈する必要があるかどうか、またはどのように解釈するかについて制限を設けていません。それらを格納するメカニズムを提供するだけです。

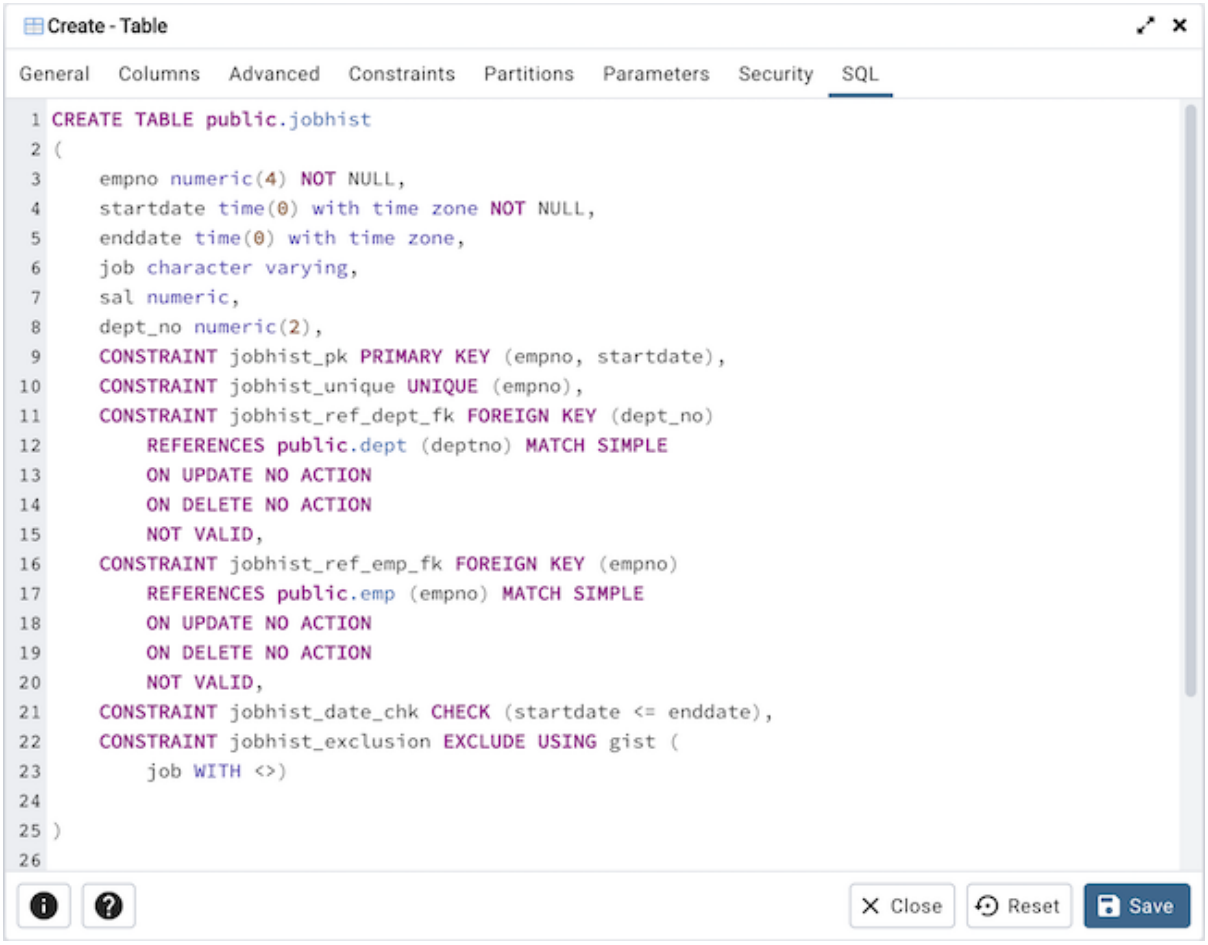
Add アイコン（+）をクリックして、追加のセキュリティラベルを割り当てます。セキュリティラベルを破棄するには、行の左側にあるゴミ箱アイコンをクリックし、 *DeleteRow* ポップアップで削除を確認します。

SQL タブをクリックして続行します。

テーブル ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.10.1 例

以下は、 **テーブル** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



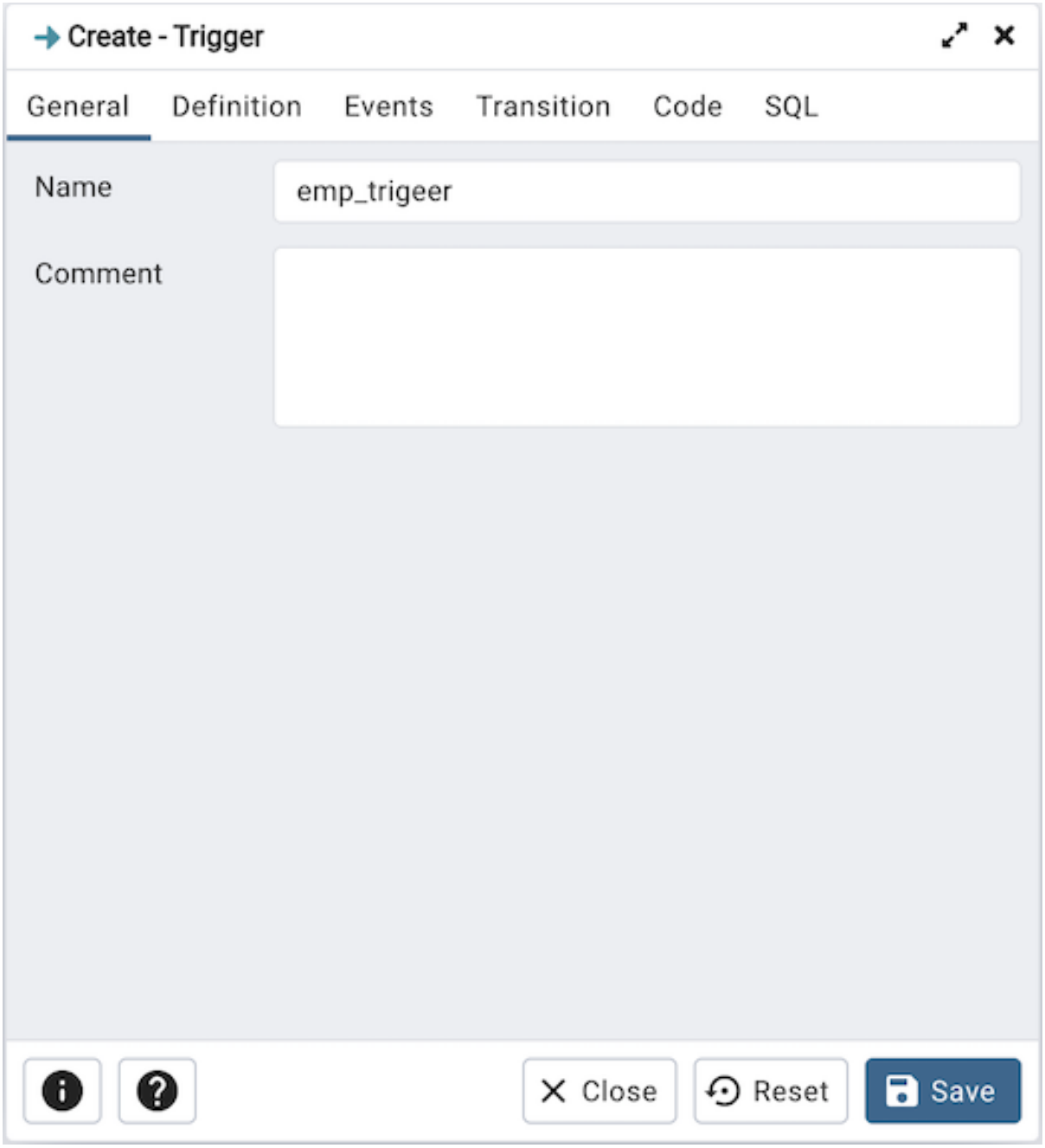
示されている例は、*jobhist* という名前のテーブルの作成を示しています。6 つの列と、*empno* と *startdate* 列に主キー制約があります。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.11 Trigger Dialog

トリガー ダイアログを使用して、トリガーを作成するか、既存のトリガーを変更します。トリガーは、特定のイベントが発生したときに指定された関数を実行します。

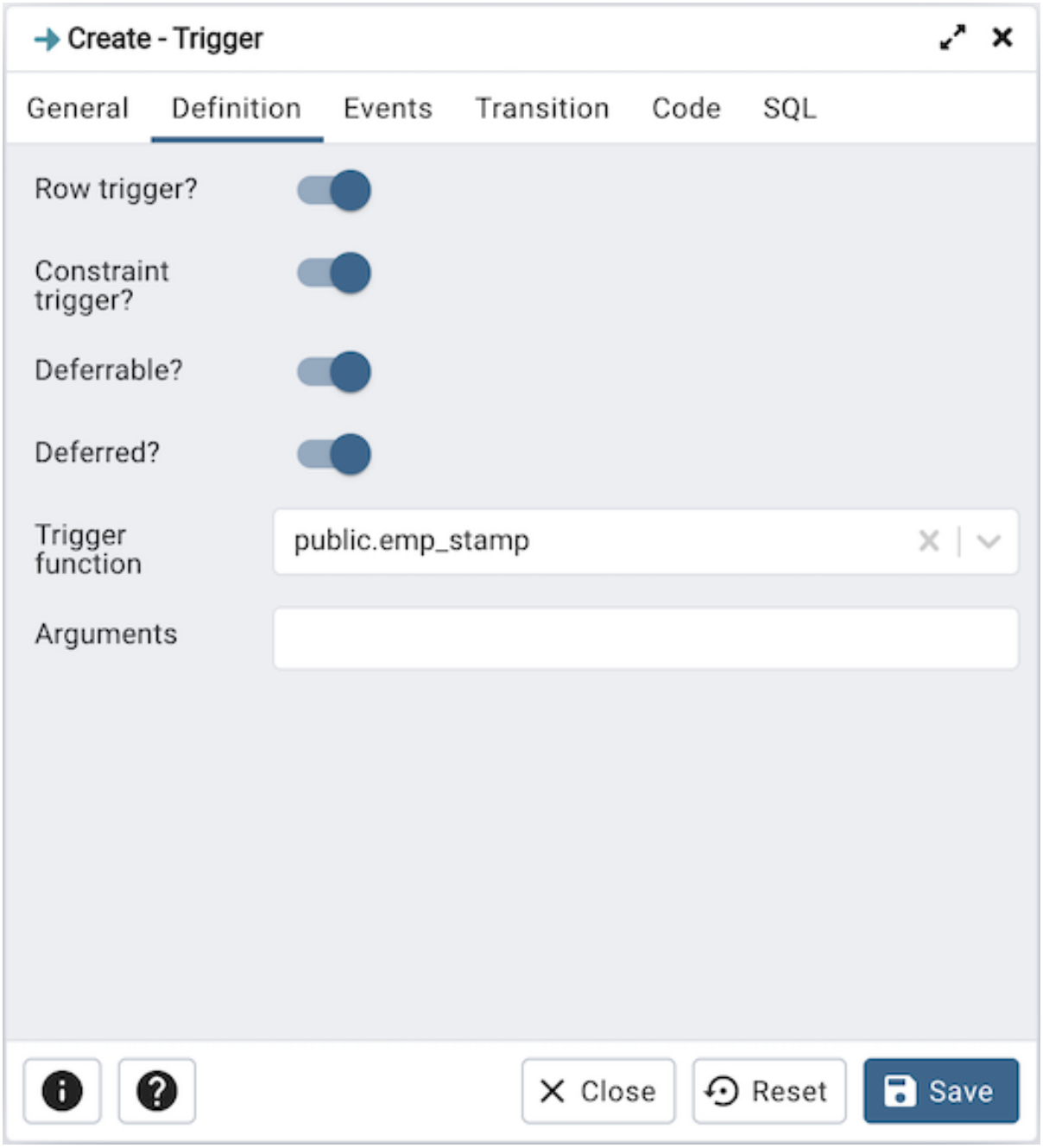
トリガー ダイアログは、次のダイアログタブを介してトリガーの開発を編成します。 **一般** 、 **定義** 、 **イベント** 、および **コード** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、トリガーを識別します。

- *Name* フィールドを使用して、トリガーの説明的な名前を追加します。これは、同じテーブルの他のトリガーの名前とは異なる必要があります。名前は *pgAdmin* ツリーコントロールに表示されます。同じ種類の複数のトリガーが同じイベントに対して定義されている場合、名前のアルファベット順に起動されることに注意してください。
- トリガーに関するメモを **コメント** フィールドに保存します。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、トリガーを定義します。

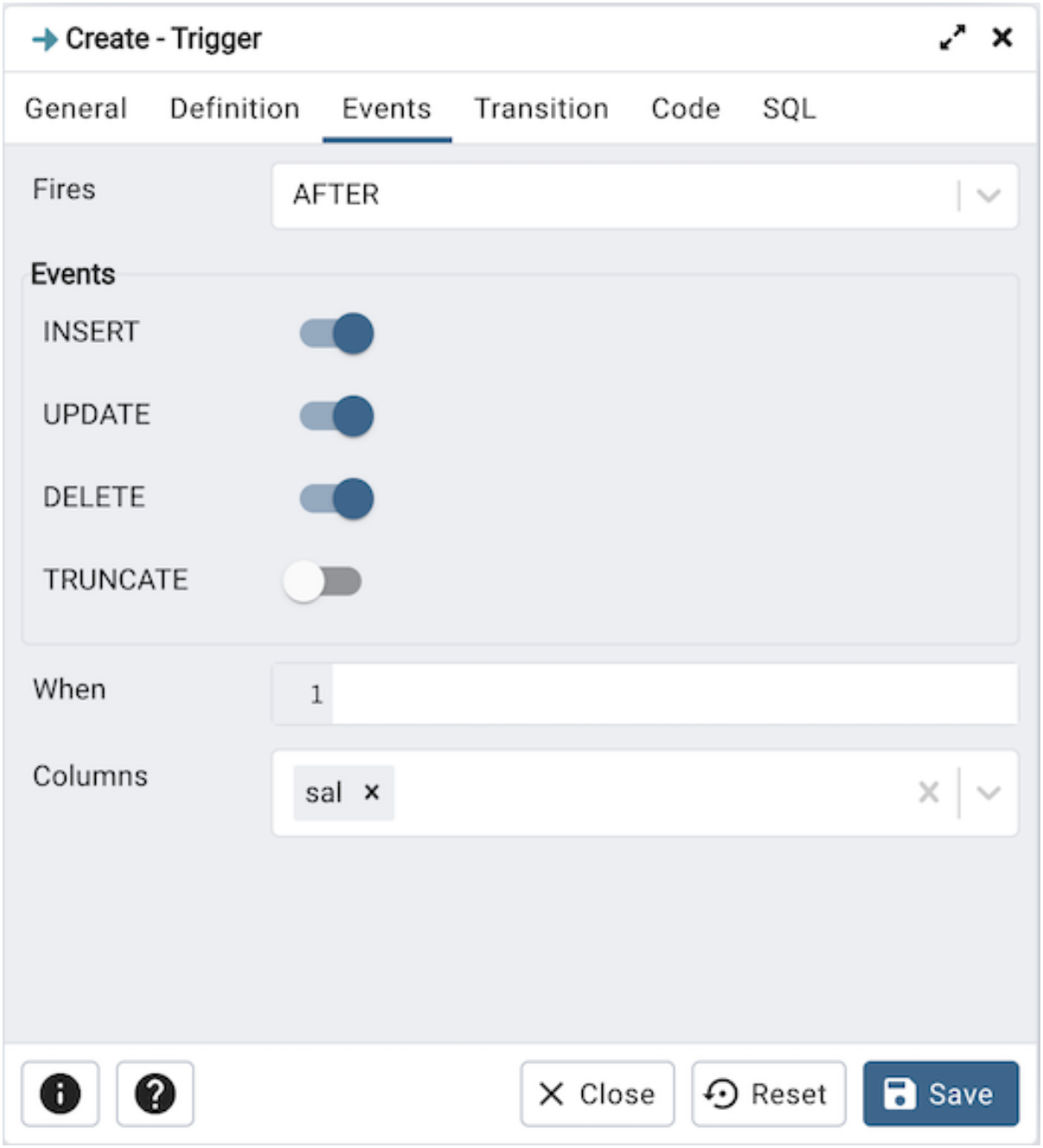
- **行トリガー***?* スイッチを **いいえ** の位置に移動して、テーブルの各行でトリガーが発火しないようにします。デフォルトは **はい** です。
- *Constrainttrigger?* スイッチを *Yes* の位置に移動して、トリガーがコンストレイントトリガーであることを指定します。
- 有効になっている場合、 *Deferrable?* スイッチを *Yes* の位置に移動して、制約トリガーが遅延可能であり、ステートメントの最後まで延期できるように指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約トリガーのタイミングをトリガーイベントを発生させるステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。

- **トリガー関数** の横にあるドロップダウンリストボックスを使用して、トリガー関数またはプロシージャを選択します。
- *Arguments* フィールドを使用して、トリガーが実行されるときに関数に引数のオプション（コンマ区切り）リストを提供します。引数はリテラル文字列定数です。



- **トリガー有効** フィールドは、トリガーが作成されるとトリガーダイアログで使用できます。4 つのオプションから 1 つを選択できます。

[**イベント**] タブをクリックして続行します。



[**イベント**] タブのフィールドを使用して、トリガーを起動する方法とタイミングを指定します。

- *Fires* フィールドの横にあるドロップダウンリストボックスを使用して、トリガーが指定されたイベントの *BEFORE* または *AFTER* かを決定します。デフォルトは *BEFORE* です。
- トリガーを呼び出すイベントの種類を選択します。イベントタイプを選択するには、イベントの横にあるスイッチを *YES* の位置に動かします。サポートされているイベントタイプは、*INSERT* 、*UPDATE* 、 *DELETE* 、および *TRUNCATE* です。
- *When* フィールドを使用して、トリガーを呼び出すブール条件を提供します。
- 列固有のトリガーを定義する場合は、 *Columns* フィールドを使用して、トリガーのターゲットとなる列を指定します。

[**コード**] タブをクリックして続行します。



Code フィールドを使用して、トリガーが起動したときに呼び出される追加のコードを指定します。

SQL タブをクリックして続行します。

トリガー ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.11.1 例

以下は、 **トリガー** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



この例は、 *emp_trigger* という名前のトリガーの作成を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

6.12 Unique Constraint Dialog

一意の制約 ダイアログを使用して、指定したテーブルの一意の制約を定義します。一意性制約は、列または列のグループに含まれるデータが、テーブル内のすべての行で一意であることを保証します。

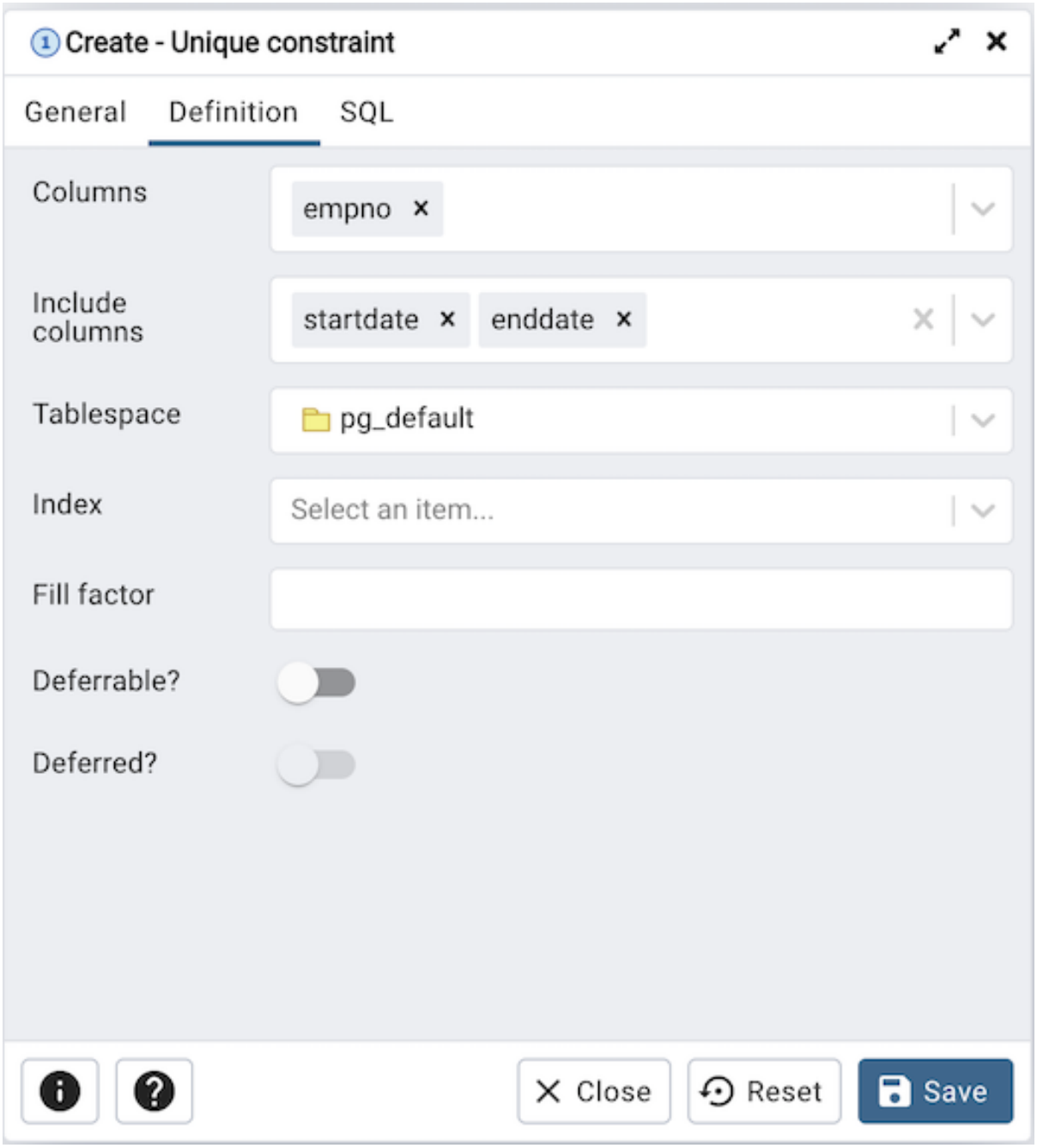
ユニークな制約 ダイアログは、次のダイアログタブを介してユニークな制約の開発を整理します。 **一般** と **定義** 。 *SQL* タブには、ダイアログ選択によって生成された SQL コードが表示されます。



[**全般**] タブのフィールドを使用して、一意の制約を識別します。

- *Name* フィールドを使用して、一意性制約の説明的な名前を追加します。名前は *pgAdmin* ツリーコントロールに表示されます。

[**定義**] タブをクリックして続行します。



[**定義**] タブのフィールドを使用して、一意の制約を定義します。

- *Columns* フィールド内をクリックし、ドロップダウンリストボックスから 1 つ以上の列名を選択します。選択を削除するには、列名の左側にある *x* をクリックします。ユニーク制約は、同じテーブルに対して定義された主キー制約とは異なる必要があります。制約用に選択された列は個別である必要があります。
- *Includecolumns* フィールドを使用して、制約の *INCLUDE* 句の列を指定します。このオプションは Postgres11 以降で使用できます。
- *Tablespace* フィールドのドロップダウンリストボックスから、一意性制約が存在するテーブルスペースの名前を選択します。
- **インデックス** フィールドのドロップダウンリストボックスからインデックスの名前を選択します。

このフィールドはオプションです。一意性制約を追加すると、制約にリストされている列または列のグループに一意の B ツリーインデックスが自動的に作成され、列に NOTNULL のマークが付けられます。

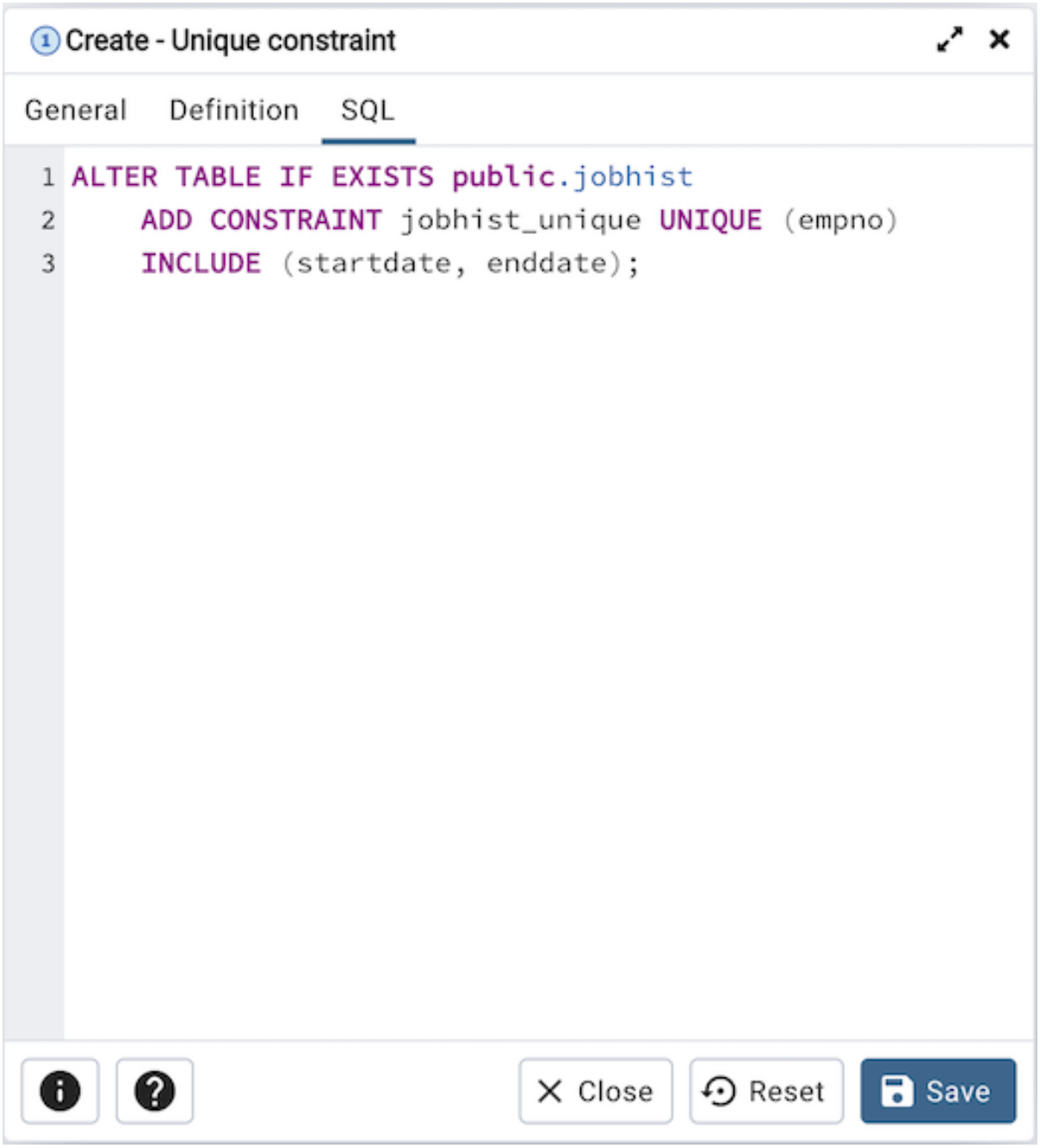
- *FillFactor* フィールドを使用して、テーブルとインデックスの FILLFACTOR を指定します。テーブルのフィルファクターは、10〜100 のパーセンテージです。デフォルトは 100（完全なパッキング）です。
- *Deferrable?* スイッチを *Yes* の位置に移動して、制約が遅延可能であり、ステートメントの最後まで延期できることを指定します。デフォルトは **いいえ** です。
- 有効になっている場合、 *Deferred?* スイッチを *Yes* の位置に移動して、制約のタイミングをステートメントの最後に延期することを指定します。デフォルトは **いいえ** です。

SQL タブをクリックして続行します。

一意の制約 ダイアログに入力すると、SQL コマンドが生成されます（以下の例を参照）。確認には *SQL* タブを使用します。再アクセスするかタブを切り替えて、SQL コマンドを変更します。

6.12.1 例

以下は、 **ユニークな制約** ダイアログでのユーザー選択によって生成された sql コマンドの例です。



ここに示す例は、*jobhist* テーブルの *empno* 列に *jobhist_unique* という名前の一意性制約を作成する方法を示しています。

- *Info* ボタン（i）をクリックして、オンラインヘルプにアクセスします。
- *Save* ボタンをクリックして作業を保存します。
- 作業を保存せずに終了するには、 **閉じる** ボタンをクリックします。
- *Reset* ボタンをクリックして、構成パラメーターを復元します。

第7章　Management Basics

pgAdmin は、サーバー管理機能の実行に役立つポイントアンドクリックのダイアログを提供します。ダイアログは、名前付け復元ポイントの管理、ユーザー権限の付与、VACUUM、ANALYZE、および REINDEX 機能の実行などのタスクを簡素化します。

7.1 Add Named Restore Point Dialog

名前付き復元ポイントの追加 ダイアログを使用して、回復ファイルで使用するサーバーの状態の名前付きスナップショットを取得します。名前付き復元ポイントを作成するには、サーバーの postgresql.conf ファイルで *wal_level* 値を *replica* 、 *logical* 、または *minimal* に指定する必要があります。復元ポイントを作成するには、データベースのスーパーユーザーである必要があります。



復元ポイント名 ウィンドウが開いたら、 **追加する復元ポイントの名前を入力してください** フィールドを使用して、復元ポイントに説明的な名前を付けます。

復元ポイントをリカバリターゲットとして使用する方法の詳細については、 PostgreSQL documentation を参照してください。

- *OK* ボタンをクリックして、復元ポイントを保存します。
- 作業を保存せずに終了するには、 *Cancel* ボタンをクリックします。

7.2 Change Password Dialog

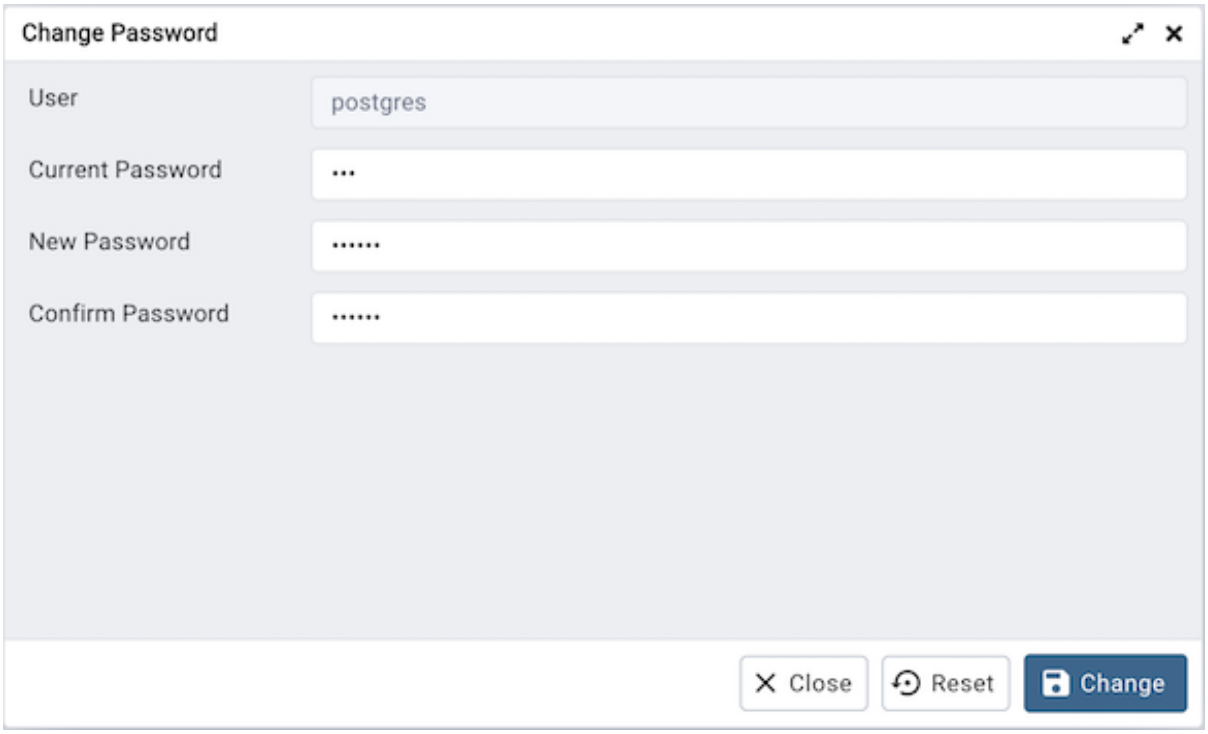
「安全」な環境でも、データを保護するためにパスワードを定期的に変更することをお勧めします。職場では、適切なパスワードポリシーを適用しないと、データ保護法に違反する可能性があります。

パスワードを選択するときは、次のガイドラインを考慮してください。

- パスワードが十分な長さであることを確認してください。6 文字は、パスワードの絶対最小文字数である必要があります。

- パスワードが辞書攻撃に対して開かれていないことを確認します。大文字、小文字、数字を組み合わせて使用し、単語や名前は避けてください。覚えやすいが聞き慣れない頭字語のフレーズを使って、各単語の最初の文字を使用することを検討してください。
- パスワードは定期的に変更してください。少なくとも 90 日ごとに交換してください。

上記のガイドラインは出発点と考えてください。それらは包括的なリストではなく、 **will not guarantee security** 。



[パスワードの変更] ダイアログを使用して、パスワードを変更します。

- *User* フィールドに表示される名前は、パスワードを変更するロールです。ツリーコントロールで強調表示されているのは、サーバー接続に関連付けられているロールです。
- ロールに関連付けられたパスワードを **現在のパスワード** フィールドに入力します。
- **新しいパスワード** フィールドに目的のパスワードを入力します。
- *ConfirmPassword* フィールドに新しいパスワードを再入力します。

保存 ボタンをクリックしてパスワードを変更します。

パスワードを変更せずに終了するには、 **閉じる** ボタンをクリックします。

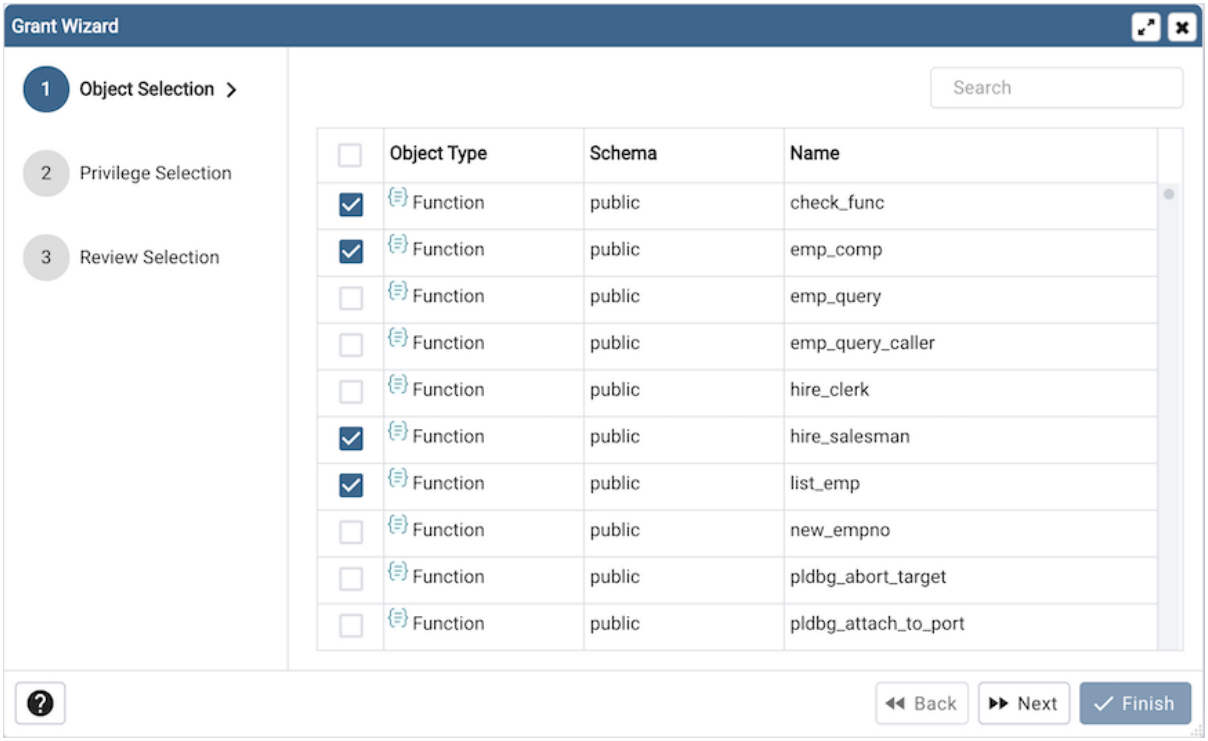
Reset ボタンをクリックして、値をリセットします。

7.3 Grant Wizard

GrantWizard ツールは、ポイントアンドクリック環境で 1 つ以上のデータベースオブジェクトの特権を管理できるグラフィカルインターフェイスです。検索ボックス、ドロップダウンリスト、およびチェックボックスを使用すると、データベースオブジェクト、ロール、および権限をすばやく選択できます。

ウィザードは、 *ObjectSelection* 、 *PrivilegesSelection* 、および *ReviewSelection* の一連のウィンドウを介して特権管理を編成します。 *ReviewSelection* ウィンドウには、ウィザードの選択によって生成された SQL コードが表示されます。

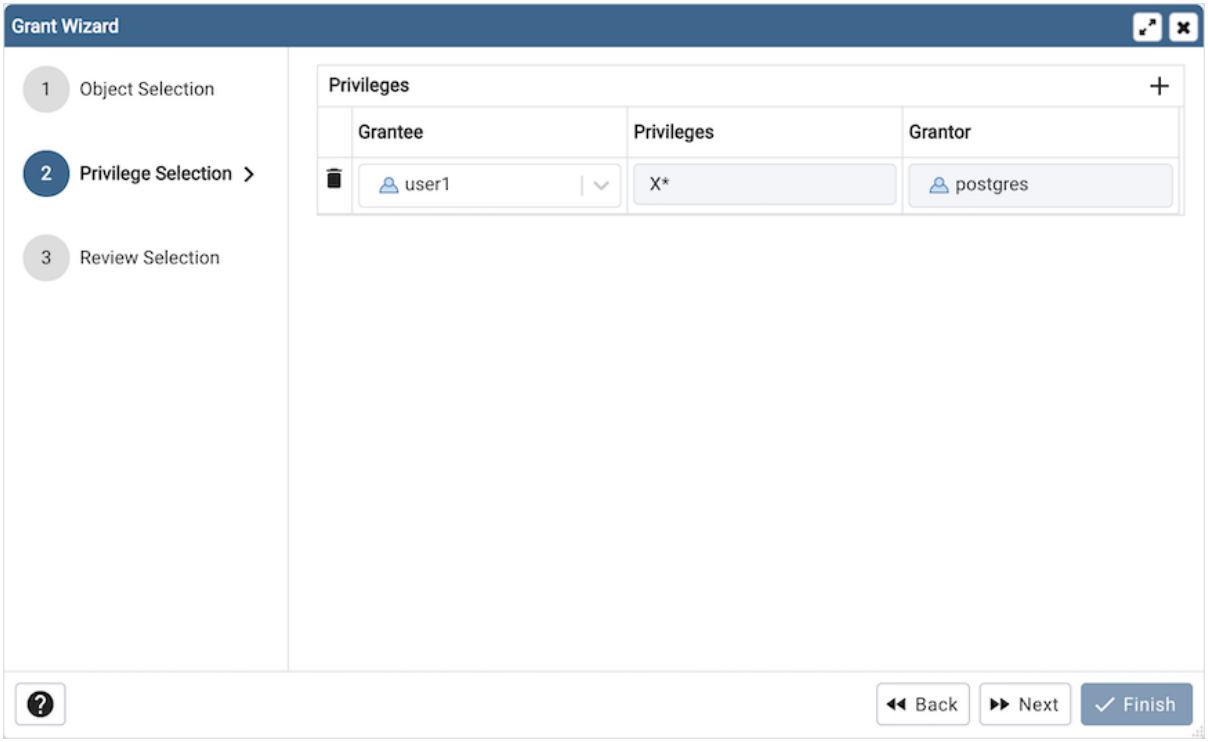
GrantWizard ツールを起動するには、 *pgAdmin* ツリーコントロールでデータベースオブジェクトを選択し、メニューバーの *Tools* に移動して *GrantWizard* オプションをクリックします。



ObjectSelection ウィンドウのフィールドを使用して、権限を変更するオブジェクトを選択します。[**オブジェクトの種類または名前で検索**] フィールドを使用してデータベースオブジェクトを見つけるか、スクロールバーを使用してすべてのアクセス可能なオブジェクトのリストをスクロールします。

- 表の各行には、オブジェクト識別子がリストされています。左の列のチェックボックスをオンにして、オブジェクトを GrantWizard のターゲットとして含めます。テーブルには次が表示されます。
  - *ObjectType* フィールドのオブジェクトタイプ
  - オブジェクトが存在するスキーマ
  - *Name* フィールドのオブジェクト名。

[**次へ**] ボタンをクリックして続行するか、[*X*] ボタンをクリックして、特権を変更せずにウィザードを閉じます。



特権の選択 ウィンドウのフィールドを使用して、特権を付与します。WITHGRANTOPTION で特権を付与すると、被付与者はオブジェクトに対する特権を他の人に付与する権利を有します。その後で WITHGRANTOPTION が取り消されると、その被付与者から（直接または一連の付与を介して）そのオブジェクトへのアクセスを受け取ったロールは、そのオブジェクトに対する特権を失います。

- **追加** アイコン（+）をクリックして、一連の特権を割り当てます。
- *Grantee* フィールドのドロップダウンリストボックスからロールの名前を選択します。
- *Privileges* フィールド内をクリックします。1 つ以上の特権の左側にあるチェックボックスをオンにして、選択した特権を指定されたユーザーに付与します。データベースオブジェクトに対して特権が以前に付与されている場合、グループまたはユーザーの特権のチェックを外すと、その特権が取り消されます。
- 特権を付与するためのデフォルトの付与者である現在のユーザーが *Grantor* フィールドに表示されます。
- **追加** アイコン（+）をクリックして、一連の特権を別のロールに割り当てます。特権を破棄するには、行の左側にあるごみ箱アイコンをクリックし、 **行の削除** ダイアログで削除を確認します。

データベースオブジェクトに対する特権の付与の詳細については、 PostgreSQL core documentation を参照してください。

続行するには **次へ** ボタン、追加のデータベースオブジェクトを選択または選択解除するには **戻る** ボタン、特権を変更せずにウィザードを閉じるには *X* ボタンをクリックします。

GrantWizard ツールに入力すると、SQL コマンドが生成されます。コマンドは、 *ReviewSelection* ウィンドウで確認できます（以下の例を参照）。

7.3.1 例

以下は、 *GrantWizard* ツールでのユーザー選択によって生成された sql コマンドの例です。

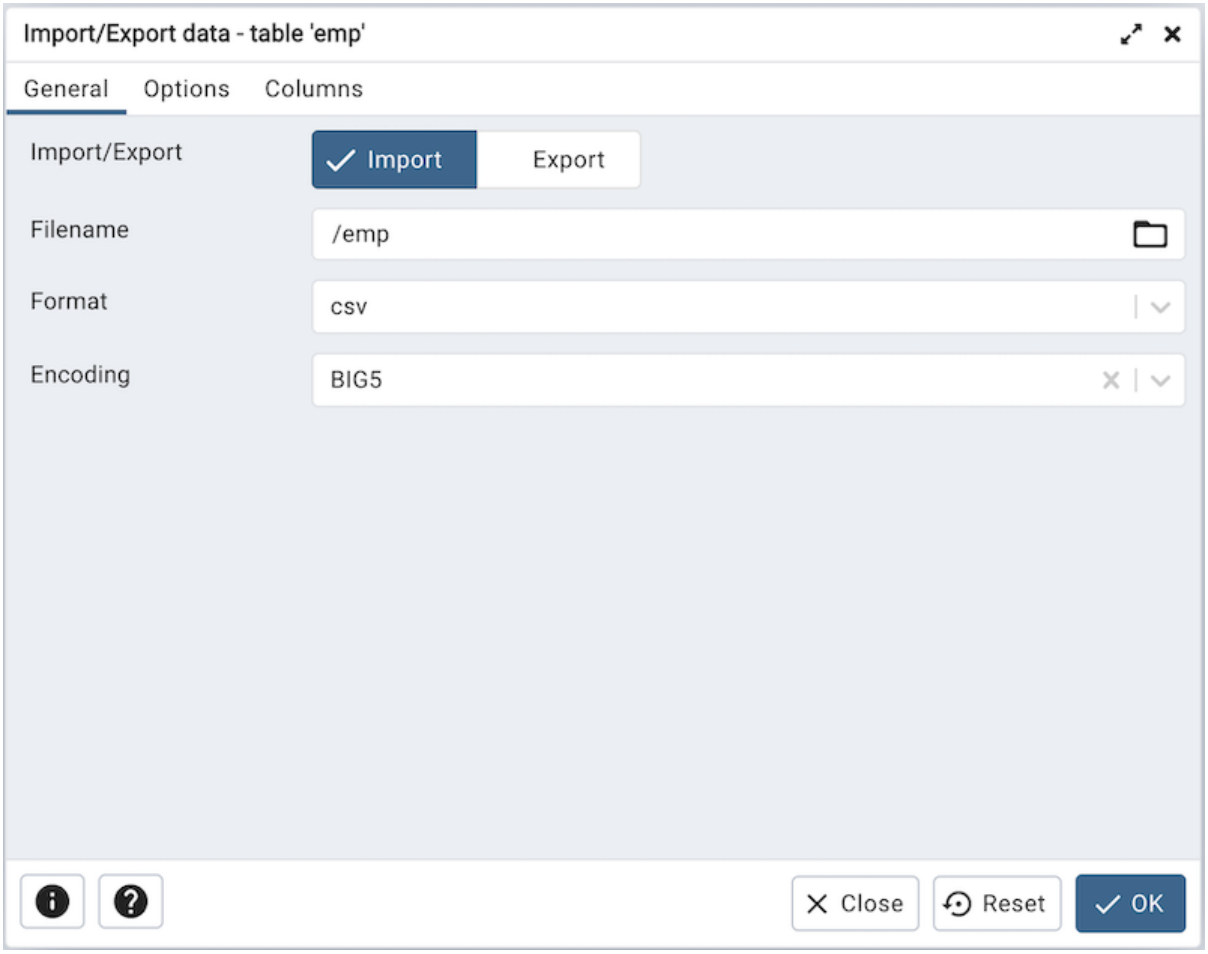


- **戻る** ボタンをクリックして、追加のデータベースオブジェクト、ロール、および特権を選択または選択解除します。
- *Finish* ボタンをクリックして選択を保存し、ウィザードを終了します。

7.4 Import/Export Data Dialog

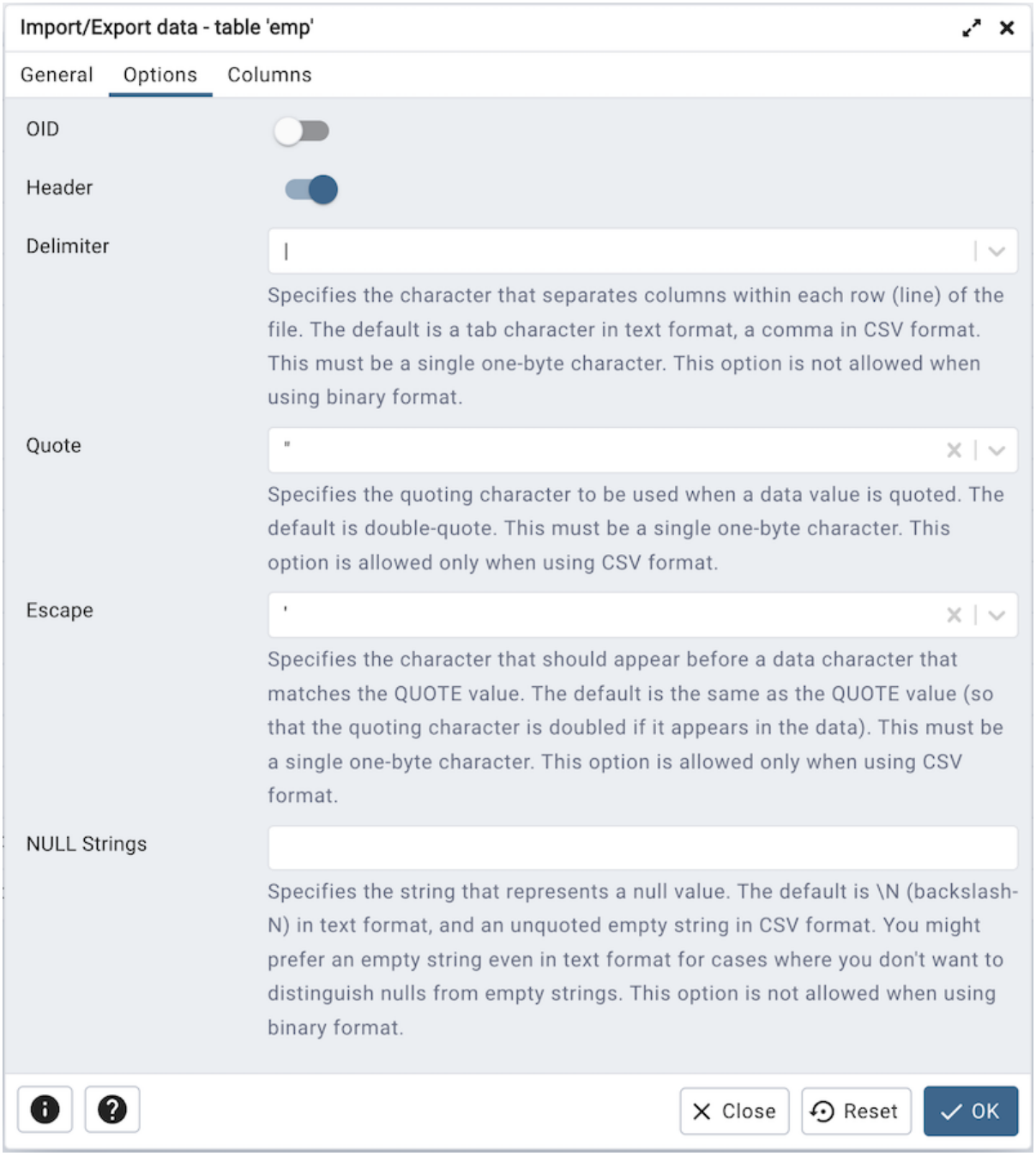
データのインポート/エクスポート ダイアログを使用して、テーブルからファイルにデータをコピーしたり、ファイルからテーブルにデータをコピーしたりします。

データのインポート/エクスポート ダイアログは、 **一般** 、 **オプション** 、および **列** タブを介してデータのインポート/エクスポートを構成します。



[**全般**] タブのフィールドを使用して、インポートとエクスポートの設定を指定します。

- **インポート/エクスポート** スイッチを **インポート** の位置に移動して、サーバーがファイルからテーブルにデータをインポートすることを指定します。デフォルトは **インポート** です。
- **ファイル名** フィールドにソースまたはターゲットファイルの名前を入力します。必要に応じて、右側にある **ブラウザー** アイコン（省略記号）を選択して、ディレクトリに移動し、ファイルを選択します。
- **フォーマット** フィールドのドロップダウンリストボックスを使用して、ファイルの種類を指定します。選択：
  - .bin ファイルの場合は **バイナリ** 。
  - .csv ファイルの場合は *csv* 。
  - .txt ファイルの場合は **テキスト** 。
- **エンコーディング** フィールドのドロップダウンリストボックスを使用して、文字エンコーディングの種類を指定します。

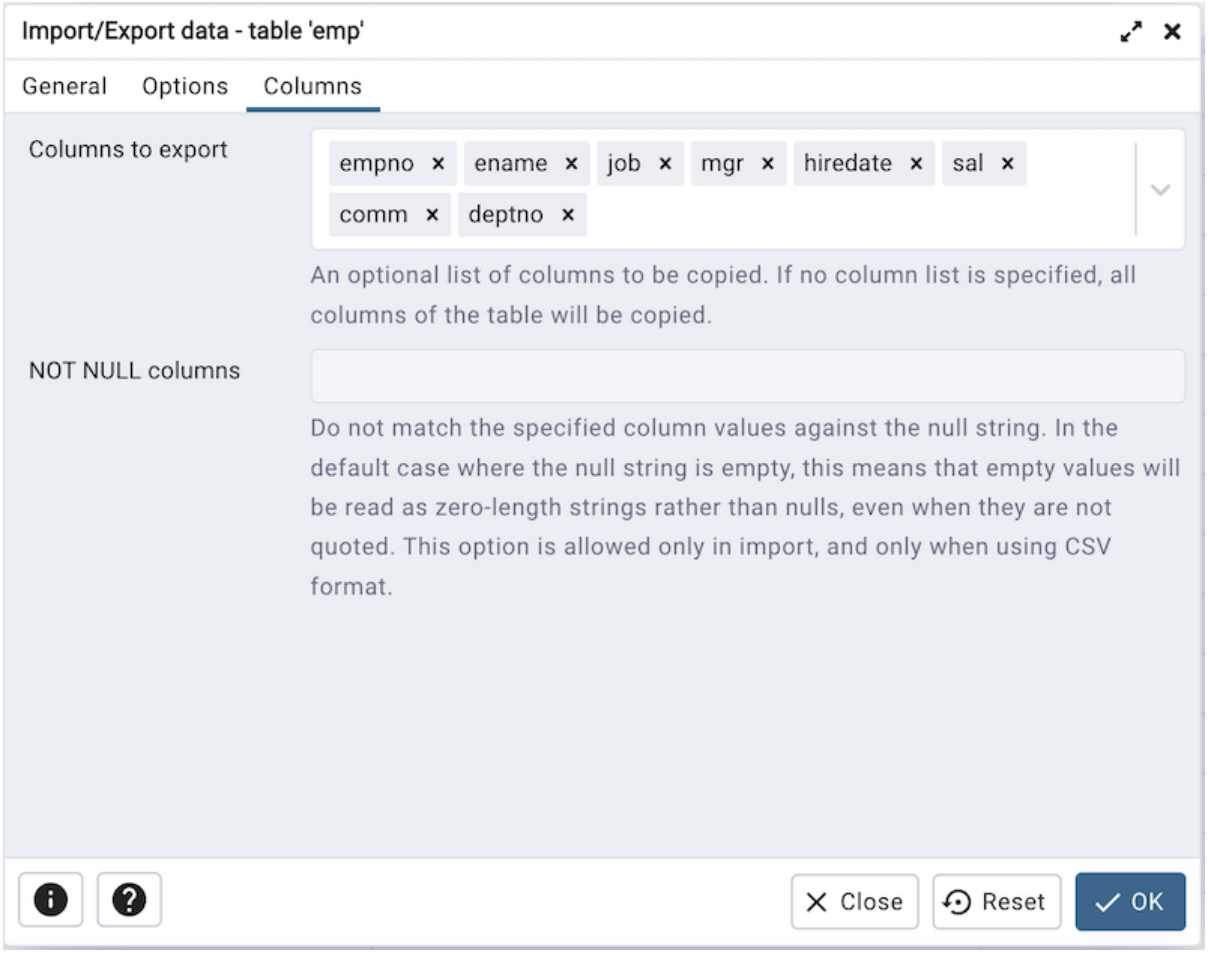


- [**オプション**] タブのフィールドを使用して、追加情報を指定します。
  - *OID* スイッチを *Yes* の位置に移動して、*OID* 列を含めます。 *OID* はシステム割り当ての値であり、変更できません。デフォルトは **いいえ** です。
  - **ヘッダー** スイッチを **はい** の位置に移動して、データ行にテーブルヘッダーを含めます。テーブルヘッダーを含めると、ファイルの最初の行に列名が含まれます。
  - データをエクスポートする場合は、**区切り文字** フィールドにターゲットファイル内の列を区切る区切り文字を指定します。区切り文字には、コロン、セミコロン、垂直バー、またはタブを使用できます。
  - *Quote* フィールドで使用される引用符を指定します。引用符は、文字列列のみ（つまり、数値列は引用符で囲まれません）またはデータ型に関係なくすべての列に適用できます。引用符には、

一重引用符または二重引用符を使用できます。

- *Escape* フィールドの *QUOTE* 値と一致するデータ文字の前に表示する文字を指定します。
- *NULL* **文字列** フィールドを使用して、ソースまたはターゲットファイル内で NULL 値を表す文字列を指定します。

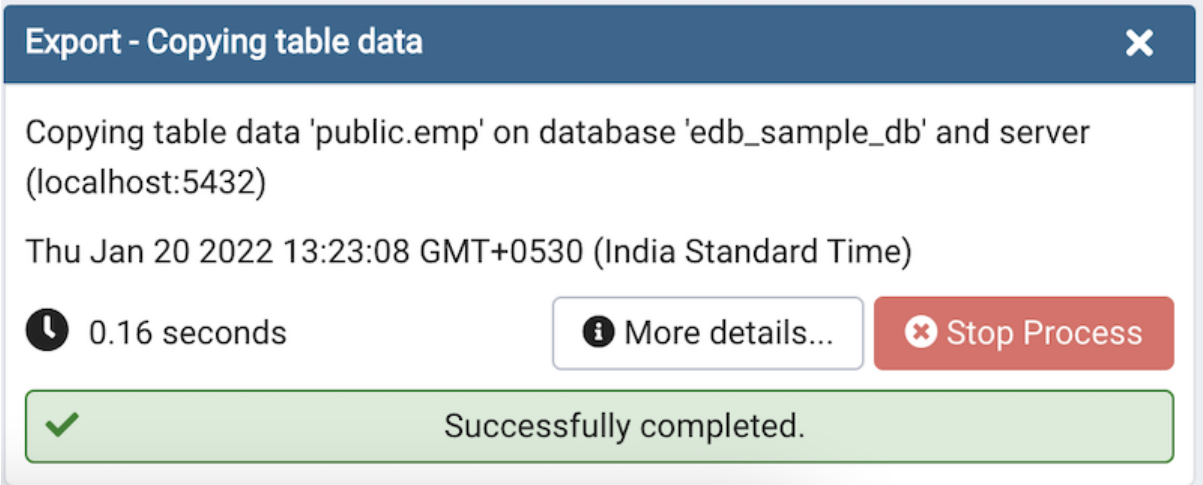
[**列**] タブをクリックして続行します。



[**列**] タブのフィールドを使用して、インポートまたはエクスポートする列を選択します。

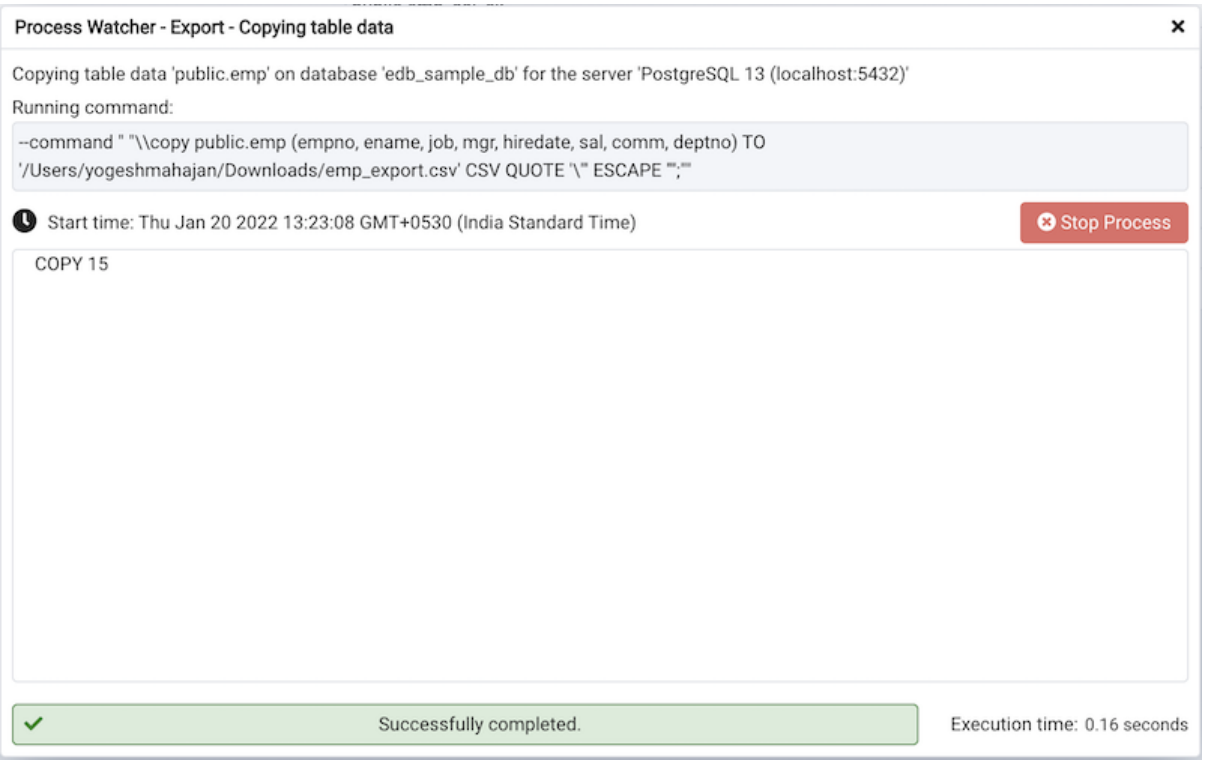
- *Columnstoexport/import* フィールド内をクリックして、ドロップダウンリストボックスから 1 つ以上の列の選択を解除します。選択を削除するには、列名の左側にある *x* をクリックします。フィールド内の空のスポットをクリックして、ドロップダウンリストにアクセスします。
- 有効になっている場合、 *NOTNULL* **列** フィールド内をクリックして、NULL 値のチェックを行わない 1 つ以上の列を選択します。列を削除するには、列名の左側にある *x* をクリックします。

データのインポート/エクスポート ダイアログが完了したら、 *OK* ボタンをクリックしてインポートまたはエクスポートを実行します。pgAdmin は、バックグラウンドプロセスが完了すると通知します。



Stop Process ボタンを使用して、インポート/エクスポートプロセスを停止します。

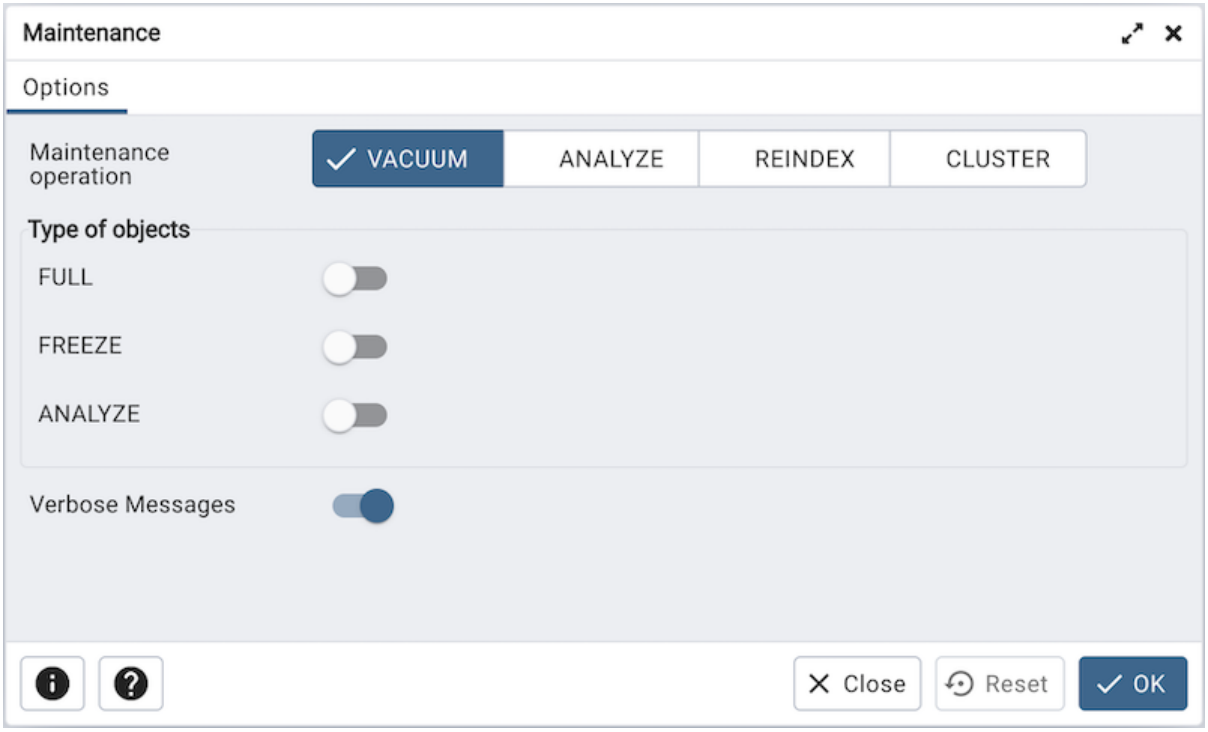
通知の **ここをクリックして詳細を表示する** リンクを使用して *ProcessWatcher* を開き、インポートまたはエクスポートを実行したコマンドの実行に関する詳細情報を確認します。



注釈: サーバーモード で *pgAdmin* を実行している場合、|sm_icon|ProcessWatcher ウィンドウのアイコンをクリックして、StorageManager でファイルの場所を開きます。 *Storage Manager* を使用して、クライアントマシンにバックアップファイルをダウンロードできます。

7.5 Maintenance Dialog

メンテナンス ダイアログを使用して、データベースまたは選択したデータベースオブジェクトを VACUUM、ANALYZE、REINDEX、またはクラスター化します。



このユーティリティはアドホックなメンテナンスの目的には役立ちますが、定期的に自動 VACUUM ジョブを実行することをお勧めします。

メンテナンス操作 の横にあるボタンを選択して、メンテナンスの種類を指定します。

- *VACUUM* をクリックして、選択したデータベースまたはテーブルをスキャンし、デッドタプルが使用するストレージを再利用します。
  - *FULL* スイッチを *Yes* の位置に移動して、デッドスペースのない完全に新しいバージョンのテーブルファイルを書き込みます。デフォルトは **いいえ** です。
  - *FREEZE* スイッチを *Yes* の位置に移動して、テーブルのデータをこれ以上更新しないときにフリーズします。デフォルトは **いいえ** です。
  - *ANALYZE* スイッチを *Yes* の位置に移動して、テーブルの内容が十分に変更されたときは常に ANALYZE コマンドを発行します。デフォルトは **いいえ** です。
- *ANALYZE* をクリックして、クエリプランナーが使用する保存された統計を更新します。これにより、クエリオプティマイザーは最適なパフォーマンスを得るために最速のクエリプランを選択できます。
- 異常なデータパターンの挿入が原因でインデックスが縮退した場合、*REINDEX* をクリックしてインデックスを再構築します。これは、たとえば、インデックス値が増加する行を挿入し、低いインデックス値を削除した場合に発生します。
- *CLUSTER* をクリックして、選択したテーブルをクラスター化するように PostgreSQL に指示します。

プロセス出力からステータスメッセージを除外するには、 **詳細メッセージ** スイッチを **いいえ** の位置に移動します。デフォルトでは、ステータスメッセージが含まれます。

ダイアログが完了したら、 *OK* をクリックしてバックグラウンドプロセスを開始します。メンテナンス操作を実行せずにダイアログを終了するには、 *Cancel* をクリックします。

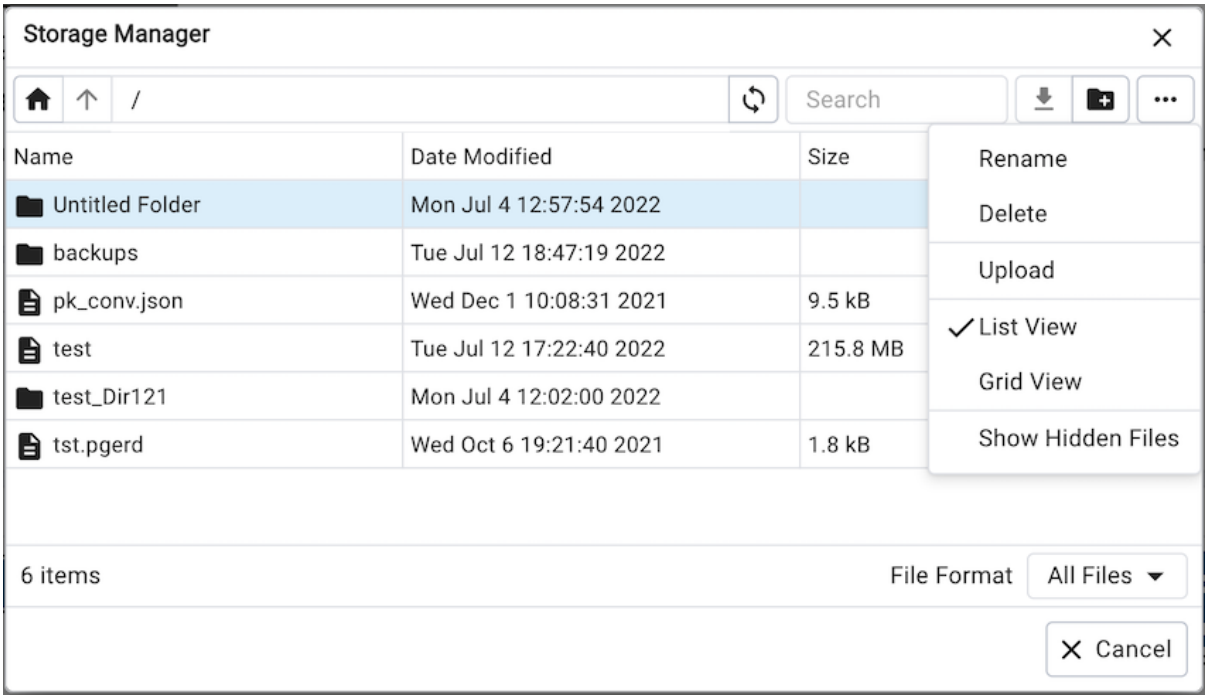
pgAdmin はメンテナンスプロセスをバックグラウンドで実行します。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

7.6 Storage Manager

ストレージマネージャー は、システムのストレージデバイスの管理に役立つ機能です。 **ストレージマネージャー** を使用して次のことができます。

- オペレーティングシステムファイルをダウンロード、アップロード、または管理します。この機能を使用するには、クライアントマシンで *pgAdmin* が **サーバーモード** で実行されている必要があります。
- クライアントマシンに **バックアップ** または **エクスポート** ファイル（カスタム、tar、およびプレーンテキスト形式）をダウンロードします。
- テーブルのダンプファイルをダウンロードします。

ツール メニューから **ストレージマネージャー** にアクセスできます。



ストレージマネージャー ウィンドウの上部にあるアイコンを使用して、ストレージを管理します。

`Home` アイコン|home|を使用してくださいホームディレクトリに戻ります。

`Up Arrow` アイコンを使用する|uparrow|前のディレクトリに戻ります。

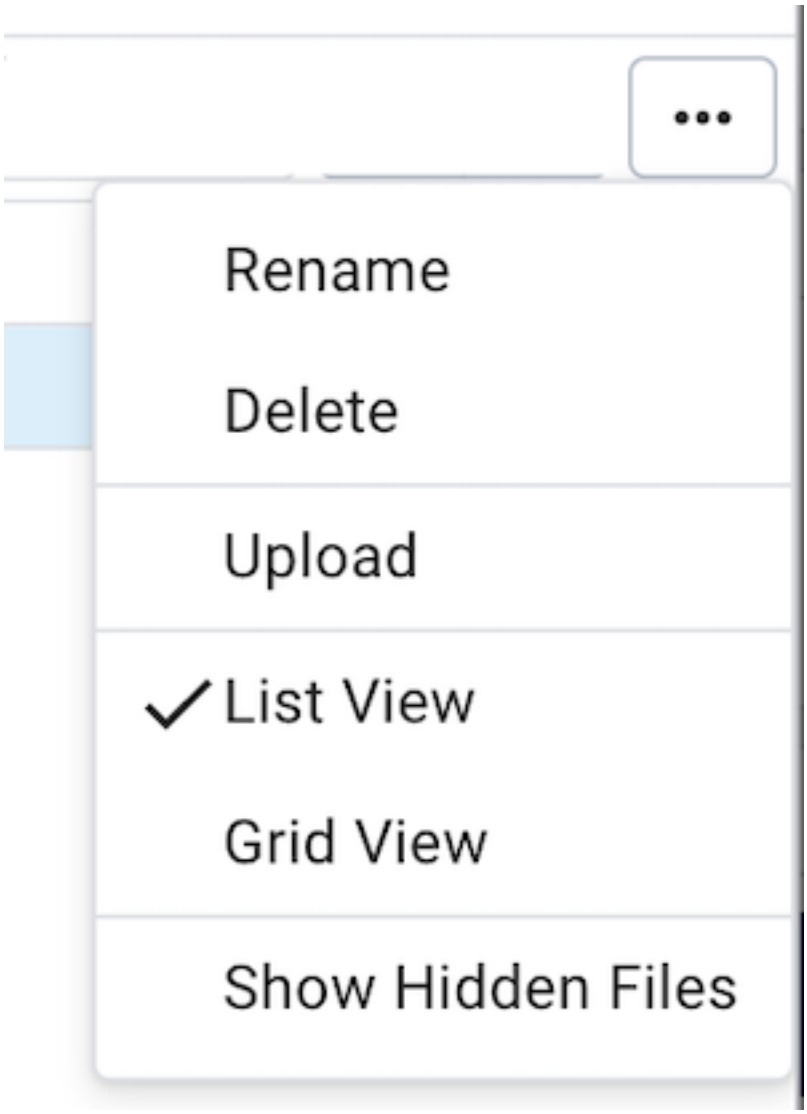
`Refresh` アイコンを使用|refresh|使用可能な最新のファイルを表示します。

`Download` アイコン|download|を選択しますをクリックして、選択したファイルをダウンロードします。

`New Folder` アイコン|フォルダー|を使用する新しいフォルダーを追加します。

フォーマット ドロップダウンリストを使用して、表示するファイルの形式を選択します。 *sql* 、 *csv* 、または *AllFiles* から選択します。

7.6.1 その他のオプション



メニュー	行動
名前を変更	ファイル/フォルダーの名前を変更するには、 **名前の変更** オプションをクリックします。
削除	ファイル/フォルダーの名前を変更するには、 **削除** オプションをクリックします。
アップロード	**アップロード** オプションをクリックして、現在のフォルダーに複数のファイルをアップロードします。
リストビュー	**リストビュー** オプションをクリックして、すべてのファイルとフォルダーをリストビューに表示します。

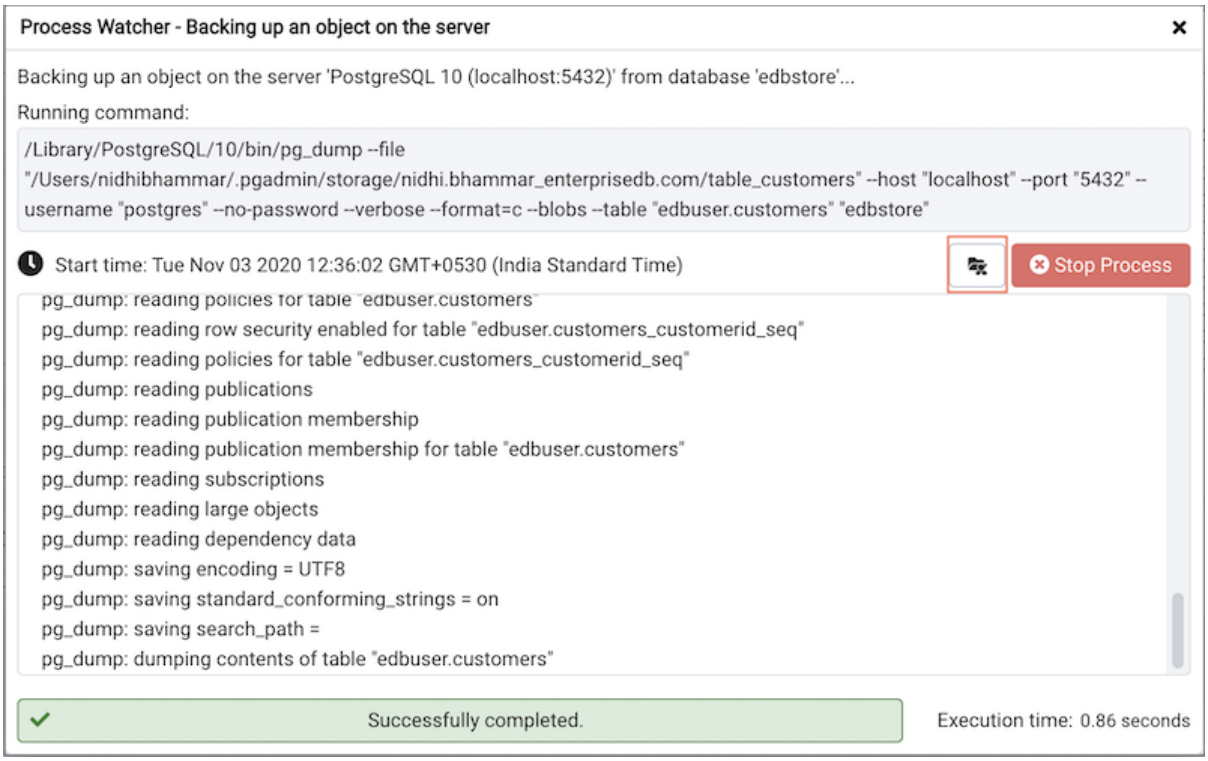
次のページに続く

表 1 – 前のページからの続き

メニュー	行動
グリッドビュー	**グリッドビュー** オプションをクリックして、すべてのファイルとフォルダーをグリッドビューで表示します。
隠しファイルを表示	**隠しファイルを表示** オプションをクリックして、隠しファイルとフォルダーを表示します。

Backup Dialog 、 *Backup Dialog* 、または *Backup Dialog* で取得したバックアップが正常に完了したら、 **ストレージマネージャー** を介してバックアップファイルをダウンロードすることもできます。

バックアップが正常に完了したら、アイコンをクリックして、 **プロセスウォッチャー** ウィンドウの **ストレージマネージャー** で現在のバックアップファイルを開きます。



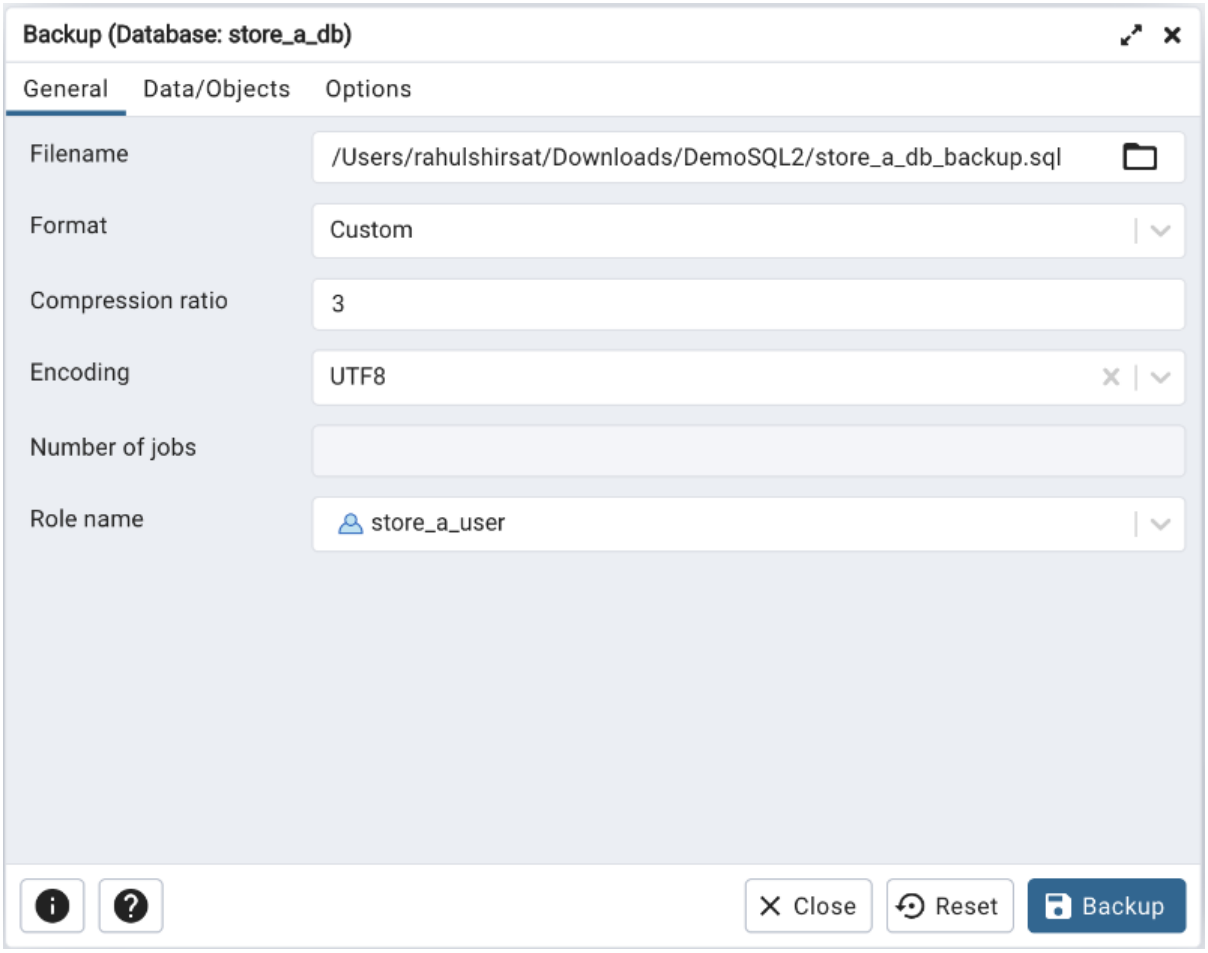
第8章　Backup and Restore

強力でありながら使いやすいバックアップと復元ツールは、pg_dump、pg_dumpall、および pg_restore を使用して、開発環境で使用するデータベースまたはデータベースオブジェクトのバックアップを作成する簡単な方法を提供します。

8.1 Backup Dialog

pgAdmin は *pg_dump* ユーティリティを使用して、プレーンテキストまたはアーカイブ形式でバックアップを簡単に作成できます。次に、クライアントアプリケーション（ *psql* や **クエリツール** など）を使用してプレーンテキストバックアップファイルを復元するか、Postgres *pg_restore* ユーティリティを使用してアーカイブされたバックアップを復元します。 *pg_dump* ユーティリティには、バックアップするすべてのデータベースオブジェクトへの読み取りアクセスが必要です。

単一のテーブル、スキーマ、またはデータベース全体をバックアップできます。 *pgAdmin* ツリーコントロールでバックアップソースの名前を選択し、右クリックしてコンテキストメニューを開き、 *Backup...* を選択して *Backup* ダイアログを開きます。選択したオブジェクトの名前がダイアログのタイトルバーに表示されます。



[**全般**] タブのフィールドを使用して、バックアップのパラメーターを指定します。

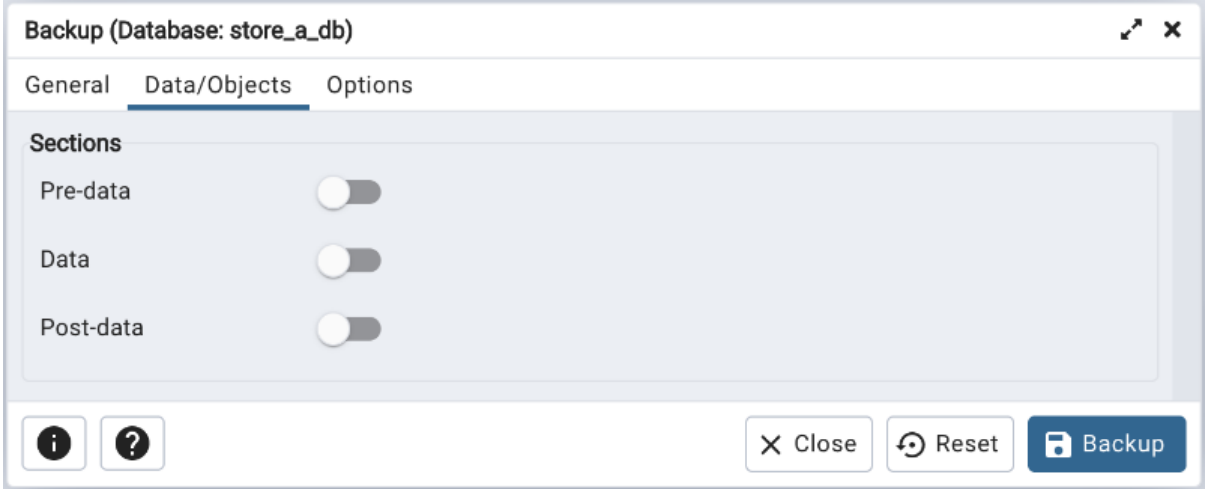
- **ファイル名** フィールドにバックアップファイルの名前を入力します。必要に応じて、右側の **ブラウザー** アイコン（...）を選択してディレクトリに移動し、アーカイブを含むファイルを選択します。
- **フォーマット** フィールドのドロップダウンリストボックスを使用して、アプリケーションに最適なフォーマットを選択します。各フォーマットには長所と短所があります。
  - *Custom* を選択して、 *pg_restore* で使用してデータベースのコピーを作成できるカスタムアーカイブファイルを作成します。カスタムアーカイブファイル形式は *pg_restore* で復元する必要があります。この形式では、バックアップファイルから復元するデータベースオブジェクトを選択できます。デフォルトで圧縮されているため、中規模から大規模のデータベースには **カスタム** アーカイブ形式をお勧めします。
  - *Tar* を選択して、 *pg_restore* で復元できる tar アーカイブファイルを生成します。tar 形式は圧縮をサポートしていません。
  - [*Plain*] を選択して、プレーンテキストのスクリプトファイルを作成します。プレーンテキストのスクリプトファイルには、 *psql* コマンドラインで実行してデータベースオブジェクトを再作成し、テーブルデータをロードできる SQL ステートメントとコマンドが含まれています。プレーンテキストのバックアップファイルは、 *psql* プログラムを使用してデータベースオブジェクトを復元する前に、必要に応じてテキストエディターで編集できます。小規模なデータベースには通常、 **プレーン** 形式をお勧めします。BLOB のスクリプトダンプはお勧めしません。スクリプト内の SQL コマンドは、データベースを最後に保存した状態に再構築します。プレーン

テキストスクリプトを使用して、別のマシンで、または（変更を加えて）データベースを再構築できます。

- *Directory* を選択して、 *pg_restore* での使用に適したディレクトリ形式のアーカイブを生成します。このファイル形式は、ダンプされるテーブルと BLOB ごとに 1 つのファイルと、ダンプされたオブジェクトを *pg_restore* が読み取ることができる形式で説明する **目次** ファイルを含むディレクトリを作成します。この形式はデフォルトで圧縮されています。

- **圧縮率** フィールドを使用して、バックアップの圧縮レベルを選択します。ゼロの値を指定すると、圧縮を使用しません。最大圧縮値 9 を指定します。tar アーカイブは圧縮をサポートしていないことに注意してください。
- **エンコーディング** ドロップダウンリストボックスを使用して、アーカイブに使用する文字エンコード方法を選択します。
- **ジョブの数** フィールド（該当する場合）を使用して、並列バックアップで同時にダンプされるテーブルの数を指定します。
- *Rolename* の横にあるドロップダウンリストボックスを使用して、バックアップを所有するロールを指定します。

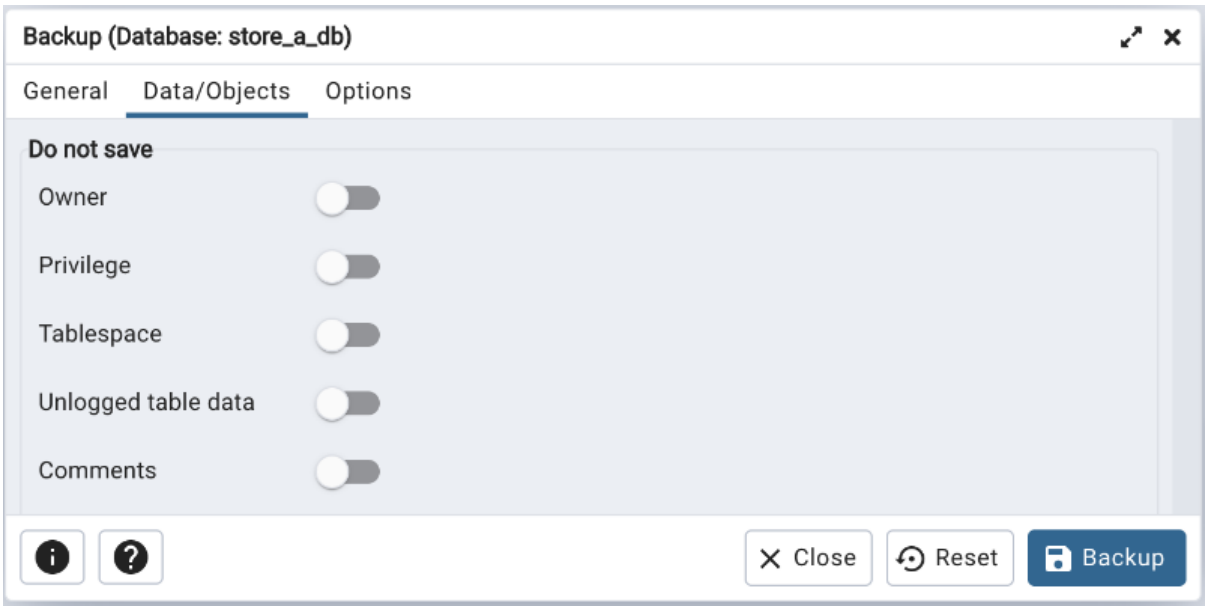
[**データ/オブジェクト**] タブをクリックして続行します。 **データ/オブジェクト** タブのフィールドを使用して、 *pg_dump* に対応するデータまたは pgAdmin オブジェクトに関連するオプションを提供します。



- **Sections** フィールドボックスのスイッチを移動して、バックアップするオブジェクトの一部を選択します。
  - **プレデータ** の横にあるスイッチを右の位置に移動して、データまたはポストデータ項目リストに含まれていないすべてのデータ定義項目を含めます。
  - *Data* の横にあるスイッチを右の位置に移動して、実際のテーブルデータ、ラージオブジェクトのコンテンツ、シーケンス値をバックアップします。
  - *Post-data* の横にあるスイッチを右側に移動して、検証済みのチェック制約以外のインデックス、トリガー、ルール、および制約の定義を含めます。



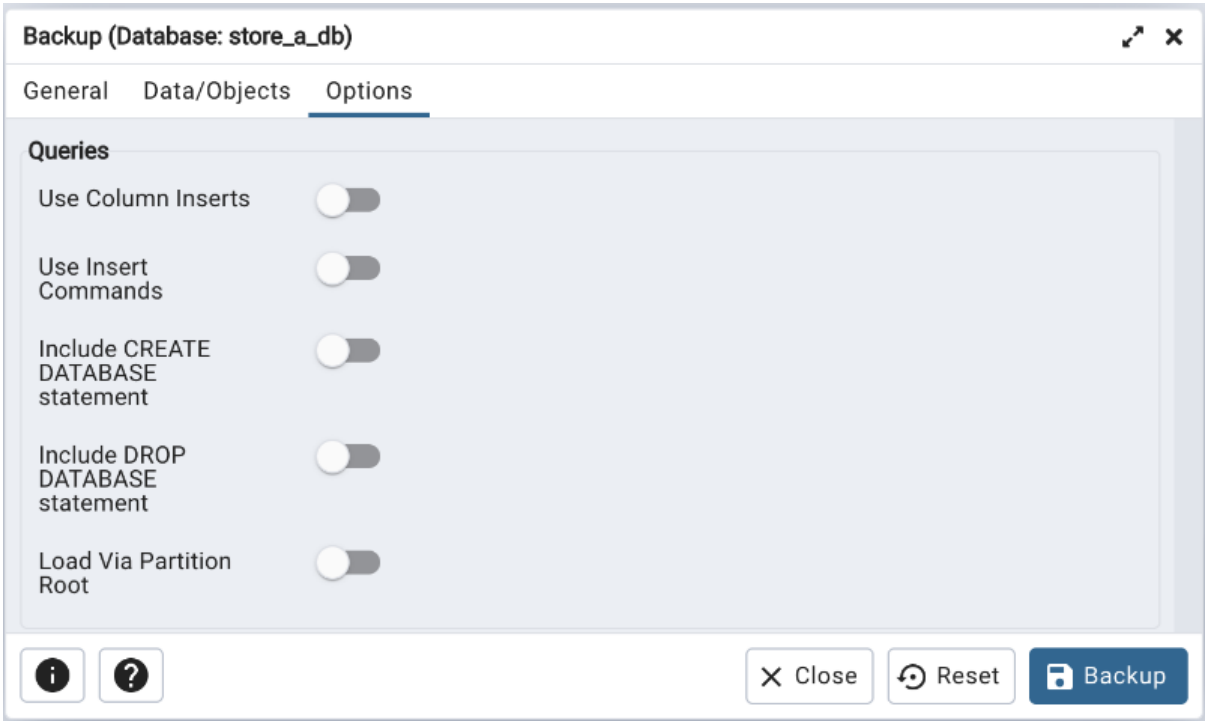
- **Type of objects** フィールドボックスのスイッチを移動して、バックアップするオブジェクトの種類に関する詳細を指定します。
  - **データのみ** の横にあるスイッチを右の位置に移動して、データへのバックアップを制限します。
  - スイッチを **スキーマのみ** の横に移動して、バックアップをスキーマレベルのデータベースオブジェクトに制限します。
  - *Blobs* の横にあるスイッチを左側に移動して、バックアップから大きなオブジェクトを除外します。



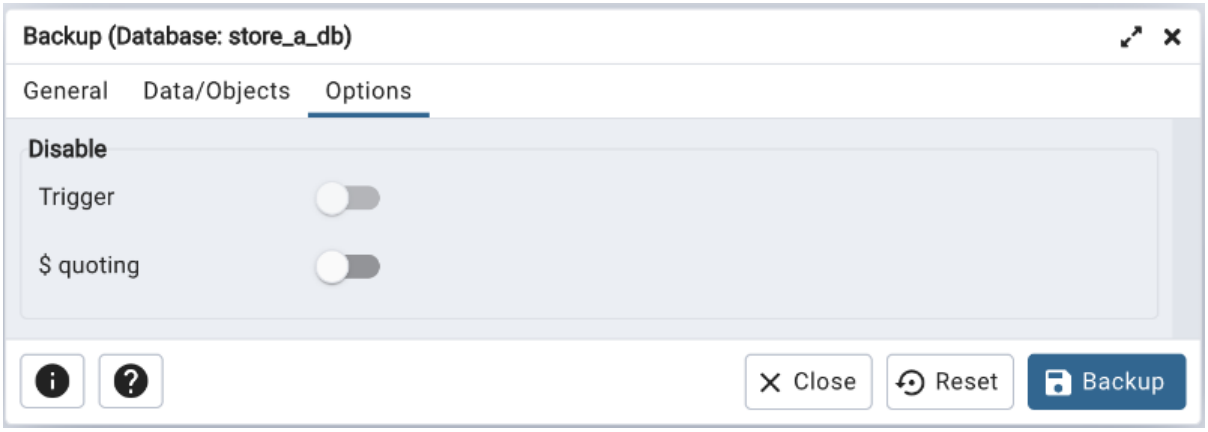
- **Do not save** フィールドボックスのスイッチを移動して、バックアップに含まれないオブジェクトを選択します。
  - *Owner* の横にあるスイッチを右の位置に移動して、オブジェクトの所有権を設定するコマンドを除外します。
  - *Privilege* の横にあるスイッチを右側に移動して、アクセス権限を作成するコマンドを除外します。
  - *Tablespace* の横にあるスイッチを右の位置に移動して、テーブルスペースを除外します。

- **ログに記録されていないテーブルデータ** の横にあるスイッチを右の位置に移動して、ログに記録されていないテーブルのコンテンツを除外します。
- *Comments* の横にあるスイッチを右の位置に移動して、コメントを設定するコマンドを除外します。 **Note:** このオプションは、11 以上のデータベースサーバでのみ表示されます。

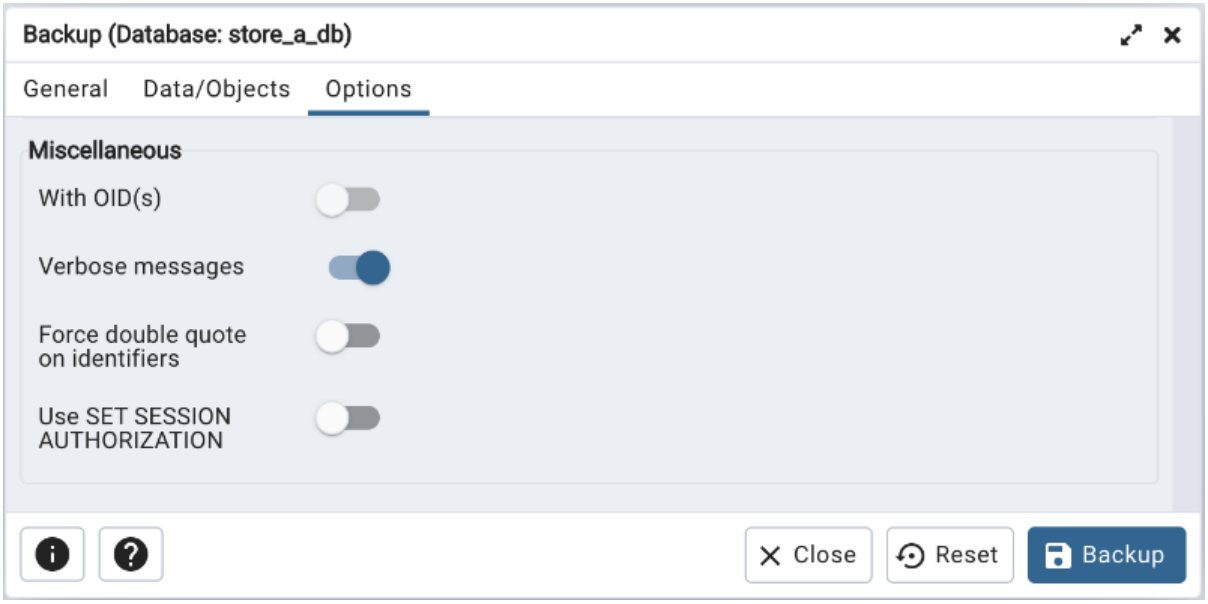
[**オプション**] タブをクリックして続行します。これらの追加フィールドを使用して、ddl ステートメント、詳細メッセージを含める、または *pg_dump* オプションに対応する setsessionauthorization を使用するなどのオプションを指定します。



- **Queries** フィールドボックスのスイッチを移動して、バックアップに含めるステートメントの種類を指定します。
  - *UseColumnInserts* の横にあるスイッチを右側に移動して、データを INSERT ステートメントの形式でダンプし、明示的な列名を含めます。注：これにより、バックアップからの復元が遅くなる場合があります。
  - *UseInsert* **コマンド** の横にあるスイッチを右側に移動して、COPY コマンドを使用するのではなく、INSERT ステートメントの形式でデータをダンプします。注：これにより、バックアップからの復元が遅くなる場合があります。
  - *IncludeCREATEDATABASEstatement* の横にあるスイッチを右側に移動して、バックアップの復元時に新しいデータベースを作成するコマンドをバックアップに含めます。
  - *IncludeDROPDATABASE* **ステートメント** の横にあるスイッチを右側に移動して、バックアップ中にオブジェクトを再作成する前に同じ名前の既存のデータベースオブジェクトを削除するコマンドをバックアップに含めます。
  - *LoadViaPartitionRoot* の横にあるスイッチを右側に移動するため、パーティションテーブルの COPY または INSERT ステートメントをダンプするときは、パーティション自分自身ではなく、それを含むパーティション階層のルートをターゲットにします。 **Note:** このオプションは、11 以上のデータベースサーバでのみ表示されます。



- **Disable** フィールドボックスのスイッチを移動して、バックアップから除外するステートメントのタイプを指定します。
  - **トリガー**（データのみのバックアップを作成するときにアクティブ）の横にあるスイッチを右の位置に移動して、データのロード中にターゲットテーブルでトリガーを無効にするコマンドを含めます。
  - *$quoting* の横にあるスイッチを右の位置に移動して、ファンクション本体内でドルクォートを有効にします。無効にすると、ファンクション本体は SQL 標準の文字列構文を使用して引用符で囲まれます。



- **Miscellaneous** フィールドボックスのスイッチを移動して、その他のバックアップオプションを指定します。
  - *OIDs* の横にあるスイッチを右の位置に移動して、各テーブルのテーブルデータの一部としてオブジェクト識別子を含めます。
  - **詳細メッセージ** の横にあるスイッチを左側に移動して、 *pg_dump* に詳細メッセージを除外します。
  - **識別子の二重引用符を強制する** の横にあるスイッチを右の位置に移動して、すべての識別子を強制的に引用します。

  - *UseSETSESSIONAUTHORIZATION* の横にあるスイッチを右側に移動して、（ALTEROWNER コマンドの代わりに）SETSESSIONAUTHORIZATION コマンドを使用してオブジェクトの所有権を判断するステートメントを含めます。

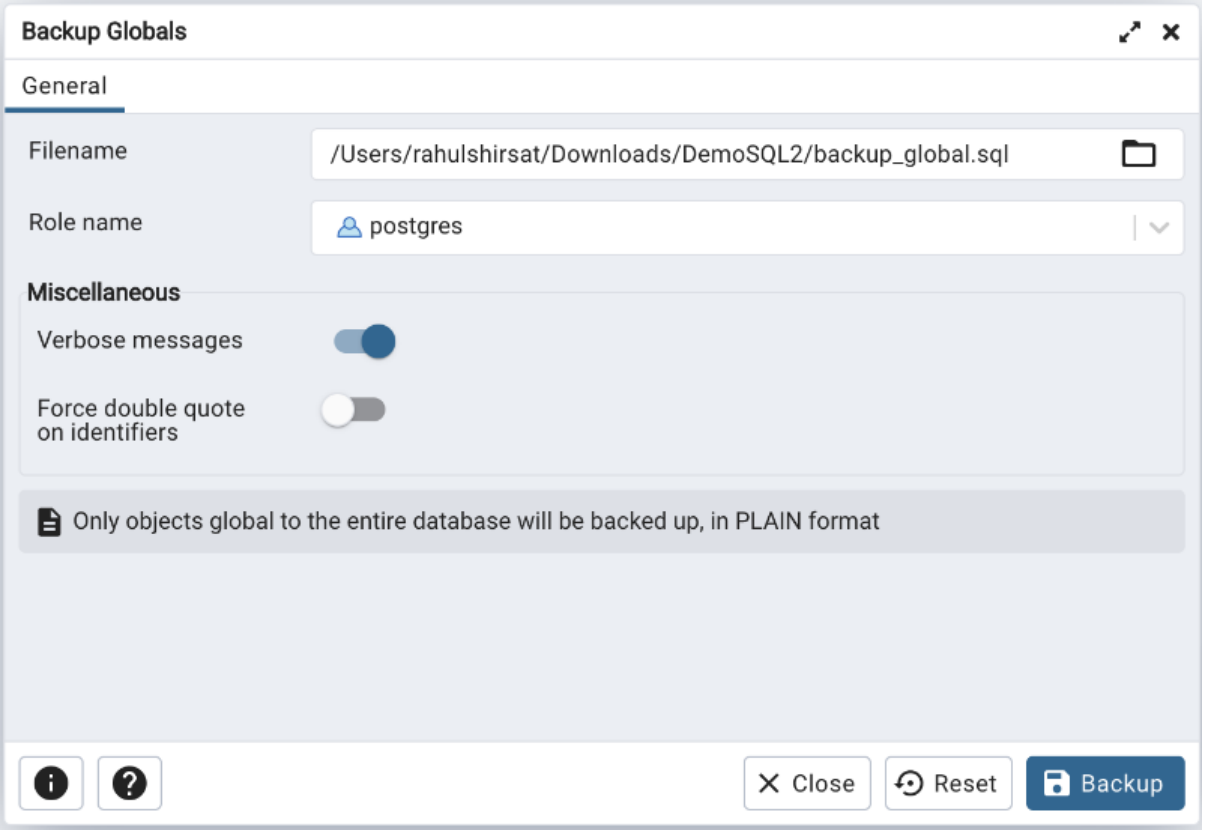
pg_dump コマンドに組み込まれる詳細を指定したら：

- **バックアップ** ボタンをクリックして、 **バックアップ** ダイアログでの選択に基づいてバックアップをビルドするコマンドを実行します。
- 作業を保存せずに終了するには、 *Cancel* ボタンをクリックします。

pgAdmin はバックアッププロセスをバックグラウンドで実行します。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

8.2 Backup Globals Dialog

BackupGlobals ダイアログを使用して、クラスター内のすべてのデータベースオブジェクトと、それらのデータベースで共有されるグローバルオブジェクトを再作成するプレーンテキストスクリプトを作成します。グローバルオブジェクトには、テーブルスペース、ロール、およびオブジェクトプロパティが含まれます。pgAdmin **クエリツール** を使用して、プレーンテキストスクリプトを再生し、バックアップ内のオブジェクトを再作成できます。



[**全般**] タブのフィールドを使用して、次を指定します。

- **ファイル名** フィールドにバックアップファイルの名前を入力します。必要に応じて、右側にある **ブラウザー** アイコン（省略記号）を選択してディレクトリに移動し、アーカイブを含むファイルを選択します。

- **ロール名** の横にあるドロップダウンリストボックスを使用して、選択したサーバーで接続特権を持つロールを指定します。このロールは、バックアップ中の認証に使用されます。

Miscellaneous フィールドボックスのスイッチを移動して、バックアップに含めるステートメントの種類を指定します。

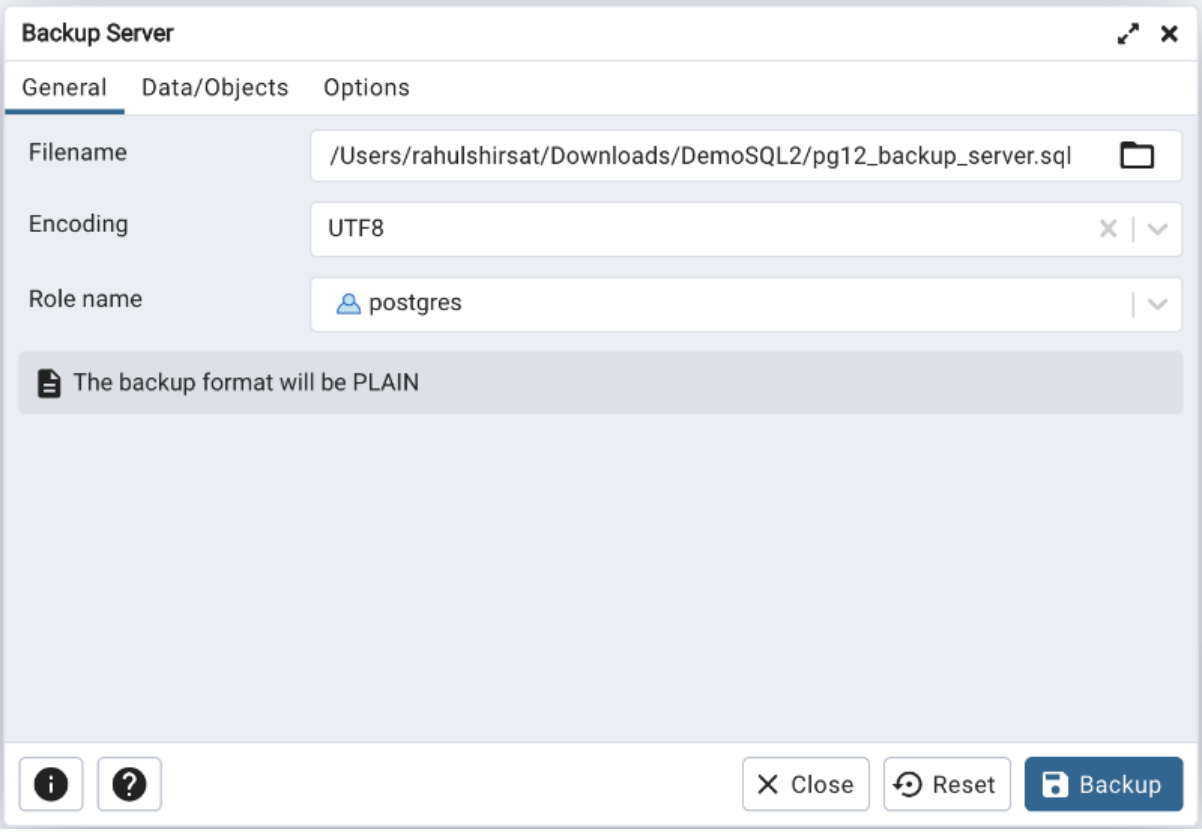
- **詳細メッセージ** スイッチを **いいえ** の位置に移動して、バックアップからステータスメッセージを除外します。デフォルトは **はい** です。
- 大文字と小文字を変更せずに識別子に **強制的に二重引用符を** スイッチを **はい** の位置に移動します。デフォルトは **いいえ** です。

Backup ボタンをクリックして、選択に基づいてコマンドをビルドして実行します。作業を保存せずに終了するには、 *Cancel* ボタンをクリックします。

pgAdmin はバックアッププロセスをバックグラウンドで実行します。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

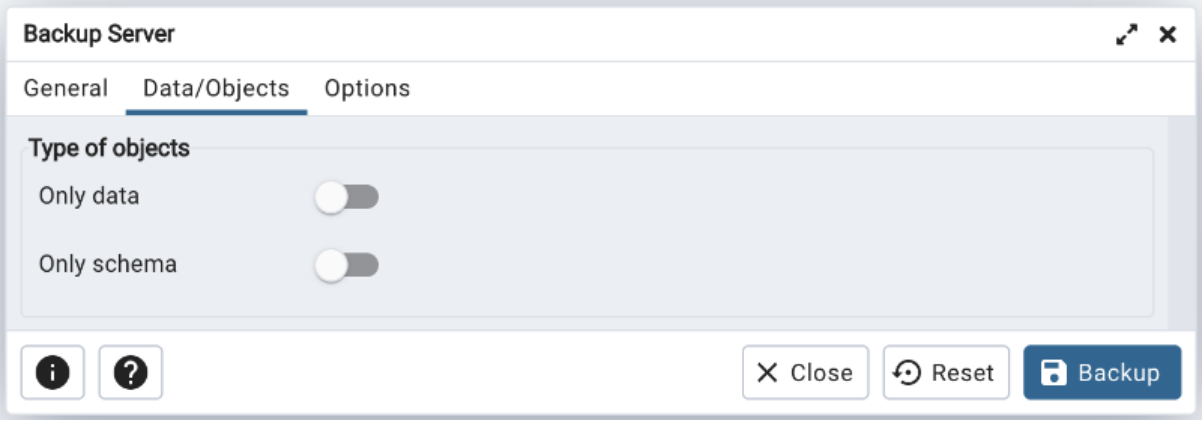
8.3 Backup Server Dialog

バックアップサーバー ダイアログを使用して、選択したサーバーを再作成するプレーンテキストスクリプトを作成します。pgAdmin **クエリツール** を使用して、プレーンテキストスクリプトを再生し、サーバーを再作成できます。



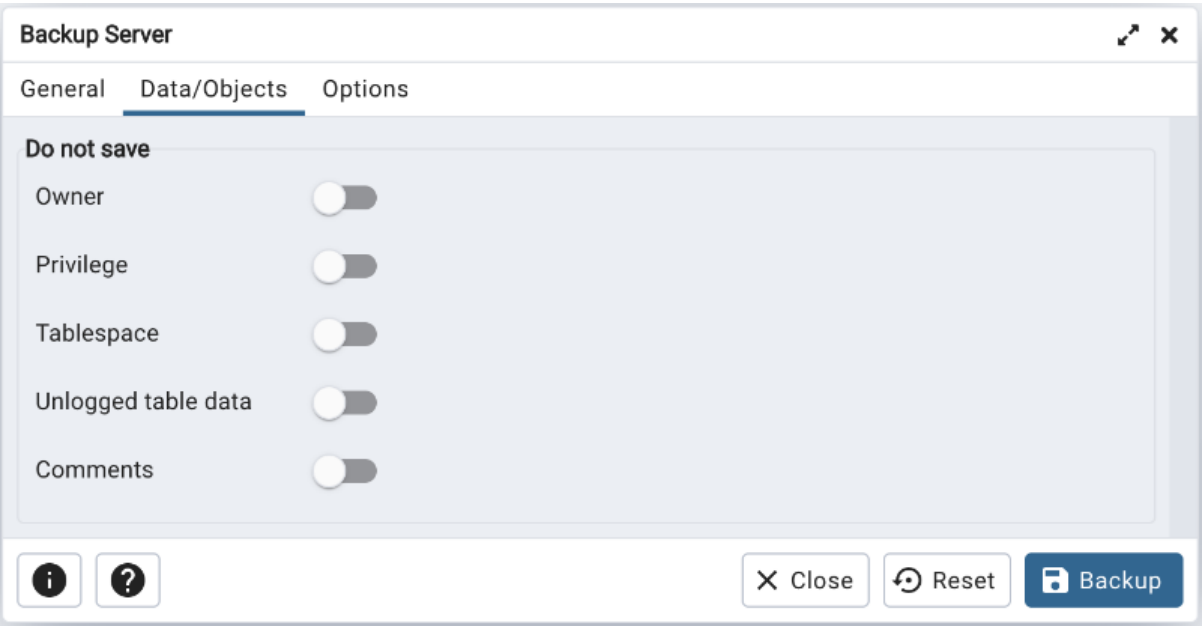
[**全般**] タブのフィールドを使用して、次を指定します。

- **ファイル名** フィールドにバックアップファイルの名前を入力します。必要に応じて、右側にある **ブラウザー** アイコン（省略記号）を選択してディレクトリに移動し、アーカイブを含むファイルを選択します。
- **エンコーディング** ドロップダウンリストボックスを使用して、アーカイブに使用する文字エンコード方法を選択します。 **Note:** このオプションは、11 以上のデータベースサーバでのみ表示されます。
- **ロール名** の横にあるドロップダウンリストボックスを使用して、選択したサーバーで接続特権を持つロールを指定します。このロールは、バックアップ中の認証に使用されます。



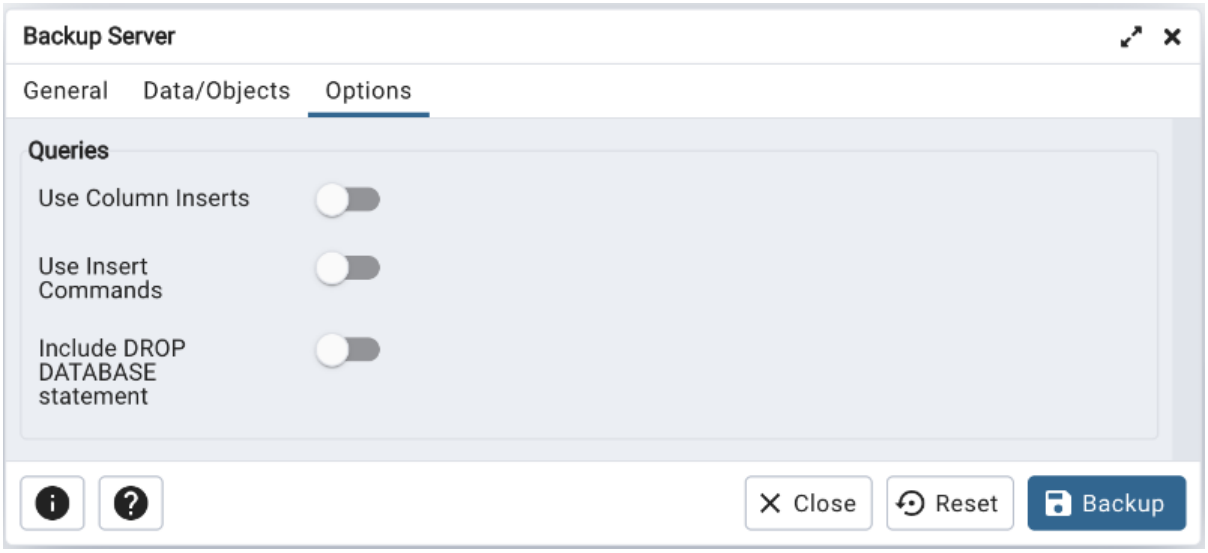
[**データ/オブジェクト**] タブをクリックして続行します。 **データ/オブジェクト** タブのフィールドを使用して、 *pg_dump* に対応するデータまたは pgAdmin オブジェクトに関連するオプションを提供します。

- **Type of objects** フィールドボックスのスイッチを移動して、バックアップするオブジェクトの種類に関する詳細を指定します。
  - **データのみ** の横にあるスイッチを右の位置に移動して、データへのバックアップを制限します。
  - スイッチを **スキーマのみ** の横に移動して、バックアップをスキーマレベルのデータベースオブジェクトに制限します。

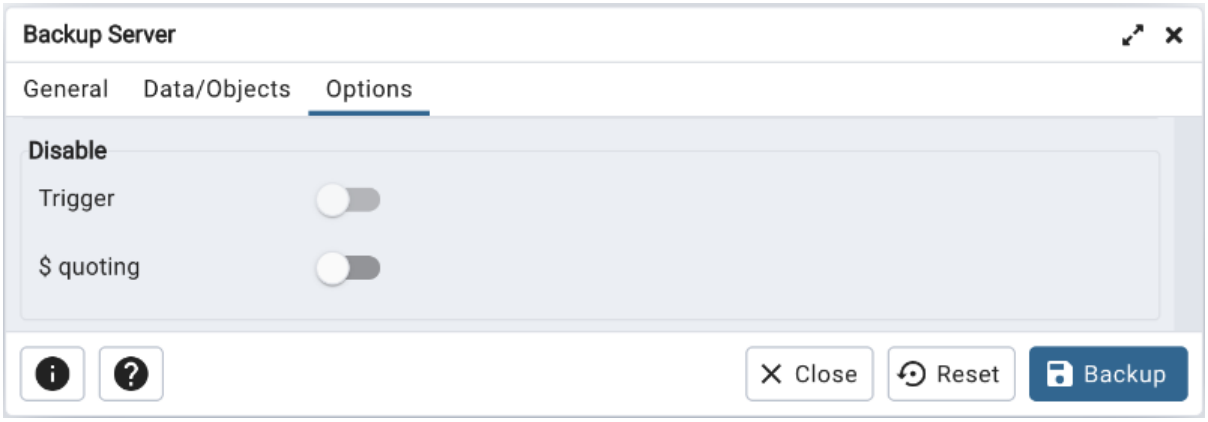


- **Do not save** フィールドボックスのスイッチを移動して、バックアップに含まれないオブジェクトを選択します。

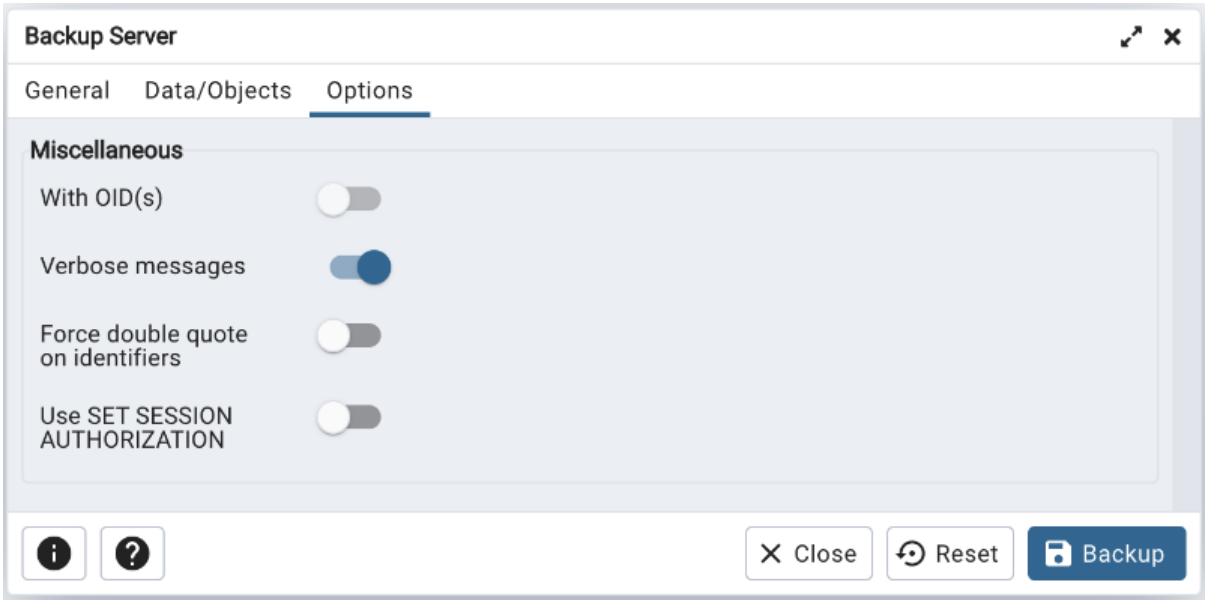
- *Owner* の横にあるスイッチを右の位置に移動して、オブジェクトの所有権を設定するコマンドを除外します。
- *Privilege* の横にあるスイッチを右側に移動して、アクセス権限を作成するコマンドを除外します。
- *Tablespace* の横にあるスイッチを右の位置に移動して、テーブルスペースを除外します。
- **ログに記録されていないテーブルデータ** の横にあるスイッチを右の位置に移動して、ログに記録されていないテーブルのコンテンツを除外します。
- *Comments* の横にあるスイッチを右の位置に移動して、コメントを設定するコマンドを除外します。 **Note:** このオプションは、11 以上のデータベースサーバでのみ表示されます。



- **Queries** フィールドボックスのスイッチを移動して、バックアップに含めるステートメントの種類を指定します。
  - *UseColumnInserts* の横にあるスイッチを右側に移動して、データを INSERT ステートメントの形式でダンプし、明示的な列名を含めます。注：これにより、バックアップからの復元が遅くなる場合があります。
  - *UseInsert* **コマンド** の横にあるスイッチを右側に移動して、COPY コマンドを使用するのではなく、INSERT ステートメントの形式でデータをダンプします。注：これにより、バックアップからの復元が遅くなる場合があります。
  - *IncludeDROPDATABASE* **ステートメント** の横にあるスイッチを右側に移動して、バックアップ中にオブジェクトを再作成する前に同じ名前の既存のデータベースオブジェクトを削除するコマンドをバックアップに含めます。



- **Disable** フィールドボックスのスイッチを移動して、バックアップから除外するステートメントのタイプを指定します。
  - **トリガー**（データのみのバックアップを作成するときにアクティブ）の横にあるスイッチを右の位置に移動して、データのロード中にターゲットテーブルでトリガーを無効にするコマンドを含めます。
  - *$quoting* の横にあるスイッチを右の位置に移動して、ファンクション本体内でドルクォートを有効にします。無効にすると、ファンクション本体は SQL 標準の文字列構文を使用して引用符で囲まれます。



- **Miscellaneous** フィールドボックスのスイッチを移動して、その他のバックアップオプションを指定します。
  - *OIDs* の横にあるスイッチを右の位置に移動して、各テーブルのテーブルデータの一部としてオブジェクト識別子を含めます。
  - **詳細メッセージ** の横にあるスイッチを左側に移動して、 *pg_dump* に詳細メッセージを除外します。
  - **識別子の二重引用符を強制する** の横にあるスイッチを右の位置に移動して、すべての識別子を強制的に引用します。

- *UseSETSESSIONAUTHORIZATION* の横にあるスイッチを右側に移動して、（ALTEROWNER コマンドの代わりに）SETSESSIONAUTHORIZATION コマンドを使用してオブジェクトの所有権を判断するステートメントを含めます。

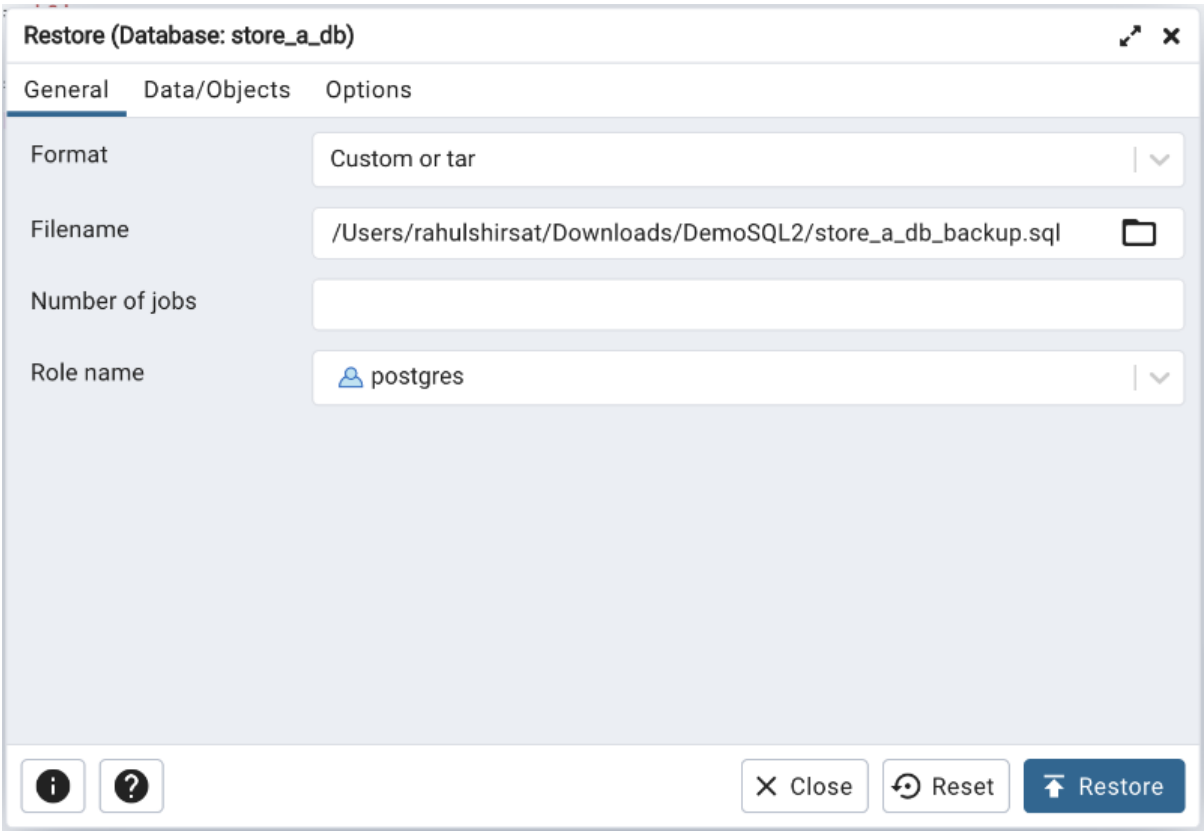
Backup ボタンをクリックして、選択に基づいてコマンドをビルドして実行します。作業を保存せずに終了するには、 *Cancel* ボタンをクリックします。

pgAdmin はバックアッププロセスをバックグラウンドで実行します。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

8.4 Restore Dialog

Restore ダイアログは、pgAdmin *Backup* ダイアログで取得したカスタム、tar、またはディレクトリ形式のバックアップを使用して、データベースまたはデータベースオブジェクトを再作成する簡単な方法を提供します。 *Backup* ダイアログは、pg_dump クライアントユーティリティのオプションを呼び出します。 *Restore* ダイアログは、pg_restore クライアントユーティリティのオプションを呼び出します。

You can use the *Query Tool* to play back the script created during a plain-text backup made with the *Backup* dialog. For more information about backing up or restoring, please refer to the documentation for pg_dump or pg_restore.



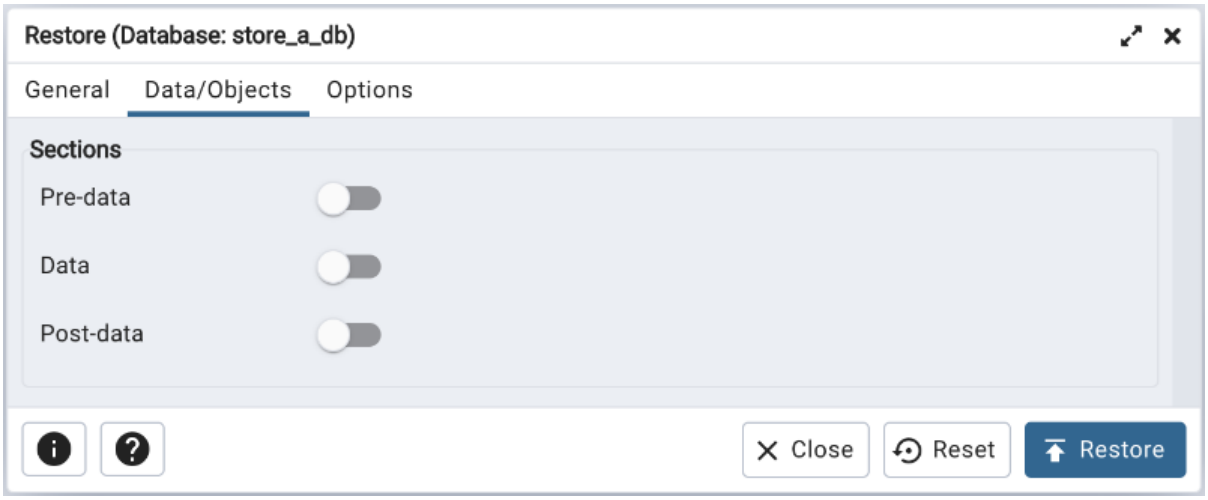
[**全般**] タブのフィールドを使用して、復元プロセスに関する一般的な情報を指定します。

- **フォーマット** フィールドのドロップダウンリストボックスを使用して、バックアップファイルの形式を選択します。
  - カスタムアーカイブファイルから復元する **カスタムまたは** *tar* を選択して、バックアップされたオブジェクトのコピーを作成します。

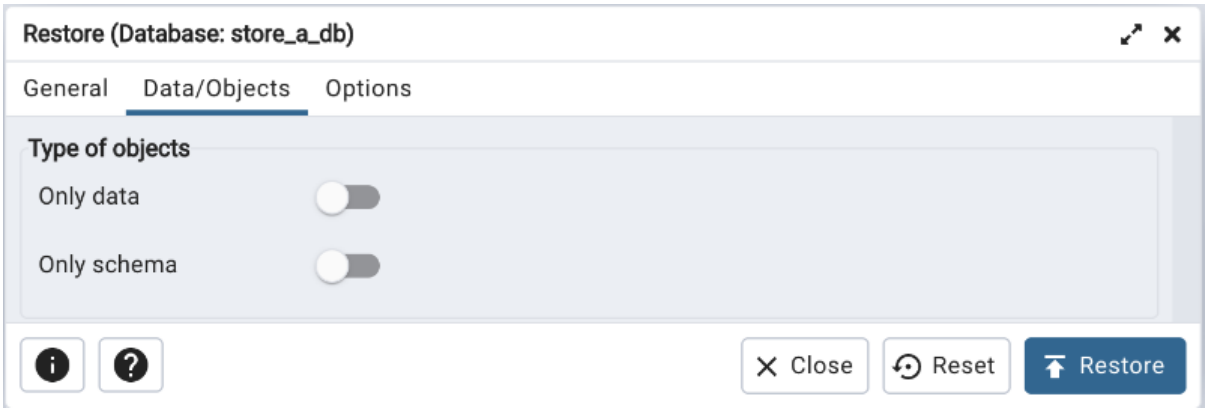
  - 圧縮されたディレクトリ形式のアーカイブから復元するには、 **ディレクトリ** を選択します。

- **ファイル名** フィールドにバックアップファイルへの完全なパスを入力します。必要に応じて、右側の **ブラウザー** アイコン（省略記号）を選択してディレクトリに移動し、アーカイブを含むファイルを選択します。
- **ジョブ数** フィールドを使用して、pg_restore が複数の（同時）ジョブを使用して復元を処理するかどうかを指定します。各ジョブは、サーバーへの個別の接続を使用します。
- *Rolename* の横にあるドロップダウンリストボックスを使用して、復元プロセス中にサーバーでの認証に使用されるロールを指定します。

[**データ/オブジェクト**] タブをクリックして続行します。 **データ/オブジェクト** タブのフィールドを使用して、データまたは *pg_restore* オプションに対応する pgAdmin オブジェクトを指定します。

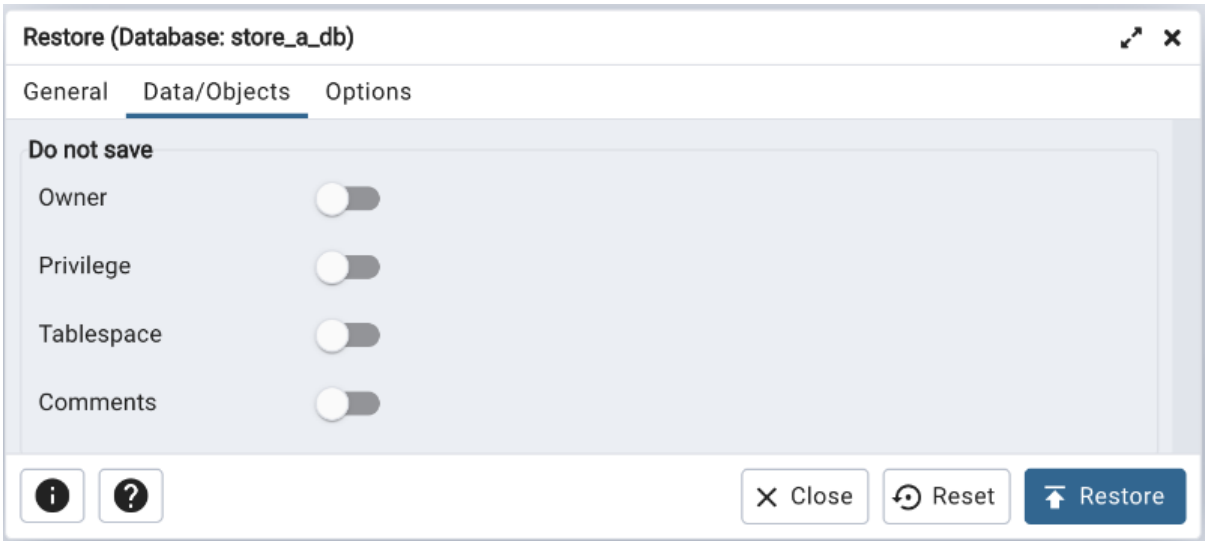


- **Sections** ボックスのスイッチを使用して、復元するコンテンツを指定します。
  - **プレデータ** の横にあるスイッチを右の位置に移動して、データまたはポストデータ項目リストに含まれていないすべてのデータ定義項目を復元します。
  - *Data* の横にあるスイッチを右の位置に移動して、実際のテーブルデータ、ラージオブジェクトの内容、およびシーケンス値を復元します。
  - *Post-data* の横にあるスイッチを正しい位置に移動して、インデックス、トリガー、ルール、および制約（検証済みのチェック制約を除く）の定義を復元します。



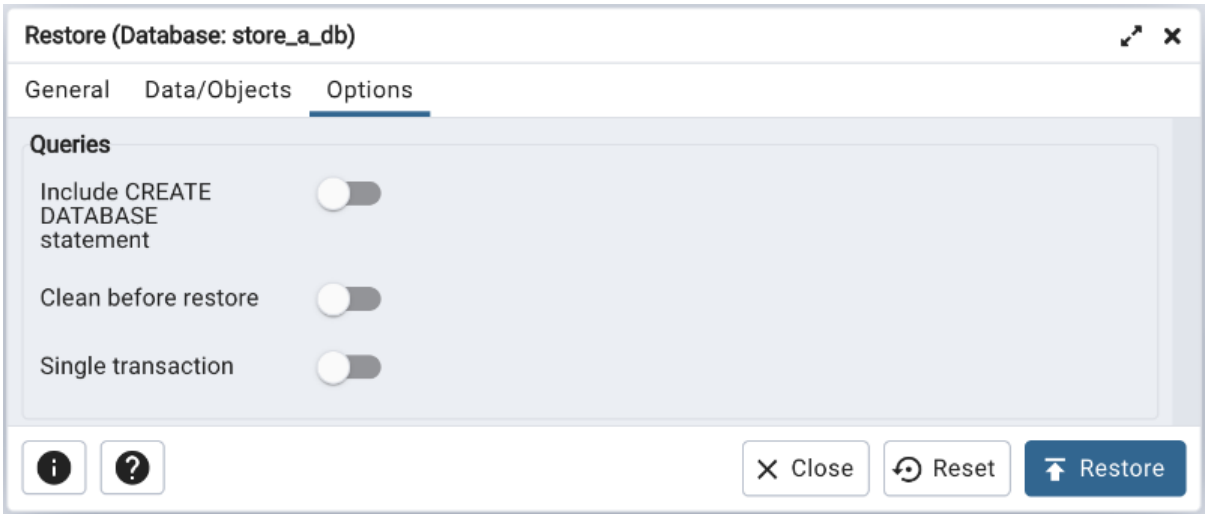
- **Type of objects** ボックスのスイッチを使用して、復元するオブジェクトを指定します。

- **データのみ** の横にあるスイッチを右の位置に移動して、復元をデータに制限します。
- [**スキーマのみ**] の横にあるスイッチを移動して、復元をスキーマレベルのデータベースオブジェクトに制限します。



- **Do not save** ボックスのスイッチを使用して、復元しないオブジェクトを指定します。
  - *Owner* の横にあるスイッチを右の位置に移動して、オブジェクトの所有権を設定するコマンドを除外します。
  - *Privilege* の横にあるスイッチを右側に移動して、アクセス権限を作成するコマンドを除外します。
  - *Tablespace* の横にあるスイッチを右の位置に移動して、テーブルスペースを除外します。
  - *Comments* の横にあるスイッチを右の位置に移動して、コメントを設定するコマンドを除外します。 **Note:** このオプションは、11 以上のデータベースサーバでのみ表示されます。

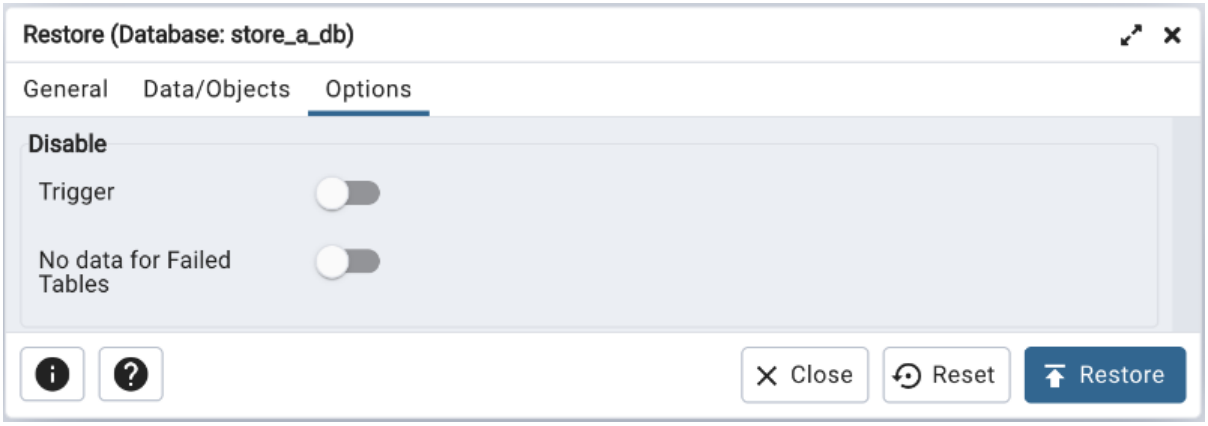
[**オプション**] タブをクリックして続行します。これらの追加フィールドを使用して、復元前のクリーニング、詳細メッセージ、または *pg_restore* オプションに対応するセッション承認の使用を指定します。



- **Queries** ボックスのスイッチを使用して、復元に含めるステートメントのタイプを指定します。
  - *IncludeCREATEDATABASE* **ステートメント** の横にあるスイッチを右側に移動して、復元を実行

する前に新しいデータベースを作成するコマンドを含めます。

- *Cleanbeforerestore* の横にあるスイッチを右の位置に移動して、復元する前に既存の各データベースオブジェクト（およびデータ）を削除します。
- *Singletransaction* の横にあるスイッチを右の位置に移動して、単一のトランザクションとして復元を実行します（つまり、発行されたコマンドを *BEGIN/COMMIT* でラップします）。これにより、すべてのコマンドが正常に完了するか、変更が適用されません。このオプションは *--exit-on-error* を意味します。



- **Disable** ボックスのスイッチを使用して、復元から除外するステートメントのタイプを指定します。
  - **トリガー**（データのみの復元を作成するときにアクティブ）の横にあるスイッチを右の位置に移動して、データのロード中にターゲットテーブルでトリガーを無効にするコマンドを含めます。
  - *NodataforFailedTables* の横にあるスイッチを右側に移動して、トリガーに失敗したデータを無視します。



- **Miscellaneous/Behavior** ボックスのスイッチを使用して、その他の復元オプションを指定します。
  - **詳細メッセージ** の横にあるスイッチを左に移動して、 *pg_restore* に詳細メッセージを除外します。
  - *UseSETSESSIONAUTHORIZATION* の横にあるスイッチを右側に移動して、（ALTEROWNER コマンドの代わりに）SETSESSIONAUTHORIZATION コマンドを使用してオブジェクトの所有権を判断するステートメントを含めます。

- *Exitonerror* の横にあるスイッチを右側に移動して、SQL コマンドの送信でエラーが発生した場合に *pg_restore* に復元を終了するように指示します。デフォルトでは続行し、復元の最後にエラーの数を表示します。

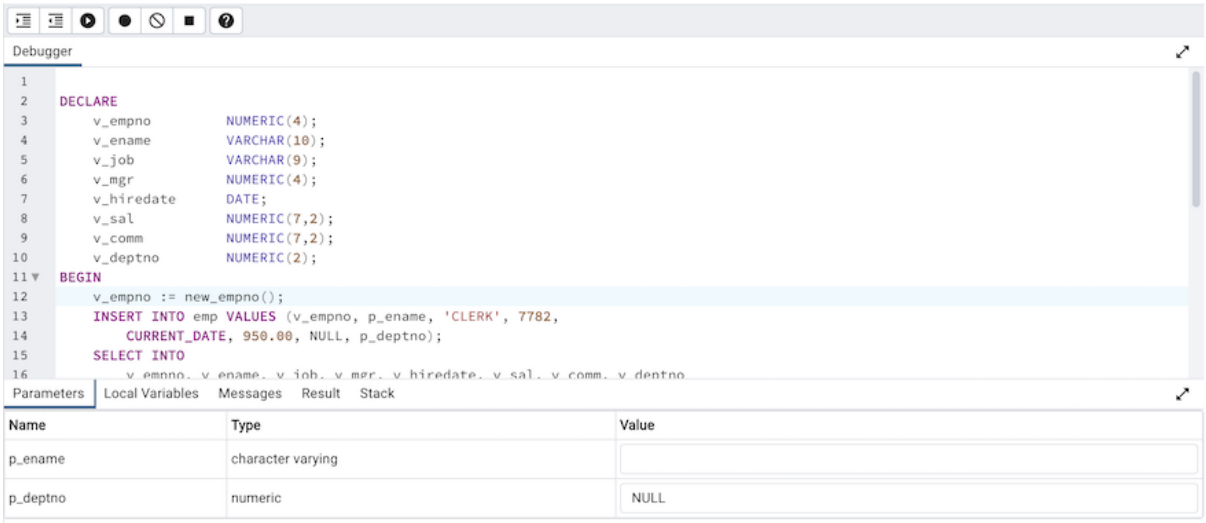
pg_restore コマンドに組み込む詳細を指定したら、 *Restore* ボタンをクリックしてプロセスを開始するか、*Cancel* ボタンをクリックして作業を保存せずに終了します。復元が成功したかどうかを確認するポップアップが表示されます。

pgAdmin は復元プロセスをバックグラウンドで実行します。 *Processes* タブで実行中のステータスとログを使用して、すべてのバックグラウンドプロセスを表示できます

第9章　Developer Tools

pgAdmin *Tools* メニューには、複雑な SQL コマンドの実行と分析、データの管理、および PL/SQL コードのデバッグに使用できる強力な開発者ツールのリストが表示されます。

9.1 Debugger



デバッガを使用して、PostgreSQL の PL/pgSQL ファンクション、および EDBPostgresAdvancedServer の EDB-SPL ファンクション、ストアドプロシージャ、パッケージをデバッグできます。デバッガーは、PostgreSQL インストールの拡張機能として利用でき、AdvancedServer の一部として配布されます。デバッガーを使用するには、スーパーユーザー権限が必要です。

デバッガーを使用する前に、 *postgresql.conf* ファイルを変更して、サーバーサイドのデバッガーコンポーネントを *shared_preload_libraries* パラメーターの値に追加する必要があります。

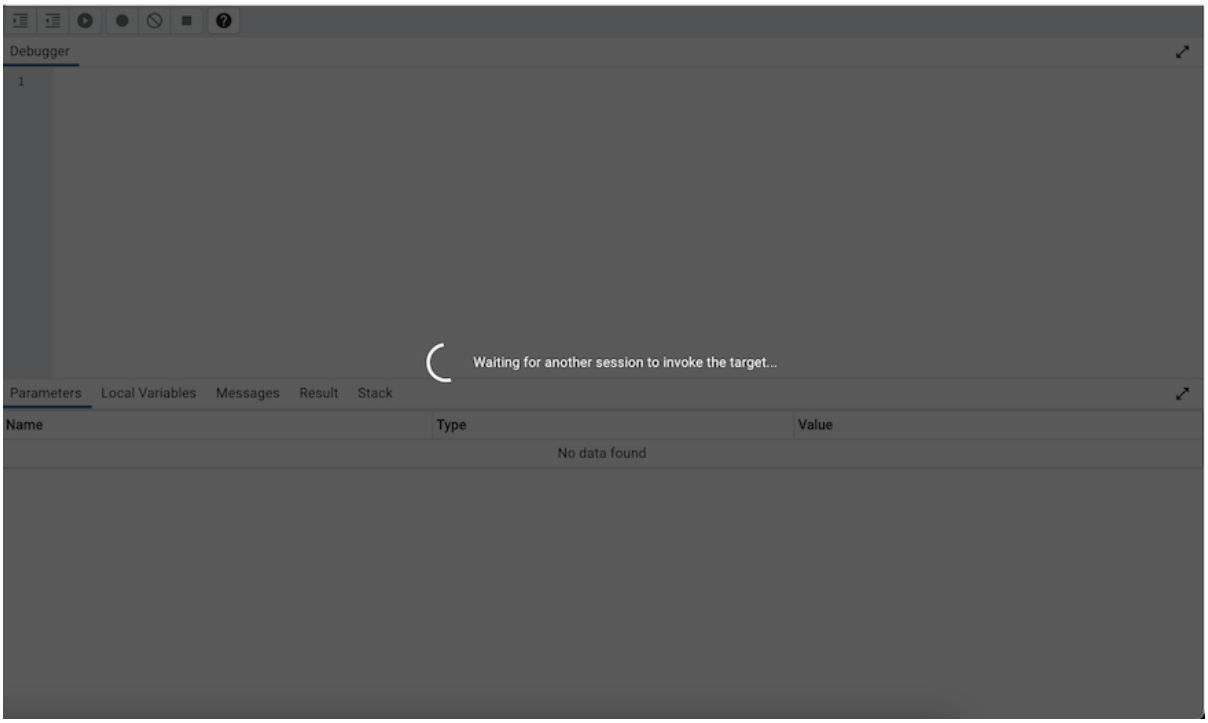
```
shared_preload_libraries='$libdir/plugin_debugger'
```

shared_preload_libraries パラメーターを変更した後、サーバーを再起動して変更を適用します。

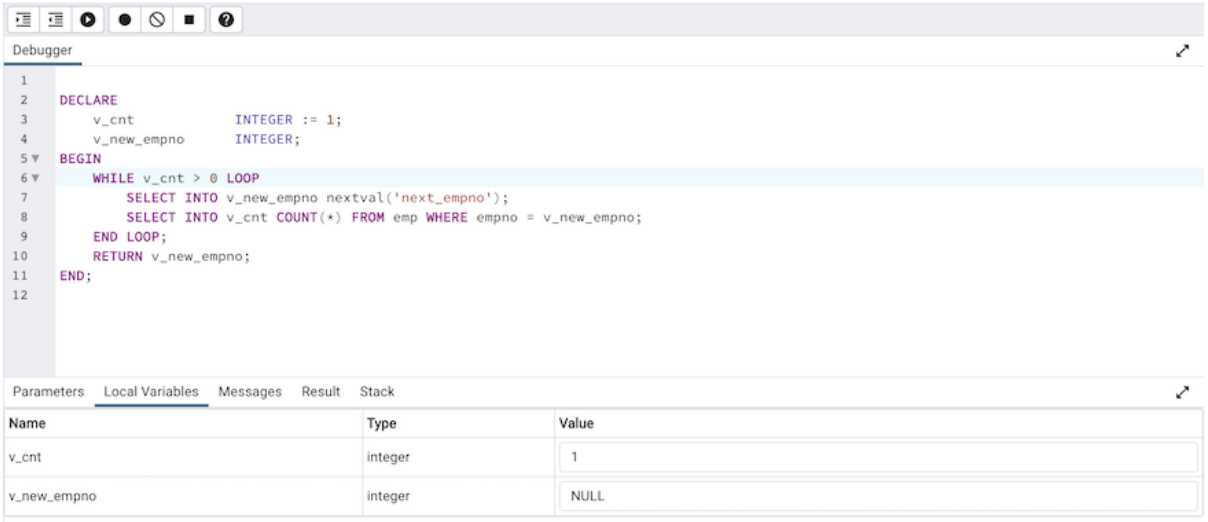
デバッガーは、ターゲット関数またはプロシージャのインコンテキストデバッグまたは直接デバッグのいずれかに使用できます。インコンテキストデバッグにデバッガーを使用する場合、プログラムの最初の行にブレークポイントを設定します。セッションがターゲットを呼び出すと、制御がデバッガーに転送されます。ダイレクトデバッグを使用する場合、デバッガーはターゲットに必要なパラメーターの入力を求め、コードをステップ実行できます。

9.1.1 インコンテキストデバッグ

プログラムの最初の行にブレークポイントを設定するには、デバッグするオブジェクトの名前を右クリックし、*Debugging* サブメニューから *Setbreakpoint* を選択します。デバッガーウィンドウが開き、別のセッションがプログラムを呼び出すのを待ちます。

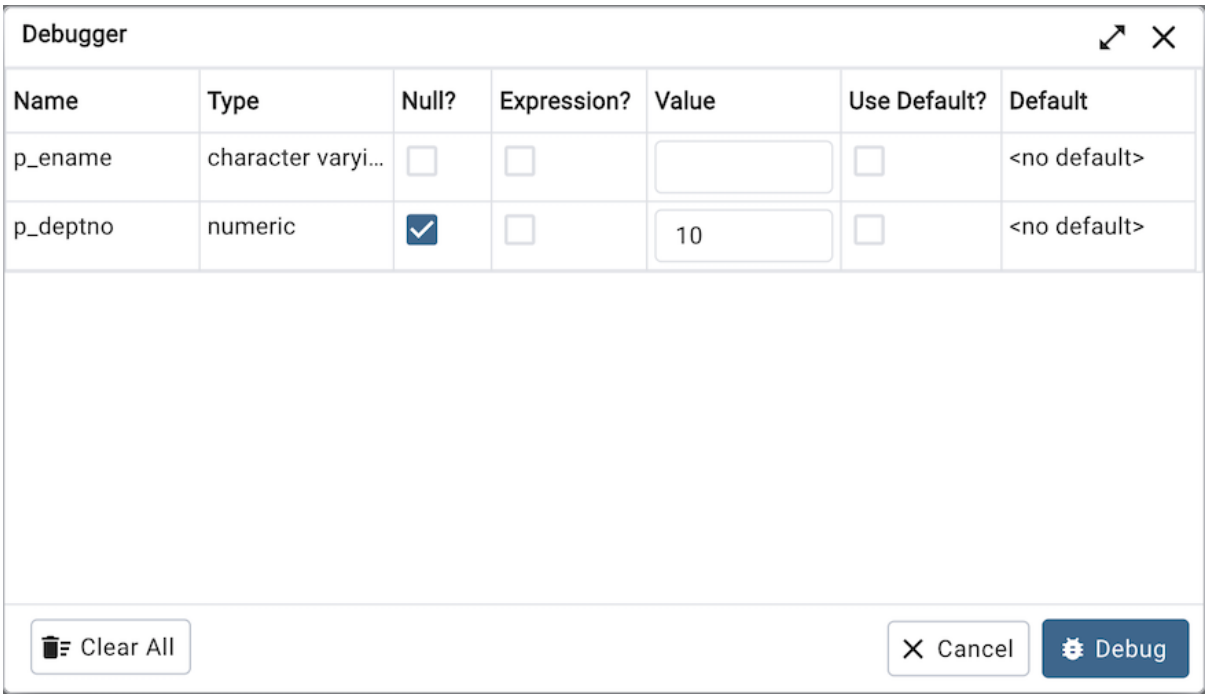


別のセッションがターゲットを呼び出すと、デバッガーにコードが表示され、ブレークポイントを追加したり、行ごとにステップ実行したりできます。もう一方のセッションは、デバッグが完了するまで中断されます。その後、制御はセッションに戻ります。



9.1.2 ダイレクトデバッグ

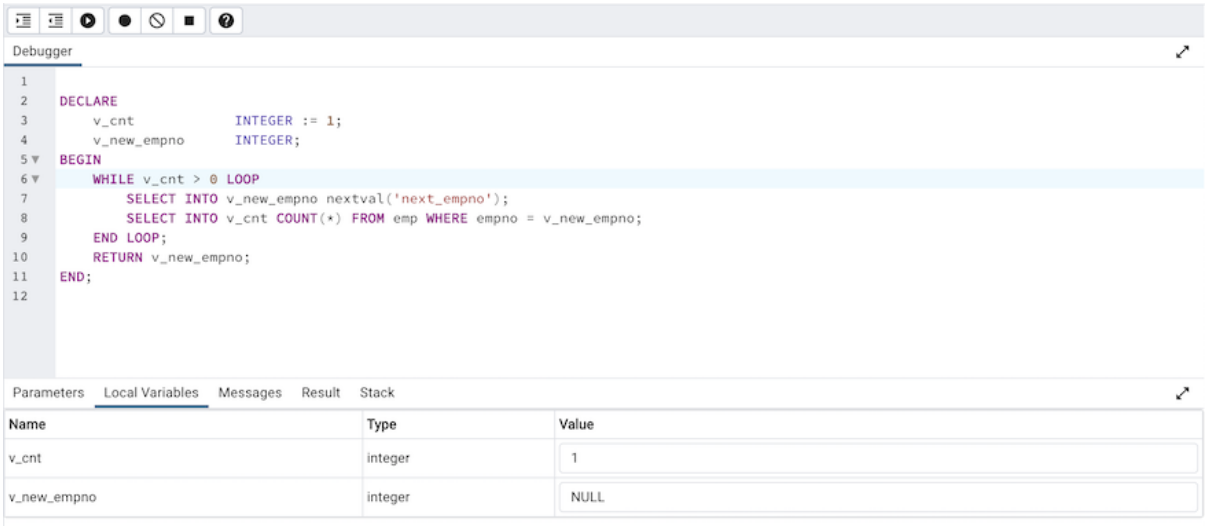
直接デバッグにデバッガを使用するには、pgAdmin ツリーコントロールでデバッグするオブジェクトの名前を右クリックし、 *Debugging* サブメニューから *Debug* を選択します。デバッガーウィンドウが開き、プログラムに必要な値の入力を求めます。



デバッガー ダイアログのフィールドを使用して、各パラメーターの値を指定します。

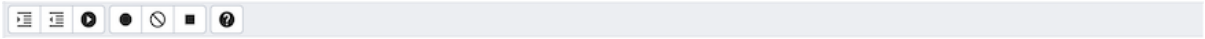
- *Name* フィールドには、仮パラメーター名が含まれています。
- *Type* フィールドには、パラメーターのデータ型が含まれています。
- *Null?* チェックボックスをオンにして、パラメーターが NULL 値であることを示します。
- Value フィールドに式が含まれる場合は、 *Expression?* チェックボックスをオンにします。
- *Value* フィールドを使用して、プログラムに渡されるパラメーター値を指定します。パラメーター値を入力するときは、グリッドの適切なセルに値を入力するか、セルを空のままにして NULL を表すか、「」（2 つの一重引用符）を入力して空の文字列を表すか、2 つのリテラル文字列を入力します一重引用符、/'/' を入力します。PostgreSQL8.4 以降では、可変引数関数パラメーターがサポートされています。これらは、値のコンマ区切りリストとして入力し、必要に応じて引用符で囲んだり、キャストしたりできます。
- [デフォルトを使用しますか?] チェックボックスをオンにして、プログラムが [デフォルト値] フィールドの値を使用することを示します。
- **デフォルト値** フィールドには、パラメーターのデフォルト値が含まれています。

プログラムに必要な値を指定し、[*Debug*] ボタンをクリックしてプログラムのステップ実行を開始します。ここで指定された引数の値が保存されます。次にダイアログが開いたときに、値が事前に入力されます。値をクリアするには、 *ClearAll* ボタンを使用します。



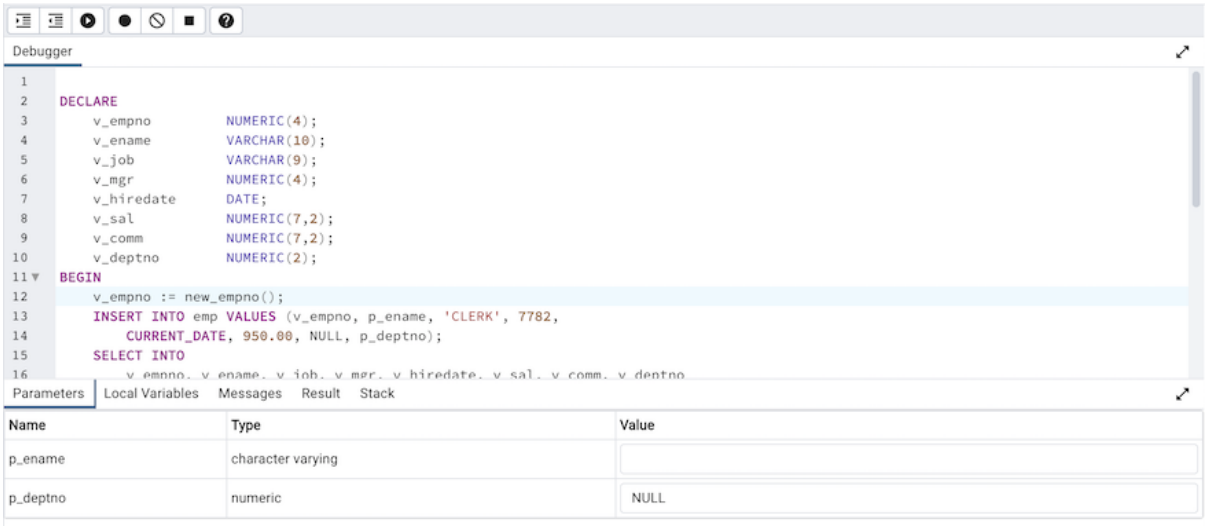
9.1.3 デバッガーの使用

デバッガーのメインウィンドウは、2 つのパネルと状況依存のツールバーで構成されています。ツールバーアイコンを使用して、ブレークポイントを管理し、コードにステップインまたはステップスルーします。アイコンに関連付けられたオプションを識別するツールチップのアイコンにカーソルを合わせます。ツールバーのオプションは次のとおりです。



オプション	アクション
ステップイン	**ステップイン** アイコンをクリックして、現在強調表示されているコード行を実行します。
ステップオーバー	**ステップオーバー** アイコンをクリックしてコード行を実行し、コードによって呼び出されるサブファンクションをステップオーバーします。サブファンクションは実行されますが、ブレークポイントが含まれていない限りデバッグされません。
続行/開始	**続行/スタート** アイコンをクリックして強調表示されたコードを実行し、プログラムがブレークポイントに到達するか完了するまで続行します。
トグルブレークポイント	**トグルブレークポイント** アイコンを使用して、ブレークポイントを有効または無効にします（ブレークポイントを削除せずに）。
すべてのブレークポイントをクリアする	**すべてのブレークポイントをクリア** アイコンをクリックして、プログラムからすべてのブレークポイントを削除します。
ストップ	*Stop* アイコンをクリックして、プログラムの実行を停止します。
ヘルプ	**ヘルプ** アイコンをクリックして、デバッガーのドキュメントを開きます。

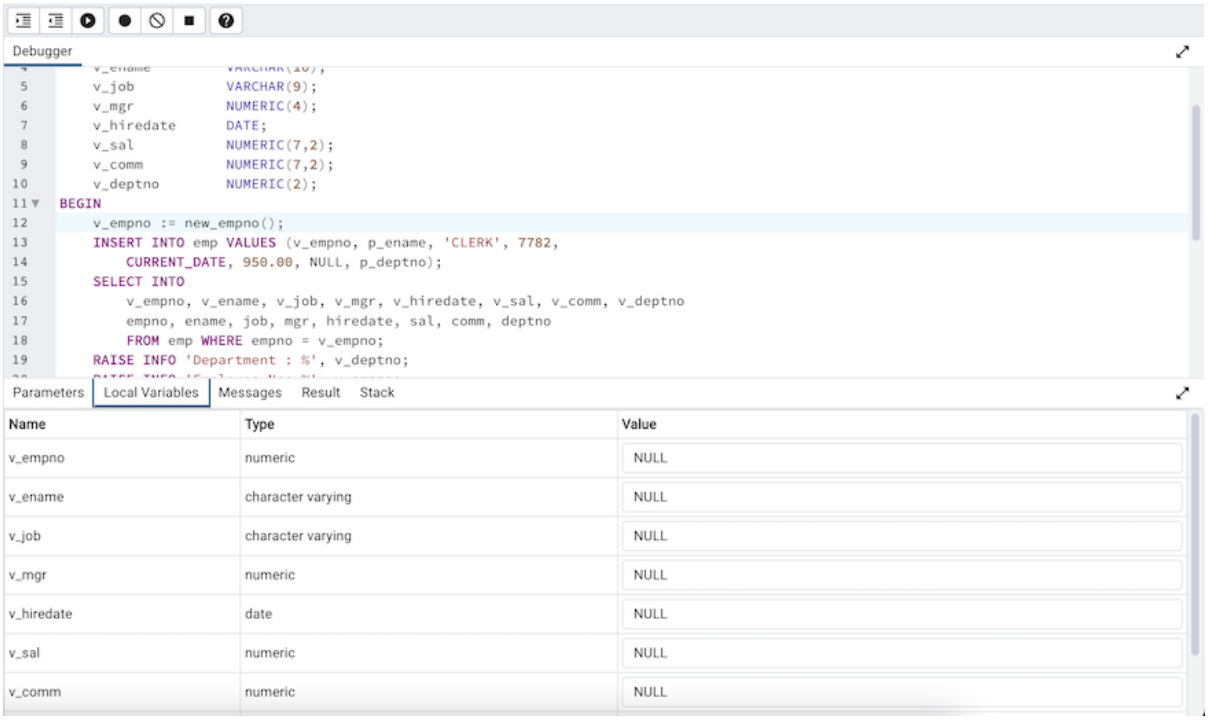
デバッガーウィンドウの上部パネルには、プログラム本体が表示されます。行番号の横にある灰色の余白をクリックして、ブレークポイントを追加します。上部パネルで強調表示されている行は、実行しようとしている行です。



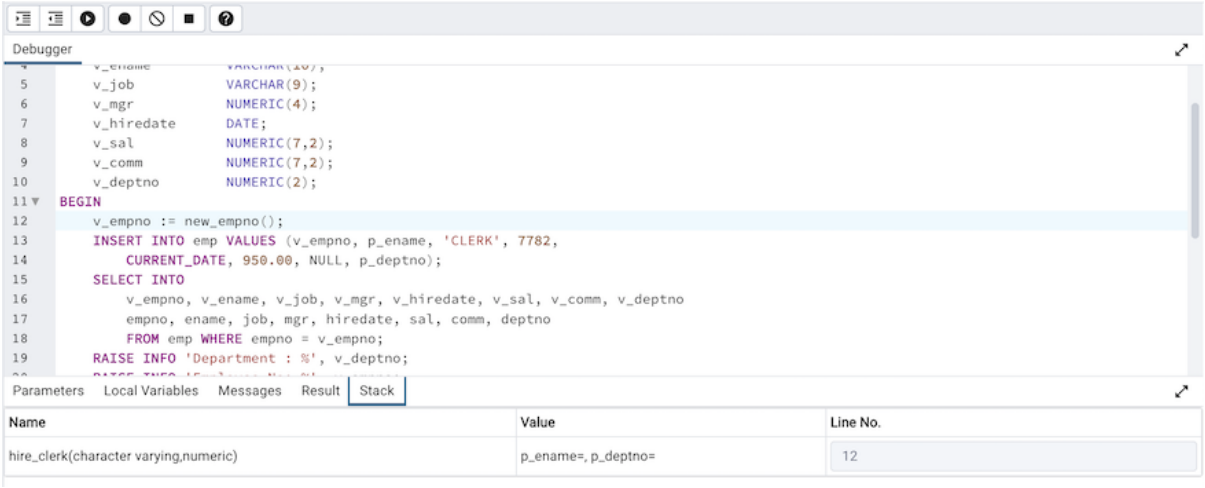
デバッガーウィンドウの下部パネルには、プログラムに関する情報を確認できる一連のタブがあります。

- [**パラメーター**] タブには、各パラメーターの値が表示されます。
- [**ローカル**] 変数タブには、プログラム変数の現在の値が表示されます。
- [**メッセージ**] タブには、サーバーから返されたメッセージ（エラー、警告、および情報メッセージ）が表示されます。
- プログラムが完了すると、[**結果**] タブにサーバーメッセージが表示されます。
- [**スタック**] タブには、呼び出されたがまだ完了していない関数のリストが表示されます。

プログラムをステップ実行すると、[**ローカル変数**] タブに各変数の現在の値が表示されます。



サブルーチンにステップインすると、[**スタック**] タブに、各呼び出し元の名前、各呼び出し元のパラメーター値（ある場合）、および各呼び出し元内の行番号を含むコールスタックが表示されます。



呼び出し元を選択してそのスタックフレームにフォーカスを変更し、上部パネルに呼び出し元の状態を表示します。

プログラムが完了すると、サーバーから返されたメッセージが [**結果**] タブに表示されます。プログラムでエラーが発生した場合、[**メッセージ**] タブに詳細が表示されます。

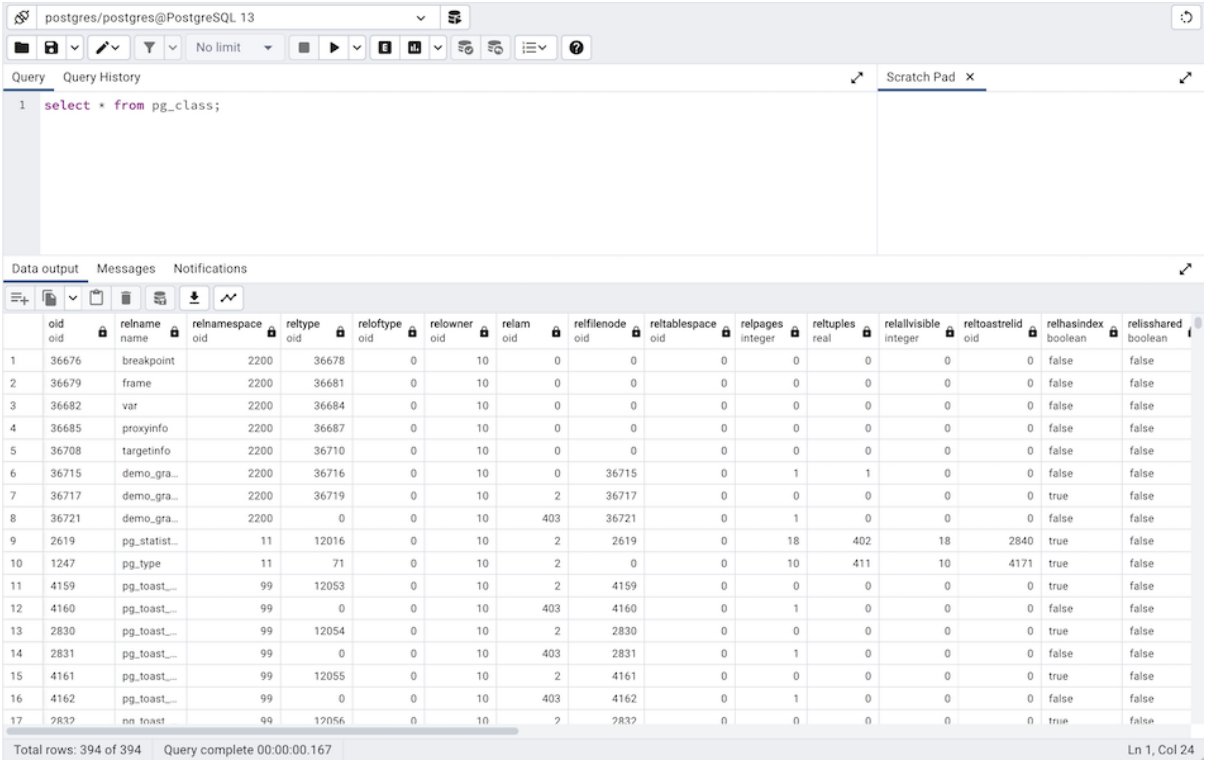


9.2 Query Tool

クエリツールは、任意の SQL コマンドを実行して結果セットを確認できる強力で機能が豊富な環境です。クエリツールには、[**ツール**] メニューの **クエリツール** メニューオプション、またはブラウザツリーコントロールのノードを選択するコンテキストメニューからアクセスできます。クエリツールを使用すると、次のことができます。

- アドホック SQL クエリを発行します。
- 任意の SQL コマンドを実行します。
- *updatable* の場合、SELECT クエリの結果セットを編集します。

- ユーザーが構成した現在の接続とトランザクションのステータスを表示します。
- 出力パネルに表示されたデータを CSV ファイルに保存します。
- テキスト、グラフィカル形式、またはテーブル形式（<ahref="https://explain.depesz.com">https://explain.depesz.comに類似）で SQL ステートメントの実行計画を確認します。
- SQL ステートメントに関する分析情報を表示します。



個々のタブでクエリツールの複数のコピーを開くことができます。クエリツールのコピーを閉じるには、タブバーの右上隅にある *X* をクリックします。

クエリツールには 2 つのパネルがあります。

- 上部パネルには *SQLEditor* が表示されます。パネルを使用して、クエリーを入力、編集、または実行できます。また、セッションで実行されたクエリを表示するために使用できる **履歴** タブと、編集中にテキストスニペットを保持するために使用できる **スクラッチパッド** も示しています。スクラッチパッドが閉じている場合、SQL エディターおよび他のパネルを右クリックし、新しいパネルを追加することにより、再度開く (またはさらに開く) ことができます。
- 下部のパネルには、 *DataOutput* パネルが表示されます。タブ付きパネルには、クエリによって返された結果セット、クエリの実行プランに関する情報、クエリの実行に関連するサーバーメッセージ、およびサーバーから受信した非同期通知が表示されます。

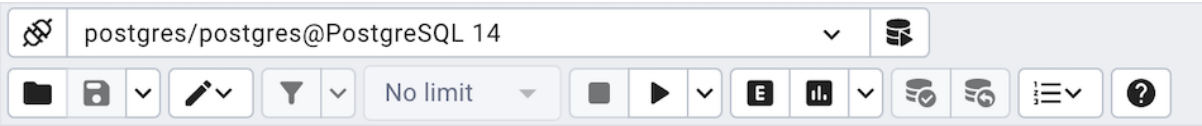
9.2.1 ツールバー

ツールバーについては、次のサブセクションで説明します。

Query Tool Toolbar

クエリツール ツールバーは、頻繁に実行されるタスクへのショートカットを提供する状況依存のアイコンを使用します。アイコンが強調表示されている場合、オプションは有効です。アイコンがグレー表示されている場合、タスクは無効になっています。

注釈: *Query Tool* と *Query Tool* ツールは、実際には同じツールの異なる動作モードです。一部のコントロールはどちらのモードでも無効になります。



pgAdmin のアイコンにカーソルを合わせると、アイコンの機能を説明するツールチップが表示されます。

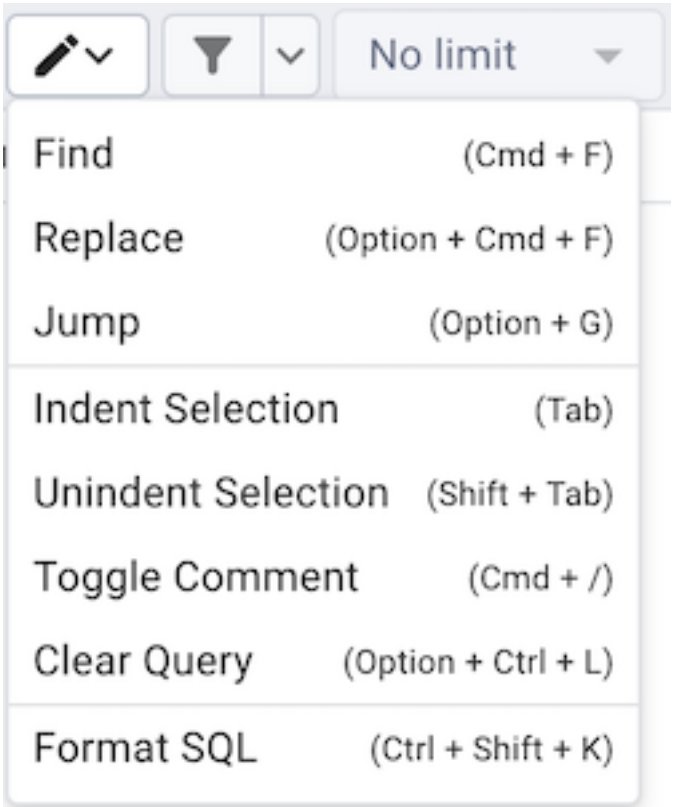
ファイルオプション

アイコン	行動	ショートカット
ファイルを開く	*OpenFile* アイコンをクリックして、以前に保存したクエリを SQLEditor に表示します。	アクセスキー +O
ファイルを保存	**保存** アイコンをクリックして、以前に保存したクエリをすばやく保存するか、 **保存** メニューにアクセスします。 • *Save* を選択して、SQLEditor パネルで選択した内容をファイルに保存します。 • **名前を付けて保存** を選択して新しいブラウザーダイアログを開き、SQL エディターパネルで選択したコンテンツを保存する新しい場所を指定します。	アクセスキー +S

フィルタ/制限オプション

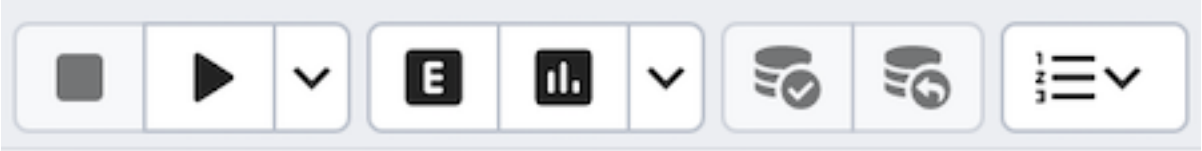
アイコン	行動	ショートカット
フィルター	**フィルター** アイコンをクリックして、 **データの表示/編集モード** でデータのフィルタリングと並べ替えの基準を設定します。下矢印をクリックして、他のフィルター処理と並べ替えオプションにアクセスします。 • *SQLFilter* では、フィルタリング条件として SQL クエリを入力できます。 *DataSorting* では、列を選択して並べ替える順序を指定できます。 • **選択項目でフィルター** をクリックして、選択したセルの値を含む行のみを表示します。 • **選択で除外** をクリックして、選択したセルの値を含まない行のみを表示します。 • **並べ替え/フィルターの削除** をクリックして、以前に選択した並べ替えまたはフィルター処理のオプションを削除します。	アクセスキー +F
リミットセレクター	**制限セレクター** で値を選択して、データセットのサイズを行数に制限します。	アクセスキー +R

クエリ編集オプション



アイコン	行動	ショートカット
編集	[**編集**] メニューを使用して、SQL エディターに表示されたコードを検索、置換、または移動します。	
	Find を選択して検索対象を指定し、SQLEditor のコンテンツを検索します。	Cmd+F
	Replace を選択して、ターゲットの個々の出現を見つけて（プロンプトで）置き換えます。	Option+Cmd+F（MAC）Ctrl+Shift+F（その他）
	Jump を選択して、検索ターゲットの次の出現に移動します。	Alt+G
	インデント選択 を選択して、現在選択されているテキストをインデントします。	タブ
	現在選択されているテキストからインデントを削除するには、*UnindentSelection* を選択します。	Shift+Tab
	ToggleComment を選択して、SQL スタイルで選択を含む行をコメント化/コメント解除します。	Cmd+/
	ClearQuery を選択して、クエリエディターウィンドウをクリアします。	Cmd+。
	FormatSQL を選択して、選択した SQL をフォーマットするか、選択されていない場合はすべての SQL をフォーマットします	Shift+Cmd+K

クエリの実行



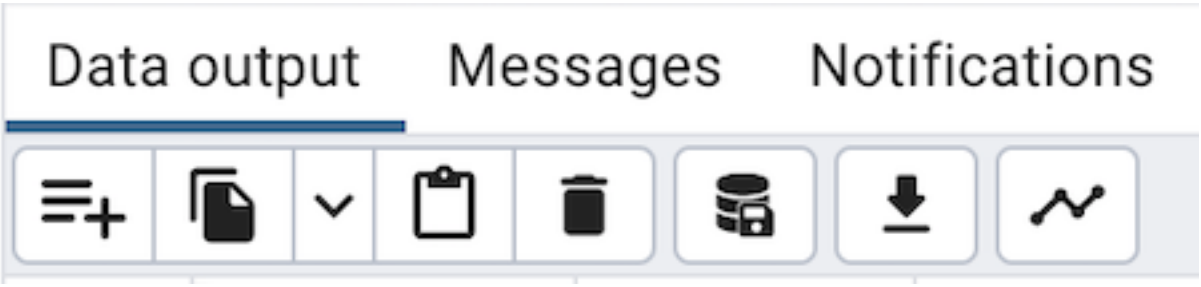
アイコン	行動	ショートカット
ストップ	*Stop* アイコンをクリックして、現在実行中のクエリの実行をキャンセルします。	アクセスキー+Q

次のページに続く

表 4 – 前のページからの続き

アイコン	行動	ショートカット
実行/更新	**実行/更新** アイコンをクリックして、SQL エディターパネルで強調表示されたクエリを実行または更新します。下矢印をクリックして、他の実行オプションにアクセスします。 • **エラー時に自動ロールバック***?* の横にチェックを追加して、トランザクション中にエラーが発生した場合にトランザクションを自動的にロールバックするようにサーバーに指示します。 • *Autocommit?* の横にチェックを追加して、各トランザクションを自動的にコミットするようにサーバーに指示します。トランザクションによって行われた変更は他のユーザーに表示され、クラッシュが発生した場合に持続します。	F5
説明	*Explain* アイコンをクリックして、現在のクエリの説明計画を表示します。EXPLAIN の結果は、出力パネルの **説明** タブにグラフィカルに表示され、 **データ出力** タブにテキスト形式で表示されます。	F7
分析を説明する	*Explainanalyze* アイコンをクリックして、現在のクエリで EXPLAIN-ANALYZE コマンドを呼び出します。 *ExplainOptions* メニューをナビゲートして、EXPLAIN コマンドのオプションを選択します。 • **詳細** を選択して、クエリプランに関する追加情報を表示します。 • *Costs* を選択して、各プランノードの推定スタートアップと合計コスト、および推定行数と各行の推定幅に関する情報を含めます。 • バッファー使用量に関する情報を含めるには、 *Buffers* を選択します。 • 起動時間とクエリの各ノードで費やされた時間に関する情報を含めるには、 **タイミング** を選択します。 • クエリプランに関する概要情報を含めるには、[**概要**] を選択します。	Shift+F7
コミット	*Commit* アイコンをクリックして、トランザクションをコミットします。	Shift+CTRL+M
ロールバック	**ロールバック** アイコンをクリックして、トランザクションをロールバックします。	Shift+CTRL+R
マクロ	マクロを管理するには、 **マクロ** アイコンをクリックします。 **マクロの管理** オプションを使用して、マクロを作成、編集、またはクリアできます。	

データ編集オプション



アイコン	行動	ショートカット
行を追加	**行を追加** アイコンをクリックして新しい行を追加します	
コピー	**コピー** アイコンをクリックして、ヘッダーの有無にかかわらずコンテンツをコピーします。 • **コピー アイコンをクリックして、データ出力で現在強調表示されているコンテンツをコピーします** パネル。 • [ヘッダーとともにコピー] をクリックして、強調表示されたコンテンツをヘッダーとともにコピーします。	アクセスキー +C
貼り付け	**貼り付け** アイコンをクリックして、以前にコピーした行を新しい行に貼り付けます。	アクセスキー +P
削除	*Delete* アイコンをクリックして、選択した行を削除対象としてマークします。これらのマークされた行は、 **データの変更を保存** アイコンをクリックすると削除されます。	アクセスキー +D
データの変更を保存	**データの変更を保存** アイコンをクリックして、データ出力パネルのデータの変更（挿入、更新、または削除）をサーバーに保存します。	F6
結果を に保存します*	[結果をファイルに保存] アイコンをクリックして、現在のクエリの結果セットを区切り文字で区切られたテキストファイル（CSV、フィールド区切り文字がコンマに設定されている場合）として保存します。このボタンは、クエリが実行され、データグリッドに結果がある場合にのみ有効になります。SQLEditor->CSV/TXToutput の下の Preference ダイアログで CSV/TXT 設定を指定できます。	F8
グラフビジュアライザー	GraphVisualiser ボタンを使用して、クエリ結果のグラフを生成します。	

ステータスバー

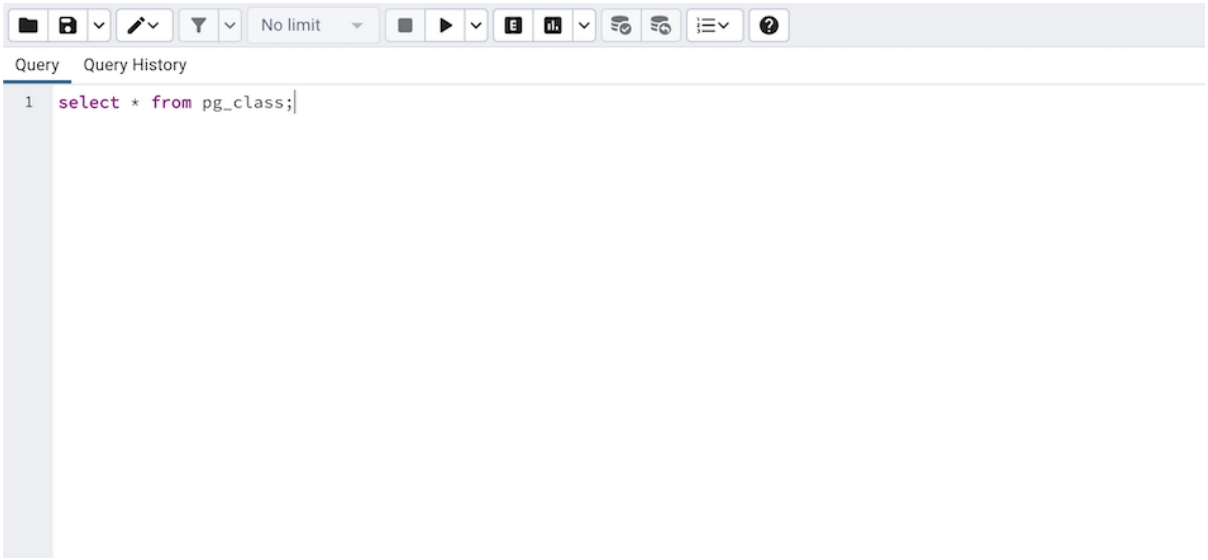


ステータスバーには次の情報が表示されます。

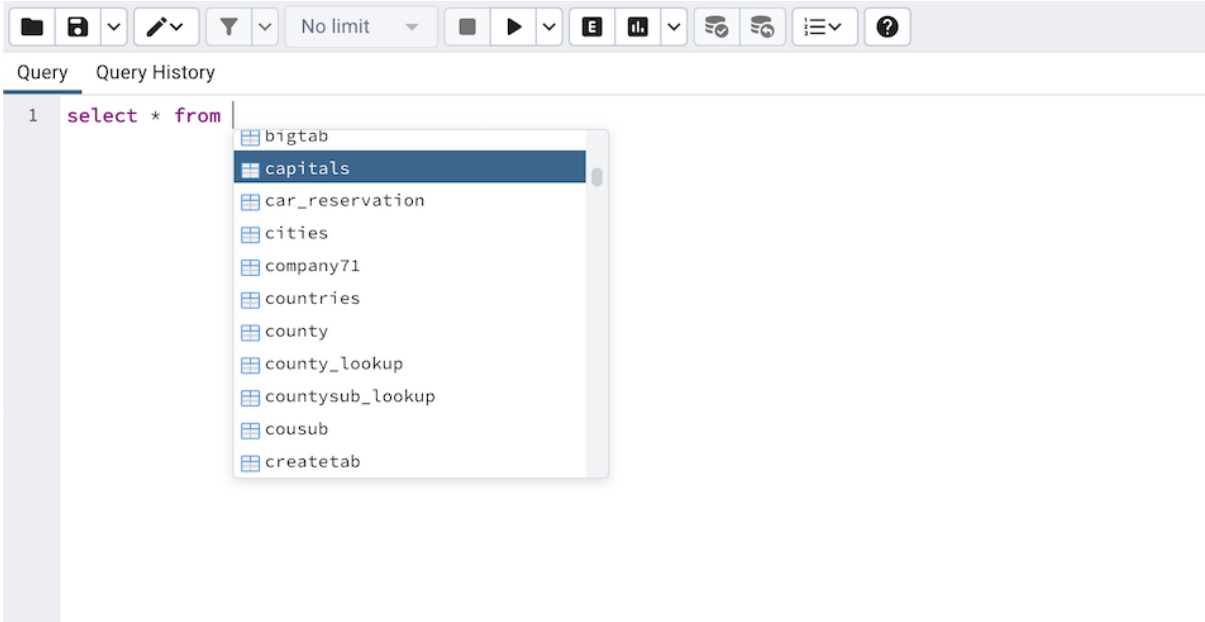
- **Total rows** ：クエリによって返された行の合計数。
- **Query complete** ：クエリが完了するまでにかかる時間。
- **Rows selected** ：データ出力パネルで選択された行数。
- **Changes staged** ：この情報は、追加、削除、および更新された行数を示しています。
- **Ln** ：[Query] タブでは、カーソルが置かれている行番号です。
- **Col** ：[Query] タブでは、カーソルが置かれている列番号です

9.2.2 SQL エディターパネル

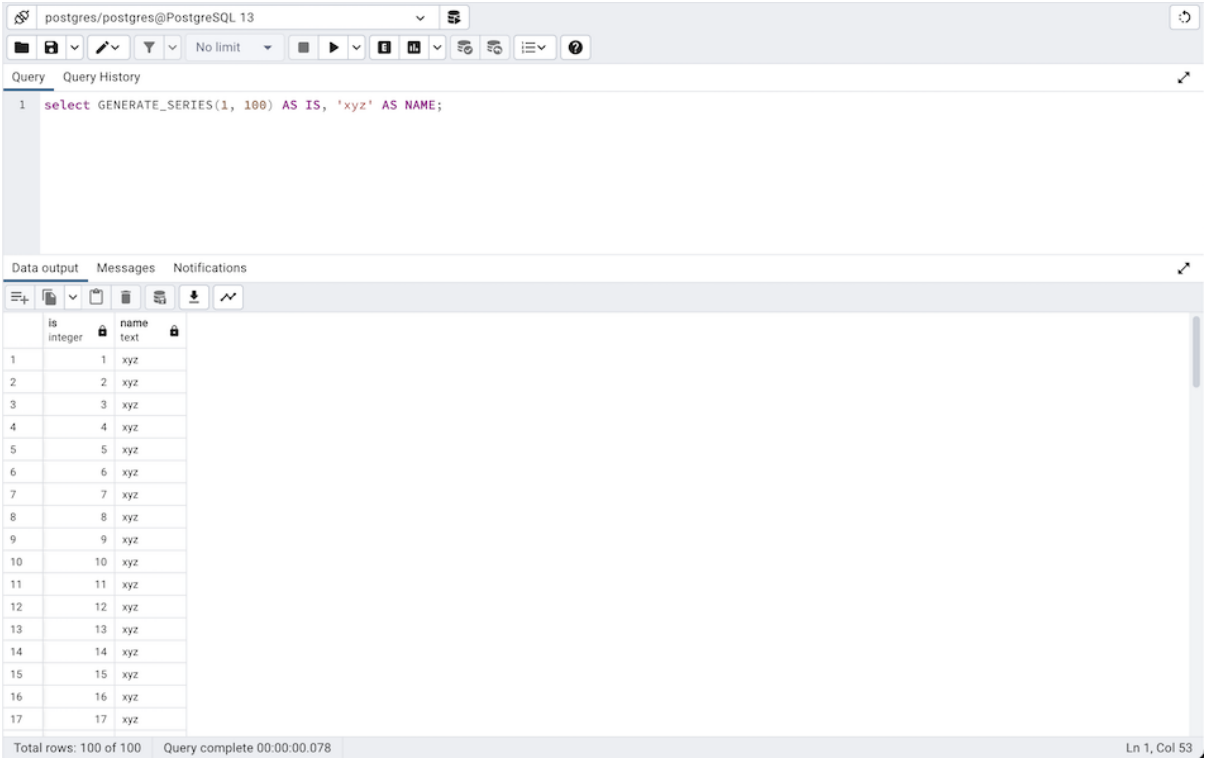
SQL **エディター** パネルは、クエリを手動で入力したり、別のソースからクエリをコピーしたり、ファイルからクエリを読み取ったりできるワークスペースです。SQL エディターは、構文の色付けとオートコンプリートを備えています。



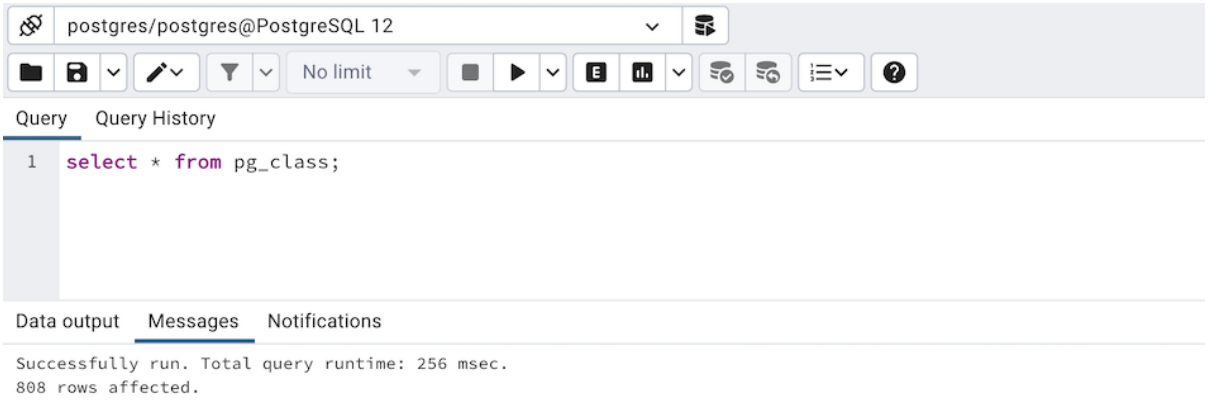
オートコンプリートを使用するには、クエリの入力を開始します。クエリエディターにクエリの次のオブジェクト名またはコマンドを提案するには、Control+Space のキーの組み合わせを押します。たとえば、/*" *SELECT/* FROM*/"（引用符はありませんが、末尾にはスペースを入れます）と入力し、Control+Space キーの組み合わせを押して、オートコンプリートオプションのポップアップメニューから選択します。



クエリを入力したら、ツールバーから **実行/更新** アイコンを選択します。SQL エディターパネルの完全な内容は、実行のためにデータベースサーバーに送信されます。SQL エディターに表示されているコードのセクションのみを実行するには、サーバーで実行するテキストを強調表示し、**実行/更新** アイコンをクリックします。



コマンドの実行時にサーバーから返されるメッセージは、[**メッセージ**] タブに表示されます。コマンドが成功すると、[**メッセージ**] タブに実行の詳細が表示されます。



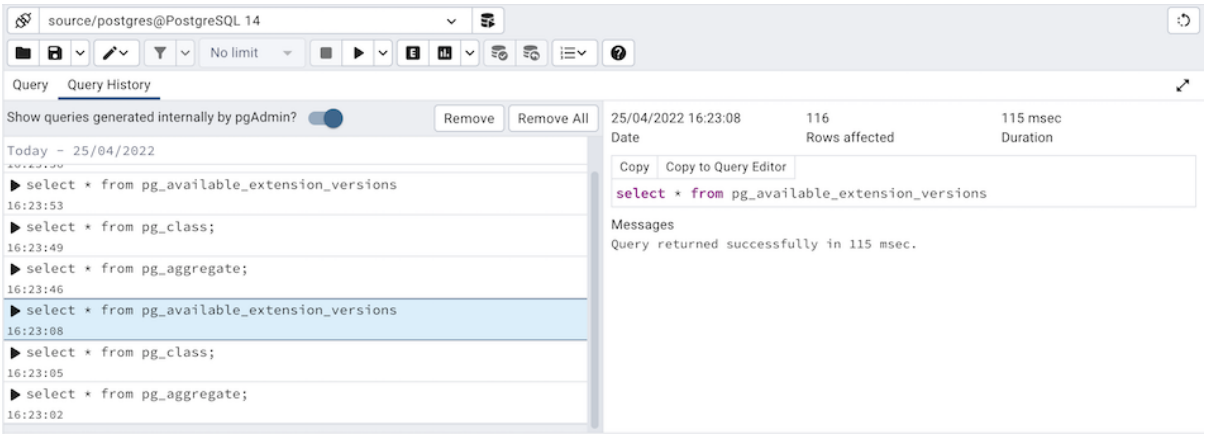
[**編集**] メニューのオプションは、コードのフォーマットとコメントに役立つ機能を提供します。

- 自動インデント機能は、Return キーを押すと、前の行と同じ深さにテキストを自動的にインデントします。
- 複数の行を選択して Tab キーを押して、インデントテキストをブロックします。
- コード内で SQL スタイルを実装または削除するか、C スタイルのコメント表記を切り替えます。

ツリービューから特定のオブジェクトを **drag and drop** すると、長いオブジェクト名を入力する時間を節約できます。オブジェクト名を含むテキストは、スキーマで完全修飾されます。必要に応じてダブルクォーテーションが追加されます。関数とプロシージャの場合、関数名とパラメーター名がクエリツールに貼り付けられます。

9.2.3 クエリ履歴パネル

[**クエリ履歴**] タブを使用して、現在のセッションのアクティビティを確認します。



[クエリ履歴] タブには、最近のコマンドに関する情報が表示されます。

- クエリが呼び出された日時。

- クエリのテキスト。
- クエリによって返された行数。
- サーバーがクエリを処理して結果セットを返すまでにかかった時間。
- サーバーから返されたメッセージ（[**メッセージ**] タブには記載されていません）。
- クエリのソース（ツールバーに対応するアイコンで示されます）。

pgAdmin によって内部的に生成されたクエリを表示または非表示にできます（「データの表示/編集」または「データの保存」操作中）。

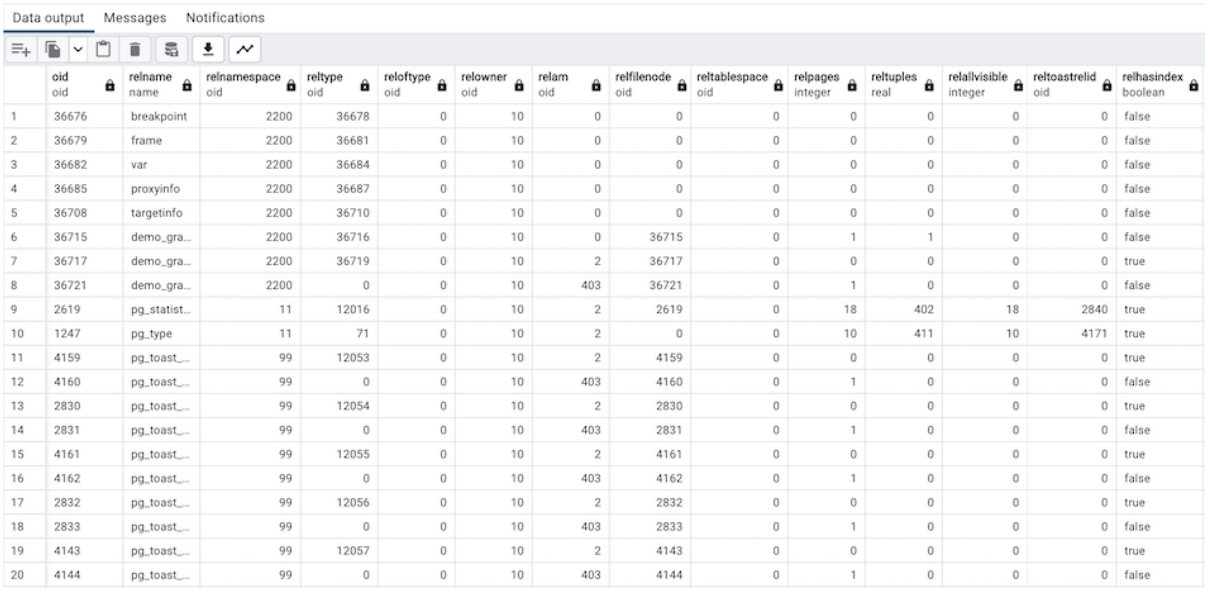
単一のクエリを選択して **削除** ボタンをクリックすると、クエリを削除できます。 **クエリ履歴** タブからすべての履歴を削除する場合は、 **すべて削除** ボタンをクリックします。

コピー ボタンを使用すると、特定のクエリをクリップボードにコピーでき、 **クエリエディターにコピー** ボタンを使用して、特定のクエリを [クエリエディター] タブにコピーできます。この操作中に、クエリエディターの既存のコンテンツはすべて削除されます。

クエリ履歴は、クエリツールモードで実行されている場合、ユーザーごとに各データベースのセッションにわたって維持されます。データの表示/編集モードでは、履歴は保持されません。デフォルトでは、データベースごとに最新の 20 個のクエリが保存されます。これは、*MAX_QUERY_HIST_STORED* **値をオーバーライドすることにより、** ``*config_local.py*`` または `config_system.py` （ `config_local.py` ドキュメントを参照）で調整できます。詳細については、 `config_local.py` セクションを参照してください。

9.2.4 データ出力パネル

[**データ出力**] パネルには、最後に実行されたクエリによって生成されたデータと統計が表示されます。


Data output Messages Notifications

	oid oid	relname name	relnamespace oid	reltype oid	reloftype oid	relowner oid	relam oid	relfilenode oid	reltablespace oid	relpages integer	reltuples real	relallvisible integer	reltoastrelid oid	relhasindex boolean
1	36676	breakpoint	2200	36678	0	10	0	0	0	0	0	0	0	false
2	36679	frame	2200	36681	0	10	0	0	0	0	0	0	0	false
3	36682	var	2200	36684	0	10	0	0	0	0	0	0	0	false
4	36685	proxyinfo	2200	36687	0	10	0	0	0	0	0	0	0	false
5	36708	targetinfo	2200	36710	0	10	0	0	0	0	0	0	0	false
6	36715	demo_gra...	2200	36716	0	10	0	36715	0	1	1	0	0	false
7	36717	demo_gra...	2200	36719	0	10	2	36717	0	0	0	0	0	true
8	36721	demo_gra...	2200	0	0	10	403	36721	0	1	0	0	0	false
9	2619	pg_statist...	11	12016	0	10	2	2619	0	18	402	18	2840	true
10	1247	pg_type	11	71	0	10	2	0	0	10	411	10	4171	true
11	4159	pg_toast_...	99	12053	0	10	2	4159	0	0	0	0	0	true
12	4160	pg_toast_...	99	0	0	10	403	4160	0	1	0	0	0	false
13	2830	pg_toast_...	99	12054	0	10	2	2830	0	0	0	0	0	true
14	2831	pg_toast_...	99	0	0	10	403	2831	0	1	0	0	0	false
15	4161	pg_toast_...	99	12055	0	10	2	4161	0	0	0	0	0	true
16	4162	pg_toast_...	99	0	0	10	403	4162	0	1	0	0	0	false
17	2832	pg_toast_...	99	12056	0	10	2	2832	0	0	0	0	0	true
18	2833	pg_toast_...	99	0	0	10	403	2833	0	1	0	0	0	false
19	4143	pg_toast_...	99	12057	0	10	2	4143	0	0	0	0	0	true
20	4144	pg_toast_...	99	0	0	10	403	4144	0	1	0	0	0	false

[**データ出力**] タブには、クエリの結果セットがテーブル形式で表示されます。次のことができます。

- 表示された結果セットから選択してコピーします。
- **実行/更新** オプションを使用して、クエリ実行情報を取得し、クエリ実行オプションを設定します。

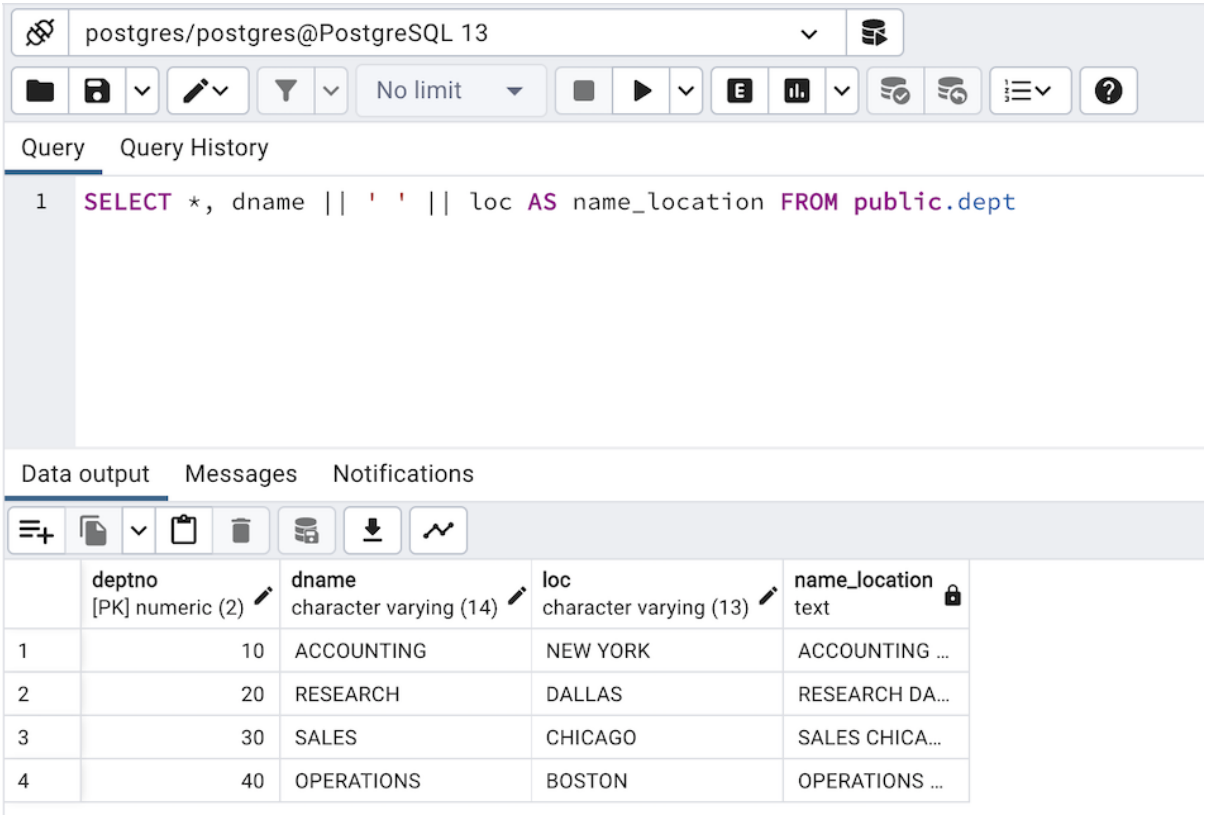
- **結果をファイルに保存** アイコンを使用して、 **データ出力** タブの内容をコンマ区切りのファイルとして保存します。
- 更新可能な場合は、SELECT クエリの結果セット内のデータを編集します。

結果セットは次の場合に更新できます。

- すべての列は、単一のテーブルから直接選択されているか、テーブルの列ではありません（2 つの列の連結など）。テーブルから直接選択した列のみが編集可能で、他の列は読み取り専用です。
- テーブルのすべての主キー列または OID が結果セットで選択されます。

名前を変更または複数回選択した列も読み取り専用です。

編集可能な列と読み取り専用の列は、列ヘッダーの鉛筆アイコンと錠前アイコンを使用して識別されます。



更新可能なクエリ結果セットを機能させるには、psycopg2 ドライバーのバージョンが 2.8 以上である必要があります。

更新可能な結果セットは、データの表示/編集モードの *Data Grid* と同じであり、同じ方法で変更できます。

自動コミットがオフの場合、データの変更は進行中のトランザクションの一部として行われます。進行中のトランザクションがない場合、新しいトランザクションが開始されます。トランザクションがコミットされない限り、データの変更はデータベースにコミットされません。

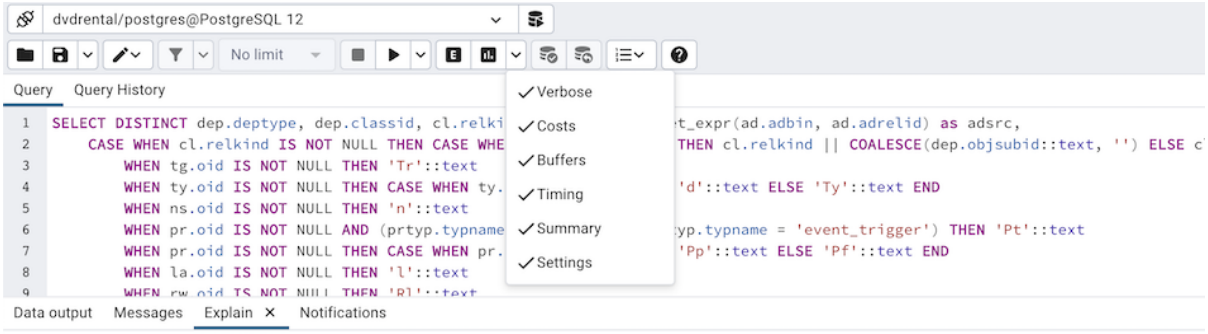
保存中にエラーが発生した場合（たとえば、NOTNULL 制約を使用して列に保存しようとした場合）、データの変更は自動的に作成された SAVEPOINT にロールバックされ、進行中のトランザクションで以前に実行されたクエリがロールバックされません。

データ出力 パネルに表示される以前のクエリまたはコマンドのすべての行セットは、別のクエリを呼び出すと破棄されます。別のクエリツールタブを開いて、以前の結果を利用できるようにします。

9.2.5 説明パネル

クエリの *Explain* または *ExplainAnalyze* プランを生成するには、ツールバーの *Explain* または *ExplainAnalyze* ボタンをクリックします。

Explain と *ExplainAnalyze* に関連するその他のオプションは、ツールバーの *ExplainAnalyze* ボタンの右側にあるドロップダウンから選択できます。



pgAdmin は *Explain[Analyze]* プランを JSON 形式で生成することに注意してください。

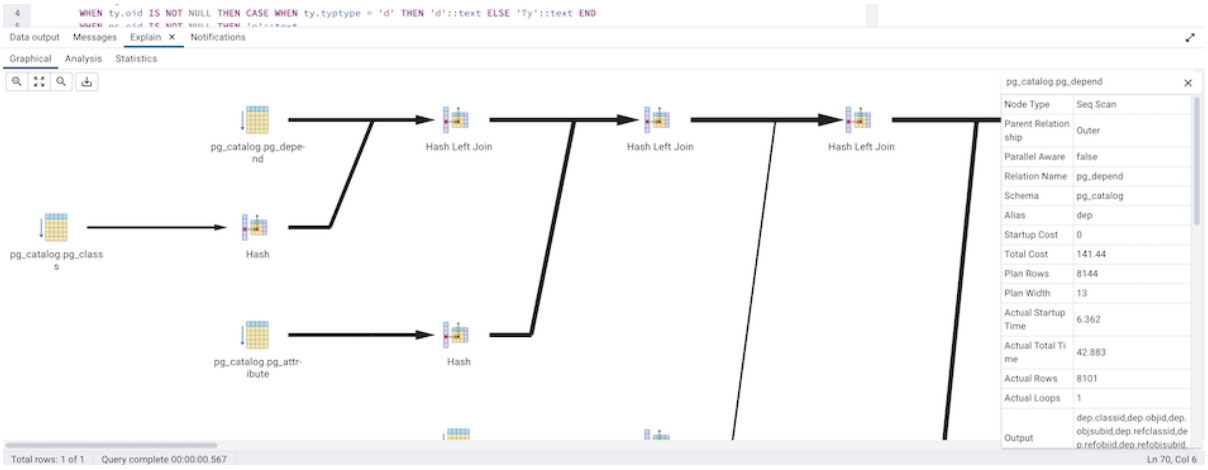
説明 計画が正常に生成されると、[説明] パネルの下に 3 つのタブ/パネルが作成されます。

- グラフィカル

EXPLAINVERBOSE はグラフィカルに表示できないことに注意してください。[**グラフィカル**] タブのノードアイコンをクリックして、そのアイテムに関する情報を確認します。選択したオブジェクトに関する情報を含むポップアップウィンドウが右側に表示されます。JIT 統計、トリガー、概要については、右上隅のボタンをクリックしてください。必要に応じて、同様のポップアップウィンドウが表示されます。

説明 キャンバスの左上隅にあるダウンロードボタンを使用して、計画を SVG ファイルとしてダウンロードします。

Note: SVG としてのダウンロードは InternetExplorer ではサポートされていません。



分析の説明 に伴うクエリプランは、 **データ出力** タブで利用できることに注意してください。

- 表

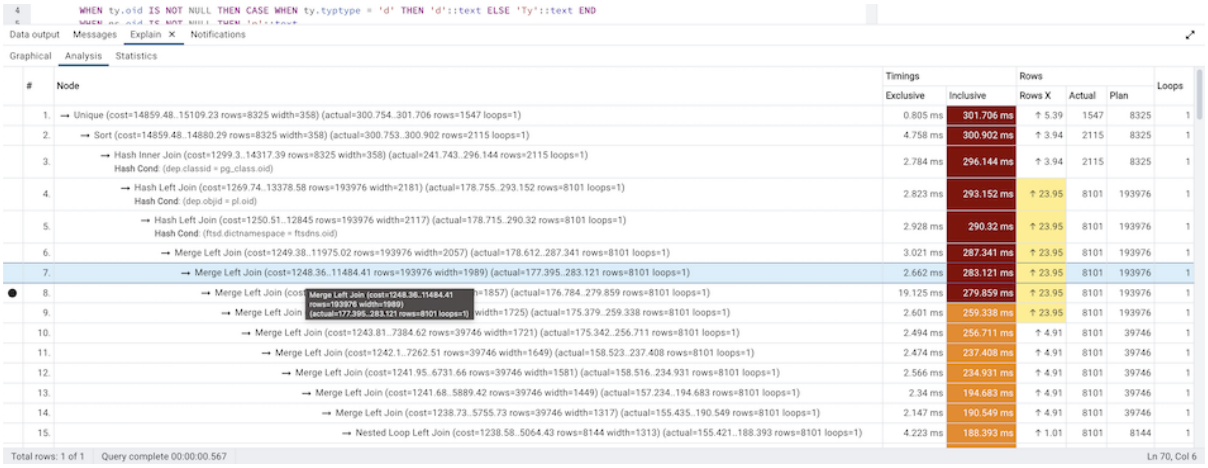
テーブル タブはプランの詳細を表形式で表示し、 *explain.depesz.com* と同様の表形式を生成します。テーブルの各行は、 **説明計画ノード** のデータを表します。これには、ノード情報、排他的タイミング、包括的

タイミング、実際の行と計画的な行の違い、実際の行、計画的な行が含まれる場合があります。

排他的、包括的、および行 X の列の背景色は、実際と計画の違いに基づいて異なる場合があります。

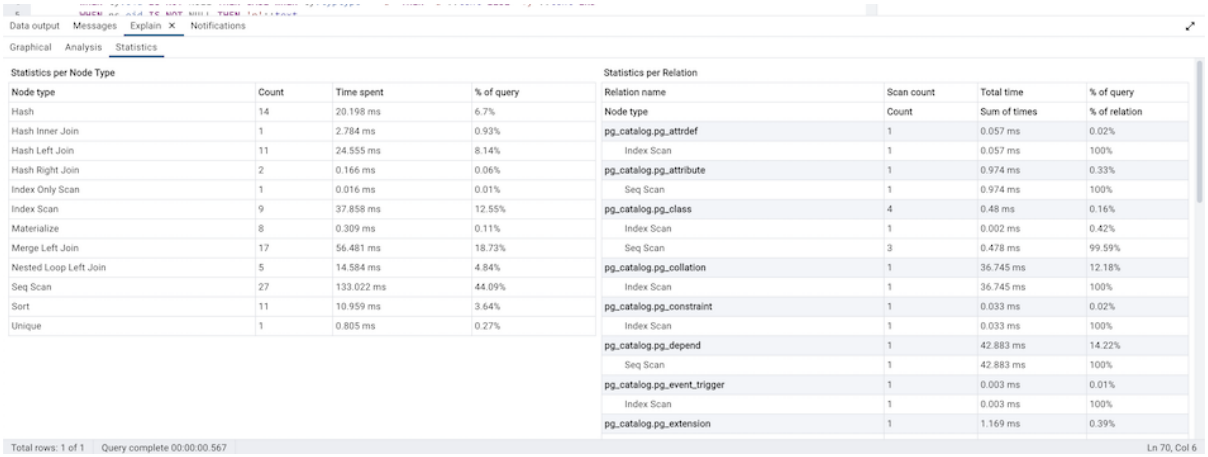
合計クエリ時間の排他的/包括的タイミングの割合が：>90-赤色>50-オレンジ（赤と黄色の間）の色>10-黄色

プランナーが行数（実際対計画）を 10 回誤って見積もった場合-黄色 100 回-オレンジ（赤と黄色の間）色-赤色



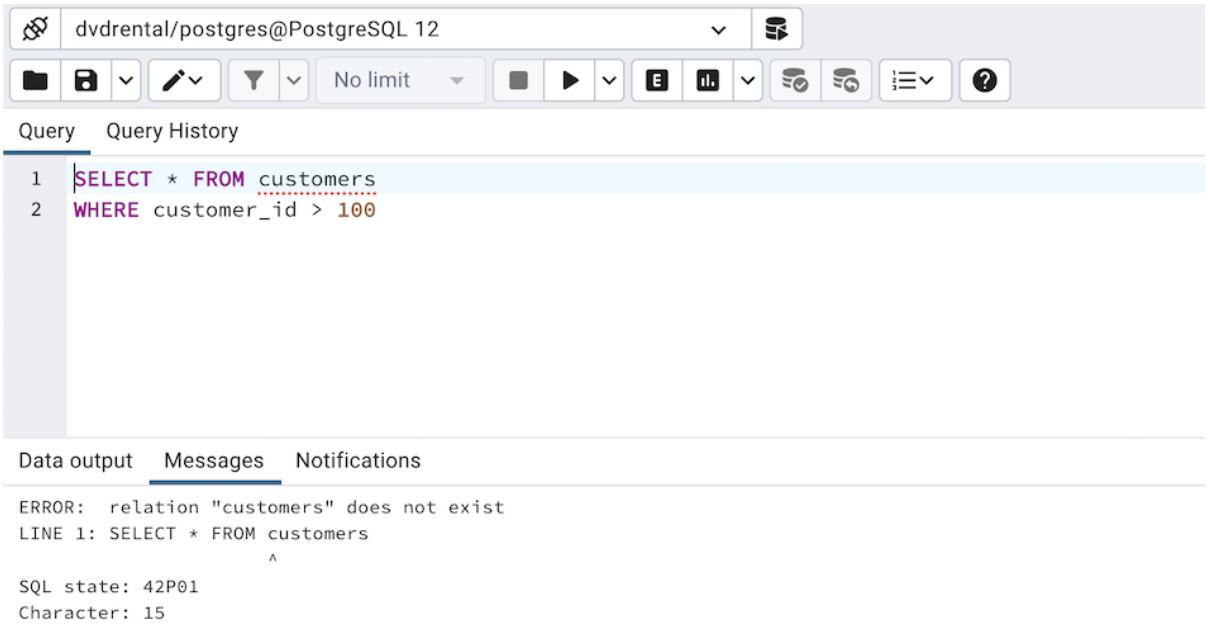
- 統計

統計 タブには 2 つのテーブルが表示されます。1. プランノードタイプごとの統計 2. テーブルごとの統計

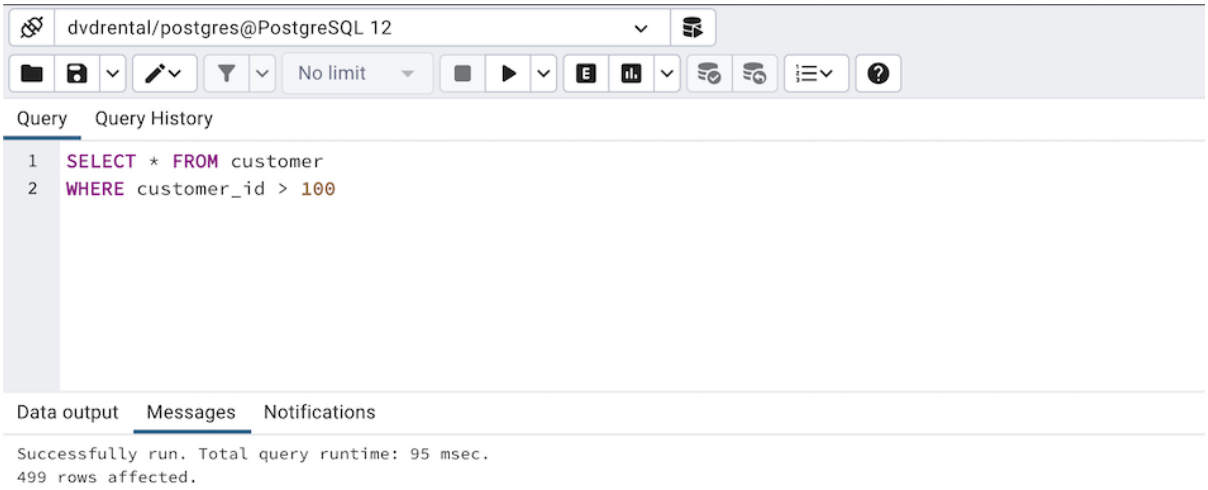


9.2.6 メッセージパネル

[**メッセージ**] タブを使用して、最後に実行されたクエリに関する情報を表示します。



サーバーがエラーを返すと、エラーメッセージが [**メッセージ**] タブに表示され、エラーの原因となった構文に下線が付き SQL。クエリが成功すると、[**メッセージ**] タブにクエリが完了するまでにかかった時間と取得された行数が表示されます。

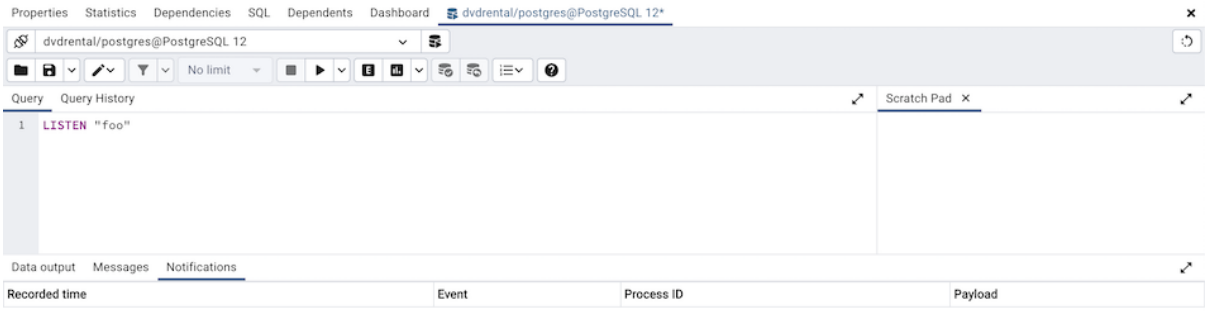


9.2.7 通知パネル

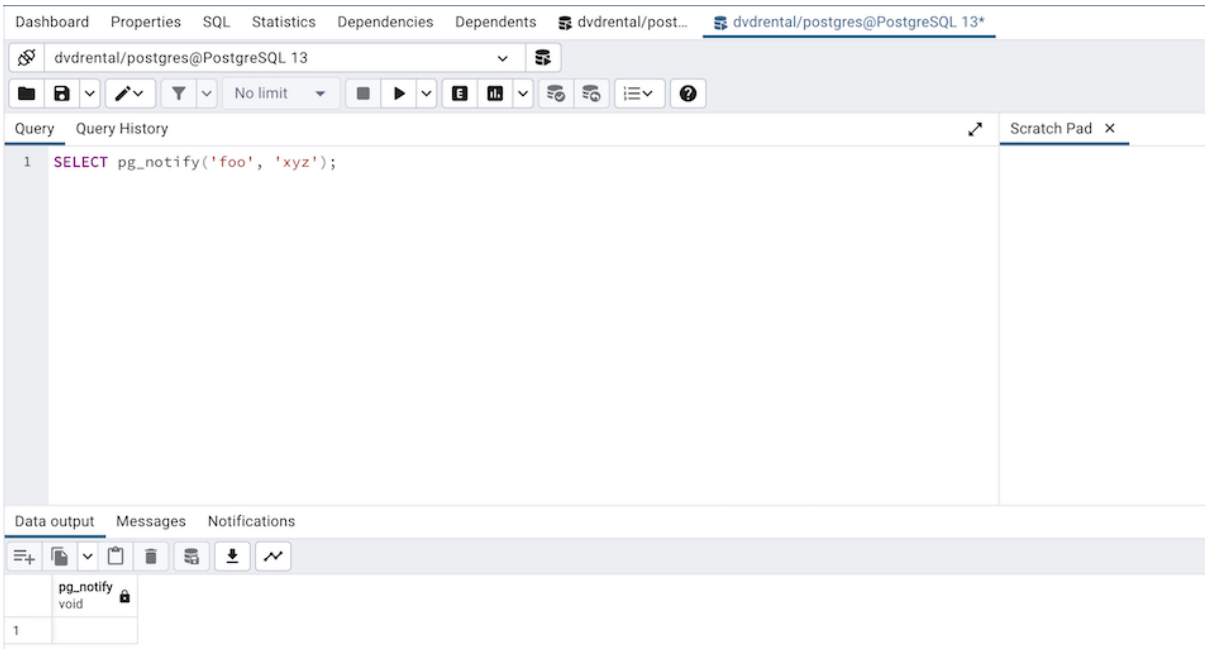
[**通知**] タブを使用して、PostgreSQL の **リッスン/通知** 機能を使用して通知を表示します。詳細については、 PostgreSQL documentation を参照してください。

例：

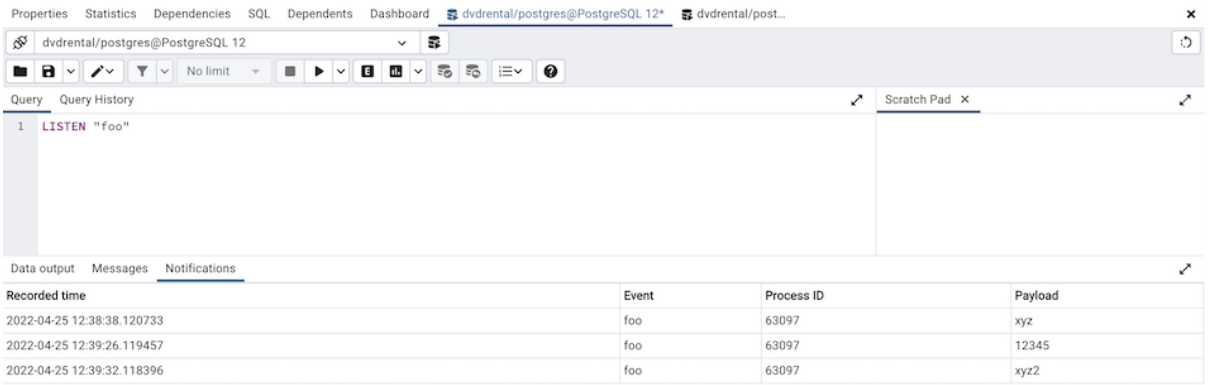
1. 最初の **クエリツール** セッションで *LISTEN/"foo/"* を実行する



2. 別の **クエリツール** セッションで、 *Notify* コマンドまたは *pg_notify* 関数を実行して、ペイロードと共にイベントの通知を送信します。

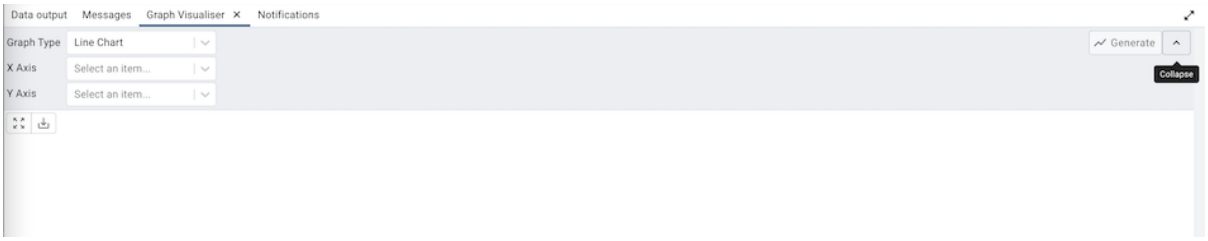


3. 最初の **クエリツール** セッションの **通知** タブに、特定のチャネルの記録時間、イベント、プロセス ID、ペイロードが表示されます。



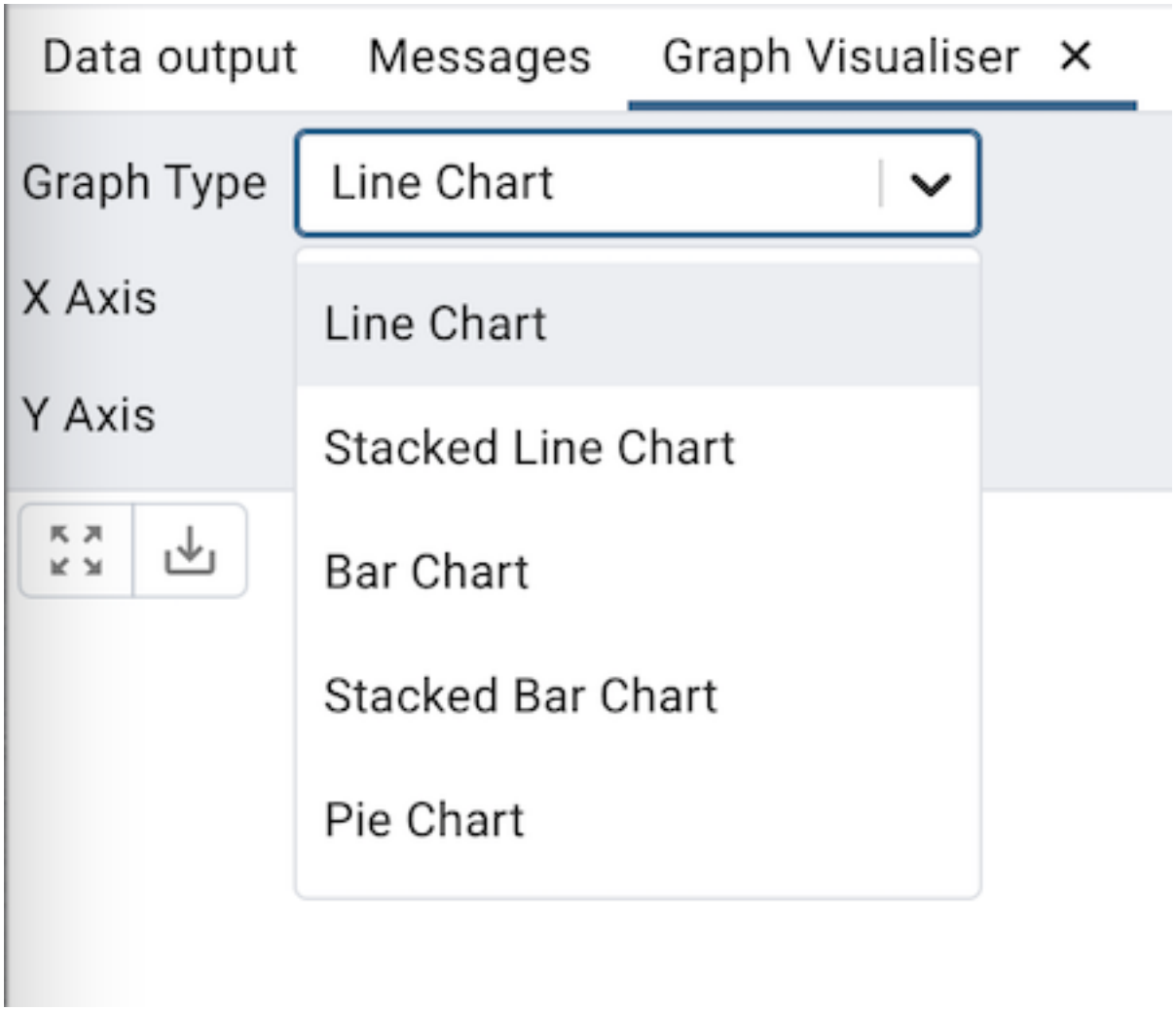
9.2.8 グラフビジュアライザーパネル

ツールバーの GraphVisualiser ボタンをクリックして、クエリ結果の **グラフ** を生成します。グラフビジュアライザーは、折れ線グラフ、積み上げ折れ線グラフ、棒グラフ、積み上げ棒グラフ、および円グラフをサポートしています。



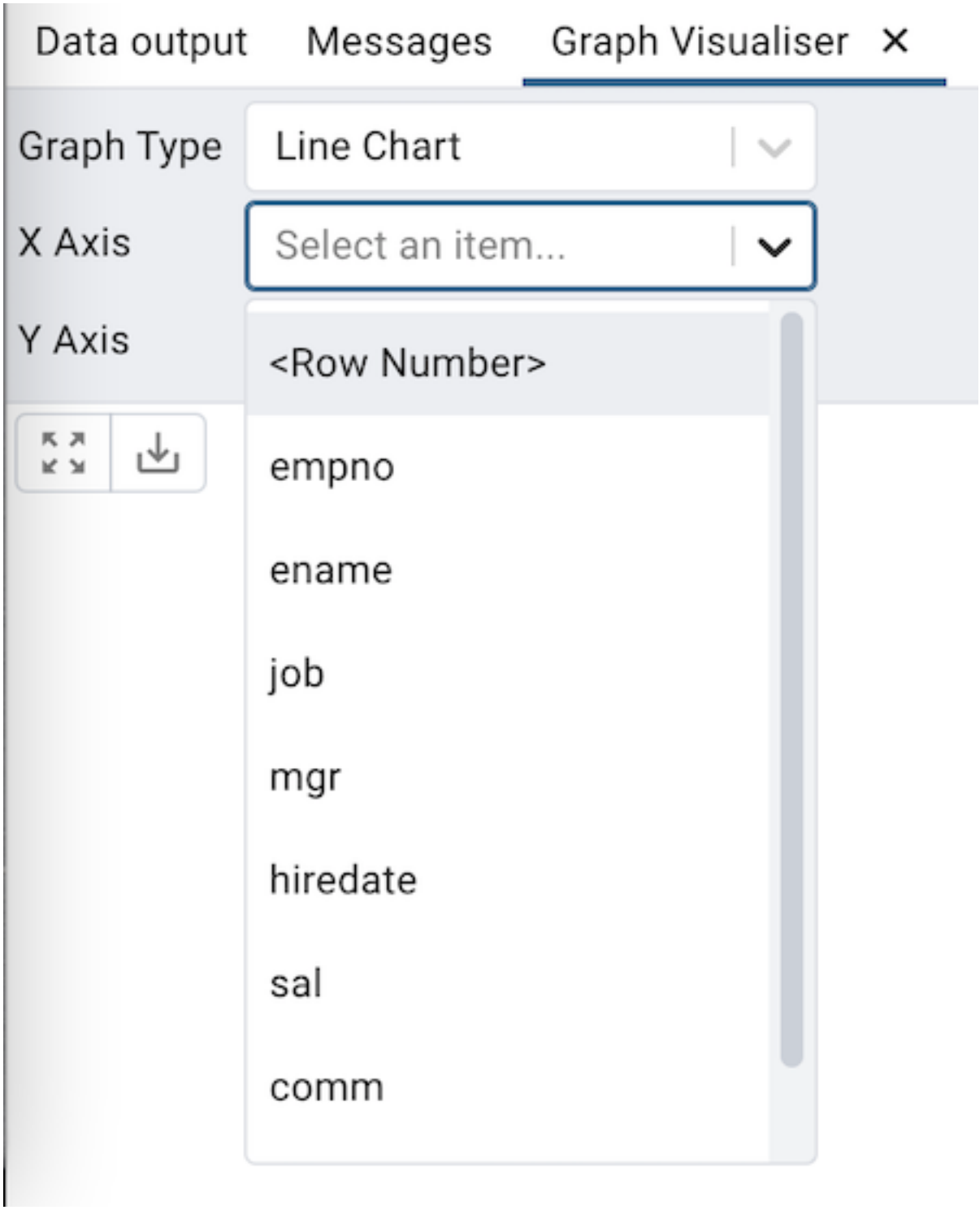
- グラフタイプ

生成するグラフの種類を選択します。



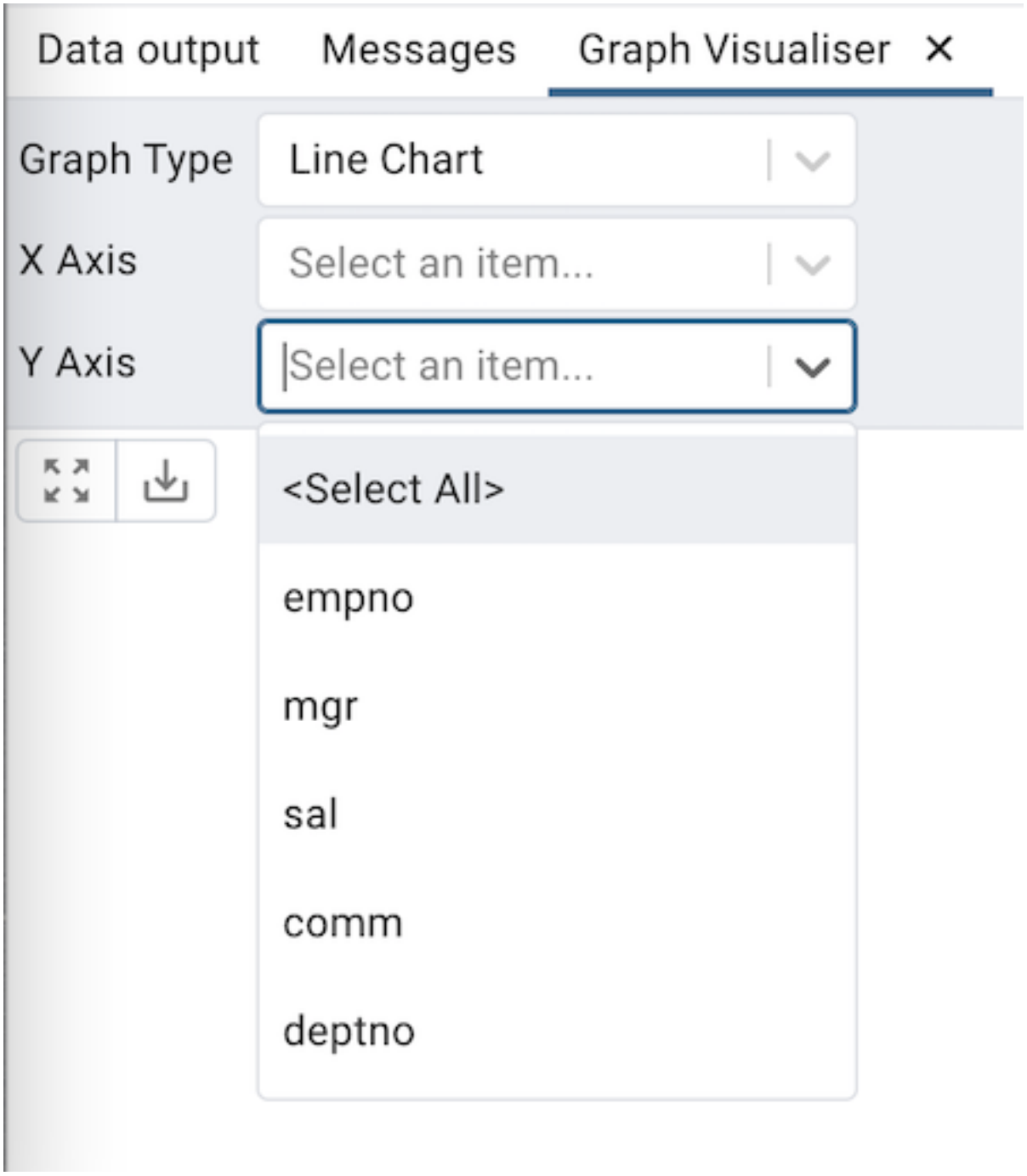
- X 軸

X 軸に値を表示する列を *X* **軸** ドロップダウンから選択します。 *<RowNumber>* オプションを選択して、行数を X 軸のラベルとして使用します。



- Y 軸

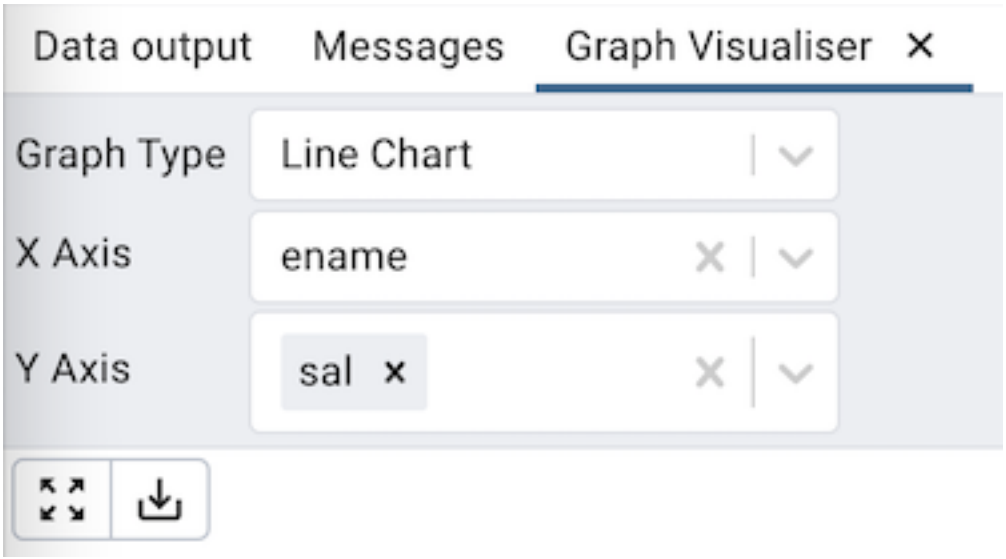
Y 軸に値を表示する列を *YAxis* ドロップダウンから選択します。ユーザーは複数の列を選択できます。ドロップダウンメニューから *<SelectAll>* オプションを選択して、すべての列を選択します。



- ダウンロードしてズームボタン

ズームは、チャート上の領域をクリックしてマウスで選択することで実行されます。 **元にズーム** ボタンは、元のズームレベルに戻ります。

ボタンバーの **ダウンロード** ボタンをクリックして、チャートをダウンロードします。



折れ線グラフ

折れ線グラフ は、[グラフタイプ] ドロップダウンから [折れ線グラフ] を選択し、X 軸と Y 軸を選択して [生成] ボタンをクリックすることで生成できます。以下は、従業員の名前と給与のグラフの例です。



Preferences Dialog で *Usedifferentdatapointstyles?* オプションを true に設定して、各グラフの線に異なるスタイルでデータポイントを表示します。

積み上げ折れ線グラフ

積み上げ折れ線グラフ は、[グラフの種類] ドロップダウンから [積み上げ折れ線グラフ] を選択し、X 軸と Y 軸を選択して [生成] ボタンをクリックすることで生成できます。



棒グラフ

棒グラフ は、[グラフタイプ] ドロップダウンから [棒グラフ] を選択し、X 軸と Y 軸を選択して [生成] ボタンをクリックすることで生成できます。



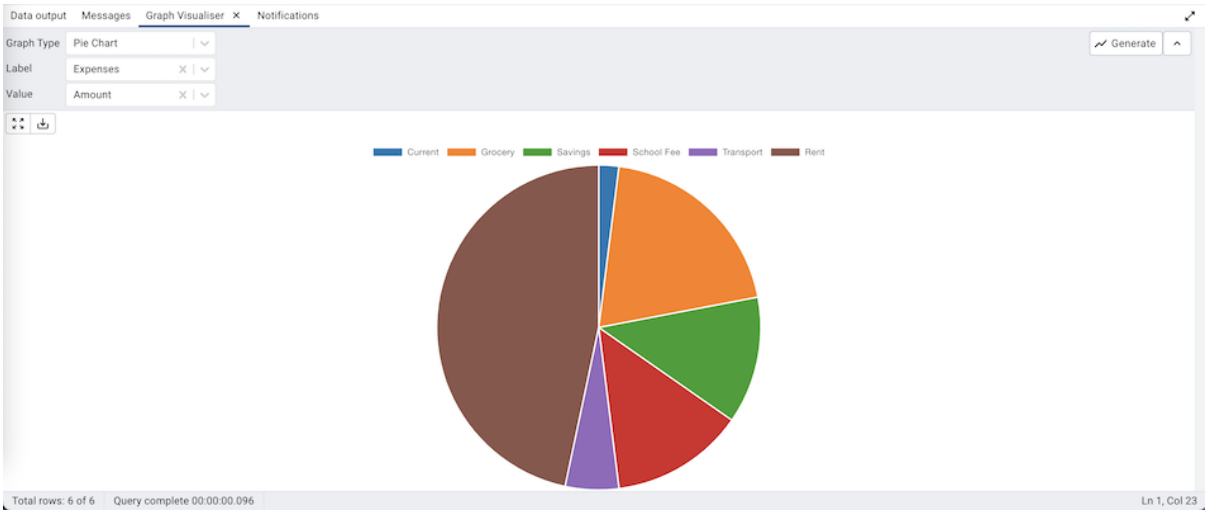
積み上げ棒グラフ

積み上げ棒グラフ は、[グラフの種類] ドロップダウンから [積み上げ棒グラフ] を選択し、X 軸と Y 軸を選択して [生成] ボタンをクリックすることで生成できます。



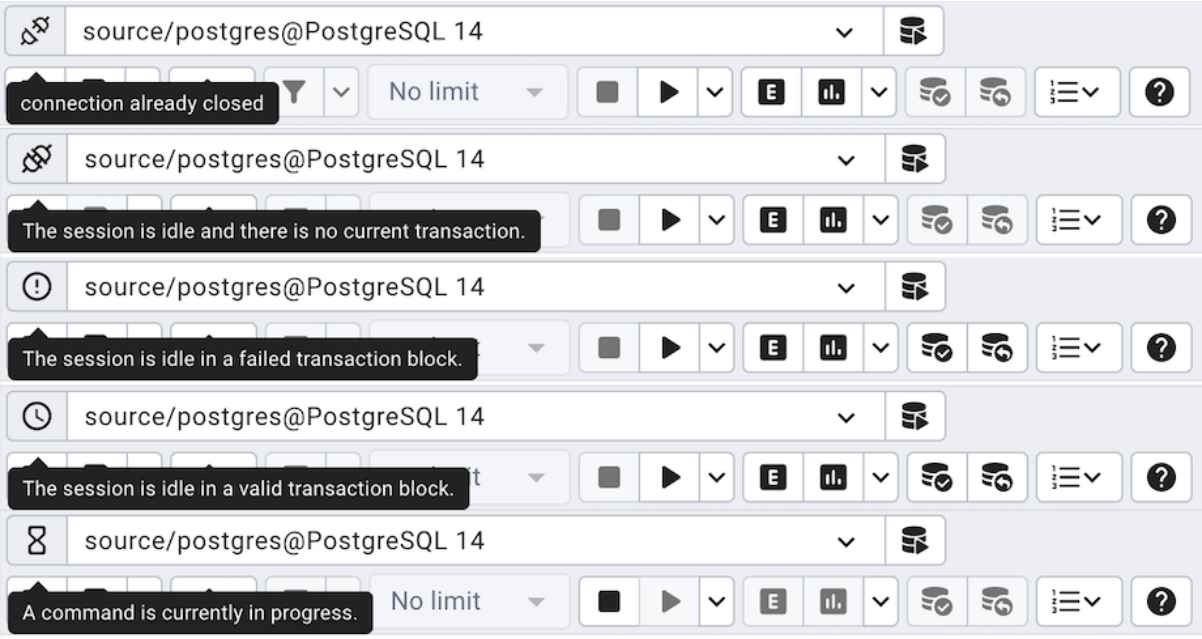
円グラフ

円グラフ は、[グラフの種類] ドロップダウンから [円グラフ] を選択し、[ラベル] と [値] を選択して、[生成] ボタンをクリックすることで生成できます。



9.2.9 接続状態

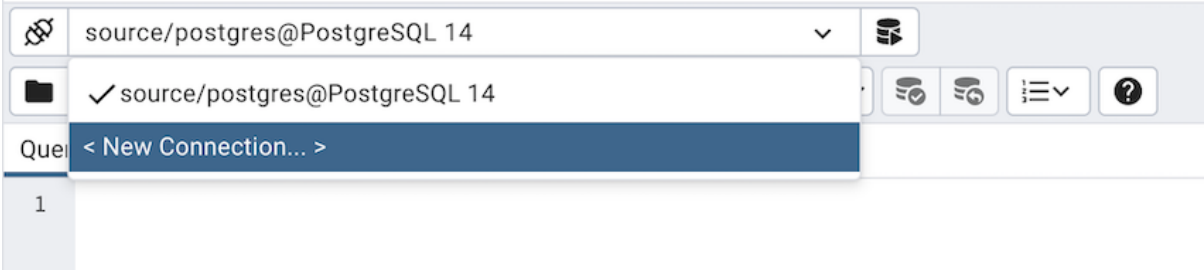
接続ステータス 機能を使用して、クエリツールのステータスアイコンをクリックして、現在の接続とトランザクションのステータスを表示します。



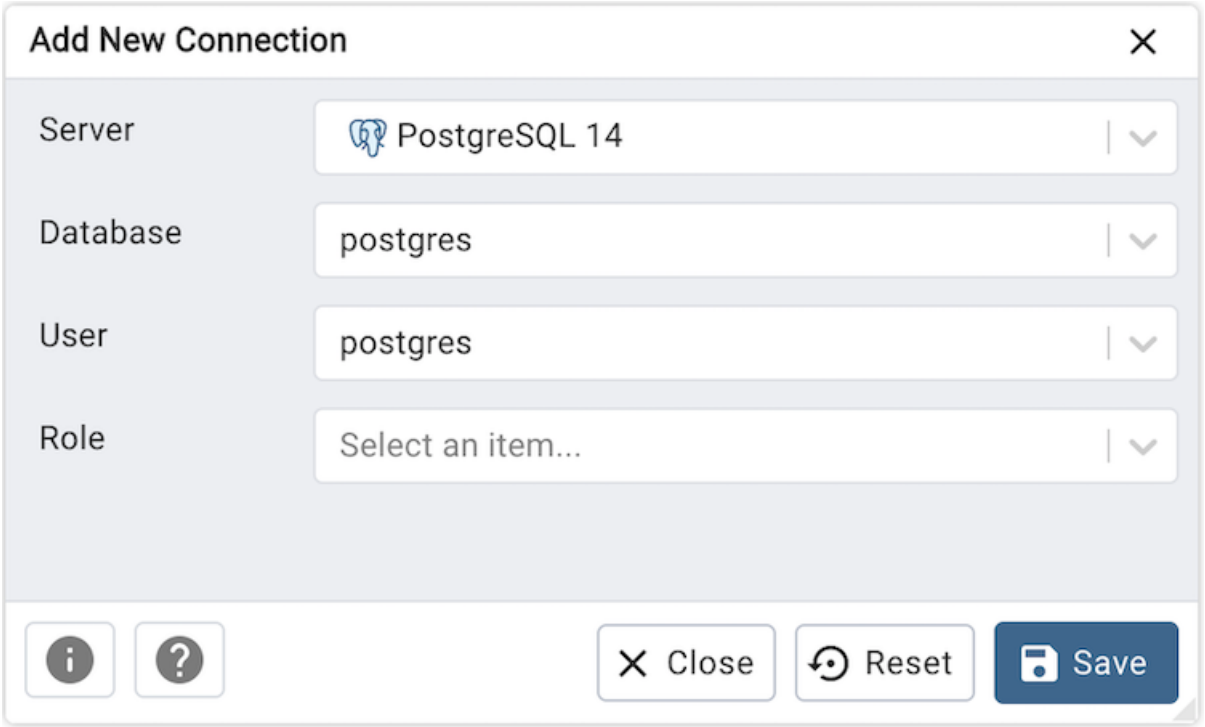
9.2.10 接続を変更する

ユーザーは、クエリツールの既存の開いているセッションから別のサーバーまたはデータベースに接続できます。

- 接続ステータスの横にある接続リンクをクリックします。
- ドロップダウンから *<NewConnection>* オプションをクリックします。



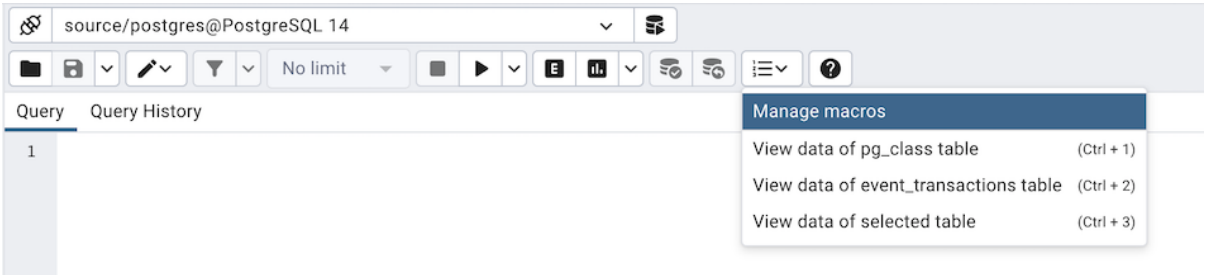
- 接続するサーバー、データベース、ユーザー、ロールを選択し、[保存] ボタンをクリックします。



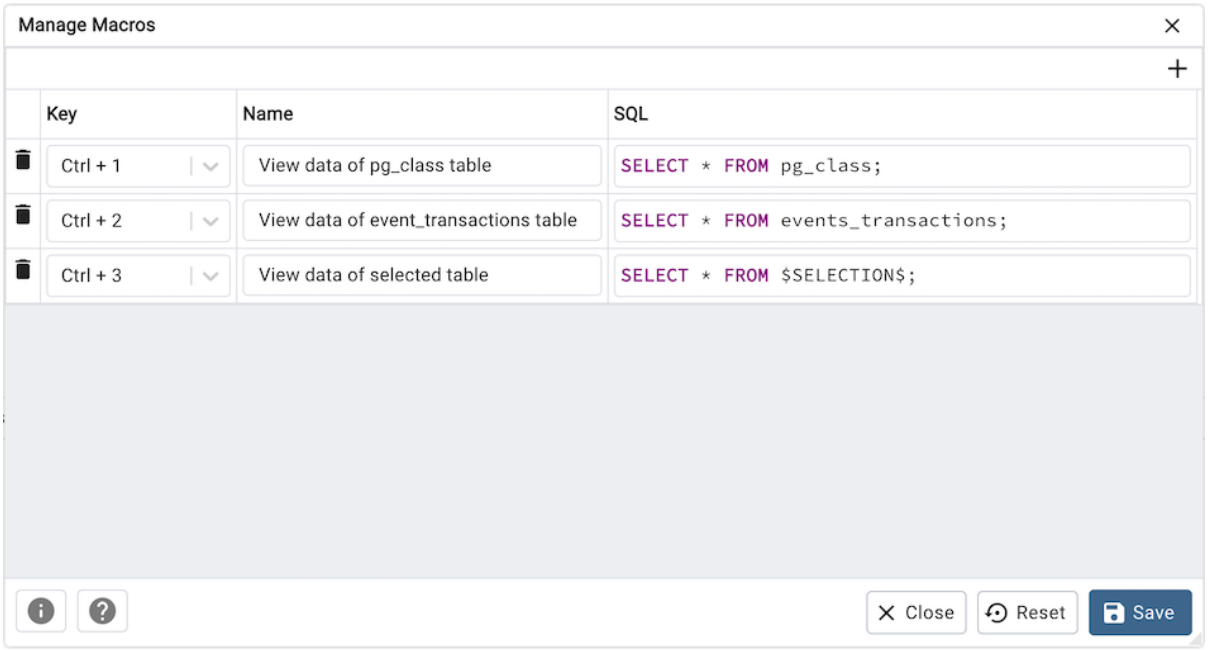
- 新しく作成された接続がオプションにリストされます。
- 接続するには、新しく作成した接続をドロップダウンリストから選択します。

9.2.11 マクロ

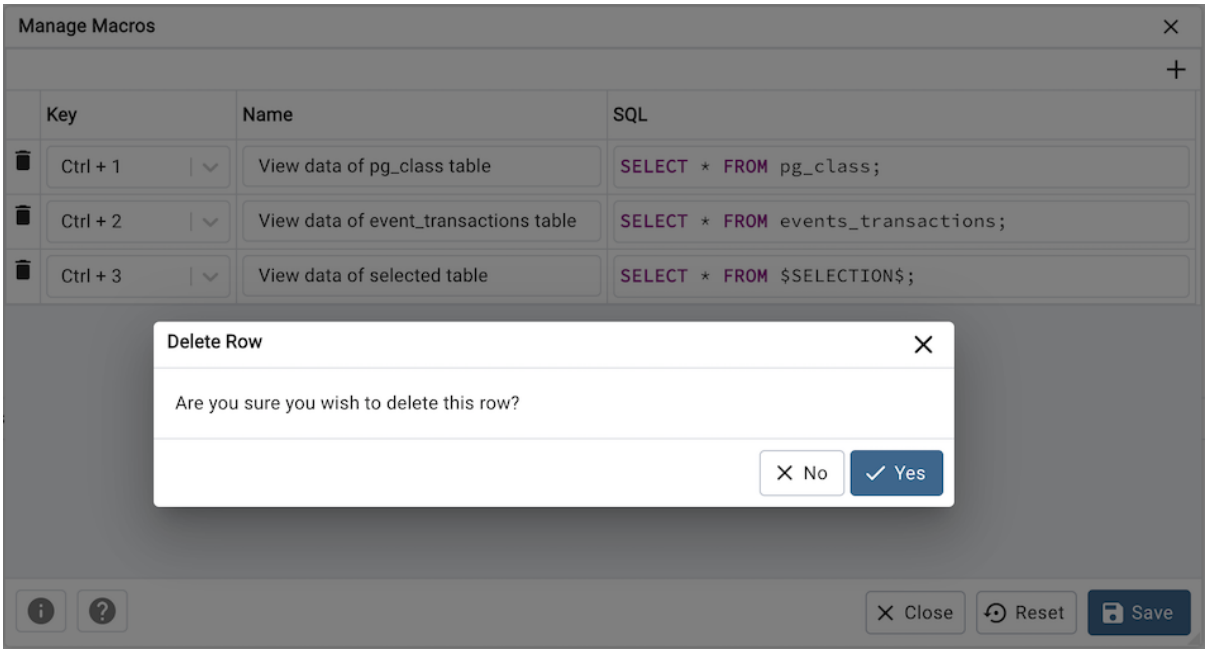
クエリツールマクロを使用すると、キーを 1 回押すだけで事前定義された SQL クエリを実行できます。事前定義されたクエリには、プレースホルダー$SELECTION$を含めることができます。マクロの実行時に、プレースホルダーはクエリツールのクエリエディターペインで現在選択されているテキストに置き換えられます。



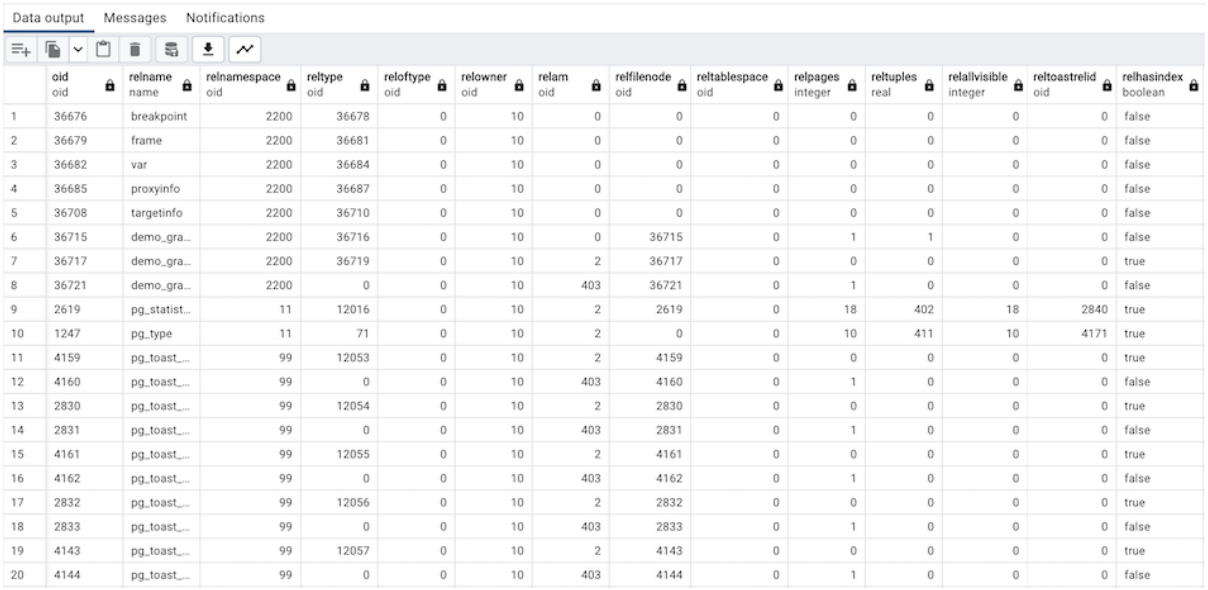
マクロを作成するには、 **クエリツール** の **マクロ** メニューから *ManageMacros* オプションを選択します。使用するキーを選択し、マクロの名前、クエリ、オプションで選択プレースホルダーを入力し、 **保存** ボタンをクリックしてマクロを保存します。



マクロを削除するには、**マクロの管理** ダイアログでマクロを選択し、**削除** ボタンをクリックします。サーバーは、マクロを削除するかどうかの確認を求めます。



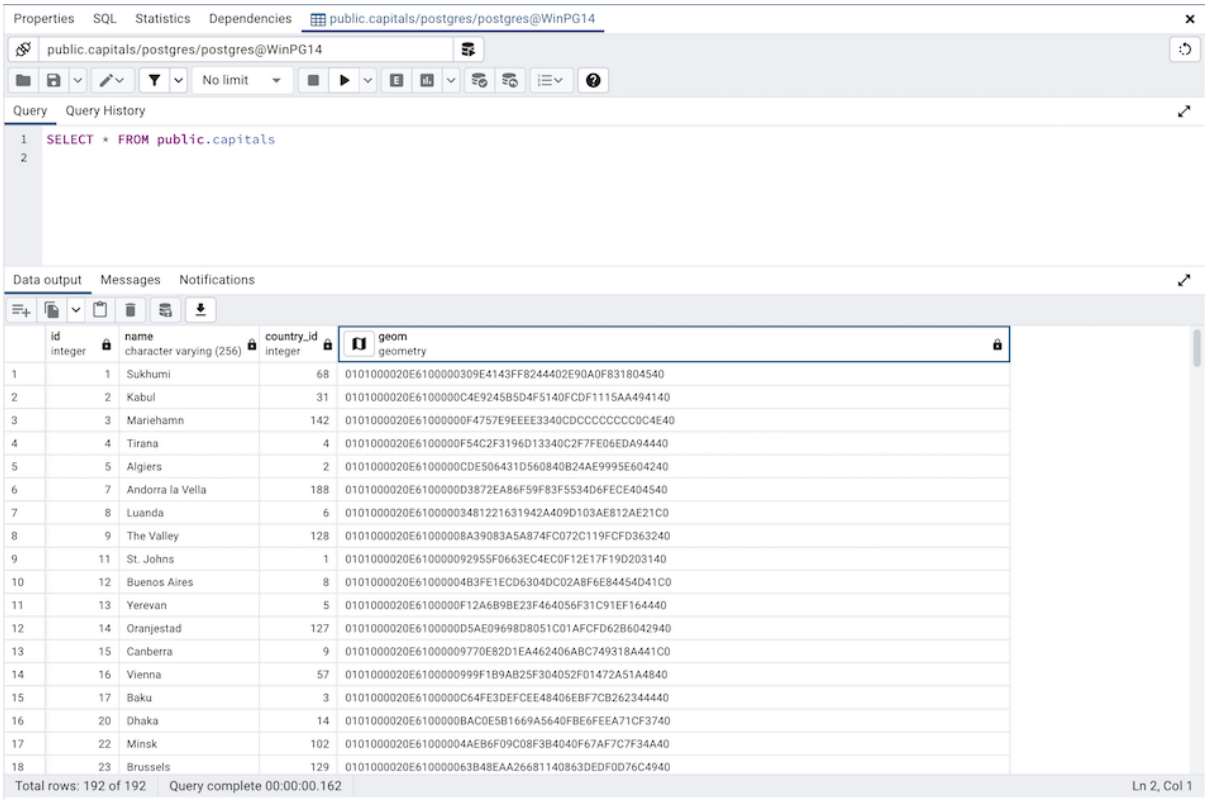
マクロを実行するには、適切なショートカットキーを選択するか、 **マクロ** メニューから選択します。


Data output Messages Notifications

	oid oid	relname name	relnamespace oid	reltype oid	reloftype oid	relowner oid	relam oid	relfilenode oid	reltablespace oid	relpages integer	reltuples real	relallvisible integer	reltoastrelid oid	relhasindex boolean
1	36676	breakpoint	2200	36678	0	10	0	0	0	0	0	0	0	false
2	36679	frame	2200	36681	0	10	0	0	0	0	0	0	0	false
3	36682	var	2200	36684	0	10	0	0	0	0	0	0	0	false
4	36685	proxyinfo	2200	36687	0	10	0	0	0	0	0	0	0	false
5	36708	targetinfo	2200	36710	0	10	0	0	0	0	0	0	0	false
6	36715	demo_gra...	2200	36716	0	10	0	36715	0	1	1	0	0	false
7	36717	demo_gra...	2200	36719	0	10	2	36717	0	0	0	0	0	true
8	36721	demo_gra...	2200	0	0	10	403	36721	0	1	0	0	0	false
9	2619	pg_statist...	11	12016	0	10	2	2619	0	18	402	18	2840	true
10	1247	pg_type	11	71	0	10	2	0	0	10	411	10	4171	true
11	4159	pg_toast_...	99	12053	0	10	2	4159	0	0	0	0	0	true
12	4160	pg_toast_...	99	0	0	10	403	4160	0	1	0	0	0	false
13	2830	pg_toast_...	99	12054	0	10	2	2830	0	0	0	0	0	true
14	2831	pg_toast_...	99	0	0	10	403	2831	0	1	0	0	0	false
15	4161	pg_toast_...	99	12055	0	10	2	4161	0	0	0	0	0	true
16	4162	pg_toast_...	99	0	0	10	403	4162	0	1	0	0	0	false
17	2832	pg_toast_...	99	12056	0	10	2	2832	0	0	0	0	0	true
18	2833	pg_toast_...	99	0	0	10	403	2833	0	1	0	0	0	false
19	4143	pg_toast_...	99	12057	0	10	2	4143	0	0	0	0	0	true
20	4144	pg_toast_...	99	0	0	10	403	4144	0	1	0	0	0	false

9.3 View/Edit Data

データを表示または変更するには、 **ブラウザー** ツリーコントロールでテーブルまたはビューの名前を右クリックします。コンテキストメニューが開いたら、 **データの表示/編集** メニューを使用して、エディターパネルに表示する行数を指定します。


Properties SQL Statistics Dependencies public.capitals/postgres/postgres@WinPG14

public.capitals/postgres/postgres@WinPG14

No limit

Query Query History

```
1  SELECT * FROM public.capitals
2
```

Data output Messages Notifications

	id integer	name character varying (256)	country_id integer	geom geometry
1	1	Sukhumi	68	0101000020E6100000309E4143FF8244402E90A0F831804540
2	2	Kabul	31	0101000020E6100000C4E9245B5D4F5140FCDF1115AA494140
3	3	Mariehamn	142	0101000020E61000000F4757E9EEEE3340CDCCCCCCCC0C4E40
4	4	Tirana	4	0101000020E6100000F54C2F3196D13340C2F7FE06EDA94440
5	5	Algiers	2	0101000020E6100000CDE506431D560840B24AE9995E604240
6	7	Andorra la Vella	188	0101000020E6100000D3872EA86F59F83F5534D6FECE404540
7	8	Luanda	6	0101000020E610000034812216319 42A409D103AE812AE21C0
8	9	The Valley	128	0101000020E61000008A39083A5A874FC072C119FCFD363240
9	11	St. Johns	1	0101000020E6100000929 55F0663EC4EC0F12E17F19D203140
10	12	Buenos Aires	8	0101000020E61000004B3FE1ECD6304DC02A8F6E84454D41C0
11	13	Yerevan	5	0101000020E6100000F12A6B9BE23F464056F31C91EF164440
12	14	Oranjestad	127	0101000020E6100000D5AE09698D8051C01AFCFD62B6042940
13	15	Canberra	9	0101000020E61000009770E82D1EA462406ABC749318A441C0
14	16	Vienna	57	0101000020E6100000999F1B9AB25F304052F01472A51A4840
15	17	Baku	3	0101000020E6100000C64FE3DEFCEE48406EBF7CB262344440
16	20	Dhaka	14	0101000020E6100000BAC0E5B1669A5640FBE6FEEA71CF3740
17	22	Minsk	102	0101000020E61000004AEB6F09C08F3B4040F67AF7C7F34A40
18	23	Brussels	129	0101000020E610000063B48EAA26681140863DEDF0D76C4940

Total rows: 192 of 192 Query complete 00:00:00.162 Ln 2, Col 1

テーブルの内容を変更するには、テーブルの各行を一意に識別できる必要があります。テーブル定義に OID または主キーが含まれていない場合、表示されるデータは読み取り専用です。ビューは編集できないことに注意してください。更新可能なビュー（ルールを使用）はサポートされていません。

エディターには、頻繁に使用するオプションにすばやくアクセスできるツールバーと、2 つのパネルに分かれた作業環境があります。

- 上部パネルには、下部パネルに表示された内容を選択するために使用された SQL コマンドが表示されます。
- 下部のパネル（データグリッド）には、テーブルまたはビューから選択されたデータが表示されます。

9.3.1 データの表示/編集ツールバー

Query Tool と *Query Tool* ツールは、実際には同じツールの異なる動作モードです。一部のコントロールはどちらのモードでも無効になります。使用可能なコントロールの説明については、 *Query Tool* を参照してください。

9.3.2 データグリッド

データグリッドの一番上の行には、各列の名前、データ型、および該当する場合は許可される文字数が表示されます。主キーの一部である列は、さらに [PK] でマークされます。

表示されたデータを変更するには：

- グリッド内の数値を変更するには、値をダブルクリックしてフィールドを選択します。表示されている正方形のコンテンツを変更します。
- グリッド内の数値以外の値を変更するには、コンテンツをダブルクリックして編集バブルにアクセスします。編集バブルの内容を変更した後、 *OK* ボタンをクリックして変更をデータグリッドに表示するか、 **キャンセル** をクリックして保存せずに編集バブルを終了します。

改行文字を入力するには、Ctrl-Enter または Shift-Enter をクリックします。改行のフォーマットは、フィールドのコンテンツが編集バブルを介してアクセスされた場合にのみ表示されます。

テーブルに新しい行を追加するには、テーブルの最後の（番号のない）行にデータを入力します。データを保存するとすぐに、行に行番号が割り当てられ、新しい空の行がデータグリッドに追加されます。

テーブルに SQLNULL を書き込むには、フィールドを空のままにします。新しい行を保存すると、サーバーはその列のデフォルト値を入力します。既存の行への変更を保存する場合、値 NULL が明示的に書き込まれます。

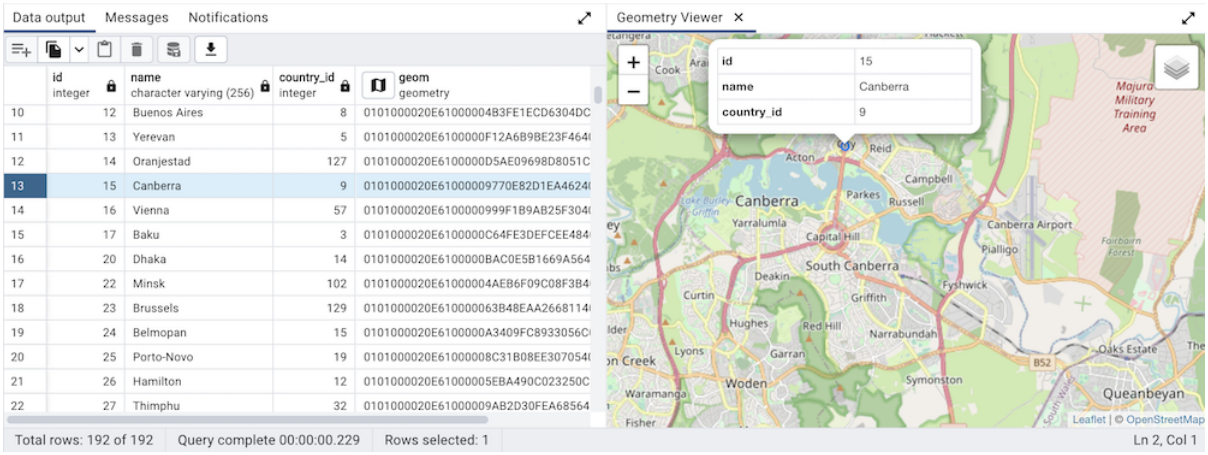
テーブルに空の文字列を書き込むには、フィールドに特殊な文字列"（2 つの一重引用符）を入力します。2 つの一重引用符のみを含む文字列をテーブルに書き込む場合は、「/'/」と入力してこれらの引用符をエスケープする必要があります

行を削除するには、 *Delete* ツールバーボタンを押します。ポップアップが開き、削除を確認します。

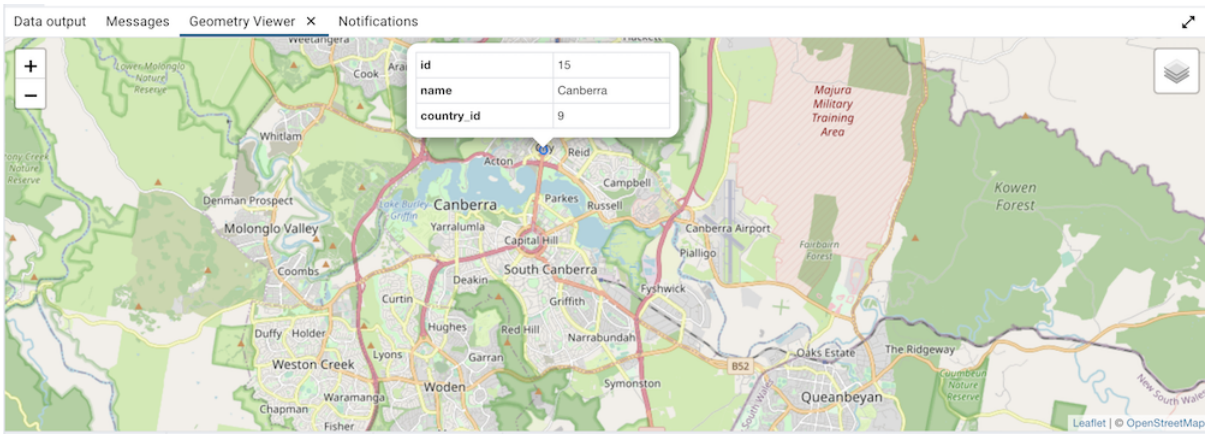
サーバーに変更をコミットするには、 **データの保存** ツールバーボタンを選択します。

Geometry Data Viewer

PostGIS がインストールされている場合、行を選択して列の [ViewGeometry] ボタンをクリックすると、地図に GIS オブジェクトを表示できます。行が選択されていない場合、データセット全体がレンダリングされます。



パネルのタイトルをドラッグして、レイアウトを調整できます。ジオメトリのプロパティをマップで直接表示するには、特定のジオメトリをクリックするだけです。

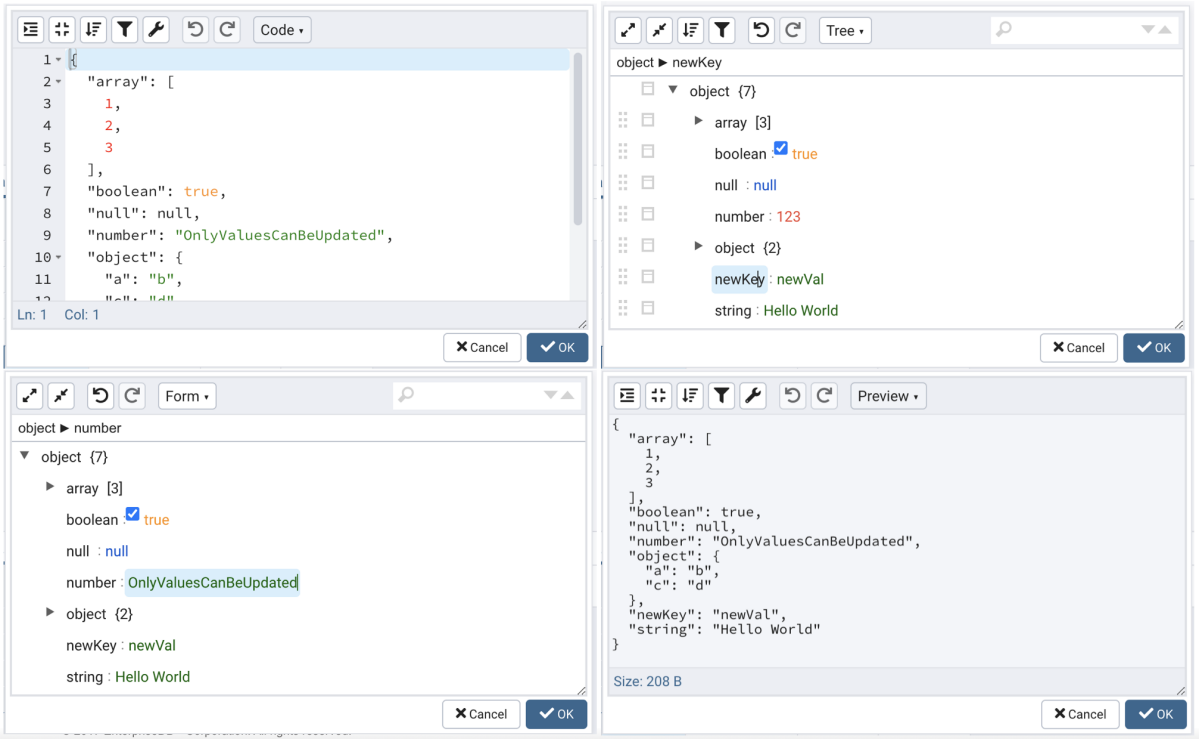


注釈:

- **サポートされているデータタイプ：** GeometryViewer は、`Point、LineString、PolygonMultiPoint、MultiLineString、MultiPolygon、GeometryCollection`を含む EWKB 形式の 2D および 3DM ジオメトリをサポートしています。
- *SRID:* 同じ列に異なる SRID を持つジオメトリがある場合、ビューアはマップ内の同じ SRID を持つジオメトリをレンダリングします。SRID=4326 の場合、OSM タイルレイヤーがマップに追加されます。
- **データサイズ**: パフォーマンス上の理由から、ビューアは合計 100000 個以下のジオメトリをレンダリングします。
- **インターネットアクセス**: GeometryViewer が正しく機能するには、インターネット接続が必要です。

JSON Data Editor

JSON/JSONB **データ** 用にビルトインの json エディターが提供されています。データグリッドの json/jsonb データ型のセルをダブルクリックすると、JSON エディターが開きます。エディターは、json データを表示および編集するためのさまざまなモードを提供します。



コードモード ：json データをフォーマットおよび圧縮する方法を提供します。また、引用符とエスケープ文字を修正し、コメントと JSONP 表記を削除して、JavaScript オブジェクトを JSON に変換する機能を提供します。

ツリーモード ：フィールドと値を変更、追加、移動、削除、および複製できます。データを検索および強調表示する機能を提供します。

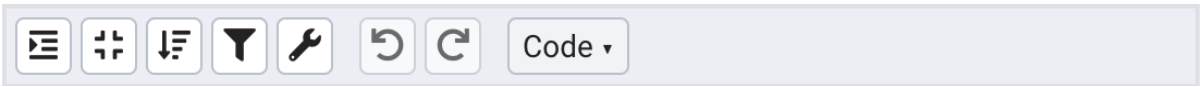
フォームモード ：編集中にデータ構造を変更しないでください。

プレビューモード ：保存する前にデータを確認する機能を提供し、現在の json データのサイズも表示します。json データもフォーマットしてコンパクトにします。

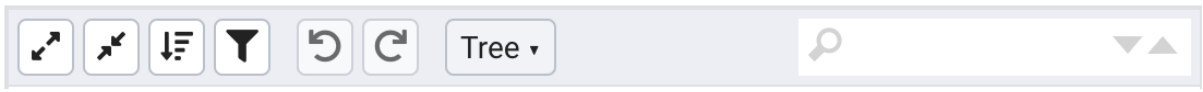
エディターツールバー

json データを操作するためのさまざまなオプションが提供されています。

コード/プレビューモード：



ツリー/フォームモード：



アイコン	行動	モードで利用可能
フォーマット *Json*	クリックして、適切なインデントで json データを *FormatJson* フォーマットします。	コード、プレビュー

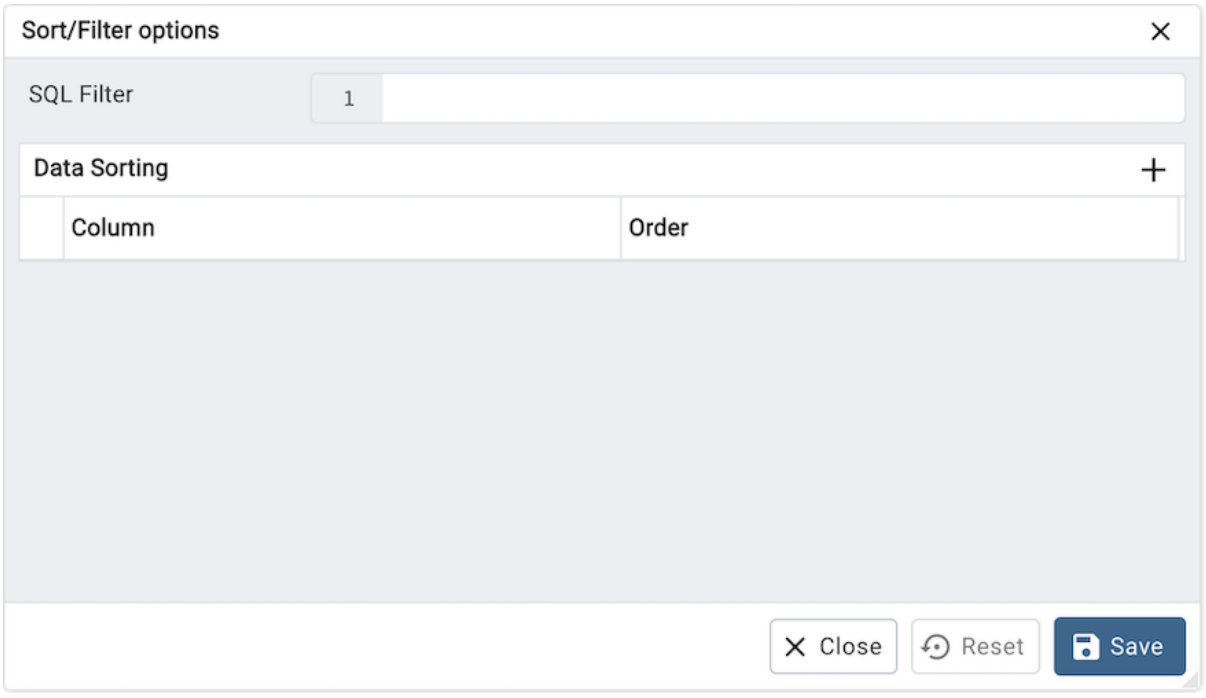
次のページに続く

表 6 – 前のページからの続き

アイコン	行動	モードで利用可能
コンパクト *Json*	クリックして *CompactJson* で圧縮された json データを取得します。	コード、プレビュー
並べ替え	**並べ替え** アイコンをクリックして、ファイルの値を使用してデータの並べ替え基準を設定します。	コード、プレビュー、ツリー
変身	クリックして **変換** し、JSME クエリ言語を使用してデータをフィルター処理します。	コード、プレビュー、ツリー
元に戻す	クリックして **元に戻す** データに対して実行された最後の操作を元に戻します。	コード、プレビュー、ツリー、から
やり直し	クリックして **やり直し** 、データに対して実行された最後のアクションを再実行します。	コード、プレビュー、ツリー、から
モード	クリックして **モード** ドロップダウンして、エディターのディパリーモードを変更します。使用可能なモードは、コード、プレビュー、ツリー、フロムです。	コード、ツリー、フロム、プレビュー
すべて展開	クリックして **すべて展開** して json データを展開します。	ツリー、から
すべて折りたたむ	クリックして **やり直し** 、json データを折りたたみます。	ツリー、から
検索ボックス	データを検索する文字列の一部/全体を入力します。	ツリー、から

9.3.3 並べ替え/フィルターオプションダイアログ

並べ替え/フィルターボタンをクリックすると、 **並べ替え/フィルターオプションダイアログ** にアクセスできます。これにより、SQL フィルターを指定して、編集グリッドウィンドウに表示されるデータとデータの並べ替えオプションを制限できます。



- *SQLFilter* を使用して、SQL フィルタリング基準を提供します。これらは、データの取得に使用されるクエリの/"WHERE/"句に追加されます。たとえば、次のように入力します。

```
id > 25 AND created > '2018-01-01'
```

- **データの並べ替え** を使用して、出力グリッドのデータを並べ替えます

データ並べ替えグリッドに新しい列を追加するには、[+] アイコンをクリックします。

- ドロップダウン **列** を使用して、並べ替える列を選択します。
- ドロップダウン *Order* を使用して、列の並べ替え順序を選択します。

グリッドから行を削除するには、ゴミ箱アイコンをクリックします。

- *Save* ボタンをクリックして作業を保存します。
- **閉じる** ボタンをクリックして、現在の変更を破棄し、ダイアログを閉じます。

View/Edit Data Filter

ブラウザパネルに表示される *FilteredRows* ツールバーボタンをクリックするか、*View/EditData->FilteredRows* コンテキストメニューオプションを選択して、 **データフィルターダイアログ** にアクセスできます。

これにより、SQL フィルターを指定して、編集グリッドウィンドウに表示されるデータを制限できます。



9.4 Schema Diff

Schema Diff は、2 つのデータベースまたは 2 つのスキーマ間でオブジェクトを比較できる機能です。 *Tools* メニューを使用して、SchemaDiff にアクセスします。

スキーマ差分機能を使用すると、次のことができます。

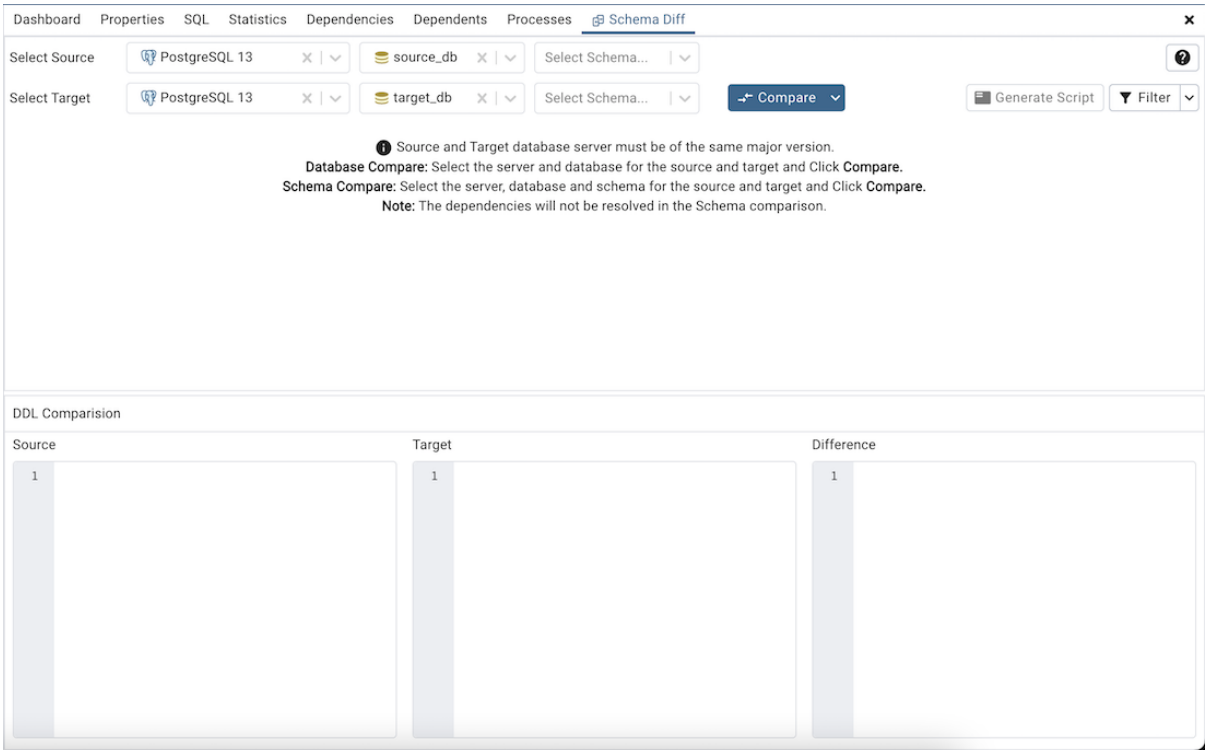
- データベースオブジェクトを（ソースからターゲットへ）比較して同期します。
- データベースオブジェクト間の違いを視覚化します。
- ターゲットデータベースオブジェクトの SQL ステートメントの違いをリストします。
- 同期スクリプトを生成します。

Note:

- ソースデータベースサーバーとターゲットデータベースサーバーは、同じメジャーバージョンである必要があります。
- 2 つの **schemas** を比較すると、依存関係は解決されません。

Tools メニューの下の *SchemaDiff* をクリックして、選択パネルを開きます。 **databases** を比較するには、ソースサーバーとターゲットサーバー、およびデータベースを選択します。 **databases** を比較するには、ソースとターゲットのサーバー、データベース、およびスキーマを選択します。オブジェクトを選択したら、 **比較** ボタンをクリックします。

個々のタブで *SchemaDiff* の複数のコピーを開くことができます。SchemaDiff のコピーを閉じるには、タブバーの右上隅にある *X* をクリックします。右クリックして/"RenamePanel/"オプションを選択すると、パネルのタイトルの名前を変更できます。



Preferences ダイアログを使用して、次を指定します。

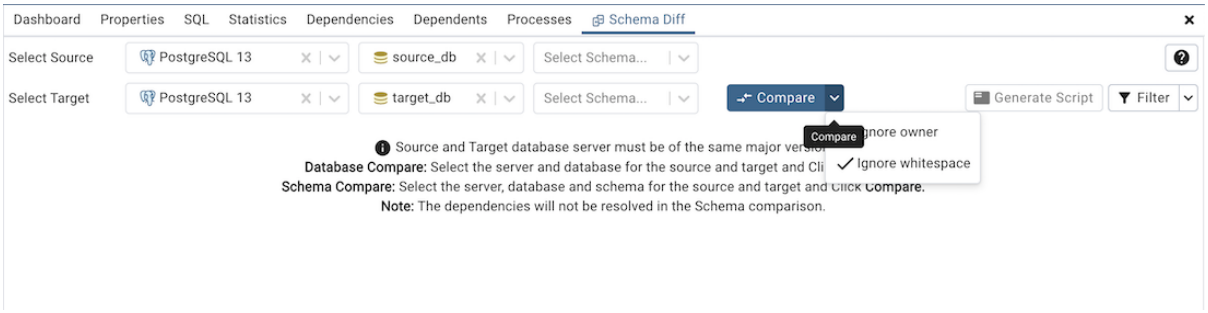
- *SchemaDiff* が新しいブラウザータブで開きます。 **新しいブラウザータブで開く** オプションを true に設定します。
- *SchemaDiff* は、文字列オブジェクトの比較中に空白を無視する必要があります。 **空白を無視** オプションを true に設定します。
- *SchemaDiff* は、オブジェクトの比較中に所有者を無視する必要があります。 **所有者を無視** オプションを true に設定します。

SchemaDiff パネルは 2 つのパネルに分かれています。オブジェクト比較パネルと DDL 比較パネル。

9.4.1 スキーマ差分オブジェクト比較パネル

オブジェクト比較パネルでは、同じメジャーバージョンのソースサーバーとターゲットサーバー、および比較するデータベースを選択できます。接続されているか切断されているかにかかわらず、ブラウザーツリーの下にリストされているサーバーを選択できます。接続されていないサーバーを選択すると、サーバーを使用する前にパスワードの入力を求められます。

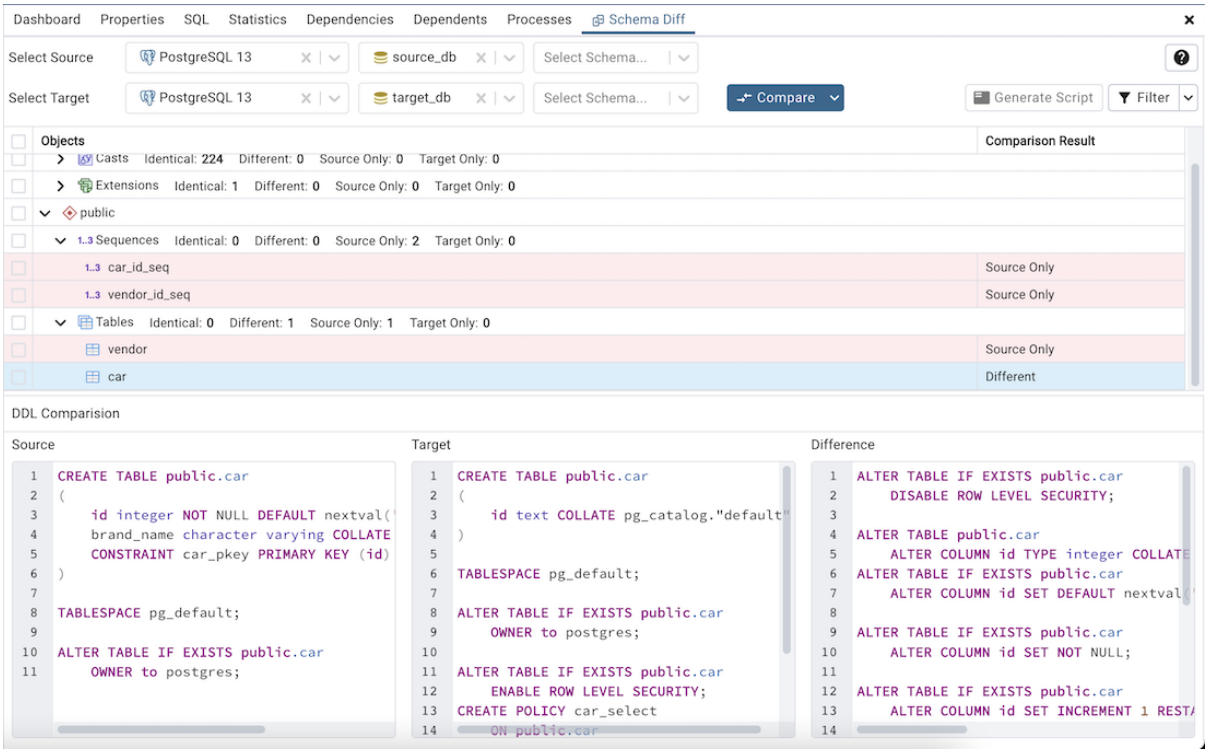
次に、比較するデータベースを選択します。データベースは同じでも異なっていてもかまいません（および同じサーバー内または別のサーバーから）。



所有者を無視し、空白を無視するには、 **比較** ボタンの近くにあるドロップダウンを使用します。

- 所有者を無視する–オブジェクトの比較中に所有者を無視する場合に選択します。
- 空白を無視–文字列オブジェクトの比較中に空白を無視する場合に選択します。ホワイトスペースには、スペース、タブ、CRLF が含まれます。

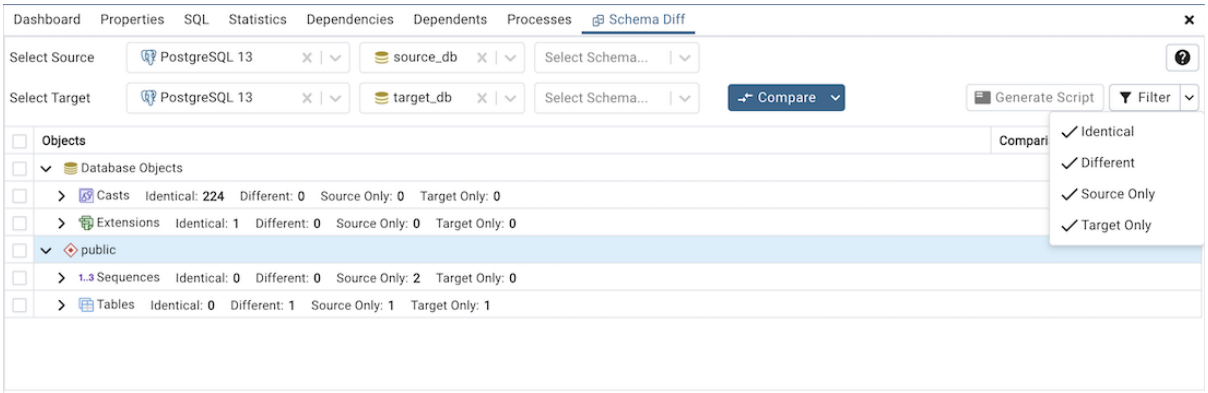
サーバーとデータベースを選択したら、 **比較** ボタンをクリックして **比較結果** を取得します。



データベースオブジェクトのドロップダウンリストを使用して、DDL ステートメントを表示します。

オブジェクト比較パネルの右上隅には、次の比較基準に基づいてデータベースオブジェクトをフィルター処理するために使用できる **フィルター** オプションがあります。

- 同一–同じ SQL ステートメントを使用してオブジェクトが両方のデータベースで見つかった場合、比較結果は同一です。
- 異なる–オブジェクトが両方のデータベースで見つかったが、SQL ステートメントが異なる場合、比較結果は異なります。
- ソースのみ–オブジェクトがソースデータベースのみで見つかった場合、比較結果はソースのみです。
- ターゲットのみ–オブジェクトがターゲットデータベースのみで見つかった場合、比較結果はターゲットのみです。



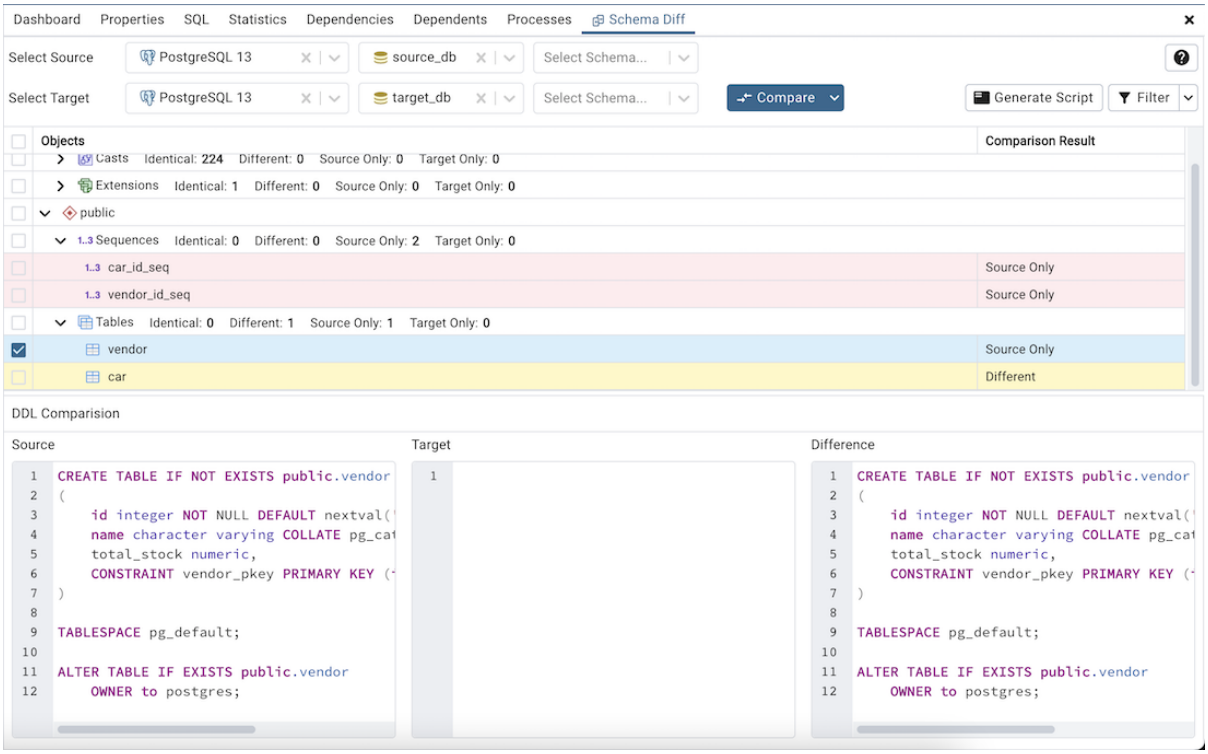
オブジェクト比較パネルでデータベースオブジェクトをクリックして、そのオブジェクトの DDL ステートメントを DDL 比較パネルに表示します。

ヘルプ アイコンをクリックして、SchemaDiff ドキュメントを開きます。

9.4.2 スキーマ相違 DDL 比較パネル

DDLCompare パネルには 3 つの列が表示されます。

- 最初の列には、ソースデータベースからのオブジェクトの DDL ステートメントが表示されます。
- 2 番目の列には、ターゲットデータベースからのオブジェクトの DDL ステートメントが表示されます。
- 3 番目の列には、ターゲットデータベースオブジェクトの SQL ステートメントの違いが表示されます。



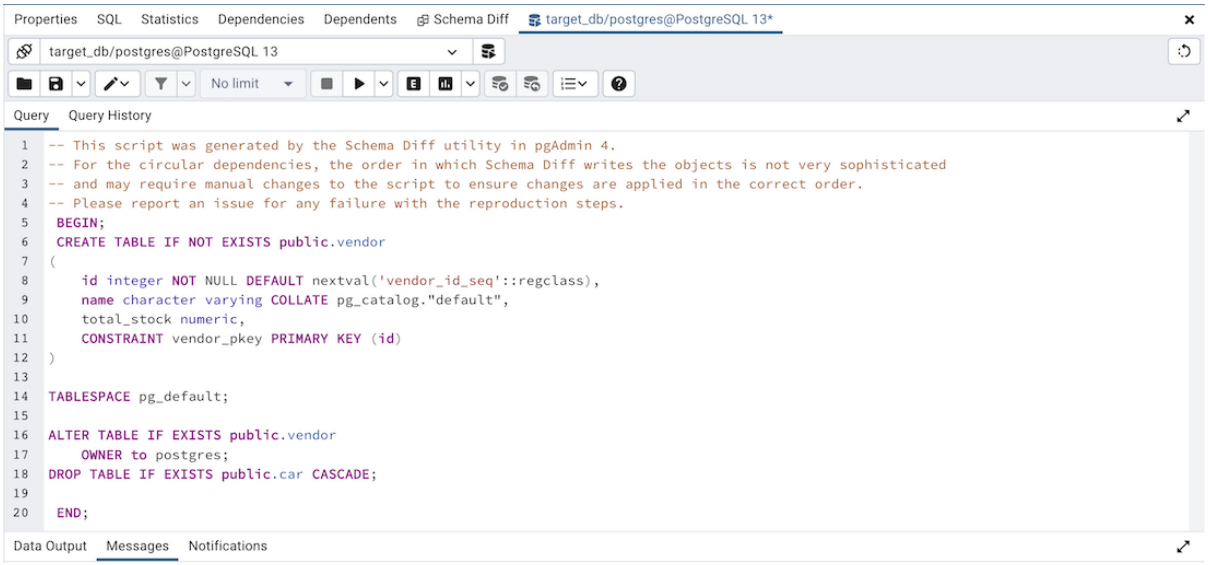
すべてのデータベースオブジェクトの DDL ステートメントを確認して、SQL ステートメントの違いを確認できます。

また、ソースデータベースオブジェクトの SQL ステートメントに基づいて、ターゲットデータベースオブジェクトで見つかった違いの SQL スクリプトを生成できます。スクリプトを生成するには、オブジェクト比較パネルでデータベースオブジェクトのチェックボックスを選択し、オブジェクト比較パネルの右上隅にある *GenerateScript* ボタンをクリックします。

データベースオブジェクトを選択し、**スクリプトの生成** ボタンをクリックして、新しいタブで **クエリツール** を開き SQL ステートメントの違いは **クエリエディター** に表示されます。

データベースオブジェクトをクリックして *DDL* **比較** パネルで生成された違いを確認し、データベースオブジェクトのチェックボックスを選択していない場合、pgAdmin は新しいタブで **クエリツール** を開きます。 **クエリエディター** に表示される SQL ステートメント。

コピー ボタンを使用して、 *DDL* **比較** パネルで生成された差分をコピーすることもできます。

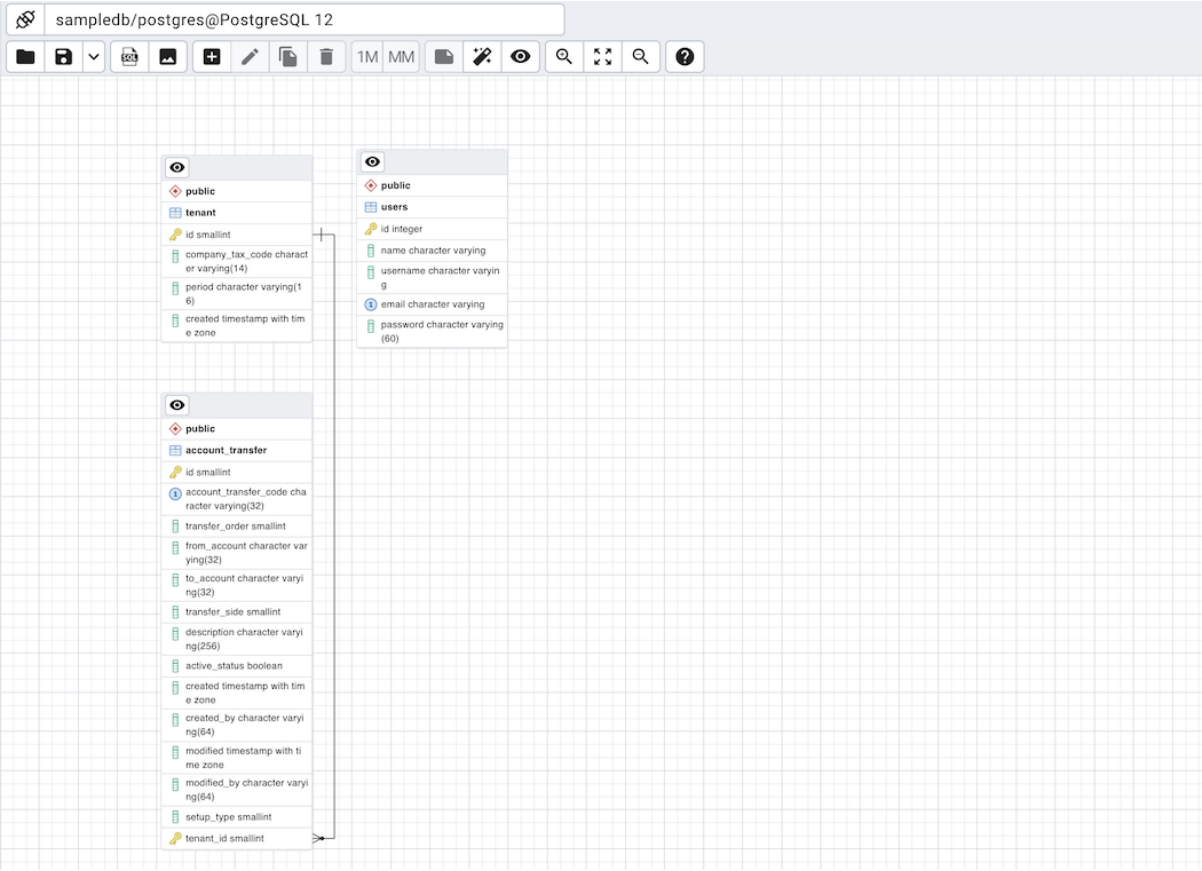


ターゲットデータベースに SQL ステートメントを適用して、データベースを同期します。

9.5 ERD Tool

エンティティ関係図（ERD）ツールは、データベースのテーブル、列、相互関係をグラフィカルに表現するデータベースデザインツールです。ERD は、データベース管理者がデータベースを開発および保守するときに従う十分な情報を提供できます。ERD ツールを使用すると、次のことができます。

- データベーステーブルとそのリレーションシップを設計および視覚化します。
- 図にメモを追加します。
- テーブルとリンクを自動配置して、より明確に視覚化します。
- ダイアグラムを保存して後で開いて、作業を続行します。
- データベースデザインからすぐに実行できる SQL を生成します。
- 既存のデータベースのデータベースダイアグラムを生成します。
- ブラウザーツリーからダイアグラムにテーブルをドラッグアンドドロップします。



ERD ツールの複数のコピーを個々のタブで同時に開くことができます。ERD ツールのコピーを閉じるには、タブバーの右上隅にある *X* をクリックします。

9.5.1 ツールバー

ERDTool ツールバーは、頻繁に実行されるタスクへのショートカットを提供する状況依存のアイコンを使用します。このオプションは、強調表示されたアイコンでは有効になっており、グレー表示されているアイコンでは無効になっています。



ツールバーのアイコンにカーソルを合わせると、アイコンの機能を説明するツールチップが表示されます。

9.5.2 ファイルオプション

アイコン	行動	ショートカット
ファイルを開く	*OpenFile* アイコンをクリックして、以前に保存した図をロードします。	Ctrl+O

次のページに続く

表 7 – 前のページからの続き

アイコン	行動	ショートカット
保存	*Save* アイコンをクリックして、以前に保存した図をすばやく保存するか、図をファイルに保存します。	Ctrl+S
名前を付けて保存	**名前を付けて保存** をクリックして新しいブラウザーダイアログを開き、図を保存する新しい場所を指定します。[名前を付けて保存] を表示するには、保存ボタンの横にある下矢印をクリックする必要があります。	Ctrl+Shift+S

9.5.3 エクスポートオプション

アイコン	行動	ショートカット
SQL **を生成する**	*GenerateSQL* アイコンをクリックして図の DDLSQL を生成し、生成された SQL を実行する準備ができているクエリツールを開きます。各 CREATETableDDL の前に DROPTableDDL ステートメントを使用する場合は、オプション *WithDROPTable* を選択できます。SQL ボタンの横にある下矢印をクリックすると、オプションが表示されます。	Option+Ctrl+S
画像をダウンロード	**画像をダウンロード** アイコンをクリックして、ERD 図を画像形式で保存します	Option+Ctrl+I

9.5.4 編集オプション

アイコン	行動	ショートカット
テーブルを追加	このボタンをクリックして、新しいテーブルをダイアグラムに追加します。クリックすると、テーブルの詳細を入力できるテーブルダイアログが開きます。	Option/Alt+Ctrl+A
テーブルを編集	このボタンをクリックして、ダイアグラム上のテーブルを編集します。クリックすると、テーブルの詳細を変更できるテーブルダイアログが開きます。これは、テーブルが選択されたときに有効になります。	Option/Alt+Ctrl+E
クローンテーブル	このボタンをクリックして、完全なテーブル構造を複製し、自動生成された名前を付けて図に配置します。	Option/Alt+Ctrl+C
ドロップテーブル/リンク	このボタンを使用して、テーブルまたはリンクをドロップできます。ドロップするには、テーブルまたはリンクを選択してこのボタンをクリックする必要があります。	Option/Alt+Ctrl+D

9.5.5 テーブル関係オプション

アイコン	行動	ショートカット
1M	このボタンをクリックして 1 対多のリレーションシップダイアログを開き、2 つのテーブル間のリレーションシップを追加します。選択したテーブルが参照テーブルになり、リンクの **多く** のエンドポイントがあります。	Option/Alt+Ctrl+O
MM	このボタンをクリックして多対多のリレーションシップダイアログを開き、2 つのテーブル間のリレーションシップを追加します。このオプションは、2 つの関連するテーブルで選択された列に基づいて新しいテーブルを作成し、それらをリンクします。	Option/Alt+Ctrl+M

9.5.6 ノードの色のオプション

アイコン	行動
塗りつぶし色	FillColor を使用して、テーブルノードの背景色を変更します。これは、グループテーブルを識別したい場合に役立ちます。設定すると、新しく追加されたすべてのテーブルが同じ色になります。
テキストの色	テキストの色を使用して、塗りつぶしの色に基づいてテーブルノードのテキストの色を変更して、テキストを読みやすくします。

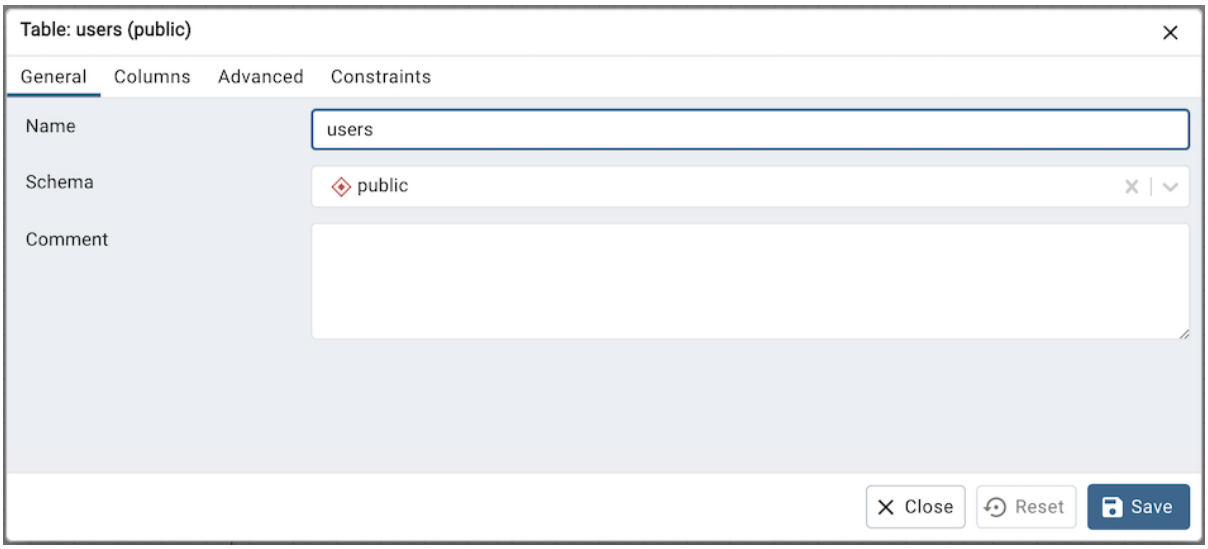
9.5.7 ユーティリティオプション

アイコン	行動	ショートカット
メモの追加/編集	このボタンをクリックして、データベースの設計中にテーブルノードに関するメモを作成します。	Option/Alt+Ctrl+N
自動整列	このボタンをクリックして、すべてのテーブルとリンクを自動配置して、見やすくします。	Option/Alt+Ctrl+L
詳細を表示	このボタンをクリックして、列の詳細の表示を切り替えます。これにより、いくつかの列の詳細を表示できます。	Option/Alt+Shift+D

9.5.8 ズームオプション

アイコン	行動	ショートカット
フィットするようにズーム	このボタンをクリックして、自動的にズームイン/ズームアウトし、すべてのテーブルをビューに合わせます。	Option/Alt+Shift+F
ズームイン	このボタンをクリックして、図をズームインします。	Option/Alt+Shift+/"+/"
ズームアウト	このボタンをクリックして、図をズームアウトします。	Option/Alt+Shift+/"-/"

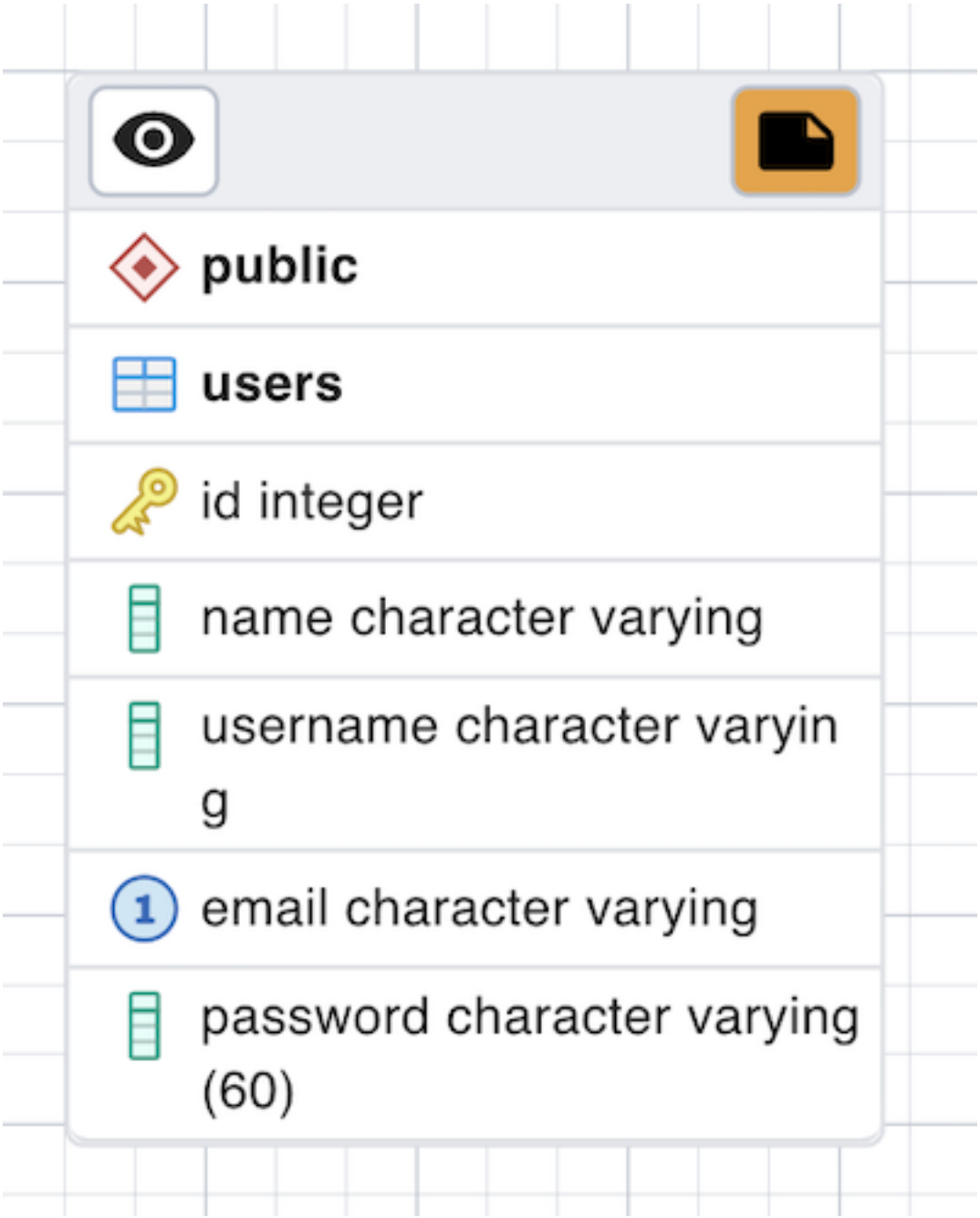
9.5.9 テーブルダイアログ



テーブルダイアログでは、次のことができます。

- テーブル構造の詳細を変更します。
- 既存のテーブルを編集したり、新しいテーブルを追加したりできます。
- さまざまなフィールドについては、 *table dialog* を参照してください。

9.5.10 テーブルノード



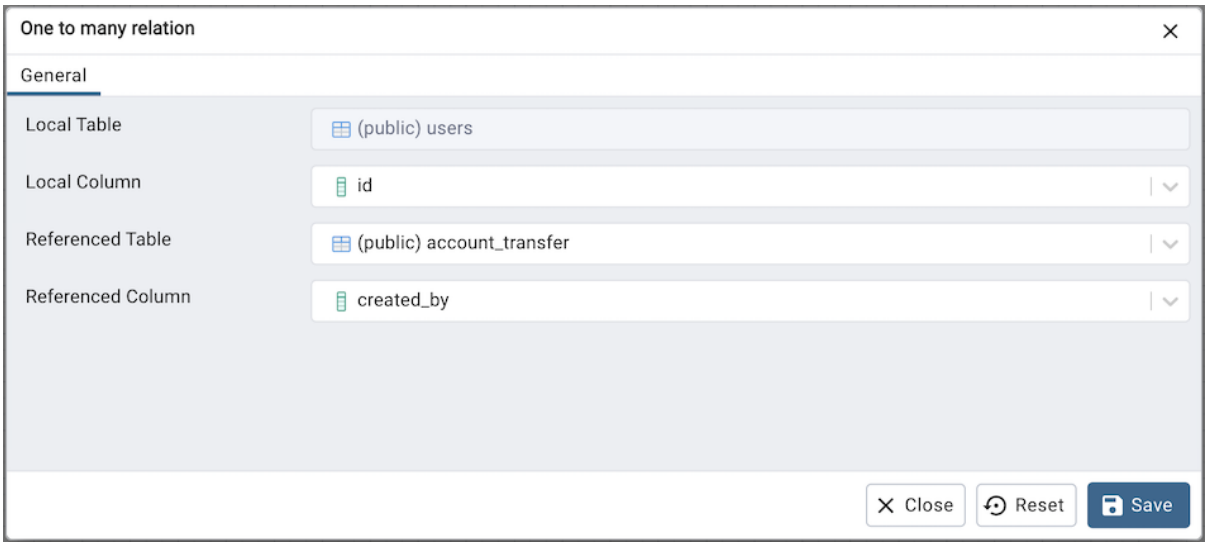
テーブルノードには、テーブルの詳細がグラフィカルに表示されます。

- トップバーには、列の詳細の表示を切り替えるために使用される **詳細トグルボタン** があります。ノートが追加された場合にのみ表示される **ノートボタン** もあります。このボタンをクリックして、ノートをすばやく変更できます。
- 最初の行は、テーブルのスキーマ名を示しています。例上の画像の *public* 。
- 2 行目はテーブル名を示しています。例上の画像の *users* 。
- テーブル名の下の他のすべての行は、テーブルの列とデータ型です。列がプライマリキーの場合、ロックキーアイコンがあります。id は上の画像の主キーです。それ以外の場合は、列アイコンが表示され

ます。

- ノードをクリックしてドラッグして、キャンバス上を移動できます。
- テーブルノードをダブルクリックするか、ツールバーの編集ボタンをクリックすると、テーブルダイアログが開き、テーブルの詳細を変更できます。さまざまなフィールドについては、 *table dialog* を参照してください。

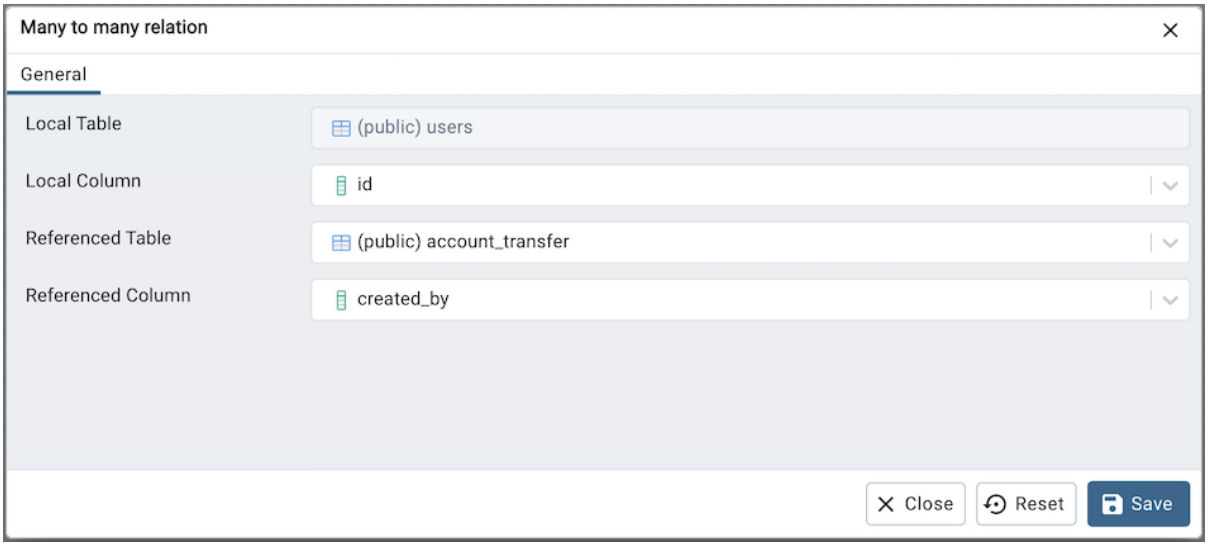
9.5.11 1 対多リンクダイアログ



1 対多のリンクダイアログでは、次のことができます。

- 2 つのテーブル間に外部キーリレーションシップを追加します。
- **ローカルテーブル** は、テーブルを参照し、 *many* のエンドポイントを持つテーブルです。
- **ローカル列** 参照する列。
- *ReferencedTable* は、参照されている *1* **つの** エンドポイントを持つテーブルです。
- **参照される列** 参照されている列。

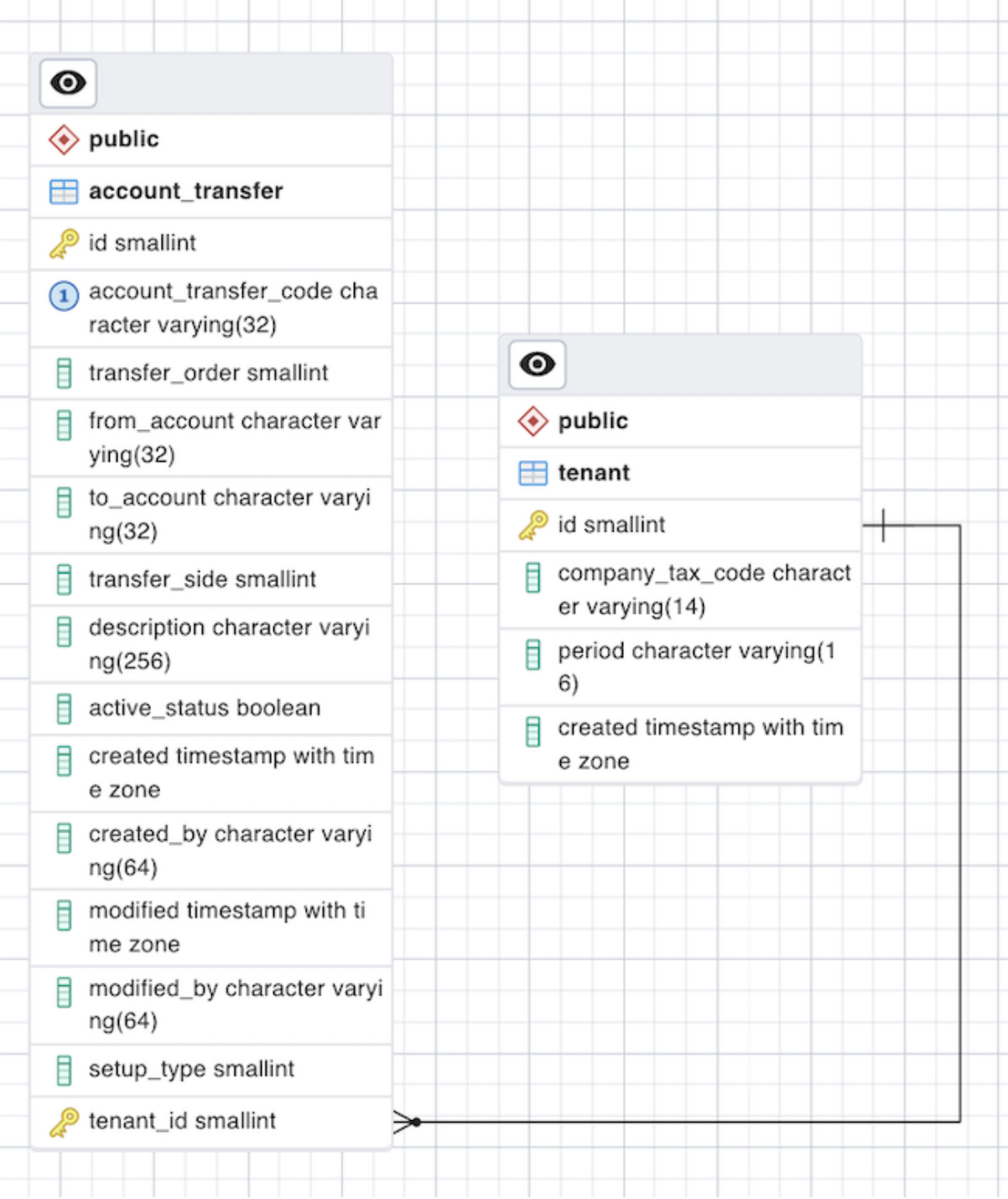
9.5.12 多対多リンクダイアログ



多対多リンクダイアログでは、次のことができます。

- 2 つのテーブル間に多対多のリレーションシップを追加します。
- 2 つのテーブルから派生した列を持つリレーションシップテーブルを作成し、それらをテーブルにリンクします。
- **左のテーブル** は、リンクされる最初のテーブルです。新しいリレーションテーブルとのリンクの *1* **つの** エンドポイントを受け取ります。
- **左の列** 最初のテーブルの列。常に主キーです。
- **右のテーブル** は、リンクされる 2 番目のテーブルです。新しいリレーションテーブルとのリンクの *1* **つの** エンドポイントを受け取ります。
- **右列** 2 番目のテーブルの列。常に主キーです。

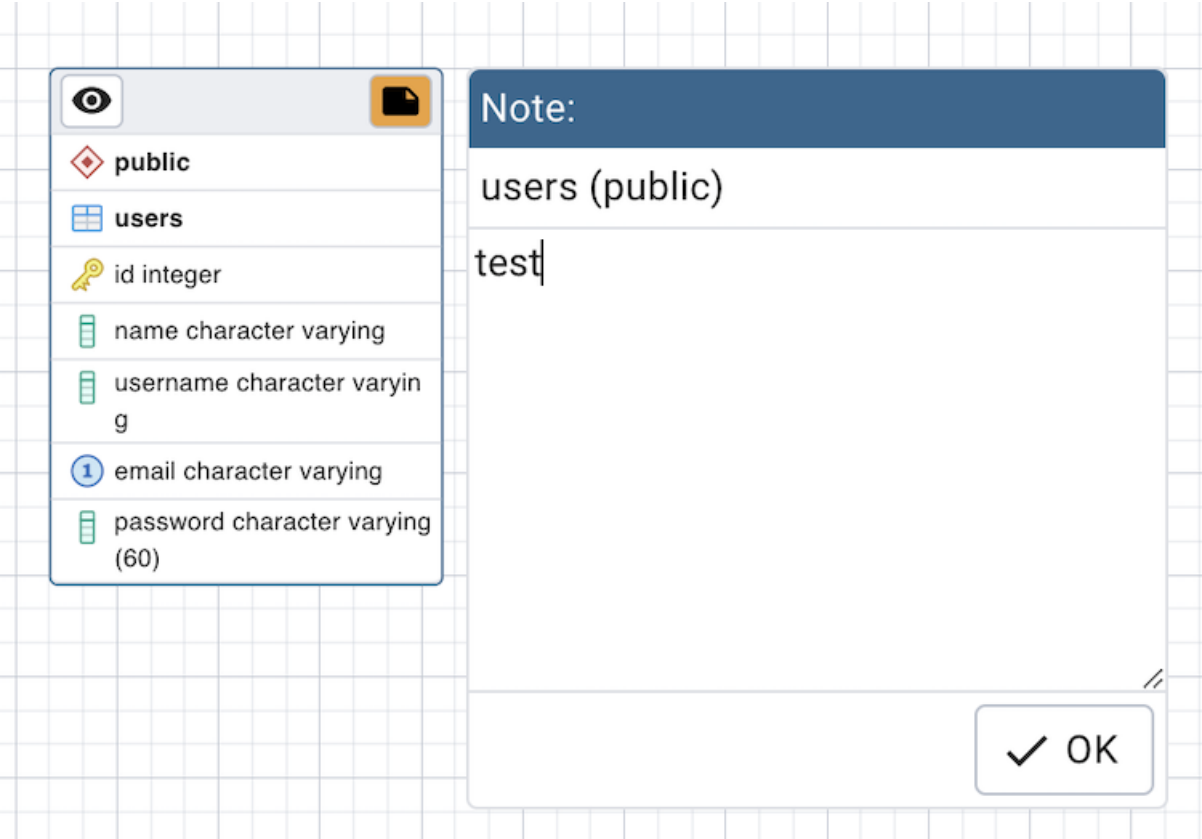
9.5.13 テーブルリンク



テーブルリンクは、テーブル間の関係を示しています。

- リンクの 1 行の端点は、参照されている列を示しています。
- リンクの 3 本線の端点は、参照する列を示しています。
- 参照されている列または参照している列のいずれかがテーブルから削除されると、リンクが削除されます。
- リンクをクリックしてドラッグして、キャンバス上を移動できます。

9.5.14 テーブルノート

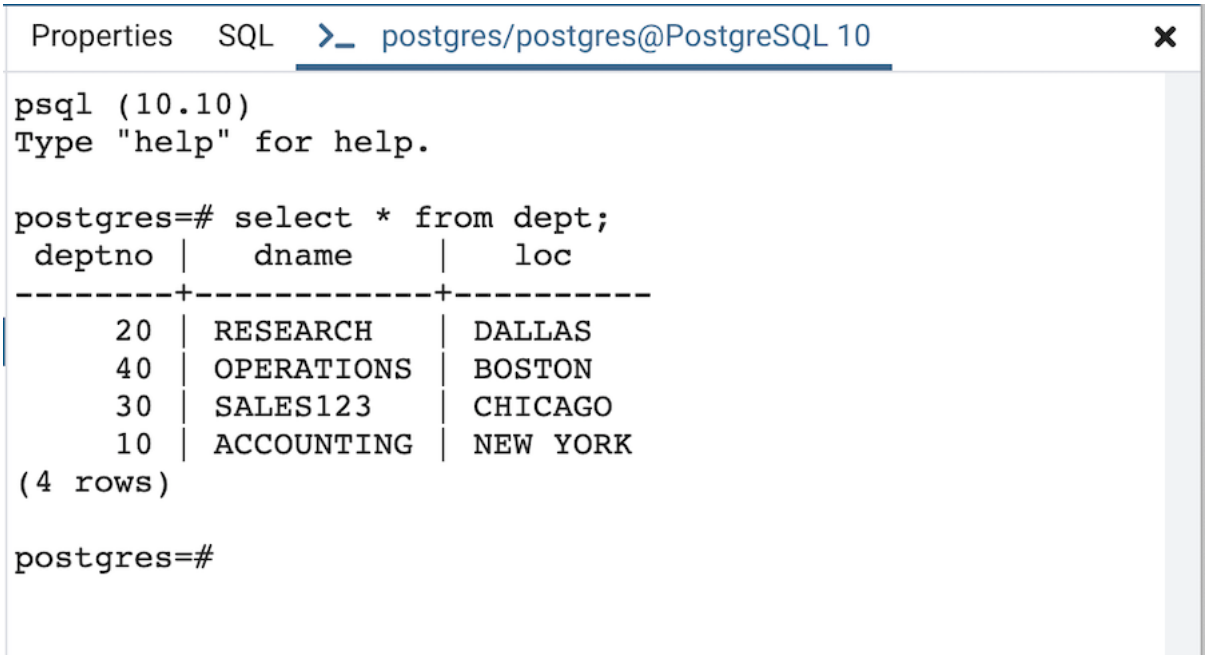


- ノートポップアップを使用して、データベースのデザイン中にノートをマークできます。
- ポップアップは、ツールバーのノートボタンを使用して開きます。
- テーブルにノートが追加されると、テーブルノードにノートボタンが表示されます。ボタンをクリックすると、ノートを確認・更新できます。

9.6 PSQL Tool

PSQL ツールを使用すると、ユーザーはブラウザーから psql コマンドラインインターフェイスを使用して、PostgreSQL または EDBAdvanced サーバーに接続できます。

- ツールまたはブラウザーツリーのコンテキストメニューから PSQL ツールを開くか、ブラウザーツリーの上部にある PSQL ツールボタンを使用します。
- PSQL はブラウザーツリーから現在接続されているデータベースに接続します。



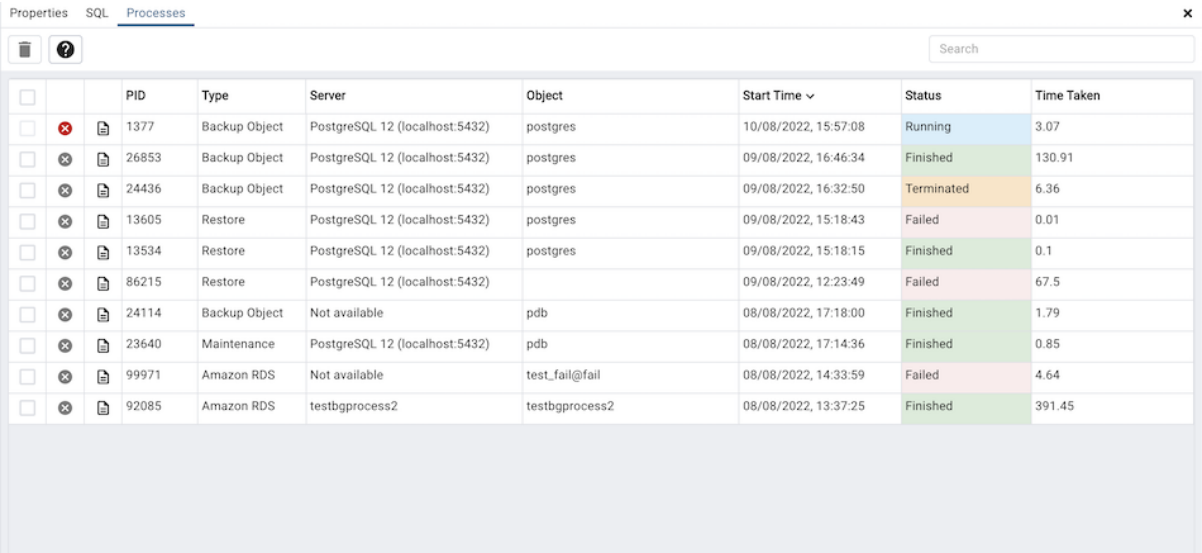
個々のタブで PSQL ツールの複数のインスタンスを同時に開くことができます。PSQL ツールを閉じるには、タブバーの右上隅にある *X* をクリックします。

Note: Windows プラットフォームでは、この機能は Windows10（1809 バージョン）、および WindowsServer2019 以降で利用できます。

Note: pgAdmin がデスクトップモードで実行されている場合、PSQL ツールを常に使用できますが、サーバーモードではデフォルトで無効になっています。これは、ユーザーが psql を介して任意のシェルコマンドを実行できるためです。これは、一部の展開ではセキュリティリスクと見なされる可能性があります。システム管理者は、*ENABLE_PSQL* オプションを *True* に設定することにより、pgAdmin 構成で PSQL ツールの使用を有効にできます。詳細については、 **Note:** を参照してください。

第10章　Processes

pgAdmin がバックグラウンドで実行する特定のタスクがあります。バックグラウンドで実行されているプロセスは、[プロセス] タブで表示できます。バックアップ、復元、メンテナンス、インポート/エクスポート、およびクラウドインスタンス作成のプロセスの詳細を示しています。



			PID	Type	Server	Object	Start Time	Status	Time Taken
☐	⊗	📄	1377	Backup Object	PostgreSQL 12 (localhost:5432)	postgres	10/08/2022, 15:57:08	Running	3.07
☐	⊗	📄	26853	Backup Object	PostgreSQL 12 (localhost:5432)	postgres	09/08/2022, 16:46:34	Finished	130.91
☐	⊗	📄	24436	Backup Object	PostgreSQL 12 (localhost:5432)	postgres	09/08/2022, 16:32:50	Terminated	6.36
☐	⊗	📄	13605	Restore	PostgreSQL 12 (localhost:5432)	postgres	09/08/2022, 15:18:43	Failed	0.01
☐	⊗	📄	13534	Restore	PostgreSQL 12 (localhost:5432)	postgres	09/08/2022, 15:18:15	Finished	0.1
☐	⊗	📄	86215	Restore	PostgreSQL 12 (localhost:5432)		09/08/2022, 12:23:49	Failed	67.5
☐	⊗	📄	24114	Backup Object	Not available	pdb	08/08/2022, 17:18:00	Finished	1.79
☐	⊗	📄	23640	Maintenance	PostgreSQL 12 (localhost:5432)	pdb	08/08/2022, 17:14:36	Finished	0.85
☐	⊗	📄	99971	Amazon RDS	Not available	test_fail@fail	08/08/2022, 14:33:59	Failed	4.64
☐	⊗	📄	92085	Amazon RDS	testbgprocess2	testbgprocess2	08/08/2022, 13:37:25	Finished	391.45

processes テーブルの列には次のものが表示されます。

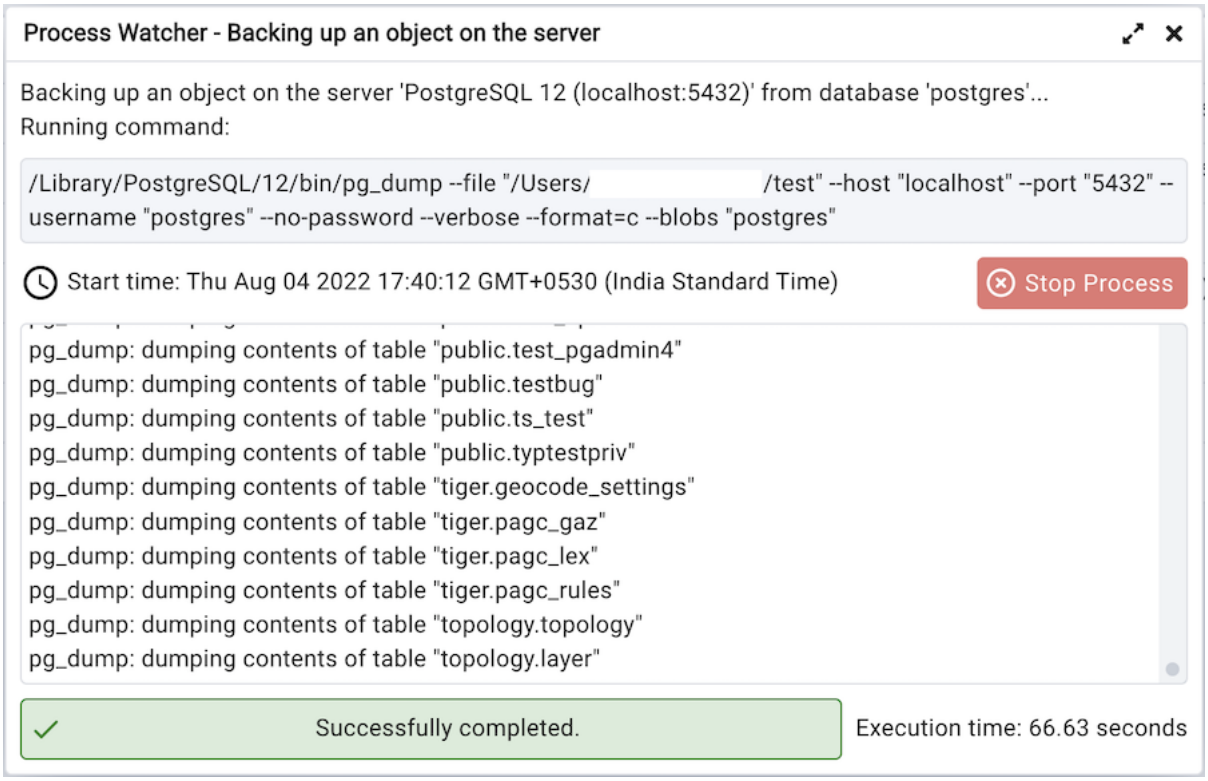
- フォークされた OS プロセスの *PID* 。
- 実行されているタスクの **タイプ** 。
- タスクの **サーバー** 名。
- **オブジェクト** は、データベース、テーブル、mview など、より多くの情報を提供するものです。
- プロセスの **開始時間** 。デフォルトで降順。
- プロセスの現在の **ステータス** 。実行中、完了、失敗、終了のいずれかです。
- 完了までの **所要時間** 。実行中は更新され続けます。

各行には 2 つのアクションボタンがあります。

- *StopProcess* ボタンを使用すると、実行中のプロセスを強制終了できます。
- **詳細** ボタンを使用すると、プロセスログやその他の詳細を表示するプロセスウォッチャーを開くことができます。

チェックボックスを選択し、上部の **削除して確認** ボタンをクリックして、プロセス情報とログをクリアすることもできます。

10.1 プロセスウォッチャー



ProcessWatcher は、プロセス/タスクに関連するすべてのアクティビティをログに記録し、トラブルシューティングのための追加情報を提供します **Stop Process** ボタンを使用して、バックアッププロセスを停止します。

注釈: サーバーモード で *pgAdmin* を実行している場合、|sm_icon|ProcessWatcher ウィンドウのアイコンをクリックして、StorageManager でファイルの場所を開きます。 *Storage Manager* を使用して、クライアントマシンにバックアップファイルをダウンロードできます。

10.2 デモを見る

第11章　pgAgent

pgAgent は、Postgres データベース用のジョブスケジューリングエージェントで、複雑なスケジュールでマルチステップのバッチまたはシェルスクリプトと SQL タスクを実行できます。

pgAgent は pgAdmin とは無関係に配布されます。pgAdminWeb サイトの download area から pgAgent をダウンロードできます。

11.1 Using pgAgent

pgAgent は、ジョブを実行および管理するスケジューリングエージェントです。各ジョブは、1 つ以上のステップとスケジュールで構成されています。2 つ以上のジョブが同時に実行されるようにスケジュールされている場合、pgAgent はジョブを並列で実行します（それぞれが独自のスレッドで）。

ステップは、一連の SQL ステートメントまたはオペレーティングシステムのバッチ/シェルスクリプトです。特定のジョブの各ステップは、前のステップが完了すると、名前の英数字順に実行されます。 *pgAgentJob* ダイアログ（ *Properties* コンテキストメニューからアクセス）のスイッチを使用して、ジョブを変更し、必要に応じて個々のステップを有効または無効にすることができます。

各ジョブは、1 つ以上のスケジュールに従って実行されます。ジョブまたはそのスケジュールが変更されるたびに、ジョブの次のランタイムが再計算されます。pgAgent の各インスタンスは、過去の次の実行時値を持つジョブについてデータベースを定期的にポーリングします。少なくとも 1 分ごとにポーリングすることにより、すべてのジョブは通常、指定された開始時刻の 1 分以内に開始されます。ジョブの次の実行時に実行されている pgAgent インスタンスがない場合、次に pgAgent が起動されるとすぐに実行され、その後に通常のスケジュールに戻ります。

pgAdmin ツリーコントロールで定義されたジョブの名前を強調表示すると、メインの pgAdmin ウィンドウの **プロパティ** タブにはジョブに関する詳細が表示され、 **統計** タブにはジョブの実行に関する詳細が表示されます。

11.1.1 セキュリティ上の懸念

pgAgent は非常に強力なツールですが、注意が必要なセキュリティ上の考慮事項がいくつかあります。

Database password -Windows 接続文字列にパスワードを含めないでください。代わりに、libpq *~/.pgpass* ファイルを使用して、pgAgent がアクセスする必要があるすべてのデータベースのパスワードを保存します。この手法の詳細は、 **Database password** にあります。

System/database access -pgAgent によって実行されるすべてのジョブは、pgAgent ユーザーのセキュリティ権限で実行されます。SQL ステップは、pgAgent がデータベースに接続するユーザーとして実行され、バッチ/シェルスクリプトは、pgAgent サービスまたはデーモンが実行されているオペレーティングシステムユー

ザーとして実行されます。このため、ジョブを作成および変更できるユーザーの制御を維持することが不可欠です。デフォルトでは、pgAgent データベースオブジェクトを作成したユーザーのみがこれを実行できます。これは通常、PostgreSQL スーパーユーザーです。

11.2 Installing pgAgent

pgAgent は、Unix システムではデーモンとして、Windows システムではサービスとして実行されます。ほとんどの場合、データベースサーバ自分自身で実行されます-このため、pgAdmin のインストール時に pgAgent は自動的に構成されません。ただし、場合によっては、同じデータベースに対して複数のシステムで pgAgent を実行する方が好ましい場合があります。個々のジョブは特定のホストをターゲットにすることも、任意のホストに実行させておくこともできます。ロックにより、複数のホストによるジョブの同じインスタンスの実行が防止されます。

11.2.1 データベース設定

pgAdmin を使用して pgAgent を管理する前に、pgAdmin に登録したメンテナンスデータベースに pgAgent 拡張機能を作成する必要があります。PostgreSQL ホストに pgAgent をインストールするには、*postgres* データベースに接続し、 *Tools* メニューからクエリツールを開きます。サーバーバージョン 9.1 以降、および pgAgent3.4.0 以降の場合、クエリウィンドウに次のコマンドを入力し、 **実行** アイコンをクリックします。

```
CREATE EXTENSION pgagent;
```

このコマンドは、「pgagent」と呼ばれるスキーマに多数のテーブルとその他のオブジェクトを作成します。

データベースには、pl/pgsql 手続き言語もインストールする必要があります。必要に応じて、PostgreSQL の CREATELANGUAGE コマンドを使用して pl/pgsql をインストールします。pl/pgsql をインストールするには、クエリウィンドウに次のコマンドを入力し、 **実行** アイコンをクリックします。

```
CREATE LANGUAGE plpgsql;
```

11.2.2 Unix でのデーモンのインストール

注釈: pgAgent は、Linux ユーザー向けの Debian/Ubuntu（DEB）および Redhat/Fedora（RPM）パッケージ、およびソースコードで入手できます。 pgAdmin Website を参照してください。詳細については。

Unix システムに pgAgent デーモンをインストールするには、通常、システム起動スクリプトを変更するための root 権限が必要です。システム起動スクリプトの変更は非常にシステム固有であるため、詳細についてはシステムのドキュメントを参照してください。

プログラム自分自身はいくつかのコマンドラインオプションを取り、そのほとんどはデバッグまたは特殊な構成にのみ必要です:

```
Usage:
  /path/to/pgagent [options] <connect-string>

options:
  -f run in the foreground (do not detach from the terminal)
  -t <poll time interval in seconds (default 10)>
  -r <retry period after connection abort in seconds (>=10, default 30)>
  -s <log file (messages are logged to STDOUT if not specified)>
  -l <logging verbosity (ERROR=0, WARNING=1, DEBUG=2, default 0)>
```

接続文字列は、標準の PostgreSQLlibpq 接続文字列です（詳細については、PostgreSQL documentation on the connection string を参照してください）。たとえば、次のコマンドラインは、ローカルホストでリッスンしているサーバーに対して pgAgent を実行し、「pgadmin」というデータベースを使用し、ユーザー「postgres」として接続します。

```
/path/to/pgagent hostaddr=127.0.0.1 dbname=postgres user=postgres
```

11.2.3 Windows でのサービスのインストール

注釈: EnterpriseDB's PostgreSQL Installers を使用する場合、pgAgent はビルド済みインストーラーで利用できます。StackBuilder アプリケーションを使用して、ダウンロードしてインストールします。この方法でインストールすると、サービスが自動的に作成されるため、以下の指示は無視できます。

---

pgAgent は、Windows システムに自分自身をサービスとしてインストールできます。使用可能なコマンドラインオプションは Unix システムのものと似ていますが、サービスに何をするかを指示する追加のパラメーターが含まれています:

```
Usage:
  pgAgent REMOVE <serviceName>
  pgAgent INSTALL <serviceName> [options] <connect-string>
  pgAgent DEBUG [options] <connect-string>

  options:
    -u <user or DOMAIN\user>
    -p <password>
    -d <displayname>
    -t <poll time interval in seconds (default 10)>
    -r <retry period after connection abort in seconds (>=10, default 30)>
    -l <logging verbosity (ERROR=0, WARNING=1, DEBUG=2, default 0)>
```

サービスは、次のようにコマンドラインから非常に簡単にインストールできます（必要に応じてパスを調整します）。

```
"C:\Program Files\pgAgent\bin\pgAgent" INSTALL pgAgent -u postgres -p secret␣
↪hostaddr=127.0.0.1 dbname=postgres user=postgres
```

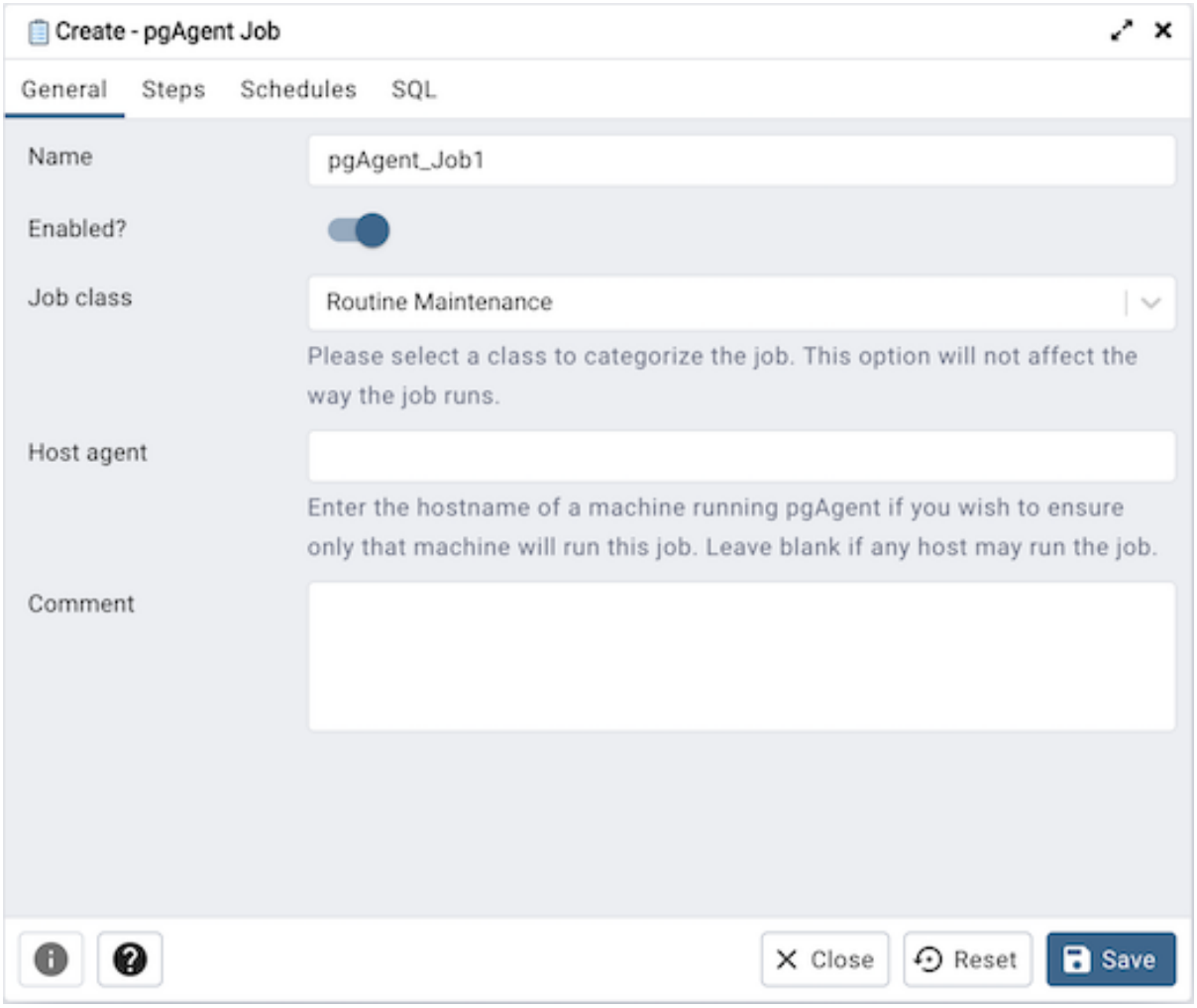
次に、 *netstartpgAgent* を使用してコマンドラインで、または **サービス** コントロールパネルアプレットからサービスを開始できます。ログ出力またはエラーは、アプリケーションイベントログに報告されます。DEBUG モードを使用して、コマンドプロンプトから pgAgent を実行できます。この方法で実行すると、ログメッセージがコマンドウィンドウに出力されます。

11.3 Creating a pgAgent Job

pgAgent は、ジョブを実行および管理するスケジューリングエージェントです。各ジョブはステップとスケジュールで構成されています。

ジョブを作成または管理するには、pgAdmin ツリーコントロールを使用して、pgAgent データベースオブジェクトが作成されたサーバーを参照します。ツリーコントロールに *pgAgentJobs* ノードが表示され、その下に現在定義されているジョブが表示されます。新しいジョブを追加するには、 *pgAgentJobs* ノードを右クリックし、コンテキストメニューから *CreatepgAgentJob...* を選択します。

pgAgent ダイアログが開いたら、 *pgAgent* **ジョブ** ダイアログのタブを使用して、pgAgent ジョブを構成するステップとスケジュールを定義します。



[*General*] タブのフィールドを使用して、ジョブに関する一般的な情報を提供します。

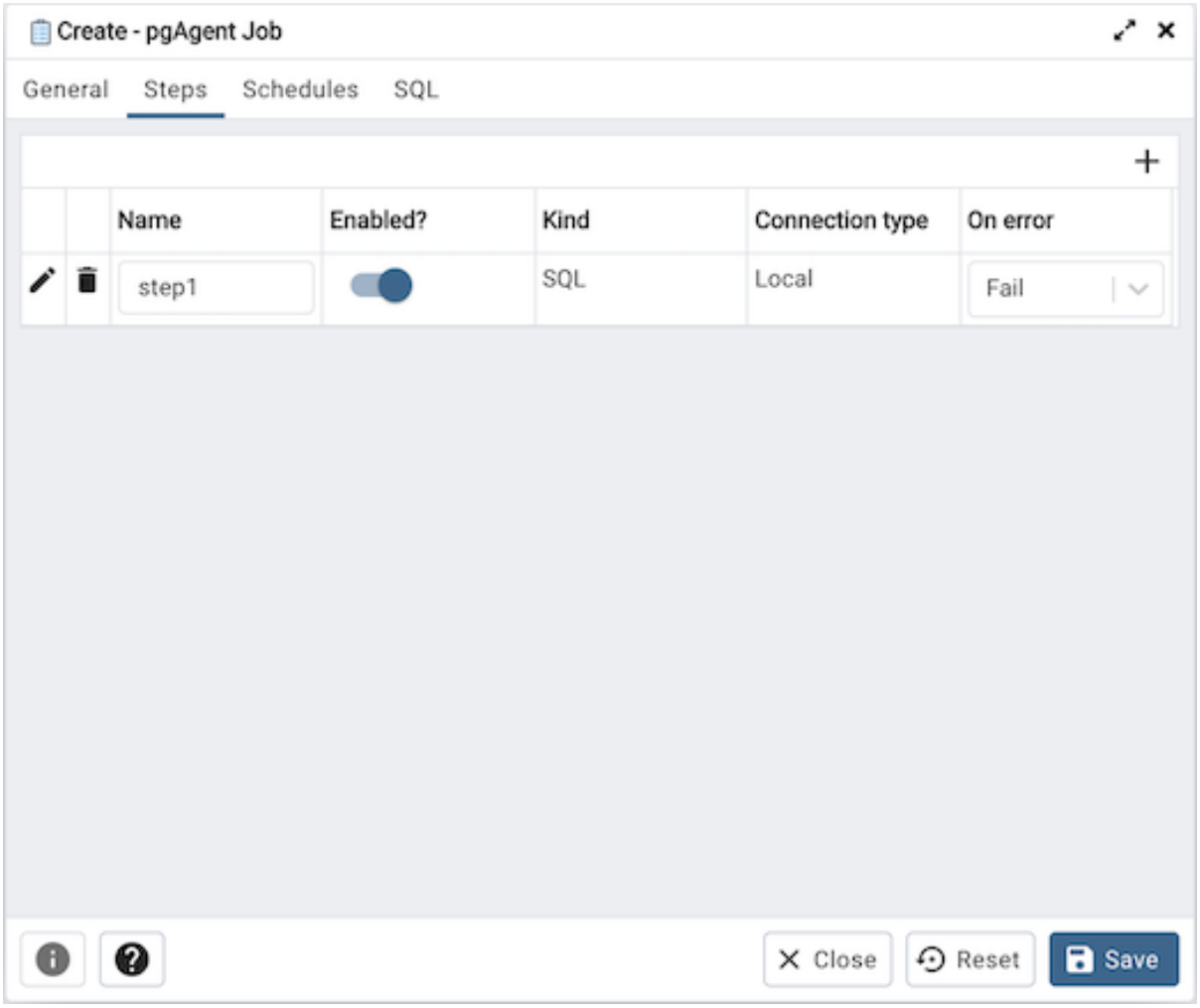
- *Name* フィールドにジョブの名前を入力します。
- *Enabled* スイッチを *Yes* の位置に移動してジョブを有効にするか、 *No* を移動します。
- **ジョブクラス** ドロップダウンを使用して、クラスを選択します（ジョブ分類用）。
- **ホストエージェント** フィールドを使用して、pgAgent を実行しているマシンの名前を指定して、そのマシンのみがジョブを実行できることを示します。すべてのマシンがジョブを実行できるように指定するには、フィールドを空白のままにします。

注釈: ジョブステップを特定のマシンにターゲット設定するためにホストエージェントに指定する値は、必ずしも明らかではありません。必要なマシンで pgAgent を実行し、スケジューラデータベースに接続したら、次のクエリを使用して、各エージェントによって報告されたホスト名を表示できます。

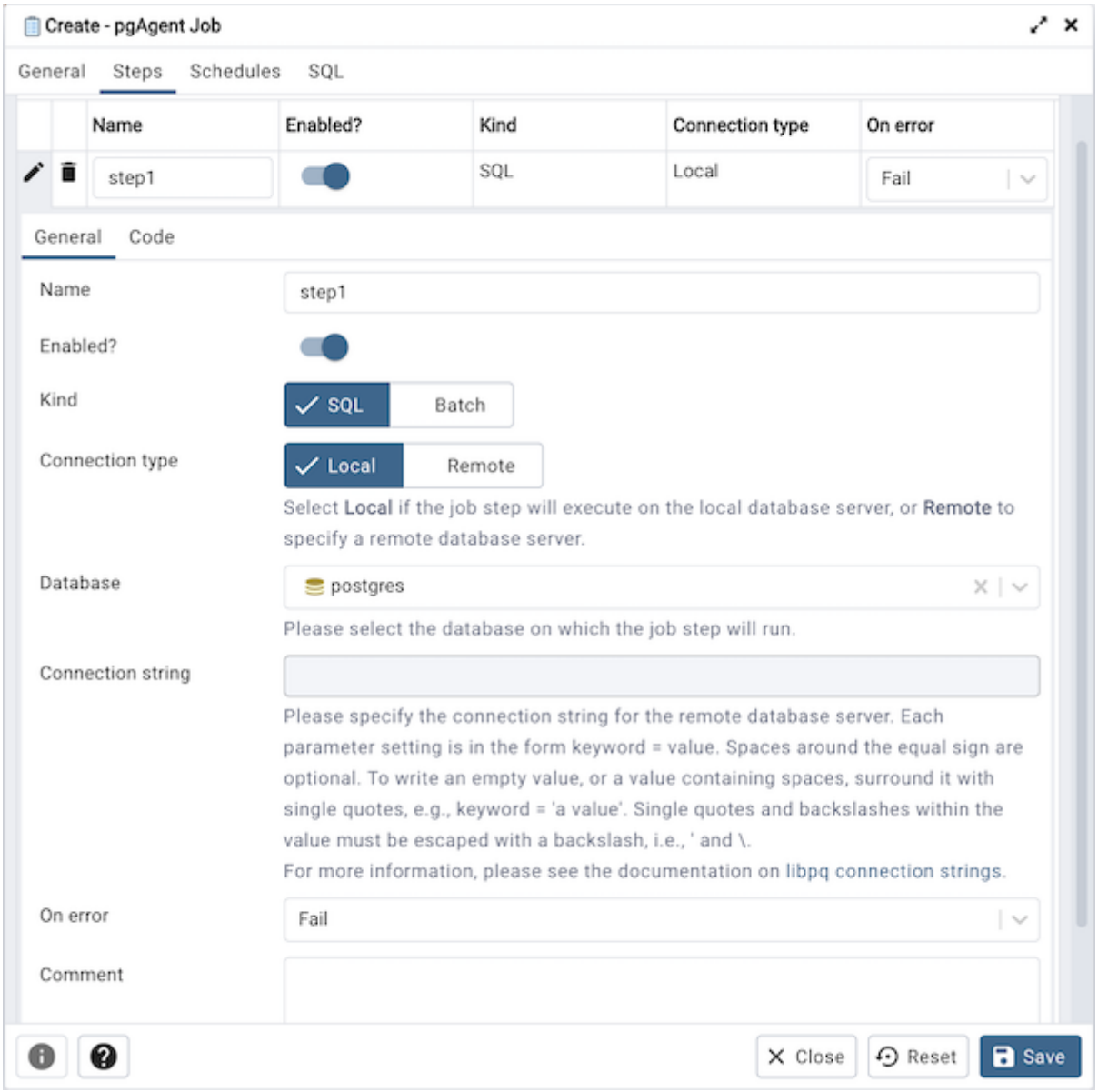
```
SELECT jagstation FROM pgagent.pga_jobagent
```

[HostAgent] フィールドのクエリで報告されたとおりにホスト名を使用します。

- **コメント** フィールドを使用して、ジョブに関するメモを保存します。



[**ステップ**] タブを使用して、ジョブが実行するステップを定義および管理します。追加アイコン（+）をクリックして、新しいステップを追加します。次に、（ヘッダーの左側にある）作成アイコンをクリックしてステップ定義ダイアログを開きます。

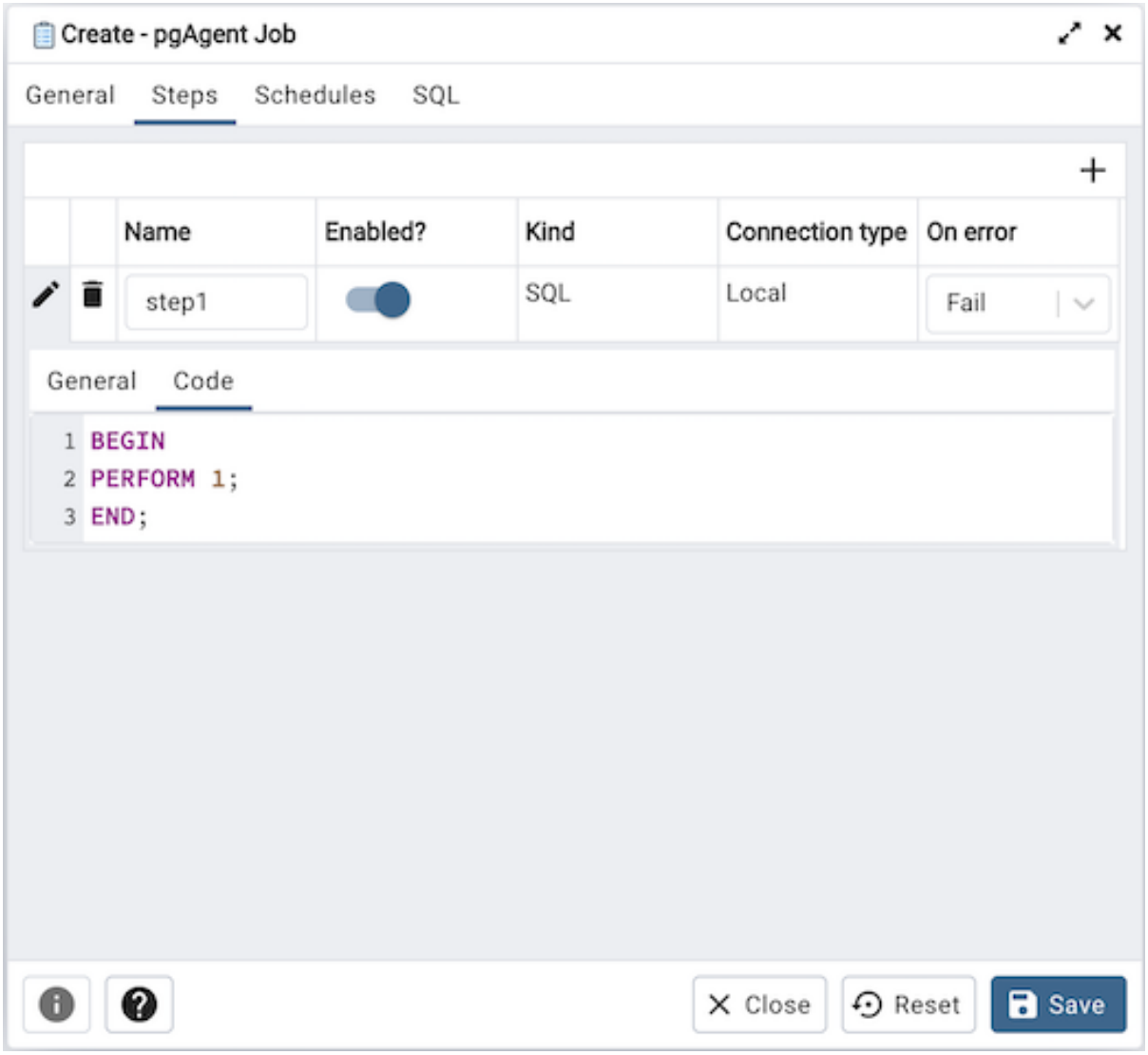


ステップ定義ダイアログのフィールドを使用して、ステップを定義します。

- *Name* フィールドにステップの名前を入力します。ステップは名前の英数字順に実行されることに注意してください。
- *Enabled* スイッチを使用して、ジョブの実行時にステップを含める（ *True* ）か、ステップを無効にします（ *False* ）。
- *Kind* スイッチを使用して、ジョブステップが SQL コード（ *SQL* ）を呼び出すか、バッチスクリプト（ *Batch* ）を呼び出すかを示します。
  - *SQL* を選択した場合、[**コード**] タブを使用してステップの SQL コードを指定します。
  - [**バッチ**] を選択した場合、[**コード**] タブを使用して、ステップ中に実行されるバッチスクリプトを指定します。

注釈: フィールド **接続タイプ** 、 **データベース** 、および **接続文字列** は、 **バッチ** はリモートサーバーで実行できないため、 *SQL* が選択された場合にのみ適用されます。

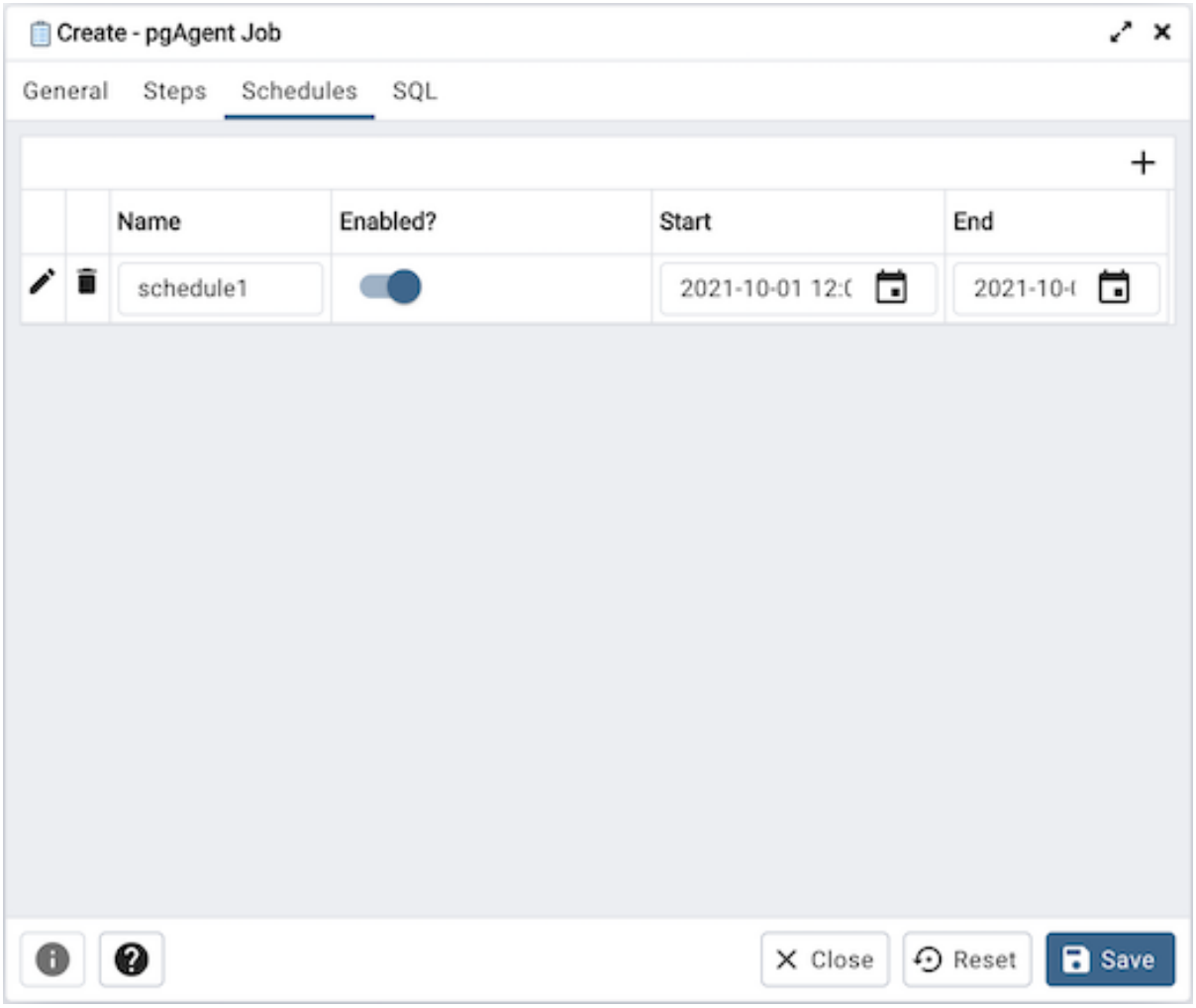
- **接続タイプ** スイッチを使用して、手順をローカルサーバー（ **ローカル** ）で実行するかリモートホスト（ **リモート** ）で実行するかを示します。ステップでリモート接続を使用するように指定した場合、 **接続文字列** フィールドが有効になり、libpq スタイルの接続文字列を指定する必要があります。
- **データベース** ドロップダウンを使用して、ジョブステップを実行するデータベースを選択します。
- **接続文字列** フィールドを使用して、ステップが実行されるリモートサーバーへの libpq スタイルの接続文字列を指定します。接続文字列の作成の詳細については、 PostgreSQL documentation を参照してください。
- **エラー時** ドロップダウンを使用して、ステップの実行中にエラーが発生した場合の pgAgent の動作を指定します。次から選択します。
  - *Fail* -このステップの処理中にエラーが発生した場合、ジョブを停止します。
  - *Success* -ステップを正常に完了したものとしてマークし、続行します。
  - *Ignore* -エラーを無視して続行します。
- *Comment* フィールドを使用して、ステップに関するコメントを入力します。



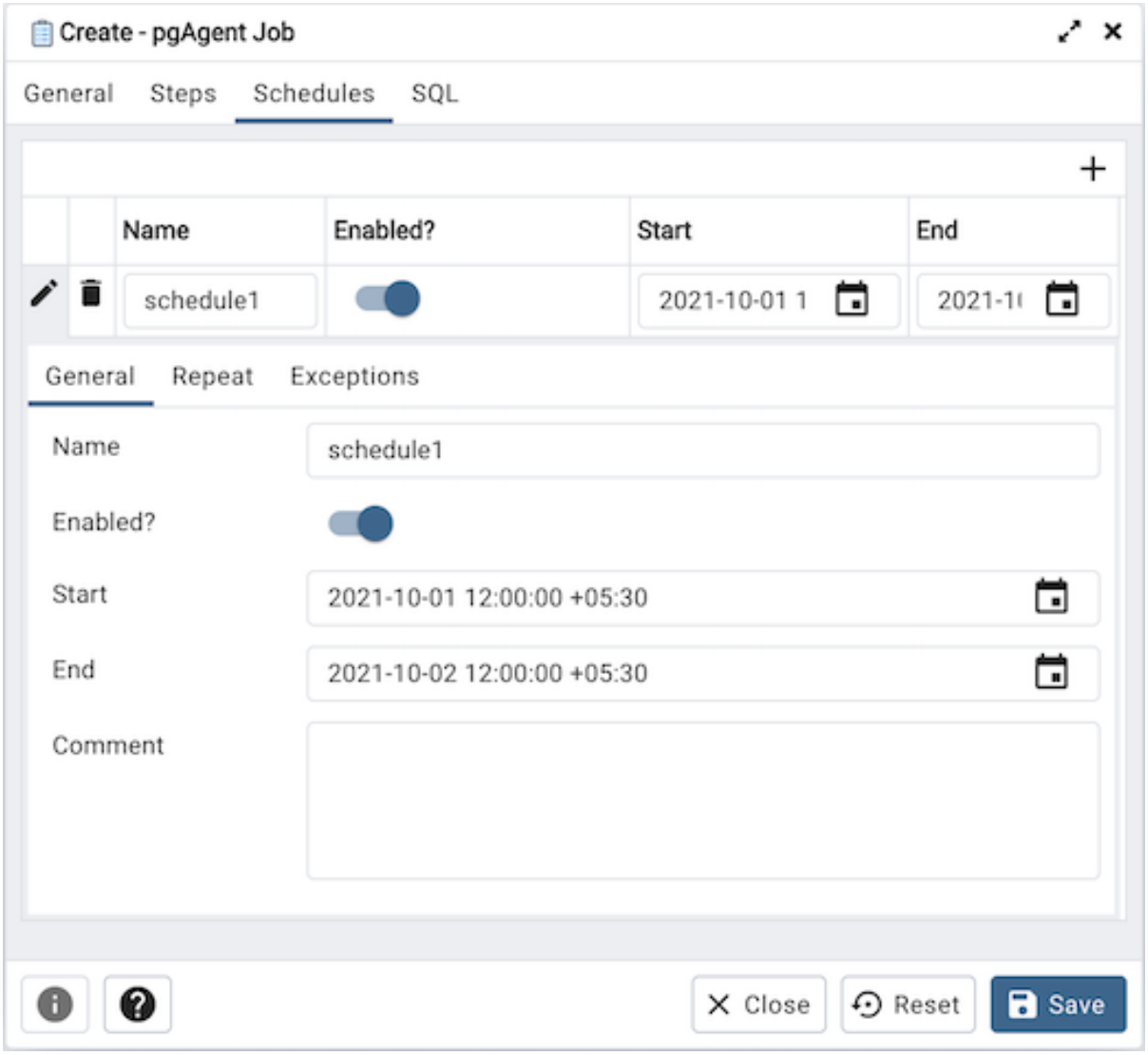
ステップ定義ダイアログの [**コード**] タブの状況依存フィールドを使用して、ステップ中に実行される SQL コードまたはバッチスクリプトを提供します。

- ステップで SQL コードを呼び出す場合は、 *SQLquery* フィールドに 1 つ以上の SQL ステートメントを入力します。
- ステップでバッチスクリプトを実行する場合は、スクリプトを **スクリプト** フィールドに入力します。Windows サーバーで実行している場合、標準のバッチファイル構文を使用する必要があります。Linux サーバーで実行する場合、最初の行で適切なインタープリターが指定されている限り、任意のシェルスクリプトを使用できます（例： *#!/bin/sh* ）。

ステップに必要な情報をすべて入力したら、作成アイコンをクリックしてステップ定義ダイアログを閉じます。追加アイコン（+）をクリックして各ステップを追加するか、[**スケジュール**] タブを選択してジョブスケジュールを定義します。



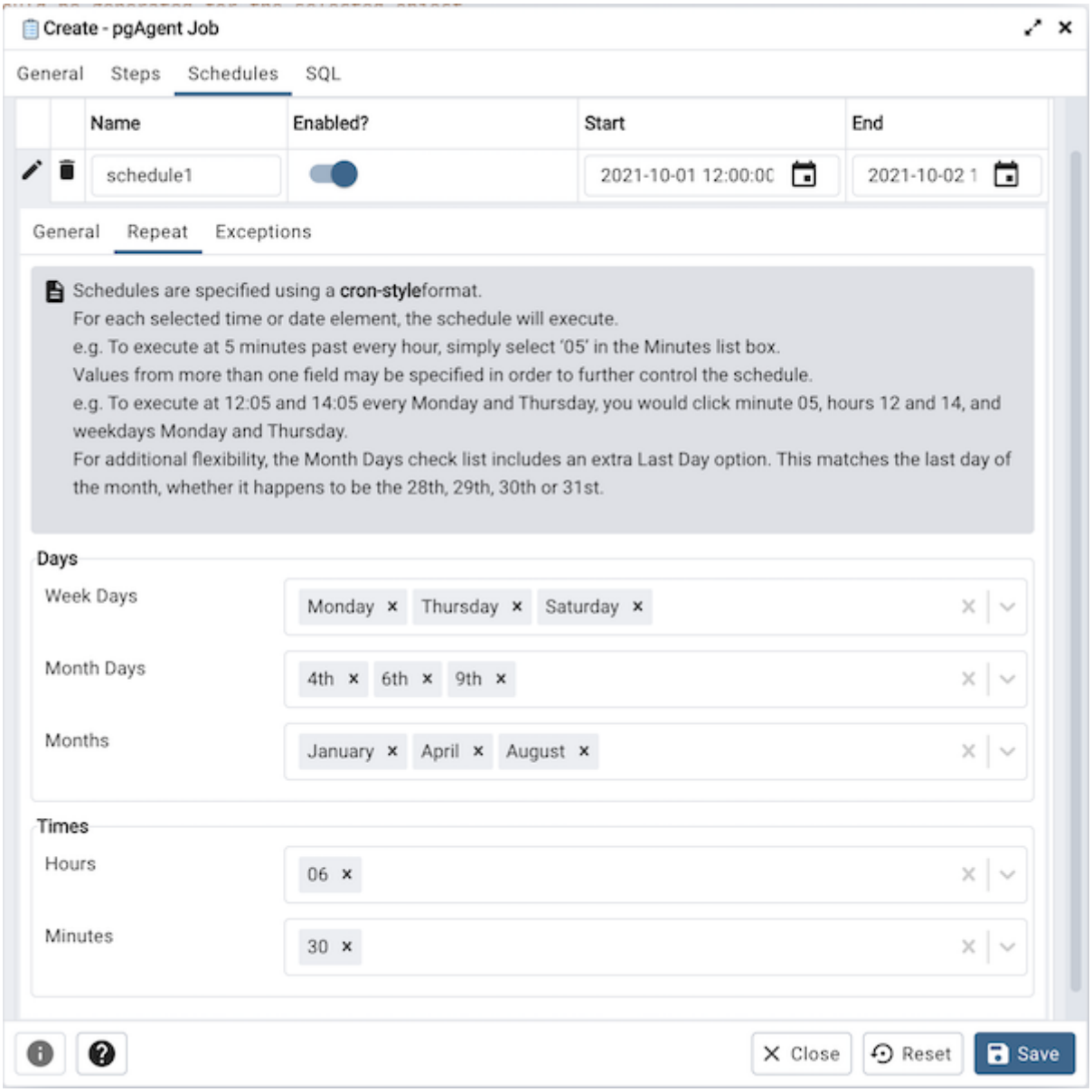
追加アイコン（+）をクリックして、ジョブのスケジュールを追加します。次に、（ヘッダーの左側にある）作成アイコンをクリックして、スケジュール定義ダイアログを開きます。



[スケジュール定義] タブのフィールドを使用して、ジョブを実行する日時を指定します。

- **名前** フィールドにスケジュールの名前を入力します。
- *Enabled* スイッチを使用して、pgAgent がスケジュールを使用するか（ *Yes* ）、スケジュールを無効にします（ *No* ）。
- *Start* フィールドのカレンダーセレクターを使用して、スケジュールの開始日時を指定します。
- *End* フィールドのカレンダーセレクターを使用して、スケジュールの終了日時を指定します。
- **コメント** フィールドを使用して、スケジュールに関するコメントを入力します。

[**繰り返し**] タブを選択して、スケジュールを実行する日を定義します。



[**繰り返し**] タブのフィールドを使用して、スケジュールに関する詳細を cron 形式で指定します。ジョブは、[**繰り返し**] タブで選択された各日付または時刻要素に実行されます。

フィールド内をクリックして、そのフィールドの有効な値のリストを開きます。特定の値をクリックして、その値をフィールドで選択した値のリストに追加します。フィールドから値をクリアするには、フィールドの右側にある X をクリックします。

Days ボックス内のフィールドを使用して、ジョブを実行する日を指定します。

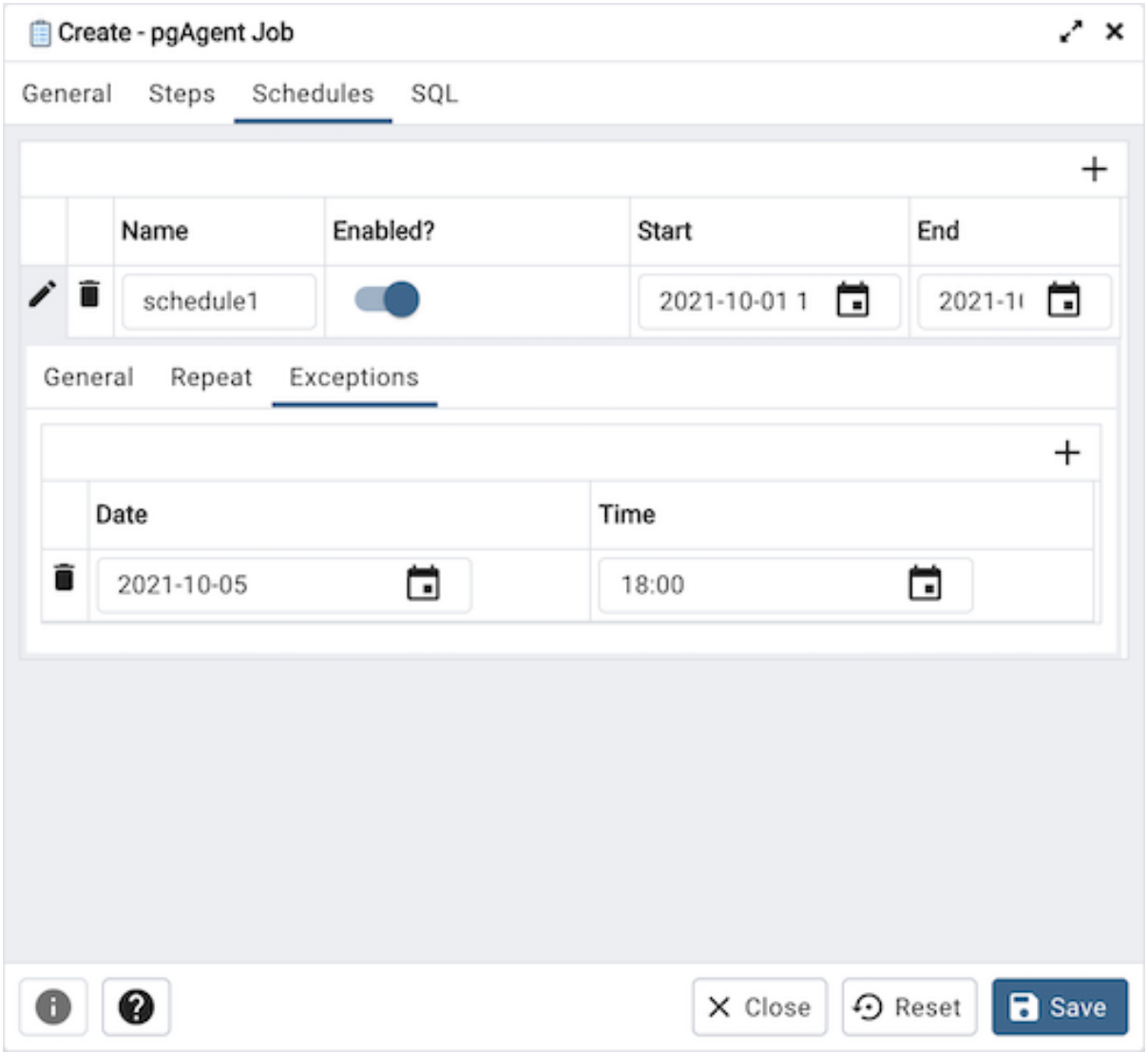
- *WeekDays* フィールドを使用して、ジョブを実行する日を選択します。
- *MonthDays* フィールドを使用して、ジョブを実行する日数を選択します。 **最終日** を指定して、日付に関係なく、その月の最終日にジョブを実行することを示します。
- *Months* フィールドを使用して、ジョブを実行する月を選択します。

Times ボックス内のフィールドを使用して、ジョブを実行する時刻を指定します。

- *Hours* フィールドを使用して、ジョブを実行する時間を選択します。

- *Minutes* フィールドを使用して、ジョブを実行する分を選択します。

[**例外**] タブを選択して、スケジュールを実行 **しない** 日を指定します。

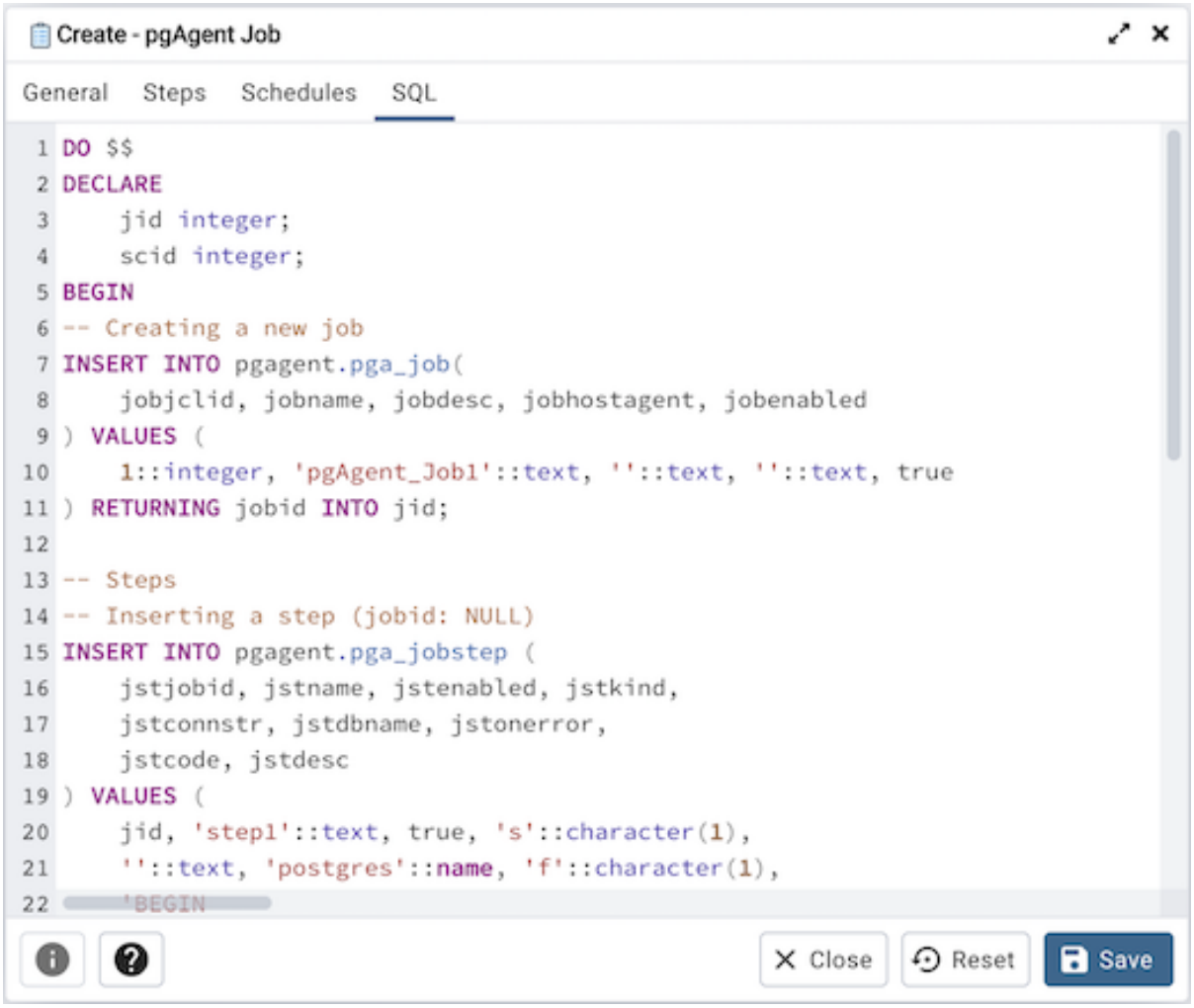


[*Exceptions*] タブのフィールドを使用して、ジョブを実行しない日を指定します。たとえば、祝日にジョブを実行しないようにすることができます。

追加アイコン（+）をクリックして例外表に行を追加し、次のことを行います。

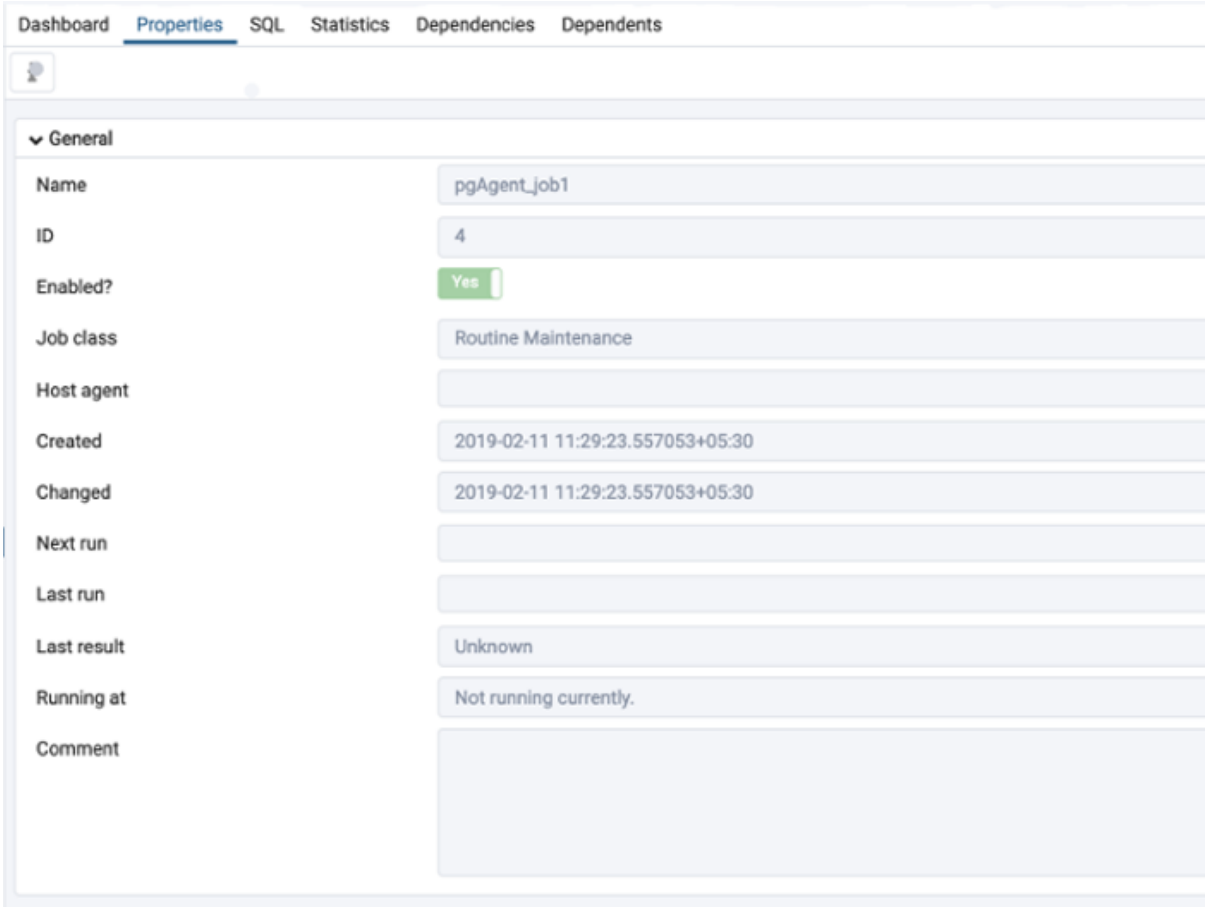
- **日付** 列内をクリックしてカレンダーセレクターを開き、ジョブを実行しない日付を選択します。 *Date* 列に *<Any>* を指定して、選択した時刻にジョブを実行しないことを示します。
- *Time* 列内をクリックして時間セレクターを開き、ジョブを実行しない時間を指定します。 *Time* 列に *<Any>* を指定して、選択した日のどの時間にもジョブを実行しないことを示します。

スケジュールの定義が終了したら、 *SQL* タブを使用して、ジョブを作成または変更するコードを確認できます。



Save ボタンをクリックしてジョブ定義を保存するか、 *Close* をクリックして保存せずにジョブを終了します。 **リセット** ボタンを使用して、ダイアログから未保存のエントリを削除します。

ジョブを保存すると、ジョブは定義されたサーバーの pgAdmin ツリーコントロールの *pgAgentJobs* ノードの下にリストされます。pgAdmin のメインウィンドウの **プロパティ** タブには、選択したジョブの高レベルの概要が表示され、 **統計** タブにはジョブの各実行の詳細が表示されます。



既存のジョブを変更したり、ジョブに関する詳細情報を確認したりするには、ジョブ名を右クリックし、コンテキストメニューから **プロパティ** を選択します。

第12章　pgAdmin Project Contributions

pgAdmin は、開発プロセスへの参加をお勧めするオープンソースプロジェクトです。pgAdmin プロジェクトへの貢献の詳細については、pgAdmin メーリングリスト pgadmin-hackers@postgresql.org の開発者に連絡して、機能強化やバグ修正に関するアイデアについて話し合ってください。

以下のセクションでは、pgAdmin クライアントの開発、改善、およびメンテナンスに使用される開発プロセスに関する詳細情報を見つけます。

12.1 Submitting Pull Requests

pgAdmin の機能またはバグ修正を開発する前に、常にメーリングリスト pgadmin-hackers@postgresql.org の開発者に連絡して、計画について話し合う必要があります。これにより、他の人があなたが何をしているのかを知り、作業の重複を避けることができ、大規模な変更の場合、後で却下される可能性のあるコードを書くことに時間を費やす前に、あなたのアイデアを議論して洗練する機会がコミュニティに与えられます。

リリースの tarball ではなく、GIT ソースコードリポジトリからのソースコードのチェックアウトに対して常に変更を開発する必要があります。これにより、ブランチで最新のコードで作業していることが保証され、プルリクエストを正しく生成しやすくなります。

2022 年 9 月の時点で、pgAdmin ソースリポジトリは https://github.com/pgadmin-org/pgadmin4 にあります。比較的単純な変更の一般的なワークフローは次のとおりです。

1. GitHub の pgAdmin4 プロジェクトにアクセスし、 *Fork* ボタンをクリックして、GitHub アカウントに独自の開発リポジトリを作成します。
2. 次のようなコマンドを使用して、ソースコードのローカルコピーをチェックアウトします。

```
$ git clone https://github.com/<your GitHub username>/pgadmin4.git
```

3. ローカル開発環境で変更を開発およびテストします。
4. 変更を確認し、完全にチェックして pgAdmin *Coding Standards* を満たしていることを確認し、*Coding Standards* に対して確認して、リジェクトされる可能性を最小限に抑えます。
5. 変更に満足したら、適切なメッセージでコミットします。先頭に 1 行の概要を含め、必要に応じて、概要の後の空白行に続けて段落を挿入します。
6. 変更をリポジトリのフォークにプッシュします。
7. GitHub に戻り、*pgadmin-org/pgadmin4 master* ブランチに対して新しいプルリクエストを作成します。

> **注釈:** 各プルリクエストには、単一のバグ修正または機能を単一のコミットとして含める必要があります。必要に応じて、送信する前に、大規模な変更または機能の複数のコミットを単一のコミットに押し込む必要があります。

8. あなたのプルリクエストは、開発チームの 1 人以上のメンバーによってレビューされ、受け入れられるか、要求された変更とともに返送されます。場合によっては、変更が適切と見なされないと拒否される場合があります。これが、重要な時間を費やす前にチームと作業について話し合うことが重要である理由です。

より複雑な変更については、pgAdmin リポジトリのフォークで **フィーチャーブランチ** を使用し、それを使用してプルリクエストを作成することもできます。

> **注釈:** これは、ワークフローの簡単な例です。代わりに GitHub CLI などの他のツールを使用することもできます。ただし、そのようなツールとワークフローを文書化することは pgAdmin ドキュメントの範囲外です。

12.2 Code Overview

pgAdmin の大部分は、バックエンドで Flask フレームワークを使用して作成され、フロントエンドで CSS3、Bootstrap、jQuery を使用して作成された PythonWeb アプリケーションです。NWjs（NodeWebkit）を使用して作成された Web アプリケーションよりもデスクトップアプリケーションを好むユーザー向けに、デスクトップランタイムも含まれています。

12.2.1 ランタイム

ランタイムは、ブラウザーとスタンドアロンアプリケーションを作成する Python サーバーを統合する NWjs に基づいています。ソースコードは、ソースツリーの **/runtime** ディレクトリにあります。

12.2.2 ウェブアプリケーション

Web アプリケーションは pgAdmin の大部分を形成し、ソースツリーの **/web** ディレクトリにあります。メインファイルは **pgAdmin4.py** で、ビルトインのスタンドアロン Web サーバーの実行、または実稼働用の WSGI アプリケーションとして使用できます。

構成

コアアプリケーション構成は **config.py** にあります。このファイルには、アプリケーションの構成可能な設定とその使用の説明が含まれています。Web サーバーに展開する前に、さまざまな設定を構成することが不可欠です。これらは **config.py** または **config.py** （ **config.py** ドキュメントを参照）でオーバーライドして、メイン構成ファイルの変更を回避できます。

ユーザー設定

デスクトップモードで実行している場合、pgAdmin にはデスクトップユーザーに使用される単一のデフォルトのユーザーアカウントがあります。サーバーモードで実行している場合、アプリケーションを使用する前にログインする必要があるユーザーは無制限です。pgAdmin は **Flask-Security** モジュールを利用してアプリケーションのセキュリティとユーザーを管理し、セルフサービスのパスワードリセットやパスワード変更などのオプションを提供します。

デスクトップモードでもサーバーモードでも、各ユーザーの設定は SQLite またはユーザーアカウントの保存にも使用される外部データベースに保存されます。これは、最初にデータベースファイルとスキーマを作成する **setup.py** スクリプトを使用して作成され、最初のユーザーアカウント（管理者権限を持つ）とデフォルトのサーバーグループを追加します。 **setup.py** テーブルは、ユーザー構成設定をキーと値で保存するためにも使用されます。必須ではありませんが、設定キー（または名前）は通常、Linux または Mac の pgAdmin3 設定ファイルと同様に、スラッシュを使用して値を人工的に名前空間にフォーマットします。

setup.py を実行する前にローカル構成をセットアップする必要があることに注意してください。ローカル構成は、特にデスクトップモードとサーバーモードに関して、スクリプトがデータベースを設定する方法を決定します。

12.2.3 pgAdmin コア

pgAdmin の中心は **pgadmin** パッケージです。これには、Jinja エンジンで使用されるグローバルに利用可能な HTML テンプレートと、マルチプルのモジュールで使用される画像、Javascript、CSS ファイルなどのグローバルな静的ファイルが含まれています。

パッケージの作業は、そのコンストラクター **__init__.py** で処理されます。これは、ロギングと認証の設定、他のモジュールの動的ロード、および他のいくつかのタスクを担当します。

12.2.4 モジュール

モジュールを追加することにより、機能のユニットが pgAdmin に追加されます。これらはクラスの Python オブジェクトインスタンスであり、 **web/pgadmin/utils.py** にある PgAdminModule クラス（Flask ブループリント実装）を継承します。他のモジュール（主にデフォルトモジュールであるブラウザー）が利用するさまざまなフックポイントを提供します。

モジュールとして認識されるには、Python パッケージを作成する必要があります。これは以下を行う必要があります。

1) **web/pgadmin/** ディレクトリ内に配置され、

2) pgadmin.utils.PgAdminModule クラスを実装します

3) そのパッケージ内の特定のクラスを表すインスタンス変数（一般に **blueprint** という名前）。

各モジュールは、実装するブループリントの **template** および **template** ディレクトリを定義できます。名前の衝突を避けるには、テンプレートは指定されたテンプレートディレクトリ内の、モジュール自分自身の名前を付けたディレクトリに保存する必要があります。たとえば、 **template** モジュールはそのテンプレートを **template** に保存します。これは、2 番目のモジュール名を省略できる静的ファイルには適用されません。

ブループリントの定義に加えて、 **views** モジュールは通常、クライアントの要求に応じてレンダリングされるすべてのビューを定義します。これらのビューに RESTAPIURL を提供する必要があります。これらには、適切なルートデコレータとセキュリティデコレータを含める必要があります。Flask の MethodView と同じアプローチを使用する NodeView クラスを見てください。 **views** にあります。この特定のクラスは、ブラウザーノードでさまざまな操作の RESTAPIURL を作成するために使用されます。すなわち、リスト、作成、更新、削除、子の取得、そのノードの統計/逆 SQL/dependencies/dependents リストの取得など。他の目的にも同じクラスを使用できます。そのクラスを継承し、メンバー変数 operations、parent_ids、ids、node_type をオーバーロードして、PgAdminModule インスタンスにノードビューとして登録するだけです。

ほとんどの pgAdmin モジュールは、PgAdminModule クラスが提供する **hooks** も実装します。これは、モジュールをアプリケーションの残りの部分に統合するためのフックポイントを提供します。たとえば、フックは、レンダリングされたページに含める必要がある CSS ファイル、または含めるメニューオプションとその動作を呼び出し元に指示します。フックポイントが必要ない場合は、存在する必要はありません。それらを利用する前に、それらが存在することを確認するのは呼び出し元の責任です。

現在実装されているフックは次のとおりです。

```
class MyModule(PgAdminModule):
    """
    This is class implements the pgadmin.utils.PgAdminModule, and
    implements the hooks
    """


    ...

    def get_own_stylesheets(self):
        """
        Returns:
            list: the stylesheets used by this module, not including any
                  stylesheet needed by the submodules.
        """
        return [url_for('static', 'css/mymodule.css')]

    def get_own_javascripts(self):
        """
        Returns:
            list of dict:
            - contains the name (representation for this javascript
              module), path (url for it without .js suffix), deps (array of
              dependents), exports window object by the javascript module,
```

(次のページに続く)

(前のページからの続き)

```
            and when (would you like to load this javascript), etc
            information for this module, not including any script needed
            by submodules.
        """
        return [
            {
                'name': 'pgadmin.extension.mymodule',
                'path': url_for('static', filename='js/mymodule'),
                'exports': None,
                'when': 'server'
                }
            ]

    def get_own_menuitems(self):
        """
        Returns:
            dict: the menuitems for this module, not including
                  any needed from the submodules.
        """
        return {
            'help_items': [
                MenuItem(
                    name='mnu_mymodule_help',
                    priority=999,
                    # We need to create javascript, which registers itself
                    # as module
                    module="pgAdmin.MyModule",
                    callback='about_show',
                    icon='fa fa-info-circle',
                    label=gettext('About MyModule'
                    )
                ]
            }
    def get_panels(self):
        """
        Returns:
            list: a list of panel objects to add implemented in javascript
                  module
        """
        return []
    ...



blueprint = MyModule('mymodule', __name__, static_url_path='/static')
```

pgAdmin モジュールには、必要に応じて、目的を果たすために必要な追加の Python モジュールを含めることができます。他の動的にロードされるモジュールを参照することもできますが、定義されたフックポイントを使用し、特定のモジュールが存在しない場合は正常に失敗する必要があります。

12.2.5 ノード

ノードはモジュールに非常に似ており、ブラウザのツリービュー上の個々のノードまたはコレクションオブジェクトを表します。ノードモジュールとして認識されるには、Python パッケージ（JavaScript モジュールとともに）を作成する必要があります。これは以下を行う必要があります。

1) **web/pgadmin/browser/** ディレクトリ内に配置され、

2) BrowserPluginModule を実装し、必要な RESTAPI を公開するノードビューを登録します

3) クラスオブジェクトのインスタンス

12.2.6 フロントエンド

pgAdmin は、フロントエンドの実装に JavaScript を広く使用しています。require.js を使用して遅延ロード（または、必要な場合にのみロード）、UI のルックアンドフィールにブートストラップと MaterialUI、選択したノードのプロパティを生成/ダイアログを作成する React を使用します。各モジュールを可能な限り小さな塊に分割しました。すべての JavaScript モジュールをロードする必要があるわけではありません（つまり、データベースの JavaScript モジュールのロードは、サーバーノードが完全にロードされた場合にのみ意味があります）。js などを使用して、コードをよりよく理解してください。

12.3 Coding Standards

pgAdmin は複数のテクノロジーと複数の言語を使用しており、それぞれに独自のコーディング標準があります。

12.3.1 全般

すべての言語で、インデントは 4 つのスペースで作成し、必要に応じて過度に長い行を折り返して、小さなディスプレイで読めるようにします（多くの場所で 80 文字が使用されていますが、これは必要な最大サイズではありません現代のディスプレイ）。通常、行は 120 文字以下にする必要があります。

コード自分自身の簡単なレビューから明らかではない場合、その目的や動作を説明するために、コメントをすべてのコードに含める必要があります。

12.3.2 CSS3

CSS3 は、アプリケーション全体のスタイルとレイアウトに使用されます。そのプロセスを支援するために Bootstrap フレームワークが広範囲に使用されていますが、追加のスタイルを時々作成する必要があります。

ほとんどのカスタムスタイリングは、フックを介してロードするモジュールに含めるように静的スタイルシートをアドバタイズする個々のモジュールから来ます。

スタイルのオーバーライド（たとえば、Bootstrap のルックアンドフィールを変更する）は、通常、アプリケーションのメインの静的ファイルディレクトリにある **overrides.css** ファイルにあります。

スタイル設定は、常に外部スタイルシートを介して HTML にインラインで適用しないでください。スタイルの使用法または目的を説明するコメントを適切に含める必要があります。

スタイルは、1 行に 1 つずつ明確に指定する必要があります。例：

```
/* iFrames should have no border */
iframe {
    border-width: 0;
}

/* Ensure the codemirror editor displays full height gutters when resized */
.CodeMirror, .CodeMirror-gutters {
    height: 100% !important;
}
```

すべてのスタイルシートは CSS3 に準拠している必要があります。

12.3.3 HTML5

HTML5 はアプリケーション全体のページ構造に使用され、ほとんどの場合、Flask の Jinja2 テンプレートエンジンによってテンプレートからレンダリングされます。

すべての HTML は HTML5 に準拠している必要があります。

12.3.4 Javascript

クライアント側のコードは、jQuery とさまざまなプラグインを使用して Javascript で記述されます。コードの多くは静的ファイルからレンダリングされますが、Jinja2 を使用してテンプレートからレンダリングされるコード（多くの場合、ユーザー設定をインジェクトする）またはモジュールフックからその場で構築されるコードもあります。

典型的な Javascript 関数は次のようにフォーマットされます（これはテンプレートからの抜粋です）。

```
// Delete a server group
function delete_server_group(item) {
    alertify.confirm(
```

(次のページに続く)

(前のページからの続き)

```
        'Delete server group?',
        'Are you sure you wish to delete the server group "{0}"?'.replace('{0}', tree.
↪getLabel(item)),
        function() {
            var id = tree.getId(item)
            $.post("{{ url_for('NODE-server-group.delete') }}", { id: id })
                .done(function(data) {
                    if (data.success == 0) {
                        report_error(data.errormsg, data.info);
                    } else {
                        var next = tree.next(item);
                        var prev = tree.prev(item);
                        tree.remove(item);
                        if (next.length) {
                            tree.select(next);
                        } else if (prev.length) {
                            tree.select(prev);
                        }
                    }
                }
            )
        },
        null
    )
}
```

アンダースコア文字を使用してすべての小文字の単語を区切り、説明的な関数名と、短いが説明的な小文字の変数名を使用していることに注意してください。

注釈: バージョン 3.0 以降、新しいコードまたはリファクタリングされたコードは、ES6 の機能と規則を使用して記述する必要があります。

12.3.5 パイソン

バックエンドの Web サーバーには Python を使用しています。すべてのコードは Python2.7 と互換性があり、 PEP 8 で定義された公式の Python コーディング標準に従っている必要があります。必要なファイルヘッダーとともに関数の例を以下に示します。

```
##########################################################################
#
# pgAdmin 4 - PostgreSQL Tools
#
```

(次のページに続く)

(前のページからの続き)

```
# Copyright (C) 2013 - 2023, The pgAdmin Development Team
# This software is released under the PostgreSQL Licence
#
##########################################################################

"""Integration hooks for server groups."""

from flask import render_template, url_for
from flask.ext.security import current_user

from pgadmin.settings.settings_model import db, ServerGroup

def get_nodes():
    """Return a JSON document listing the server groups for the user"""
    groups = ServerGroup.query.filter_by(user_id=current_user.id)

    value = ''
    for group in groups:
        value += '{"id":%d,"label":"%s","icon":"icon-server-group","inode":true},' \
                 % (group.id, group.name)

    value = value[:-1]

    return value
```

12.4 Code Review Notes

このドキュメントには、pgAdmin に含めるために送信された変更のレビュープロセス中にチェックされる多くの標準項目がリストされています。

- すべてのコードが pgAdmin *Coding Standards* に従っていることを確認します。
- すべてのコードに、必要に応じて単体テストのカバレッジと API/機能テストのカバレッジがあることを確認します。
- 著作権の年は、正確で適切な形式である必要があります（毎年の一括更新を簡単にするため）。開始日は常に 2013 年、終了年は現在の年にする必要があります。

  Copyright(C)2013-2023、pgAdmin 開発チーム

- 著作権ヘッダーの直後に空白行があることを確認します。
- 関数、クラス、モジュールの PyDoc コメントを含めます。ノードモジュールは/"/"/"XXXX ノードを実装します/"/"/"
- 生成された SQL には、先頭または末尾に空白行がなく、適切なフォーマットのために 4 つのスペースインデントが一貫して使用されていることを確認します。

- Slony オブジェクトを特別なケースにしないでください。pgAdmin3 とは異なり、pgAdmin4 は Slony を直接認識しません。
- モジュールをコピーして貼り付ける場合は、コメントが適切に更新されていることを確認してください。
- すべてのコメントを読んで、それらが理にかなっていることを確認し、コードに役立つコメントを提供します。
- フィールドラベルは両方とも PostgreSQL の用語を使用するだけでなく、説明的であることを確認します。良い例は、FTS テンプレートの/"Init/"フィールドです。Init は PG 用語ですが、その後に単語/"Function/"を追加すると、はるかに説明的になります。
- 可能な限りコードを再利用しますが、適切に中央の場所に分解します。変更が必要な場合を除き、コピーして貼り付けないでください。
- コードを読みやすいようにフォーマットします。コードの論理的なチャンクを空白行で分割し、適切にコメントして、コードのさまざまなセクションが何を対象としているかを説明します。
- フォームの検証が正しく機能し、エラーの表示方法が他のダイアログと一貫していることを確認します。
- スキーマまたは所有者フィールドを使用したダイアログで、必要に応じてデフォルト値を現在のスキーマ/ユーザーに事前設定します。一般に、一般的なまたは適切なデフォルト値がある場合は、それらをユーザーのフィールドに入力します。
- 1 つのプルリクエスト==1 つの機能。変更で既存のインフラストラクチャを修正/更新する必要がある場合、通常は別のプルリクエストにある方が簡単です。複数の新機能または無関係な変更を含むプルリクエストは拒否される可能性があります。
- 変更が完全に機能し、動作することを確認します。コミットを意図していない進行中の（WIP）変更を送信したい場合は、プルリクエストを送信する代わりに、リポジトリフォークへのリンクまたはパッチとして pgadmin-hackers@postgresql.org メーリングリストに送信します、それが WIP であることを明確に述べ、まだ機能していないことを指摘します。

12.5 Translations

pgAdmin supports multiple languages using the Flask-Babel Python module. A list of supported languages is included in the **web/config.py** configuration file and must be updated whenever languages are added or removed with ISO 639-1 (two letter) language codes. The codes are named **$LANG** in this document.

12.5.1 翻訳マーキング

文字列は、Python コード（**gettext()** を使用）または Jinja テンプレート（**gettext()** を使用）のいずれかで翻訳用にマークできます。これがどのように達成されるかを示す例をいくつか示します。

Python：

```
errormsg = gettext('No server group name was specified')
```

ジンジャ：

```
<input type="submit" value="{{ _('Change Password') }}">
```

```
<title>{{ _('%(appname)s Password Change', appname=config.APP_NAME) }}</title>
```

```
define(['sources/gettext', ...], function(gettext, ...){
    ...
    var alert = alertify.prompt(
        gettext('Password Change'),
        gettext('New password for %(userName)s', {userName: 'jsmith' }),
        ...
    )
})
```

12.5.2 更新とマージ

アプリケーションに新しい文字列が追加されるたびに、テンプレートカタログ（**web/pgadmin/messages.pot**）と既存の翻訳カタログ（**web/pgadmin/messages.pot**）を更新してコンパイルする必要があります。これは、pgAdmin の Python 仮想環境の **web/pgadmin/messages.pot** ディレクトリから次のコマンドを使用して実現できます。

```
(pgadmin4) user$ pybabel extract -F babel.cfg -o pgadmin/messages.pot pgadmin
```

テンプレートを更新したら、既存のメッセージカタログにマージする必要があります。

```
(pgadmin4) user$ pybabel update -i pgadmin/messages.pot -d pgadmin/translations
```

最後に、メッセージカタログをコンパイルして使用できます。

```
(pgadmin4) user$ pybabel compile -d pgadmin/translations
```

12.5.3 新しい言語の追加

新しい言語を追加するのは簡単です。まず、言語名と識別子を **web/config.py** に追加します:

```
# Languages we support in the UI
LANGUAGES = {
    'en': 'English',
    'zh': 'Chinese (Simplified)',
    'de': 'German',
    'pl': 'Polish'
}
```

次に、pgAdmin の Python 仮想環境のソースツリーにある **web** ディレクトリから新しいメッセージカタログを作成します。

```
(pgadmin4) user$ pybabel init -i pgadmin/messages.pot -d pgadmin/translations -l $LANG
```

第13章　リリースノート

pgAdmin リリースノートには、各リリースの機能と改善点に関する情報が記載されています。このページには、メジャーリリースとマイナー（バグ修正）リリースのリリースノートが含まれています。以下のリストからバージョンを選択して、リリースノートを表示します。

13.1 バージョン6.19

発売日：2023-01-12

このリリースには、pgAdmin4v6.18 のリリース以降の多数のバグ修正と新機能が含まれています。

13.1.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13、14 および 15

EDB Advanced Server ：10、11、12、13、14 および 15

13.1.2 新機能

Issue #5569 -BigAnimal クラウド展開の AWS プロバイダーのサポートを追加しました。

13.1.3 ハウスキーピング

Issue #5563 -YouTube ビデオデモリンクを適切な pgAdmin ドキュメントに追加できるようにします.

Issue #5615 -React を使用するように pgAdmin メインメニューバーを書き換えます.

13.1.4 バグ修正

Issue #5487 -共有サーバーで誤ったパスワードが使用される問題を修正しました。

Issue #5541 -10,000 以上のノード/オブジェクトのレンダリング中にブラウザツリーがフリーズしないことを確認します。

Issue #5542 -コレクションノードにスキーマが 1 つだけ存在する場合にスキーマノードを更新する問題を修正しました。

Issue #5559 -スクロール時のツリーのちらつきの問題を修正しました。

Issue #5577 -列の文字列のデフォルト値が作成スクリプトで引用符で囲まれる必要がある問題を修正しました。

Issue #5586 -Web サーバーと内部認証のセットアップの問題を修正します。

Issue #5613 -所有者以外のユーザーが pgAdmin にアクセスできるように、appbundle に正しい権限があることを確認します。

Issue #5622 -スキーマ差分の一部のケースで所有者無視フラグが機能しない問題を修正しました。

Issue #5626 -ツリーノードで実行されるアクションがコンテキストメニューオプションを更新する必要がある問題を修正しました。

Issue #5627 -トリガーのコンテキストメニューの下のサブメニューが正しく有効/無効になっていることを確認します。

Issue #5640 -boto3 と botocore を最新バージョンに更新します。

Issue #5641 -列が 3 未満の場合、GeometryViewer にポップアップが表示されない問題を修正しました。

Issue #5647 -行数の通知が自動的に消えていた問題を修正しました。

Issue #5661 -選択ドロップダウンの境界線の問題を修正しました。

Issue #5666 -Azure や Google などの oAuth2 プロバイダーでログインするときに発生したメタデータエラーに/"jwks_uri/"が見つからない問題を修正しました。

Issue #5675 -入力ボックスが空の場合に名前を追加しようとすると、名前変更パネルがフォーカスを失う問題を修正しました。

13.2 バージョン 6.18

発売日：2022-12-19

このリリースには、pgAdmin4v6.17 のリリース以降の多数のバグ修正と新機能が含まれています。

13.2.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13、14 および 15

EDB Advanced Server ：10、11、12、13、14 および 15

13.2.2 新機能

Issue #4088 -リレーションシップを選択するときの ERD の機能強化。

Issue #5503 -デスクトップモードにネイティブメニューのサポートを追加しました。

13.2.3 ハウスキーピング

13.2.4 バグ修正

Issue #5453 -クエリツールのセッションでトランザクション ID が見つからない問題を修正しました。

Issue #5470 -SQL のパーティションテーブルでテーブルスペースが見つからない問題を修正しました。

Issue #5536 -選択したノードが同じでも、タブを変更するとプロパティタブが更新される問題を修正しました。

Issue #5551 -二重引用符でオートコンプリートが期待どおりに機能しない問題を修正しました。

Issue #5564 -テーブル統計がサイズでソートされていることを確認します。

Issue #5598 -サーバーノード情報を更新すると、保存されたパスワードのクリアメニューが削除される問題を修正しました。

Issue #5603 -外部構成データベースでマスターパスワードが正しく設定されなかった問題を修正しました。

Issue #5606 -PG-15 の照合作成スクリプトのエラーを修正しました。

Issue #5629 -BigAnimal の認証が中断された問題を修正しました。

Issue #5637 -BigAnimal が Oracle 互換データベースタイプの PG バージョン 11-14 を表示することを確認します。

Issue #5645 -タイプミスを修正しました。

13.3 バージョン 6.17

発売日：2022-12-02

このリリースには、pgAdmin4v6.16 のリリース以降の多くのバグ修正と新機能が含まれています。

13.3.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13、14 および 15

EDB Advanced Server ：10、11、12、13、14 および 15

13.3.2 新機能

13.3.3 ハウスキーピング

Issue #5147 -BigAnimalAPI バージョンを V2 に更新します。

Issue #5493 -バックボーンとアンダースコアのすべてのトレースを削除します。

13.3.4 バグ修正

Issue #5488 -スキーマ差分ツールの外部キーに間違ったスキーマが表示される問題を修正しました。

Issue #5495 -デスクトップモードのクエリ履歴の日付形式が pgadmin サーバーのロケールの形式と一致していることを確認します。

Issue #5505 -CSV の引用文字の長さが 1 を超えると CSV ファイルがダウンロードされない問題を修正しました。

Issue #5513 -構成ファイルで個別に指定されていない場合、DATA_DIR 依存フォルダー/ファイルが指定された DATA_DIR 内に自動的に作成されます。

Issue #5539 -ユーザーが理解しやすいようにエラーメッセージを改善しました。

Issue #5548 -結果グリッドからデータを編集するときにエディターの位置が間違っている問題を修正しました。

Issue #5575 -PostgreSQL サービスに登録されているサーバーでクエリツールが正常に起動されていることを確認します。

Issue #5593 -許可され認証されたユーザーのみがバイナリパスを検証できるようにします。

13.4 バージョン 6.16

発売日：2022-11-18

このリリースには、pgAdmin4v6.15 のリリース以降の多くのバグ修正と新機能が含まれています。

13.4.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13、14 および 15

EDB Advanced Server ：10、11、12、13、14 および 15

13.4.2 新機能

Issue #1832 -pgAdmin の構成を外部データベースに保存するためのサポートが追加されました。

Issue #4756 -テーブルの ERD を生成する機能を追加しました。

Issue #5468 -pgAdmin ユーザー名に使用される Oauth2 クレームを構成する可能性を追加します。

13.4.3 ハウスキーピング

13.4.4 バグ修正

Issue #2174 -サーバーノードを展開したときに、ブラウザーツリーが選択したノードまで自動スクロールするようにします。

Issue #4841 -スキーマの比較に REST の代わりに SocketIO を使用します。

Issue #5066 -クエリ結果をダウンロードするときに、ユーザーが CSV フィールドの区切り文字/CSV 引用符としてカスタム文字を使用できることを確認します。

Issue #5058 -クエリツールの [並べ替え/フィルター] ダイアログで、[保存] ボタンがデフォルトで無効になっていることを確認します。

Issue #5098 -テーブルのプロパティダイアログが開いたときに保存ボタンが有効になる問題を修正します。

Issue #5122 -ノードの更新時にスピナーがブラウザーツリーに表示されることを確認します。

Issue #5149 -データベースへの接続が許可されていない場合、GenerateERD オプションが非表示になっていることを確認します。

Issue #5206 -ブラウザーのサイズが変更されたときに選択ドロップダウンの位置を変更します。

Issue #5281 -二重引用符で始まるオブジェクトでオートコンプリートが適切に機能することを確認します。

Issue #5344 -pgAdmin ルートに SCRIPT_NAME プレフィックスが必要であることを確認します。

Issue #5424 -サーバーモードでサーバーとの SSL 接続を試行する前に、キーファイルに適切な権限が設定されていることを確認します。

Issue #5429 -ロールのパラメーターが表示されない問題を修正しました。

Issue #5452 -コンテナー展開ドキュメントには、server.json ファイル形式を含める必要があります。

Issue #5455 -依存関係タブが PG15 で機能しない問題を修正しました。

Issue #5458 -検索オブジェクトのブラウザーパス列に完全なパスが表示されていることを確認します。

Issue #5463 -Enter キーを押してデータを編集しようとしたときに結果グリッドが正しく機能しない問題を修正しました。

Issue #5465 -wcDocker パネルを閉じるときに画面がフリーズする問題を修正しました。

Issue #5473 -不正な正規表現が原因でオートコンプリートが正しく機能しない問題を修正しました。

Issue #5475 -新しいタブで開いたクエリ/ERD ツールを閉じると、「閉じるまたは更新時に確認」設定が無視される問題を修正しました。

Issue #5507 -pgadmin がリバースプロキシを尊重しない問題を修正しました。

Issue #5521 -pgAdmin4 サーバーがサブディレクトリから実行されている場合の SocketIO 呼び出しを修正しました。

Issue #5522 -ロードファイルのパスがストレージディレクトリの子であることを確認します。

Issue #5533 -クエリツールを開くときに共有サーバーのユーザー名を使用します。

Issue #5535 -外部データベースを使用しているときに「save_password」列が共有サーバーのエラーをスローする問題を修正しました。

Issue #5537 -権限が拒否された場合の ERD に正しいエラーメッセージが表示されることを確認します。

13.5 バージョン 6.15

発売日：2022-10-20

このリリースには、pgAdmin4v6.14 のリリース以降の多数のバグ修正と新機能が含まれています。

13.5.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13、14 および 15

EDB Advanced Server ：10、11、12、13、14 および 15

13.5.2 新機能

Issue #3491 -AWSRDS または AzureDB の IAM トークンベースの認証のサポートを追加しました。

Issue #4392 -ERD ツールのテーブルノードに背景の塗りつぶし色を指定するサポートを追加しました。

Issue #4994 -ERD ツールでドラッグアンドドロップを使用してテーブルの列を並べ替えることができます。

Issue #4997 -ERD ツールで DROP テーブル DDL を使用して SQL を生成するオプションを追加します。

Issue #5304 -AWS デプロイメントに高可用性オプションを追加しました。

Issue #5390 -コンテナ内の Gunicornlimit_request_line パラメーターを公開します。デフォルトは最大 8190 に設定されています。

13.5.3 ハウスキーピング

Issue #5065 -ERD でデータベーステーブルのデータを取得するために、REST の代わりに SocketIO を使用します.

Issue #5293 -ツールチップがアプリケーション全体で一貫していることを確認します。

Issue #5357 -Python の'Six' パッケージを完全に削除します。

13.5.4 バグ修正

Issue #5101 -スキーマ差分でスキーマを比較するときに、ACLS の一貫した順序を保証します。

Issue #5132 -結果のグリッド列が最初の実行時に事前設定として幅を取る必要があることを確認します。

Issue #5133 -バックアップ中に発生し、指定されたパスに SSL 証明書/キーが見つからない問題を修正しました。

Issue #5145 -OAuth2 ログイン中に表示される断続的なエラーを修正しました。

Issue #5167 -複数のユーザーが同時に PSQL ツールを開いたときに、psqlrc ファイルへのパスが正しいことを確認します。

Issue #5188 -ユーザーがプロシージャのデバッガーを停止した場合、続行/スタートボタンを無効にする必要があります。

Issue #5210 -Alt+Shift+Q を押したときに、クエリツールが適切なユーザーで新しいタブを作成することを確認します。

Issue #5212 -通知のすべての通知に閉じるボタンを追加しました。

Issue #5249 -クエリツールから選択したテキストを検索/置換ボックスに表示する機能を追加しました。

Issue #5261 -新しい行がユーザー管理に追加されたときに検索フィルターをクリアする必要があることを確認します。

Issue #5262 -ユーザーを作成するときに、ユーザー管理ダイアログで大文字と小文字が異なる同じメールアドレスが許可されないことを確認します。

Issue #5277 -XSS の脆弱性の問題を修正しました。

Issue #5296 -タブ変更時にクエリツールの結果セットのスクロール位置が保持されるようにします。

Issue #5308 -列を空にする場合、列のデフォルト値を使用する必要があることを確認します。

Issue #5327 -ユーザーが Safari で特権を選択できない問題を修正しました。

Issue #5332 -ブラウザーツリーからデータベースノードを更新中に表示されるコンソール警告を修正しました。

Issue #5338 -大きなデータで [結果をファイルに保存] ボタンをクリックしてもプロンプトが表示されない問題を修正しました。

Issue #5352 -複数の電子メール属性を持つユーザーの LDAP 認証中に発生するエラーを修正しました。

Issue #5364 -通知がすぐに消える問題を修正しました。

Issue #5367 -パスワードのデコード中に例外が発生した場合に正しい値が返されることを確認します。

Issue #5368 -ファイルマネージャーからファイルをダウンロードする際の問題を修正しました。

Issue #5386 -認証ソースが OAuth2 または Kerberos の場合、ログインフォームが非表示であることを確認します。

Issue #5397 -パスワード回復リンクが機能していない問題を修正しました。

Issue #5402 -ウィンドウのサイズ変更時にブラウザツリーのスクロールバーが表示されるようにする。

Issue #5405 -クロスサイトスクリプティングの脆弱性を修正しました。

Issue #5427 -フィルタリングされた行が機能していない問題を修正しました。

13.6 バージョン 6.14

発売日：2022-09-22

このリリースには、pgAdmin4v6.13 のリリース以降の多数のバグ修正と新機能が含まれています。

13.6.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13 および 14

EDB Advanced Server ：10、11、12、13 および 14

13.6.2 新機能

13.6.3 ハウスキーピング

Issue #4059 -React にスキーマ diff を移植します。（RM#6133）

Issue #4060 -バックグリッドとバックフォームを削除します。（RM#6134）

Issue #5035 -ERD ツールの残りのコンポーネントを React に移植します。（RM#7343）

Issue #5120 -キーボードショートカットのドキュメントを更新しました。（RM#7446）

Issue #5260 -pgAdmin から Alertify を完全に削除します。（RM#7619）

Issue #5263 -検索オブジェクトダイアログを React に移植します。（RM#7622）

13.6.4 バグ修正

Issue #5144 -BigAnimal 認証が中止された場合、API 呼び出しを停止する必要があることを確認します。（RM#7472）

Issue #5209 -bin パスの移行が原因で pgAdmin を起動できない問題を修正しました。（RM#7557）

Issue #5230 -パラメーター「preexec_fn」がサポートされなくなったためにバックアップが機能しない問題を修正しました。（RM#7580）

Issue #5250 -所有権を変更した後にブラウザーツリーを更新する必要があることを確認します。（RM#7607）

Issue #5274 -展開中のクラウドサーバーをクリックすると表示されるエラーメッセージを修正しました。（RM#7636）

Issue #5275 -ユーザーが列のないテーブルに RLS ポリシーを作成した場合、間違った SQL が違いとして表示される問題を修正しました。（RM#7637）

Issue #5282 -ダンプサーバー機能が setup.py から動作することを確認します。（RM#7644）

Issue #5284 -Import/Export サーバーメニューオプションが表示されていることを確認します。（RM#7646）

Issue #5286 -サーバーモードでのパスワード変更の API テストケースを修正しました。（RM#7648）

Issue #5287 -データのインポート/エクスポートの前に追加された列が非表示になる問題を修正しました。（RM#7649）

Issue #5292 -JSON エディターのテキストエリアがダイアログでサイズ変更されない問題を修正しました。（RM#7656）

Issue #5299 -setup.py を実行してサーバーをロード/ダンプするときの ModuleNotFoundError を修正しました。（RM#7663）

Issue #5323 -言語選択「ブラジル」を「ポルトガル語（ブラジル）」に置き換えます。（RM#7693）

Issue #5325 -特殊文字を含むサーバー名がプロセスタブに正しく表示されない問題を修正しました。（RM#7695）

Issue #5333 -変数が列に追加されると ERD がエラーをスローする問題を修正しました。（RM#7709）

Issue #5342 -ERD テーブルへの変更を保存する際のエラーを修正しました。

Issue #5343 -ユーザーが pgAdminURL を開いたときのリダイレクトの脆弱性を修正します。

13.7 バージョン 6.13

発売日：2022-08-25

このリリースには、pgAdmin4v6.12 のリリース以降の多数のバグ修正と新機能が含まれています。

13.7.1 サポートされているデータベースサーバー

PostgreSQL ：10、11、12、13 および 14

EDB Advanced Server ：10、11、12、13 および 14

13.7.2 新機能

Issue #3709 -すべてのバックグラウンドプロセスを別のパネルに表示するサポートを追加しました。

Issue #7387 -EPAS の既存のトリガー関数からトリガーを作成するサポートが追加されました。

13.7.3 ハウスキーピング

Issue #7344 -React へのポートロールの再割り当てダイアログ。

Issue #7345 -React にユーザー管理ダイアログを移植します。

Issue #7404 -プロセスウォッチャーを React に移植します。

Issue #7462 -サポートされていないバージョンのデータベースサーバーの SQL ファイルを削除します。

Issue #7567 -React にダイアログを移植します。

Issue #7568 -ユーザーパスワードと 2FA ダイアログを React にポートします。

Issue #7590 -所有権の変更ダイアログを React にポートします。

Issue #7595 -コンテナーのベースイメージを Alpine3.16(Python3.10.5 を使用) に更新します。

Issue #7602 -PalletsWerkzeugv2.1.0 以下での HTTP リクエストの不適切な解析を修正しました（CVE-2022-29361）。

13.7.4 バグ修正

Issue #7452 -クリップボードアクセスが提供されていない場合にエラーがスローされることを確認し、行のコピーショートカットを変更します。

Issue #7468 -一部のクエリの実行後に [履歴] タブが空白になり、エラーが表示される問題を修正しました。

Issue #7481 -シーケンスにユーザー権限を追加するときに OWNEDBY が誤って NONE に設定される問題を修正しました。

Issue #7497 -Azure 展開の適切な場所に表示されるエラーメッセージの問題を修正しました。

Issue #7521 -クエリを保存するときにクエリエディターがフォーカスを失う問題を修正しました（Alt+s）。

Issue #7527 -Postgres14.4 の API テストケースを修正しました。

Issue #7540 -名前変更パネルが表示/編集パネルで機能することを確認します。

Issue #7563 -クエリエディターをクリアした後、オートコンプリートが機能しない問題を修正しました。

Issue #7573 -矢印キーを使用してコードを移動するときにオートコンプリートが表示されないようにします。

Issue #7575 -新しいクエリを作成した後に Alt-Shift-Q が機能しない問題を修正しました。

Issue #7579 -結果グリッドで行をコピーして貼り付けても、ブール値がデフォルトに設定されない問題を修正しました。

Issue #7586 -完全な列を選択するときのジオメトリのレンダリングの問題を修正しました。

Issue #7587 -information_schema および pg_catalog ノードの子が表示されることを確認します。

Issue #7591 -スキーマオブジェクトの比較中に、列/"none/"が存在しない問題を修正しました。

Issue #7596 -スキーマの差分が RLS ポリシーの変更を取得しない問題を修正しました。

Issue #7608 -クラウド展開ウィザードが、そのオプションが選択されていない場合でも高可用性でクラスターを作成する問題を修正しました。

Issue #7611 -ビュー定義が変更されたときに、schemadiff がビューの所有権を維持するようにします。

Issue #7614 -pgAdmin からログアウトするときに cryptkey が見つからない問題を修正しました。

Issue #7616 -パスワードが Azure デプロイと一致しない場合、次へのボタンを無効にする必要があります。

Issue #7617 -Azure クラウドのデプロイが失敗した問題を修正しました。

Issue #7625 -スペイン語の翻訳のタイプミスを修正しました。

Issue #7630 -トリガー関数定義が変更された場合は、スキーマ差分でトリガーを削除して再作成します。

Issue #7632 -ユーザーが OSX で AzureCLI を使用して認証できなかった問題を修正しました。

Issue #7633 -マスターパスワードとサーバーパスワードのダイアログの入力コントロールにオートフォーカスがあることを確認します。

Issue #7641 -ピン Flask-SocketIO<=v5.2.0。最新バージョンは、実稼働環境での Werkzeug をサポートしていません。

13.8 バージョン 6.12

発売日：2022-07-28

このリリースには、pgAdmin4v6.11 のリリース以降の多くのバグ修正と新機能が含まれています。

13.8.1 新機能

Issue #4488 -クエリツールでキーを押したときにオートコンプリートをトリガーするオプションを追加しました。

Issue #4607 -ユーザーがストレージマネージャーからファイル/フォルダーを削除できるようにします。

Issue #7389 -ユーザーがファイル/ストレージマネージャー内を検索できるようにします。

Issue #7486 -クエリツールで積み上げ線、棒、および積み上げ棒チャートを使用してグラフを視覚化するためのサポートが追加されました。

Issue #7487 -クエリツールで円グラフを使用したグラフの視覚化のサポートが追加されました。

13.8.2 ハウスキーピング

Issue #7313 -ファイル/ストレージマネージャーを React に移植します。

Issue #7341 -パスワード変更ダイアログを React にポートします。

Issue #7342 -React へのポートマスターパスワードダイアログ。

Issue #7492 -動的モジュールのロードを削除し、静的ロードに置き換えます。

Issue #7546 -名前付けた復元ポイントダイアログを React にポートします。

13.8.3 バグ修正

Issue #7428 -データのインポート/エクスポートダイアログでユーザーが設定した設定を保持します。

Issue #7471 -スプラッシュスクリーンを移動できることを確認します。

Issue #7508 -インデックスのコメントが表示されない問題を修正しました。

Issue #7512 -メッセージタブから通知が消えないことを確認します。

Issue #7517 -実行が完了したら、デバッグの開始ボタンを有効にします。

Issue #7518 -パネルが閉じられた後にダッシュボードグラフ API が呼び出されないことを確認します。

Issue #7519 -選択したすべてのセルにジオメトリが表示されることを確認します。

Issue #7520 -最初のレコードを非表示にする JSON エディターの問題を修正しました。

Issue #7522 -サーバーモードでの AzurePostgreSQL デプロイのサポートが追加されました。

Issue #7523 -テーブルヘッダーの統計のタイプミスを修正しました。

Issue #7524 -保存ダイアログで新しいフォルダーを作成できない問題を修正しました。

13.9 バージョン 6.11

発売日：2022-06-30

このリリースには、pgAdmin4v6.10 のリリース以降の多数のバグ修正と新機能が含まれています。

13.9.1 新機能

Issue #2647 -デバッガーのブレークポイント領域のマウスオーバー表示を追加しました。

Issue #2648 -Debugger パネルに検索テキストオプションを追加しました。

Issue #7178 -MicrosoftAzure に PostgreSQL サーバーを展開する機能が追加されました。

Issue #7332 -DockerSecret を使用してパスワードを Docker イメージに渡すためのサポートを追加しました。

Issue #7351 -オプション「テンプレートデータベースを表示しますか?」を追加しました「システムオブジェクトを表示しますか?」の設定に関係なくテンプレートデータベースを表示します。

Issue #7485 -クエリツールで折れ線グラフを使用したグラフの視覚化のサポートが追加されました。

13.9.2 ハウスキーピング

Issue #6132 -デバッガーを React に移植します。

Issue #7315 -Traefikv2 コンテナ展開のドキュメントを更新します。

Issue #7411 -pgcli を最新リリース 3.4.1 に更新します。

Issue #7469 -Chartjs を最新の 3.8.0 にアップグレードします。

13.9.3 バグ修正

Issue #7423 -クエリツールに通知をオフにする設定がない問題を修正しました。

Issue #7440 -「ConnectToServer」ダイアログに入力されたパスワードが切り捨てられる問題を修正しました。

Issue #7441 -クエリツールのタブを切り替えるときに、クエリエディターにフォーカスがあることを確認します。

Issue #7443 -クエリツールで「スペースを使用」が機能しない問題を修正しました。

Issue #7453 -データベースの制限が機能しない問題を修正しました。

Issue #7460 -新しいインデックスの作成中に pgAdmin がスタックする問題を修正しました。

Issue #7461 -ユーザーが新しい接続に切り替えてクエリツールを閉じたときに接続が閉じられない問題を修正しました。

Issue #7468 -「履歴がありません」を表示する代わりに、JSON 情報を解析できない場合、履歴レコードをスキップします。

Issue #7502 -新しいデータベースを作成するときにエラーメッセージが表示される問題を修正しました。

Issue #7506 -Docker を使用して Azure に PostgreSQL をデプロイするときの permissiondenied エラーを修正しました。

13.10 バージョン 6.10

発売日：2022-06-02

このリリースには、pgAdmin4v6.9 のリリース以降の多数のバグ修正と新機能が含まれています。

13.10.1 新機能

Issue #7364 -ダッシュボードテーブルの列のサイズを変更する機能を追加しました。

13.10.2 ハウスキーピング

Issue #7283 -PG15 の互換性の問題が修正されました。

Issue #7337 -ポート接続サーバーパスワードダイアログを React.

13.10.3 バグ修正

Issue #6962 -ブラウザツリーのノードのオーバーラップと展開の問題を修正しました。

Issue #7002 -双方向 Unicode 文字を検出してユーザーに警告する機能を追加しました。

Issue #7347 -認証ボタンが無効になっているときに、それらのテキストがダークとハイコントラストのテーマで表示されるようにします。

Issue #7368 -BigAnimal に対して不要な API が呼び出されないようにします。

Issue #7372 -コンテナをビルドするときに常に最新のベースイメージをプルするように Docker に指示します。

Issue #7373 -ジオメトリウィンドウのズームマウススクロールが機能しない問題を修正しました。

Issue #7374 -クエリツールのドロップダウンで接続を切り替えると、背景と前景色を変更する必要がある問題を修正しました。

Issue #7376 -空のクエリエディターのファイルを開くボタンをクリックすると、未保存の変更のポップアップが表示される問題を修正しました。

Issue #7380 -クエリツールグリッドでのマルチセル選択のサポートを追加しました。

Issue #7383 -ダイアログを最大化したときに設定が保存されない問題を修正しました。

Issue #7388 -クエリが長い場合にエラーメッセージがウィンドウ全体に表示される問題を修正しました。

Issue #7393 -開いたら、エディターの位置が変更されないようにします。

Issue #7394 -単一のセルを選択したときにジオメトリが表示されない問題を修正しました。

Issue #7399 -クエリツールに欠落していたトグルケースのキーボードショートカットを追加しました。

Issue #7402 -ブラウザツリーからノードを変更したときにダッシュボードグラフを更新する必要があることを確認します。

Issue #7403 -ドメインノードを選択するときにドメインの制約に関するコメントが表示されない問題を修正しました。

Issue #7405 -react-data-grid の数値列に null 値が受け入れられることを確認します。

Issue #7408 -テーブルに長さまたは精度の配列タイプの列がある場合、テーブルの編集中に列タイプが選択されない問題を修正しました。

13.11 バージョン6.9

発売日：2022-05-12

このリリースには、pgAdmin46.8 のリリース以降の多数のバグ修正と新機能が含まれています。

13.11.1 新機能

Issue #3253 -クエリツールにステータスバーを追加しました。

Issue #3989 -水平にスクロールしたときに行番号がビューに表示されるようにします。

Issue #6830 -GISViewer ボタンを結果テーブルの左側に再配置します。

Issue #7179 -EDBBigAnimal に PostgreSQL サーバーをデプロイする機能が追加されました。

Issue #7282 -スキーマの差分パネルにオプション「所有者を無視」および「空白を無視」を追加しました。

Issue #7325 -AzureADOAUTH2 認証のサポートを追加しました。

13.11.2 ハウスキーピング

Issue #6131 -クエリツールを React に移植します。

Issue #6746 -Kerberos ドキュメントを改善.

Issue #7255 -データベースとスキーマの制限コントロールがドロップダウンとして表示されないことを確認します。

Issue #7340 -データフィルターダイアログを React に移植します。

13.11.3 バグ修正

Issue #6725 -ユーザーがマルチプルのクエリツールウィンドウをすばやく開くと、クエリツールが最小サイズで開く問題を修正しました。

Issue #6958 -作成時にストレージディレクトリにのみ権限を設定します。

Issue #7026 -Browser パネルが完全に表示されない問題を修正しました。

Issue #7168 -列名が非常に長い場合に結果テーブルのサイズを変更するための GeometryViewer ポップアップを改善しました。

Issue #7187 -ダウンロードした ERD 図が 0 バイトだった問題を修正しました。

Issue #7188 -接続バーが表示されない問題を修正しました。

Issue #7231 -cpython モジュールが壊れる可能性があるため、サーバー RPM にパッケージ化するときにバイナリを削除しないでください.

Issue #7252 -インポート/エクスポートダイアログで列が常に表示されるようにします。

Issue #7260 -クエリの実行後に空のメッセージがポップアップする問題を修正しました。

Issue #7262 -接続を変更した後にオートコンプリートが機能することを確認します。

Issue #7294 -最初の列にデータが含まれていない場合、行のコピーと貼り付けが機能しない問題を修正しました。

Issue #7296 -プロパティパネルから複数のオブジェクトを削除した後、ブラウザーツリーを更新する必要があることを確認します。

Issue #7299 -統計パネルの並べ替えの問題を修正しました。

Issue #7305 -ダッシュボードサーバーアクティビティが古いクエリをアクティブとして表示していた問題を修正しました。

Issue #7307 -同じ列に複数のシーケンスを作成すると、テーブルに重複した列が表示される問題を修正しました。

Issue #7308 -サーバーアクティビティの更新時に並べ替えが保持されるようにします。

Issue #7322 -新しいデータベースの作成中にデータの取得に失敗したエラーがスローされる問題を修正しました。

Issue #7333 -ERD のドラッグアンドドロップテーブルがエラーをスローする問題を修正しました。

Issue #7339 -更新ボタンがクリックされたときにダッシュボードの列の並べ替え順序が記憶されることを確認します。

13.12 バージョン 6.8

発売日：2022-04-07

このリリースには、pgAdmin46.7 のリリース以降の多数のバグ修正と新機能が含まれています。

13.12.1 新機能

Issue #7215 -サーバーアクティビティセッションビューにトランザクションの開始時間を追加しました。

Issue #7249 -ERD に一意のキーのサポートを追加しました。

Issue #7257 -代替 UID を使用した OpenShift でのコンテナーの実行をサポートします。

13.12.2 ハウスキーピング

Issue #7132 -React のコレクションノード、ダッシュボード、および SQL パネルの PortProperties パネル。

Issue #7149 -設定ダイアログを React に移植します。

13.12.3 バグ修正

Issue #4256 -失効ステートメントの SQL がデータベースに表示されない問題を修正しました。

Issue #5836 -LDAP スキーマ/サーバーの大文字と小文字の区別を示す新しい LDAP 認証構成パラメーターを追加します。

Issue #6960 -暗号化キーが見つからない場合にマスターパスワードダイアログが表示されることを確認します。

Issue #7059 -認証ソースが oauth2 の場合、ログアウト時にエラーが表示される問題を修正しました。

Issue #7176 -ブラウザーツリーの状態が正しく保持されない問題を修正しました。

Issue #7197 -主キーが変更されたときに外部キーの関係が更新されない問題を修正しました。

Issue #7216 -コレクションノードの統計タブで特定のフィールドの値が正しく設定されていることを確認します。

Issue #7221 -拡張子に依存するオブジェクトが SchemaDiff に表示されないようにする。

Issue #7238 -ERD で参照先のテーブルを削除しても外部キーが削除されない問題を修正しました。

Issue #7239 -更新時にテーブルノードの下に新しく追加されたテーブルが表示されない問題を修正しました。

Issue #7261 -ドキュメントのタイプミスを修正。

Issue #7263 -引数にコンマを含むデフォルト値がある場合にファンクションの差分 DDL が正しく表示されないスキーマの差分の問題を修正しました。

Issue #7264 -新しい接続ダイアログで正しいユーザーが選択されていることを確認します。

Issue #7265 -外部テーブルの DDL ステートメントにオプション「null」が表示されないスキーマの差分の問題を修正しました。

Issue #7267 -ユーザーが設定から言語を変更すると、予期しないエラーメッセージが表示される問題を修正しました。

Issue #7269 -pgAdmin4 が最新の jinja2 バージョンで動作することを確認します。

Issue #7275 -ERD ツールでテーブルを作成する際の'Cannotreadpropertiesofundefined' エラーを修正しました。

13.13 バージョン 6.7

発売日：2022-03-14

このリリースには、pgAdmin46.6 のリリース以降の多数のバグ修正と新機能が含まれています。

注釈: Security Release

このリリースには、サーバーモードで実行されている pgAdmin を使用しているときにユーザーがストレージディレクトリの外部のディレクトリにファイルをアップロードできる問題を修正するためのセキュリティ更新が含まれていることに注意してください。

標準のコンテナベースのディストリビューションを含む、サーバーモードで pgAdmin を実行しているユーザーは、できるだけ早くこのリリースにアップグレードする必要があります。

この問題は、デスクトップモードで実行しているユーザーには影響しません。

13.13.1 バグ修正

Issue #7220 -テーブルの外部キー値が異なる場合に差分 SQL が生成されないスキーマの差分の問題を修正しました。

Issue #7228 -同一のユーザーマッピングで文字列区切り文字「_$PGADMIN$_」が表示されるスキーマの差分の問題を修正しました。

Issue #7230 -pgAdmin4 がスプラッシュ画面に「StartingpgAdmin」テキストを表示するまでに 75 秒かかる問題を修正しました。

Issue #7233 -アップロードパスがストレージディレクトリの子であることを確認する (CVE-2022-0959)

13.14 バージョン 6.6

発売日：2022-03-10

このリリースには、pgAdmin46.5 のリリース以降の多数のバグ修正と新機能が含まれています。

13.14.1 新機能

Issue #7177 -AmazonRDS に PostgreSQL サーバーをデプロイする機能を追加しました。

13.14.2 ハウスキーピング

Issue #7180 -メニューの名前を「DisconnectDatabase」から「Disconnectfromdatabase」に変更します。

13.14.3 バグ修正

Issue #6956 -ユーザーマッピングが正しく比較されなかったスキーマ差分の問題を修正しました。

Issue #6991 -バインド前に STARTTLS が必要な場合、pgadmin が LDAP に接続できない問題を修正しました。

Issue #6999 -認証元が内部と ldap の場合にメールアドレスに対して毎回警告が点滅する問題を修正しました。

Issue #7105 -オートコンプリート中に親パーティションテーブルが表示されなかった問題を修正しました。

Issue #7124 -テーブルに異なる列の位置があり、列にデフォルト値があるスキーマの差分の問題を修正しました。

Issue #7152 -関数コレクションノードのコメント列を追加しました。

Issue #7172 -検証エラーが発生したときに、ユーザーがスクロールして入力できるようにします。

Issue #7173 -[UserManagement] ダイアログが開かない問題を修正しました。

Issue #7181 -ユーザーが UNIX ソケット接続で新しいサーバーを追加できることを確認します。

Issue #7186 -サーバー/データベースの接続メニューが正しく更新されなかった問題を修正します。

Issue #7202 -Flask-Security-Too が最新バージョンを使用していることを確認します。

13.15 バージョン 6.5

発売日：2022-02-11

このリリースには、pgAdmin46.4 のリリース以降の多数のバグ修正と新機能が含まれています。

13.15.1 新機能

Issue #7139 -外部 Web ブラウザーで SQL ヘルプ、ダイアログヘルプ、オンラインヘルプを開くサポートが追加されました。

13.15.2 ハウスキーピング

Issue #7016 -React へのポート依存、依存関係、統計パネル。

Issue #7017 -React へのインポート/エクスポートダイアログ。

Issue #7163 -メニューの名前を「サーバーから切断」に変更します。

13.15.3 バグ修正

Issue #6916 -ランタイムに GPU ハードウェアアクセラレーションを無効にするフラグを追加しました。

Issue #7035 -データベースサーバーへの初期接続で、接続が（閉じられた）接続を開いたままにする問題を修正しました。

Issue #7085 -複数のパーティションレベルを作成するときにパーティションテーブルが正しく表示されることを確認します。

Issue #7086 -「名前付き復元ポイントの追加」のドキュメントを修正しました。

Issue #7100 -シーケンスをスクロールするとブラウザツリーが消える問題を修正しました。

Issue #7109 -ファイル選択ダイアログのすべてのディレクトリのサイズを空白にします。

Issue #7110 -カーソルが [ユーティリティ] ダイアログの最初のオプションにあることを確認します。

Issue #7118 -JSON ファイルがストレージマネージャーから適切にダウンロードされることを確認します。

Issue #7123 -復元がスキーマの誤ったオプションを生成する問題を修正しました。

Issue #7126 -F2 ファンクションキーでブラウザーパネルのコンテンツが削除される問題を修正しました。

Issue #7127 -サーバーダイアログにホスト名の検証を追加しました。

Issue #7135 -インストーラーを実行する最小の Windows バージョンを強制します。

Issue #7136 -クエリツールが間違ったラベルを表示する問題を修正しました。

Issue #7142 -データベースの作成/変更後に警告メッセージが表示される問題を修正しました。

Issue #7145 -拡張機能の比較中に所有者を無視する必要があることを確認します。

Issue #7146 -SchemaDiff ツールでの問題を比較するイベントトリガーを修正しました。

Issue #7150 -デスクトップモードで CSV をアップロードするとエラーが発生する問題を修正しました。

Issue #7151 -復元ダイアログの固定値エラー。

Issue #7154 -クエリツールが開かれ、pgAdmin が再起動された場合にレイアウトがリセットされないようにします。

13.16 バージョン 6.4

発売日：2022-01-13

このリリースには、pgAdmin46.3 のリリース以降の多数のバグ修正と新機能が含まれています。

13.16.1 新機能

Issue #4803 -GUI からサーバーグループとサーバーをインポート/エクスポートするサポートを追加しました。

13.16.2 ハウスキーピング

Issue #7018 -React へのポートの復元ダイアログ。

Issue #7019 -React へのポートメンテナンスダイアログ。

13.16.3 バグ修正

Issue #6745 -ダイアログにエラーが表示されてもテーブルスペースが作成される問題を修正しました。

Issue #7003 -ExplainAnalyze に負の排他時間が表示される問題を修正しました。

Issue #7034 -新しい行を追加するときにデフォルト値の列が表示されない問題を修正しました。

Issue #7075 -プロシージャのヘルプが適切に表示されるようにします。

Issue #7077 -プロシージャーのリバースエンジニアリング SQL に所有者が表示されない問題を修正しました。

Issue #7078 -操作エラーメッセージにデータベースオブジェクトの名前が間違って表示される問題を修正しました。

Issue #7081 -PostgreSQL-14 関数の SQL 生成の問題を修正しました。

Issue #7093 -サブプランが他のノードと重なって、グラフィカル EXPLAIN ビューでアクセスできない問題を修正しました。

Issue #7096 -切り捨てとリセットの統計が機能することを確認します。

Issue #7102 -生成されたスクリプトがファンクション/プロシージャ/トリガー関数の不要な行末を追加するスキーマの差分の問題を修正しました。

13.17 バージョン6.3

発売日：2021-12-16

このリリースには、pgAdmin46.2 のリリース以降の多くのバグ修正と新機能が含まれています。

13.17.1 新機能

Issue #6543 -セキュリティを向上させるために二要素認証のサポートを追加しました。

Issue #6872 -macOS の PostgreSQL ライブラリとユーティリティに GSSAPI サポートを含めます。

Issue #7039 -データベースサーバーの自動検出を無効にするサポートが追加されました。

13.17.2 ハウスキーピング

Issue #6088 -Flask-BabelEx を Flask-Babel に置き換えます。

Issue #6984 -React のポートバックアップグローバル、バックアップサーバー、およびバックアップオブジェクトダイアログ。

Issue #7004 -alertifyjs 通知を React ベースの通知に置き換えました。

Issue #7010 -Flask をバージョン 2 にアップグレードします。

Issue #7053 -Alertify アラートを置き換え、React ベースのモデルダイアログで確認します。

13.17.3 バグ修正

Issue #6840 -ダウンロードしたグラフィカルな説明計画にツールチップデータが表示されない問題を修正しました。

Issue #6877 -スキーマ差分の所有者に関連する問題を修正しました。

Issue #6906 -1 行追加した後、外部キー->列で参照テーブルのドロップダウンを無効にする必要がある問題を修正しました。

Issue #6955 -サーバーグループの名前を変更するときに並べ替え順序が維持されるようにします。

Issue #6957 -いくつかの問題に関連するスキーマ差分を修正しました。

Issue #6963 -ユーザーが拡張機能の作成中にスキーマを設定できるようにする必要があります。

Issue #6978 -スクロールバーの幅を大きくします。

Issue #6986 -ユーザーがタイムスタンプパラメーターを使用して関数をデバッグできない問題を修正しました。

Issue #6989 -複数の認証ソースが定義されている場合、内部認証ソースの ChangePassword メニューオプションが表示されない問題を修正しました。

Issue #7005 -行セルを編集して保存し、スクロールしてさらに行を取得すると、オンデマンド行がエラーをスローする問題を修正しました。

Issue #7006 -Python3.10 および最新のイベントレットの依存関係によってアプリケーションが中断されないことを確認します。

Issue #7013 -インストール時に python3 との競合につながる可能性のある RPM ビルドの問題を修正します。

Issue #7015 -ツリー項目が選択されていないときにダッシュボードから [新しいサーバーの追加] を使用して新しいサーバーを作成するとエラーがスローされる問題を修正しました。

Issue #7020 -長いクエリを実行するときに、彫像メッセージがメッセージの最後の行を非表示にしないことを確認します。

Issue #7024 -Aggregate の SQL のリバースエンジニアリングが間違っている問題を修正しました。

Issue #7029 -PG14 の Atomic キーワードを使用して、ファンクション/プロシージャの SQL 定義を修正します。

Issue #7031 -USER_IDFK が使用できないテーブル user_old を参照するため、SQLite データベース定義が間違っている問題を修正しました。

Issue #7040 -Debian パッケージ制御ファイルに/"section/"を追加します。

Issue #7044 -dropzone バージョンを 5.9.3 に、Flask-SQLAlchemy を 2.5.*に更新します。

Issue #7046 -いくつかのアクセシビリティの問題を修正しました。

Issue #7048 -About ダイアログを開く際のハッシュできないタイプの問題を修正しました。

Issue #7064 -プロシージャの作成中に所有者を無効にしないでください。

Issue #7071 -確認ポップアップが Reassign/DropOwnedDialog の後ろに隠れる問題を修正しました。

13.18 バージョン 6.2

発売日：2021-11-18

このリリースには、pgAdmin46.1 のリリース以降の多くのバグ修正と新機能が含まれています。

13.18.1 新機能

Issue #3834 -表示専用モードでのアグリゲートおよびオペレーターノードのサポートを追加しました。

Issue #6953 -新しい構成パラメーター WEBSERVER_REMOTE_USER を導入することにより、ユーザーが環境ごとに REMOTE_USER 環境変数を変更できるようにします。

13.18.2 ハウスキーピング

13.18.3 バグ修正

Issue #5427 -特定のテーブルのロックが原因で実行できない操作のエラーメッセージを提供することにより、pgAdmin のフリーズの問題を修正しました。

Issue #6780 -新しく追加された列が親テーブルに存在する場合、列をマージする必要があることを確認します。

Issue #6809 -pgAdmin が適切に開かない問題を修正しました。

Issue #6832 -他の認証プロバイダーと組み合わせた場合の内部認証、プロバイダーを選択するときに内部ソースの順序が重要でないことを確認します。

Issue #6845 -継承テーブルアイコンがツリービューに適切に表示されることを確認します。

Issue #6859 -ブラウザーツリーからオブジェクトを追加してもプロパティパネルが更新されない問題を修正しました。

Issue #6896 -ユーザーがキーボードの矢印キーを使用してブラウザーツリーオブジェクトをナビゲートできることを確認します。

Issue #6905 -SSL を使用したリバースプロキシの背後でデータベースノードがロードされない問題を修正しました。

Issue #6925 -/"カスタム自動バキューム/"オプションを選択し、Autovacuum オプションを Yes または No に設定していない場合の SQL 構文エラーを修正しました。

Issue #6939 -ユーザーがサーバーグループを削除したときに確認ポップアップに古いサーバーグループ名が表示される問題を修正しました。

Issue #6944 -JSON エディターのプレビューの色がダークモードで不適切なコントラストになる問題を修正しました。

Issue #6945 -コードモードでの JSON エディターのスクロールの問題を修正しました。

Issue #6940 -管理者以外のユーザーが共有サーバーに接続しようとしたときにユーザーの詳細が表示されない問題を修正しました。

Issue #6949 -所有しているメニューの再割り当て/削除をクリックするとダイアログが開くようにする。

Issue #6954 -システムのハイコントラストモードが有効になっているときに、テーマの変更が Windows で機能することを確認します。

Issue #6972 -PG14 のバイナリパスが設定に表示されることを確認します。

Issue #6974 -検索オブジェクトに演算子と集計を追加しました。

Issue #6976 -テキストエリアのサイズを変更でき、255 文字を超える問題を修正しました。

Issue #6981 -インデックスの SQL が同じ列を複数回表示する問題を修正しました。

Issue #6988 -レイアウトが一貫性のない状態にあることを pgAdmin4 が検出した場合、レイアウトをリセットします。

13.19 バージョン 6.1

発売日：2021-10-21

このリリースには、pgAdmin46.0 のリリース以降の多数のバグ修正と新機能が含まれています。

13.19.1 新機能

Issue #4596 -ブラウザツリーにインデントガイドのサポートを追加しました。

Issue #6081 -ERD ツールの外部キー、主キーなどの高度なテーブルフィールドのサポートを追加しました。

Issue #6241 -テーブルを ERD ツールにドラッグできるようにするサポートを追加しました。

Issue #6529 -ERD ツールで SQL を生成するときにインデックスの作成を追加しました。

Issue #6657 -ウェブサーバーを介した認証のサポートを追加しました（REMOTE_USER）。

Issue #6794 -ルールを有効/無効にするサポートを追加しました。

13.19.2 ハウスキーピング

13.19.3 バグ修正

Issue #6719 -OAuth2 統合リダイレクトの問題を修正しました。

Issue #6754 -クエリツールのクエリの強調表示の色を少なくする必要があります。

Issue #6776 -ラベル「InheritsTables?」を変更しましたに' 継承されますか？プロパティパネルで誤解を招くため。

Issue #6790 -ユーザーが concurrently キーワードでインデックスを作成できない問題を修正しました。

Issue #6797 -ファンクション、プロシージャ、およびトリガーファンクションの SQL の先頭にある余分な空白行を削除します。

Issue #6802 -高度なサーバーのトリガーを編集する問題を修正しました。

Issue #6828 -検索結果から選択したオブジェクトにツリーがスクロールしない問題を修正しました。

Issue #6858 -コレクションノードのプロパティタブからのオブジェクト削除の問題を修正しました。

Issue #6876 -サーバーに接続した後にダッシュボードが更新されることを確認します。

Issue #6881 -解像度を変更してもブラウザーツリーにすべてのコンテンツが表示されない問題を修正しました。

Issue #6882 -ブラウザツリーのアルファベット順ではなく、作成順に列が表示されるようにします。

Issue #6890 -ブラウザーツリーの背景色の問題を修正しました。

Issue #6891 -ERD ツールに複合外部キーのサポートを追加しました。

Issue #6900 -アクセス方法名を一度変更すると、テーブルダイアログから排他制約を作成できない問題を修正しました。

Issue #6905 -ユーザーが HTTP リバースプロキシの背後にあるデータベースを読み込めない問題を修正しました。

Issue #6908 -クリックしてコレクションノードを更新すると、オブジェクトの数が 10 以上減少する問題を修正しました。

Issue #6912 -ブラウザーツリーのソート順のリグレッションの問題を修正しました。

Issue #6915 -SQLite データベースのサーバーテーブルに NULL の代わりに空白の文字列が保存される問題を修正しました。

Issue #6928 -MASTER_PASSWORD_REQUIRED が True に設定され、AUTHENTICATION_SOURCES が Web サーバーの場合にマスターパスワードを入力する必要があることを確認します。

Issue #6929 -ERD ツールでテーブルノードのみをドロップできることを確認します。

Issue #6930 -名前を変更すると既存のサーバーグループが消える問題を修正しました。

Issue #6935 -テーブルの列を一緒に削除して名前を変更すると、間違った SQL が生成される問題を修正しました。

13.20 バージョン 6.0

発売日：2021-10-07

このリリースには、pgAdmin45.7 のリリース以降の多数のバグ修正と新機能が含まれています。

13.20.1 新機能

Issue #4211 -シーケンスの OWNEDBY 句のサポートを追加しました。

13.20.2 ハウスキーピング

Issue #5741 -すべての CREATE および DROPDDL を再確認して、適切な「IFEXISTS」、「CASCADE」および「CREATEORREPLACE」を追加します。

Issue #6129 -ブラウザーツリーを React に移植します。

Issue #6588 -オブジェクトノードとプロパティダイアログを React に移植します。

Issue #6687 -反応する PortGrantWizard。

Issue #6692 -GPDB サポートを完全に削除します。

13.20.3 バグ修正

Issue #2097 -データベースのサイズが大きい場合に付与ウィザードが応答しない問題を修正しました。

Issue #2546 -COLLATE と opclass を使用してパーティションテーブルを作成するサポートを追加しました。

Issue #3827 -[クエリ履歴] タブで、クエリの詳細がスクロール可能であることを確認します。

Issue #6712 -ネストされた IF の場合に折りたたみと展開の矢印が一致しない問題を修正しました。

Issue #6713 -最後のメッセージがデバッガーに表示されない問題を修正しました。

Issue #6723 -ダークテーマのスタイルガイドに従って、クエリエラーの行選択の色を更新しました。

Issue #6724 -データベースでドロップカスケードボタンが有効になる問題を修正しました。

Issue #6736 -マテリアライズドビューでビューの更新オプションが機能しない問題を修正しました。

Issue #6755 -「attnum」および「edit_types」パラメーターのスキーマ差分のキーエラーの問題を修正しました。

Issue #6759 -プロパティタブのコレクションノードで RLS 名を編集できないことを確認します。

Issue #6798 -データグリッドで何かを変更すると、クエリツールの [実行] ボタンが無効になる問題を修正しました。

Issue #6834 -SQL ヘルプが EPAS サーバーで動作することを確認します。

13.21 バージョン 5.7

発売日：2021-09-09

このリリースには、pgAdmin45.6 のリリース以降の多数のバグ修正と新機能が含まれています。

13.21.1 新機能

Issue #2538 -リスタート ID を使用したテーブルの切り捨てのサポートを追加しました。

Issue #4264 -コードミラーでコードの折りたたみを大文字と小文字を区別しないようにします。

Issue #4629 -Maintenanceprocesswatcher ダイアログにデータベースとサーバーの情報を追加しました。

Issue #6495 -ERD ツールの 1 対多の関係で、参照されるテーブルをローカルテーブルと同じにすることができます。

Issue #6625 -ユーザがタブを閉じたときに適切なタブにフォーカスすることにより、タブをよりスマートに閉じます.

Issue #6691 -PSQLRC および PSQL_HISTORY 環境変数を apt に設定します。サーバーモードのユーザーストレージパス。

13.21.2 ハウスキーピング

13.21.3 バグ修正

Issue #4567 -[SQL] タブに正しく表示されないテーブルなどのオブジェクトで SQL クエリを使用して特権が取り消される問題を修正しました。

Issue #4815 -ユーザーが safari12 および 13 ブラウザーで更新されたテーブルヘッダーを貼り付けることができない問題を修正しました。

Issue #5849 -トリガー関数 SQL に「関数の作成」のみではなく、「関数の作成または置換」があることを確認します。

Issue #6419 -Windows の空白画面の問題を修正し、ウィンドウサイズを記憶するために NWjs マニフェストを使用するように変更しました。

Issue #6531 -リレーションラインがノード上にあるエクスポート画像の問題を修正しました。

Issue #6544 -PSQL ツールウィンドウの幅制限の問題を修正しました。

Issue #6564 -シーケンスを持つ列がスキーマ差分ツールで不必要に変更される問題を修正しました。

Issue #6570 -大規模なレコードのロックパネルがブロックされていないことを確認します。

Issue #6572 -データ出力パネルの表示の問題を部分的に修正します。

Issue #6620 -SchemaDiff を使用してスクリプトを生成しているときにファンクション本体の空白が適用されない問題を修正しました。

Issue #6627 -Oauth2 スコープ構成に OAUTH2_SCOPE 変数を導入しました。

Issue #6641 -ネストされたロールの場合に pgAdmin がユーザー権限を取得できるようにします。これは、AWSRDS のセッションを終了するのに役立ちます。

Issue #6663 -データベースサーバー接続時の属性「_asdict」エラーを修正しました。

Issue #6668 -JSON エディターのエラーメッセージに表示される HTML タグに関連するエラーを修正しました。

Issue #6671 -割り当て前にローカル変数「user_id」が参照された UnboundLocalError を修正しました。

Issue #6682 -「自動ロールバック?」の名前を変更しました「エラー時に自動ロールバックしますか?」

Issue #6684 -最初のレコードを非表示にする JSON エディターの問題を修正しました。

Issue #6685 -データベースを削除して次のデータベースに自動的に接続しないことを確認します。

Issue #6704 -ホスト名に特殊文字が原因で pgAdmin がログインに失敗しないようにします。

Issue #6710 -単一の close イベントに対して複数のクエリツールタブが閉じられる問題を修正しました。

13.22 バージョン 5.6

発売日：2021-08-12

このリリースには、pgAdmin45.5 のリリース以降の多数のバグ修正と新機能が含まれています。

13.22.1 新機能

Issue #4904 -メインウィンドウからクエリツールに SQL をコピーするサポートが追加されました。

Issue #5198 -JSON 列のデータを操作するときのフォーマットされた JSON ビューアー/エディターのサポートが追加されました。

13.22.2 ハウスキーピング

Issue #6622 -/"Resizebydata?/"の名前を/"Columnssizedby/"に変更し、'Columnssizedby' が'Columndata' に設定されている場合、'Maximumcolumnwidth' ボタンを無効にします。

13.22.3 バグ修正

Issue #6337 -N 回の試行後にログインアカウントがロックされることを確認します。N は、「MAX_LOGIN_ATTEMPTS」パラメーターを使用して構成できます。

Issue #6369 -デスクトップモードのセッションの有効期限を長くすることにより、古いセッションの CSRF エラーを修正しました。

Issue #6448 -ユーザーがサブスクリプションにアクセスできない場合に、「すべてのタイプ」または「サブスクリプション」で検索するときの検索オブジェクトの問題を修正しました。

Issue #6574 -PSQL の右クリックオプションで貼り付けが機能しない問題を修正しました。

Issue #6580 -修正 TypeError'NoneType' オブジェクトはサブスクリプト化できません。

Issue #6586 -オブジェクトの移動ダイアログのドロップダウンの不正なテーブルスペースオプションを修正しました。

Issue #6618 -クエリタブのタイトルが異なる問題を修正しました。

Issue #6619 -ユーザーが設定からバイナリパスを削除したときの不正なバイナリパスの問題を修正しました。

Issue #6643 -カスタムデータ型の作成中に Range データ型が選択されたときに、必要なすべてのオプションがロードされるようにします。

Issue #6650 -active_since パラメーターが None の場合のダッシュボードサーバーアクティビティの問題を修正しました。

Issue #6664 -ユーザーがロックされている場合でも、パスワードをリセットして pgAdmin にログインできる問題を修正しました。

13.23 バージョン 5.5

発売日：2021-07-15

このリリースには、pgAdmin45.4 のリリース以降の多数のバグ修正と新機能が含まれています。

13.23.1 新機能

Issue #1975 -ダッシュボードで長時間実行されるクエリを強調表示しました。

Issue #3893 -ログインロールの Reassign/DropOwned のサポートを追加しました。

Issue #3920 -クエリの実行時にクエリエディターウィンドウをブロックしないでください。

Issue #5940 -OAuth2 認証のサポートを追加しました。

Issue #6559 -設定の「Resizebydata?」オプションが True に設定されている場合に列の最大幅を提供するオプションを追加しました。

13.23.2 ハウスキーピング

13.23.3 バグ修正

Issue #4189 -データ出力パネルが切り離された後にスナップできることを確認します。

Issue #6388 -通常のキーボードレイアウトのクエリツールの置換キーボードショートカットの問題を修正しました。

Issue #6398 -クエリエディターパネルをデタッチすると空白のパネルが表示される問題を修正しました。

Issue #6427 -セルが空にならないように、クエリツールのデータグリッドで先頭の空白を削除し、「[...]」に置き換えます。

Issue #6448 -ユーザーがサブスクリプションにアクセスできない場合に、「すべてのタイプ」または「サブスクリプション」で検索するときの検索オブジェクトの問題を修正しました。

Issue #6489 -空のクエリを実行するときに [実行/更新] ボタンを無効にできない問題を修正しました。

Issue #6505 -新しい接続のドロップダウンでデフォルトのメンテナンスデータベース、自動選択、およびタブスルー機能が失われた問題を修正しました。

Issue #6536 -フォルダーダイアログでのディレクトリ選択の問題を修正しました。

Issue #6541 -設定「新しいブラウザータブで開く」が表示されることを確認します。「ENABLE_PSQL」の値に基づいてはなりません。

Issue #6547 -PSQL ツールターミナルのコピー/貼り付けの問題を修正しました。

Issue #6550 -下位互換性を維持するために flask-security-too で導入されたメール到達性チェックをデフォルトで無効にします。

Issue #6555 -「Login」キーワードのチェコ語の翻訳文字列を修正しました。

Issue #6557 -Index のプロパティに誤った列名がリストされる問題を修正しました。

13.24 バージョン 5.4

発売日：2021-06-17

このリリースには、pgAdmin45.3 のリリース以降の多くのバグ修正と新機能が含まれています。

13.24.1 新機能

Issue #1561 -設定でバイナリパスを選択するための参照ボタンを追加しました。

Issue #1591 -Package ノードの下に GrantWizard オプションを追加しました。

Issue #2341 -接続されたデータベースサーバーの PSQL を起動するサポートが追加されました。

Issue #4064 -プロパティダイアログにウィンドウの最大化/復元機能を追加しました。

Issue #5370 -さまざまなデータベースサーババージョンのバイナリパスを設定するサポートが追加されました。

Issue #6231 -[バージョン情報] ダイアログに OS、ブラウザ、構成の詳細を追加しました。

Issue #6395 -サイズと経過時間に基づいて pgAdmin ログファイルをローテーションするためのサポートが追加されました。

Issue #6524 -Windows での非管理者インストールをサポートします。

13.24.2 ハウスキーピング

Issue #4622 -Table とその子ノードの RESQL/MSQL テストケースを追加しました。

Issue #6225 -Flask-Security-Too を最新の v4 に更新しました。

Issue #6460 -壊れた/壊れた構成データベースファイルを検出するメカニズムを追加しました。

13.24.3 バグ修正

Issue #4203 -別のユーザーがデータベースの名前を変更する問題を修正しました。

Issue #6404 -クエリツールの接続文字列が「クエリツールのタブのタイトル」に従って変更されていないことを確認します。

Issue #6466 -ユーザーが作成中にログイン/ロールグループにメンバーを追加できる必要があることを確認します。

Issue #6469 -ロールのプロパティパネルでカレンダーコントロールを無効にする必要があることを確認します。

Issue #6473 -ランタイムでのブラウザーパスワードの保存を無効にします。

Issue #6478 -複数のパーティションを持つテーブルの重複 SQL の問題を修正しました。

Issue #6482 -サーバーダイアログの ForegroundColor プロパティが機能しない問題を修正しました。

Issue #6513 -サーバーモードでパスワードをリセットした後に pgAdmin が開かない問題を修正しました。

Issue #6520 -CSV 形式でデータをダウンロードするときに文字の可変フィールドに 10 進数が追加される問題を修正しました。

13.25 バージョン 5.3

発売日：2021-05-20

このリリースには、pgAdmin45.2 のリリース以降の多くのバグ修正と新機能が含まれています。

13.25.1 新機能

Issue #5954 -データ出力ウィンドウのコンテンツサイズで列の自動幅を設定するサポートが追加されました。

Issue #6158 -Kerberos 認証を介して PostgreSQL サーバーに接続するサポートが追加されました。

Issue #6397 -ERD ツールの使用中に便利なテーブルとパーティションテーブルの作成中に/"IFNOTEXISTS/"句を追加しました。

13.25.2 ハウスキーピング

13.25.3 バグ修正

Issue #4436 -プロパティダイアログの codemirror でオブジェクトのドラッグアンドドロップが正しくない問題を修正しました。

Issue #5477 -キャッシュバスト Web パックチャンクファイルのサポートを追加しました。

Issue #5555 -クエリを繰り返し実行すると、データが間違った順序で表示される問題を修正しました。

Issue #5776 -SSPI ログインを使用してサーバーに接続しているときに、パスワードの入力を求めないようにします。

Issue #6329 -子パーティションテーブルに間違った SQL が表示される問題を修正しました。

Issue #6341 -CSV ダウンロードが数値列を引用する問題を修正しました。

Issue #6355 -pgAdmin がドラッグされた外部ファイルを開くことを許可しないことを確認します。

Issue #6377 -スキーマの差分が列の DROPDEFAULT ステートメントを作成しない問題を修正しました。

Issue #6385 -バックアップと復元が共有サーバーで動作することを確認します。

Issue #6392 -フィルター「選択で除外」が null 値に対して機能しない問題を修正しました。

Issue #6399 -ユーザーが重複したパネルを追加できないことを確認します。

Issue #6407 -AdvancedServer のネストされたテーブルと可変配列タイプの作成のサポートが追加されました。

Issue #6408 -ルートの外部から setup.py を実行するときの ModuleNotFoundError を修正しました。

Issue #6409 -現在のデバッグ行が「Dark」テーマで表示されない問題を修正しました。

Issue #6413 -2 つのシーケンスが所有するブラウザーツリーに重複した列が表示される問題を修正しました。

Issue #6414 -Login/Group ロールでヘルプメッセージが正しく表示されない問題を修正しました。

Issue #6416 -ビューとマテリアライズドビューのコレクションノードのプロパティパネルにコメント列を追加しました。

Issue #6417 -ユーザーが [保存しない] ボタンをクリックしてもクエリエディターが閉じない問題を修正しました。

Issue #6420 -Quit コマンドで pgAdmin4 が完全にシャットダウンしたことを確認します。

Issue #6443 -ファイルダイアログに、選択したファイルタイプの誤ったファイルが表示される問題を修正しました。

Issue #6444 -グローバルバックアップの開始中に Kerberos チケットの有効期限が 30 分未満の場合、ユーザーに警告されない問題を修正しました。

Issue #6445 -サーバーが Kerberos を使用してまたは使用せずに接続するときに、適切な ID があることを確認します。

13.26 バージョン 5.2

発売日：2021-04-22

このリリースには、pgAdmin45.1 のリリース以降の多数のバグ修正と新機能が含まれています。

13.26.1 新機能

13.26.2 ハウスキーピング

Issue #5319 -Server モジュールのコードカバレッジと API テストケースを改善します。

13.26.3 バグ修正

Issue #4001 -クエリツールの [Notifications] タブをカバーするようにドキュメントとスクリーンショットを更新しました。

Issue #5519 -auto-commit/auto-rollback が false に設定されている場合、サーバーの切断後にクエリツールタブを閉じる必要があることを確認します。

Issue #5908 -管理マクロでショートカットキーが機能しない問題を修正しました。

Issue #6076 -サーバーのインポート中に正しいエラーがスローされず、JSON ファイルに不正/不十分なキーがある問題を修正しました。

Issue #6082 -長いクエリの実行中にユーザーが接続を変更しないようにします。

Issue #6107 -チェック制約の入力ボックスのちらつきの問題を修正しました。

Issue #6161 -クエリツールに関連するすべてのモデルダイアログで、カーソルが間違ったウィンドウに移動する問題を修正しました。

Issue #6220 -「CREATETRIGGER」の構文を修正しました。v11 以降では、「EXECUTEPROCEDURE」の代わりに「EXECUTEFUNCTION」を使用します。

Issue #6274 -LDAP 認証モジュールの文字列が翻訳可能であることを確認します。

Issue #6293 -Volatility オプションを提供するとプロシージャの作成が失敗する問題を修正しました。

Issue #6306 -選択操作の直前に削除された行を選択する際の問題を修正しました。

Issue #6325 -ストレージマネージャーのファイル形式が「すべてのファイル」であることを確認し、他のダイアログでは最後に選択した形式を記憶する必要があります。

Issue #6327 -ドメインを比較するときに、スキーマの差分で関数の依存関係を考慮する必要があることを確認します。

Issue #6333 -新しいブラウザータブで開いたときのクエリツールと ERD ツールのヘルプダイアログのサイズの問題を修正しました。

Issue #6334 -実行時にデタッチするときの SQL パネルのブラックスクリーンの問題を修正しました。

Issue #6338 -RPM および Debian のデスクトップパッケージに欠落していた依存関係「xdg-utils」を追加しました。

Issue #6344 -パーティションを選択するときの non-iterable 応答オブジェクトエラーを修正しました。

Issue #6356 -WebDEB および RPM パッケージで ApacheHTTPD 構成ファイルをマークします。

Issue #6367 -PG9.5 でテーブルのプロパティダイアログを開くと、デフォルトで [保存] ボタンが有効になる問題を修正しました。

Issue #6375 -ユーザーがデータの表示/編集オプションを使用してパーティションのデータを表示できない問題を修正しました。

Issue #6376 -ユーザーが Explain または Explainanalyze をクリックすると接続警告が表示され、データベースサーバーがブラウザーツリーから切断される問題を修正しました。

Issue #6379 -host オプションに 2 つのホストアドレスが含まれている場合、外部データラッパープロパティが表示されない問題を修正しました。

13.27 バージョン 5.1

発売日：2021-03-25

このリリースには、pgAdmin45.0 のリリース以降の多数のバグ修正と新機能が含まれています。

13.27.1 新機能

Issue #5404 -グループロールを調べるときに表示されるグループロールのメンバーであるログインロールを表示します。

Issue #6212 -コンテナー配布を x86_64 および Arm64 サポートのマルチアーキテクチャビルドにします。

Issue #6268 -デフォルトでシステムに MITKerberos をインストールする必要を回避するために、「kerberos」を Pythonwheel のオプション機能にします.

Issue #6270 -サーバーのリストを新しくインポートしたものに置き換える「--replace」オプションをサーバーのインポートに追加しました。

Issue #6271 -実行時にキーボードショートカットを使用したズームスケーリングオプションを追加しました。

13.27.2 ハウスキーピング

Issue #3976 -カタログオブジェクトにアクセスするときにスキーマ修飾を使用します。

Issue #6176 -「SaveDataChanges」アイコンをより直感的にします。

13.27.3 バグ修正

Issue #4014 -ドイツ語の設定の下の配置の問題を修正しました。

Issue #4020 -safari ブラウザーのコレクションノードの統計タブの色の問題を修正しました。

Issue #4438 -テーブル名にパーセント記号が含まれている場合、レコードの追加/更新が失敗する問題を修正しました。

Issue #4784 -autovacuum と分析のスケールファクターが 2 桁以上の小数で編集可能であることを確認します。

Issue #4847 -クエリとテーブルの ExplainAnalysis で%が 2 回表示される問題を修正しました。

Issue #4849 -Explain 分析の統計タブでテキスト「table」の名前を「relation」に変更します。

Issue #4959 -タブを切り替えた後、コレクションノードのプロパティタブが応答しない問題を修正しました。

Issue #5073 -プロパティダイアログを開いたときに、デフォルトでファンクション/プロシージャの [保存] ボタンが有効になる問題を修正しました。

Issue #5119 -ディレクトリでシンボリックリンクをハングすると、ファイルの選択ダイアログが壊れる問題を修正しました。

Issue #5467 -Windows インストールパスでアンダースコアを許可します。

Issue #5628 -UAC により昇格したユーザーとして実行される可能性があるため、Windows インストーラーの/"launchnow/"オプションを削除します。

Issue #5810 -セルデータの編集時にセルのコンテンツが自動選択されるようにします。

Issue #5869 -SQL フォーマッタが余分なタブを追加せず、SQL を正しくフォーマットしないことを確認します。

Issue #6018 -データベースのエンコードが SQL_ASCII に設定され、列の名前が ASCII 文字の場合のエンコードの問題を修正しました。

Issue #6159 -ユーザに「pg_signal_backend」ロールがある場合、ダッシュボードからセッションを強制終了できることを確認します。

Issue #6206 -キーボードショートカットを使用して、サポートされていないノードのデータの表示/編集パネルを開かないでください。

Issue #6227 -コンテナ展開を初期化するときに PGADMIN_DEFAULT_EMAIL が正常に見えることを確認します。

Issue #6228 -Linux 用の Web セットアップスクリプトを改善して、プラットフォーム検出をより堅牢でオーバーライド可能にします。

Issue #6233 -「スペースを使用しますか?」の場合、SQL フォーマッタがタブサイズを使用しないことを確認する false に設定します。

Issue #6253 -接続のホスト/IP アドレスが 127.0.0.1 の場合、ユーザーがサブスクリプションを作成できない問題を修正しました。

Issue #6259 -サブスクリプションの権限が不十分な場合に、プロパティと統計タブに適切なエラーメッセージが表示されることを確認します。

Issue #6260 -同じ IP/ホストが提供されているが、ポートが異なる場合に [CreateSlot] オプションが無効になる問題を修正しました。

Issue #6269 -ランタイムが使用する Python インタープリターがユーザーサイトパッケージを無視することを確認します。

Issue #6272 -SQL ファイルを開いたときにユーザーがクエリツールで接続を変更できない問題を修正しました。

Issue #6279 -Linux で venv アクティベーションスクリプトに正しいパスがあることを確認します。

Issue #6281 -異なる when 句でトリガーを比較すると、スキーマの差分が間違った SQL を表示する問題を修正しました。

Issue #6286 -データベースの作成中にテンプレートデータベースが表示されることを確認します。

Issue #6292 -依存タブにフォーカスがあり、パブリケーションまたはテーブルを選択する場合の文字列インデックスが範囲外になるエラーを修正しました。

Issue #6294 -ログイン/グループロールを選択すると依存タブがエラーをスローする問題を修正しました。

Issue #6307 -パブリケーションの依存タブに誤った値が表示される問題を修正しました。

Issue #6312 -ビューデータの行をコピー/貼り付けすると、ブール型に間違った値が貼り付けられる問題を修正しました。

Issue #6316 -主キーがテーブルダイアログに適切に表示されることを確認します。

Issue #6317 -ほとんどのスクリーンリーダーがトグルボタンにアクセスできることを確認します。

Issue #6322 -最小画面解像度でフルスクリーンに入るとトップメニューが消える問題を修正しました。

Issue #6323 -テーブルダイアログのセキュリティタブに付与者名が適切に表示されることを確認します。

13.28 バージョン 5.0

発売日：2021-02-25

このリリースには、pgAdmin44.30 のリリース以降の多数のバグ修正と新機能が含まれています。

13.28.1 新機能

Issue #5091 -統計、依存関係、依存タブを閉じることができ、ユーザーは「パネルの追加」オプションを使用して戻すことができます。

Issue #5912 -論理レプリケーションのサポートを追加しました。

Issue #5967 -NWjs を使用して、システムトレイと Web ブラウザではなくスタンドアロンウィンドウで pgAdmin4 を開くランタイムを実装しました.

Issue #6148 -メニュー項目とヘルプ記事のクイック検索機能を追加しました。

Issue #6153 -SchemaDiff にパブリケーションとサブスクリプションのサポートを追加しました。

Issue #6170 -ログがコンテナに保存されず、コンソールにのみ送信されることを確認します。

13.28.2 ハウスキーピング

Issue #5017 -pgAdmin4 のデフォルトの実稼働サーバーとして cheroot を使用します。

Issue #6145 -論理レプリケーションのドキュメント。

Issue #6195 -NWjs でのランタイムのドキュメント.

Issue #6196 -クイック検索サポートのドキュメント。

Issue #6207 -JS の依存関係を最新に更新しました。

13.28.3 バグ修正

Issue #5809 -エラーダイアログを閉じた後、フィルターテキストエディターにフォーカスが適切に設定されない問題を修正しました。

Issue #5871 -サービスとユーザー名の両方が指定された場合、ユーザー名が「サーバーに接続」ポップアップに表示されるようにします。

Issue #6045 -スキーマ名にエスケープ文字が含まれている場合に提案が表示されないオートコンプリートの問題を修正しました。

Issue #6087 -依存関係タブに、共有の依存関係を持つオブジェクトの複数の所有者が表示される問題を修正しました。

Issue #6117 -ユーザーがセキュリティタブから列レベルの権限を更新できない問題を修正しました。

Issue #6143 -ユーザーが削除された場合、SQLite データベースから共有サーバーのエントリが削除されない問題を修正しました。

Issue #6157 -スクロール後に削除のために選択された行の取り消し線が表示されない問題を修正しました。

Issue #6163 -ダイアグラムがキャンバスより大きい場合にのみズームしてフィットするボタンが機能する問題を修正しました。

Issue #6164 -ERD でズームアウトしすぎた場合に図が完全に消えないようにします。

Issue #6177 -Safari および Firefox で ERD 画像をダウンロードする際の問題を修正しました。

Issue #6178 -ユーザーがサーバーの背景色を変更できない問題を修正しました。

Issue #6179 -ERD ツールに GenerateSQL が 2 回表示される問題を修正しました。

Issue #6180 -ERD の「DownloadImage」オプションの欠落しているドキュメントを更新しました。

Issue #6187 -アップグレードチェックを 1 日に 1 回実行するように制限します。

Issue #6193 -新しいタブで開いた場合、未保存の変更を閉じる前に ERD が警告をスローするようにします。

Issue #6197 -ERD イメージが適切にダウンロードされない問題を修正しました。

Issue #6201 -サブスクリプションを作成するための SSL サポートを追加しました。

Issue #6208 -ログレベルが DEBUG に設定されているときにユーティリティ（バックアップ、メンテナンスなど）ジョブが失敗する問題を修正しました。

Issue #6230 -ユーザーがサブスクリプションを作成できない問題を修正しました。

Issue #6250 -DEB/RPM パッケージが互いに同じバージョンに依存していることを確認します。

13.29 バージョン 4.30

発売日：2021-01-28

このリリースには、pgAdmin44.29 のリリース以降の多数のバグ修正と新機能が含まれています。

13.29.1 新機能

Issue #1802 -基本的なテーブルフィールド、主キー、外部キー、および DDLSQL 生成による ERD ダイアグラムのサポートを追加しました。

Issue #5457 -SPNEGO を使用してブラウザーで Kerberos チケットを転送する、Kerberos 認証のサポートが追加されました。

Issue #6147 -Kerberos サポートのドキュメント。

Issue #6152 -ERD 図サポートのドキュメント.

Issue #6160 -Postfix サーバーを無効にするコンテナオプション（PGADMIN_DISABLE_POSTFIX）を追加します。

13.29.2 ハウスキーピング

Issue #5338 -pgAgent のコードカバレッジと API テストケースを改善します。

Issue #6052 -ログファイルに接続された pgAdmin ユーザと接続名を追加しました。

Issue #6079 -「text/javascript」が廃止されたため、mimetype を「text/javascript」から「application/javascript」に更新しました。

Issue #6162 -コンテナに PostgreSQL13 ユーティリティを含めます。

13.29.3 バグ修正

Issue #5282 -パーティションサブテーブルに「CountRows」オプションを追加しました。

Issue #5488 -指定されたノードをクリックすると、ツールチップの代わりにポップアップを表示することにより、説明計画の詳細を改善します。

Issue #5571 -除外制約での式のサポートを追加しました。

Issue #5829 -AUTHENTICATION_SOURCES の不正なログ情報を修正しました。

Issue #5875 -pg_upgrade 後に「template1」データベースが表示されないことを確認します。

Issue #5905 -マクロで [保存] ボタンがデフォルトで有効になっている問題を修正しました。

Issue #5906 -すべてのマクロをクリアするときに、[ManageMacros] メニューの後の余分な行を削除します。

Issue #5907 -使用可能な既存のマクロがなく、ユーザーが値を指定しない場合、「ClearAllRows」が機能しないことを確認します。

Issue #5929 -ユーザーが無効な SQL でマクロを作成すると、サーバーが切断されたというエラーメッセージが表示される問題を修正しました。

Issue #5965 -マクロクエリの結果が適切にダウンロードされることを確認します。

Issue #5973 -Login/GroupRole のアカウントパスワードの有効期限をユーザーに知らせるための適切なヘルプメッセージとプレースホルダーを追加しました。

Issue #5997 -Flask-BabelEx を最新に更新しました。

Issue #6046 -[ファイルの保存] アイコンの状態がダーティエディターのインジケーターと一致しない問題を修正しました。

Issue #6047 -すべての変更が元に戻されても、ダーティインジケーターがアクティブなままになる問題を修正しました。

Issue #6058 -クエリツールで SQL ファイルを開くときに、名前変更パネルを無効にする必要があることを確認します。

Issue #6061 -ビューの結合を囲む余分な括弧の問題を修正しました。

Issue #6065 -スキーマ差分モジュールのアクセシビリティの問題を修正しました。

Issue #6069 -クエリツールでファイルを更新する際の問題を修正しました。

Issue #6075 -管理者以外のユーザーがサービスを使用して作成された共有サーバーを表示できない問題を修正しました。

Issue #6077 -さまざまなダイアログのアクセシビリティの問題を修正しました。

Issue #6084 -同一のオブジェクトを選択したときのスキーマ差分の TypeError 例外を修正しました。

Issue #6096 -展開ドキュメントを更新しました。Gunicorn が参照された uWSGI を正しく参照します。

Issue #6098 -ユーザーが複数のレコードを削除しようとしたときにレコードが削除される問題を修正しました。

Issue #6120 -ユーザーがロール/ユーザーのアカウントの有効期限に古い日付を指定できることを確認します。

Issue #6121 -新しい接続ウィンドウにデータベースリストが表示されない問題を修正しました。

Issue #6122 -スキーマ diff に違いがない場合の情報メッセージを追加しました。

Issue #6128 -シーケンスが作成されない問題を修正しました。

Issue #6140 -ユーティリティ（バックアップ、メンテナンス）ジョブで詳細ログが表示されることを確認します。

Issue #6144 -スキーマ差分を使用して比較するときにシーケンスの現在の値を無視する必要があることを

確認します。

13.30 バージョン 4.29

発売日：2020-12-10

このリリースには、pgAdmin44.28 のリリース以降の多くのバグ修正と新機能が含まれています。

13.30.1 新機能

13.30.2 ハウスキーピング

Issue #5328 -外部テーブルのコードカバレッジと API テストケースを改善します。

Issue #5337 -ビューとマテリアライズドビューのコードカバレッジと API テストケースを改善します。

Issue #5343 -デバッガーのコードカバレッジと API テストケースを改善します。

Issue #6062 -コードカバレッジがクラスと関数の宣言をカバーすることを確認します。

13.30.3 バグ修正

Issue #5886 -AutoFKIndex が true に設定された外部キーが既に存在する場合、テーブルダイアログから新しい外部キーを追加するときに false のエラーが表示される問題を修正しました。

Issue #5943 -StorageManager でフォルダーの名前変更が適切に機能することを確認します。

Issue #5974 -新しいブラウザータブで開いたときにデバッガーのカスタムタブのタイトルが適用されない問題を修正しました。

Issue #5978 -動的タブのタイトルがデバッガーパネルに初めて適用されない問題を修正しました。

Issue #5982 -JSON が有効でないドキュメントの問題を修正しました。

Issue #5983 -新しい接続ダイアログに、サーバーの種類に基づいて適切なサーバーアイコンを追加しました。

Issue #5985 -プロセスウォッチャーダイアログが既に削除されたデータベースサーバーに対してエラーをスローする問題を修正しました。

Issue #5991 -タブの名前を変更するときにダーティインジケータ（*）が表示されないようにします。

Issue #5992 -サーバー/データベース名に特殊文字が含まれる場合にエスケープ文字が表示される問題を修正しました。

Issue #5998 -ソーススキーマに RLS のテーブルがある場合、スキーマの差分に比較の結果が表示されない問題を修正しました。

Issue #6003 -ビュー用にトリガーを作成するときに、トリガー SQL に不正な引数が表示される問題を修正しました。

Issue #6022 -共有サーバーのインポートが失敗する問題を修正しました。

Issue #6072 -bcrypt のインポート中に失敗した DLL のロードを修正しました。

13.31 バージョン 4.28

発売日：2020-11-12

このリリースには、pgAdmin44.27 のリリース以降の多数のバグ修正と新機能が含まれています。

13.31.1 新機能

Issue #3318 -クライアントサイドでユーティリティファイルをダウンロードするサポートを追加しました。
Issue #4230 -クエリツールとデバッガーのタブのタイトルを名前変更するサポートを追加しました。
Issue #4231 -動的タブサイズのサポートを追加しました。
Issue #4232 -クエリツール、データの表示/編集、およびデバッガーのタブタイトルプレースホルダーを追加しました。
Issue #5891 -schemadiff でスキーマとデータベースを比較するサポートが追加されました。

13.31.2 ハウスキーピング

Issue #5938 -StorageManager のドキュメント。

13.31.3 バグ修正

Issue #4639 -編集モードのトリガーに対して一部のフィールドを無効にする必要があることを確認します。
Issue #5736 -検証エラーメッセージが 2 回表示される問題を修正しました。
Issue #5760 -非スーパーユーザーが関数をデバッグできることを確認します。
Issue #5842 -クエリ履歴が日付/時刻の降順でリストされるようにします。
Issue #5858 -検索オブジェクト機能が大文字と小文字を区別しない文字列で動作することを確認します。
Issue #5895 -[統計] タブにトーストテーブルサイズのサフィックスが表示されない問題を修正しました。
Issue #5911 -マクロが古いバージョンの Safari および Chrome で実行されるようにします。
Issue #5914 -関数の「Estimatedrow」の値が一致しない問題を修正しました。
Issue #5919 -セキュリティ関連の機能強化を追加しました。
Issue #5923 -閉じられないタブが閉じられる問題を修正しました。
Issue #5950 -プロセスウォッチャーダイアログに長いファイル名が表示されない問題を修正しました。
Issue #5953 -別のユーザーが提供されている場合、サーバーへの接続が待機状態になる問題を修正しました。
Issue #5959 -GrantWizard に外部テーブルを含める必要があることを確認します。

13.32 バージョン 4.27

発売日：2020-10-15

このリリースには、pgAdmin44.26 のリリース以降の多数のバグ修正と新機能が含まれています。

13.32.1 新機能

Issue #1402 -マクロのサポートを追加しました。

Issue #2519 -各トリガーノードの下にトリガー機能を表示するサポートが追加されました。

Issue #3794 -ユーザーが開いているクエリツールタブからデータベース接続を変更できるようにします。

Issue #5200 -SchemaDiff ツールでオブジェクトを比較するときに所有者を無視するサポートが追加されました。

Issue #5857 -マクロサポートのドキュメントを追加しました。

13.32.2 ハウスキーピング

Issue #5330 -関数のコードカバレッジと API テストケースを改善します。

Issue #5395 -関数の RESQL/MSQL テストケースを追加しました。

Issue #5497 -オートコンプリート機能に使用される「pgcli」の最新コードをマージしました。

13.32.3 バグ修正

Issue #4806 -実行計画が使用されておらず空の場合に役立つメッセージを追加しました。

Issue #4855 -ファイルの名前を変更するとファイル拡張子が削除される問題を修正しました。

Issue #5131 -「ctrl+a」ショートカットが SQL エディターでカーソルを移動しないことを確認します。

Issue #5417 -スキーマ差分ツールの API テストケースを修正および改善しました。

Issue #5739 -インポート/エクスポート機能が SSH トンネルで動作することを確認します。

Issue #5802 -サーバーダイアログのパスワードフィールドの最大長を削除します。

Issue #5807 -列の名前を変更してから削除すると、ドロップ SQL クエリが間違った列名を取得する問題を修正しました。

Issue #5826 -デフォルトのバキューム設定が原因でスキーマの差分が同じテーブルを異なるものとして表示する問題を修正しました。

Issue #5830 -View/MView 定義の括弧が適切に配置されないリバースエンジニアリング SQL を修正しました。

Issue #5835 -ユーザーが AutoFKIndex のオプションを変更した場合の「can'texecuteanemptyquery」メッセージを修正しました。

Issue #5839 -プロパティタブから複数の拡張機能を削除できることを確認します。

Issue #5841 -サーバーがサービスを使用して接続できない問題を修正しました。

Issue #5843 -ファンクション/プロシージャのリバースエンジニアリング SQL に「PARALLELUNSAFE」オプションがない問題を修正しました。

Issue #5845 -テーブルに主キーがない場合、クエリツールが 1000 を超える行をフェッチしない問題を修正しました。

Issue #5853 -ダークテーマでの ExplainAnalyze で「RowsX」列の値が適切に表示されない問題を修正しました。

Issue #5855 -ユーザーが既存のシーケンスの開始値を変更できることを確認します。

Issue #5861 -コンテキストメニューに [サーバーの削除] オプションが表示されることを確認します。

Issue #5867 -共有サーバーの一部のプロパティが正しく更新されない問題を修正しました。

Issue #5882 -マテリアライズドビューの依存関係を取得するときの無効なリテラルの問題を修正しました。

Issue #5885 -ユーザーがマクロ名を変更できない問題を修正しました。

13.33 バージョン 4.26

発売日：2020-09-17

このリリースには、pgAdmin44.25 のリリース以降の多数のバグ修正と新機能が含まれています。

13.33.1 新機能

Issue #2042 -クエリツールに SQL フォーマッタのサポートを追加しました。

Issue #4059 -新しいクエリツールウィンドウを開くための新しいボタンをクエリツールツールバーに追加しました。

Issue #4979 -管理ユーザーの共有サーバーサポートを追加しました。

Issue #5772 -pgAdmin がサポートするよりも古いサーバーに接続するときにユーザーに警告します。

13.33.2 ハウスキーピング

Issue #5332 -列と制約（インデックス、外部キー、チェック、除外）のコードカバレッジと API テストケースを改善します。

Issue #5344 -GrantWizard のコードカバレッジと API テストケースを改善します。

Issue #5774 -テーブルのコードカバレッジと API テストケースを改善します。

Issue #5792 -共有サーバーのサポートに関するドキュメントを追加しました。

13.33.3 バグ修正

Issue #4216 -standard_conforming_strings がオフに設定されている場合、「pg」で始まるスキーマ名がブラウザツリーに表示されるようにします。

Issue #5426 -最大スペースを使用するようにジョブステップコードブロックの高さを調整しました。

Issue #5652 -「コミット」および「ロールバック」クエリツールボタンのアイコンを変更しました。

Issue #5722 -データベースが接続されている場合でも、ユーザーがデータベースを削除できることを確認します。

Issue #5732 -いくつかのアクセシビリティの問題を修正しました。

Issue #5734 -ドキュメントの GIN および GiST インデックスの説明を更新します。

Issue #5746 ---load-server が pg_services を使用する接続のロードを許可しない問題を修正しました。

Issue #5748 -テーブル作成時の外部キーの不正なリバースエンジニアリング SQL を修正しました。

Issue #5751 -サーバーの起動に通常よりも時間がかかる場合に、「構成」および「ログの表示」メニューオプションを有効にします。

Issue #5754 -外部データラッパー、外部サーバー、およびユーザーマッピングにオプションを提供すると、スキーマの差分が機能しない問題を修正しました。

Issue #5764 -誤って生成される行レベルセキュリティの SQL を修正しました。

Issue #5765 -列が javascript オブジェクトの内部関数と同じ名前を持っている場合のクエリツールの問題を修正しました。

Issue #5766 -PostgreSQL<9.3 の文字列インデックスは整数でなければならない問題を修正しました。

Issue #5773 -アプリケーションが固定ポート構成値を無視する問題を修正しました。

Issue #5775 -「setup-web.sh」が Debian10 で動作することを確認します。

Issue #5779 -トリガー DDL ステートメントのトリガー関数から不正な引数を削除します。

Issue #5794 -グラフデータセットの無限の成長を停止することにより、過度の CPU 使用率を修正しました。

Issue #5815 -ポップアップブロッカーが原因で [スクリプトを生成] ボタンをクリックすると永久にスピナーが表示される問題を修正しました。

Issue #5816 -スキーマがターゲットデータベースに存在しない場合、生成されたスクリプトに「CREATESCHEMA」ステートメントが存在することを確認します。

Issue #5820 -リソースグループを更新する際の問題を修正しました。

Issue #5833 -システムオブジェクトの表示が false に設定されているときにカスタムシーケンスが表示されない問題を修正しました。

Issue #5834 -コンテキストメニューで [サーバーグループの削除] オプションを使用できることを確認します。

13.34 バージョン 4.25

発売日：2020-08-20

このリリースには、pgAdmin44.24 のリリース以降の多数のバグ修正と新機能が含まれています。

13.34.1 新機能

Issue #3904 -React を使用して、チャートライブラリ Flotr2 を ChartJS に置き換えます。

Issue #5126 -2 つのスキーマではなく 2 つのデータベースを比較するようにスキーマ差分ツールを変更しました。

Issue #5610 ---yes コマンドラインオプションを setup-web.sh に追加して、非対話型の使用を許可します。

13.34.2 ハウスキーピング

Issue #5324 -外部サーバーとユーザーマッピングのコードカバレッジと API テストケースを改善します。

Issue #5327 -スキーマのコードカバレッジと API テストケースを改善します。

Issue #5336 -Types のコードカバレッジと API テストケースを改善します。

Issue #5700 -古い Python2 互換性コードを削除します。

Issue #5731 -fontAwesome を v4 から v5 にアップグレードします。

13.34.3 バグ修正

Issue #3767 -SQL エディターで同じファイルを保存するときに元のファイル形式を維持する必要があります。

Issue #3791 -ビューの各列のリバースエンジニアリング SQL に欠落していたコメントを追加しました。

Issue #4123 -検索パスが「public」以外に設定されている場合、デバッガーが機能しない問題を修正しました。

Issue #4361 -ユーザーがサーバーを切断しようとすると、ssh トンネルがハングする問題を修正しました。

Issue #4387 -テーブルと列の名前に特殊文字が含まれている場合、ユーザーがデータを挿入できない問題を修正しました。

Issue #4810 -テーブルが空の場合、ユーザーが新しい行を保存できない問題を修正しました。

Issue #5429 -FTS 構成ダイアログの [辞書] ドロップダウンにすべての辞書が表示されていることを確認します。

Issue #5490 -実行時構成ダイアログを非モーダルにします。

Issue #5526 -複数行のデータを含むセルをコピーして貼り付けると、複数の行が作成される問題を修正しました。

Issue #5567 -bytea からバイナリ文字列への変換がエラーになる問題を修正しました。

Issue #5604 -ユーザーがバックアップと復元を実行するとログ全体が赤いテキストで表示される問題を修正しました。

Issue #5632 -ユーザーが Identity 列の開始値を変更できることを確認します。

Issue #5646 -RLS ポリシーノードが検索オブジェクトを使用して検索可能であることを確認します。

Issue #5664 -ユーザーがビューの列のデフォルト値を設定するときに「ALTERVIEW」ステートメントが見つからない問題を修正しました。

Issue #5670 -ユーティリティダイアログにエラーメッセージが表示されない問題を修正しました。

Issue #5689 -スキーマの差分の問題を修正するための特権タイプの'ORDERBY' 句を追加しました。

Issue #5708 -コンテナ展開ドキュメントの TLS 証明書ファイル名を修正しました。

Issue #5710 -スキーマの差分でエラーをスローするトリガーとテーブルを比較するときの問題を修正しました。

Issue #5713 -カタログの DROPSQL 構文を修正しました。

Issue #5716 -データベースサーバーを切断した後も ajax 呼び出しが発生し続ける問題を修正しました。

Issue #5724 -ドキュメントでサーバーモードで実行する場合の違いのいくつかを明確化。

Issue #5730 -適切なノードを自動的に選択してスキーマの差分の依存関係を解決し、生成されたスクリプトで順序を維持します。

13.35 バージョン 4.24

発売日：2020-07-23

このリリースには、pgAdmin44.23 のリリース以降の多数のバグ修正と新機能が含まれています。

13.35.1 新機能

Issue #5235 -アプリケーションインストールの外部にある構成ファイルをサポートします。

Issue #5484 -LDAP 接続に専用ユーザーを設定することにより、異なる DN での LDAP 認証のサポートを追加しました。

Issue #5583 -スキーマレベルの制限のサポートを追加しました。

Issue #5601 -SchemaDiff に RLS ポリシーのサポートを追加しました。

Issue #5622 -RLS ポリシーの作成中に permissive/restricted ポリシータイプのサポートを追加しました。

Issue #5650 -LDAP 匿名バインディングのサポートを追加しました。

Issue #5653 -ハイコントラストテーマのサポートを追加しました。

13.35.2 ハウスキーピング

Issue #5323 -ForeignDataWrapper のコードカバレッジと API テストケースを改善します。

Issue #5326 -ドメインとドメインの制約のコードカバレッジと API テストケースを改善します。

Issue #5329 -FTS 構成、FTS パーサー、FTS ディクショナリ、および FTS テンプレートのコードカバレッジと API テストケースを改善します。

13.35.3 バグ修正

Issue #3814 -バックアップ、復元、およびインポート/エクスポートでファイル名が空の場合にエラーメッセージが表示されない問題を修正しました。

Issue #3851 -関数、プロシージャ、およびトリガー関数を生成するときにコードに適切なインデントを追加します。

Issue #4235 -設定で「usespace==true」の場合にコンテンツが削除される、選択した行のタブインデントの問題を修正しました。

Issue #5137 -数値入力フィールドの内外にフォーカスするときの保存ボタンの有効化の問題を修正しました。

Issue #5287 -ダークテーマ関連の CSS を修正し、カラーコードを変更しました。

Issue #5414 -ランタイムで QStandardPaths::AppLocalDataLocation を使用して、ランタイムログを保存する場所を決定します。

Issue #5463 -CSV ダウンロードが数値列を引用する問題を修正しました。

Issue #5470 -ホバーの背景色がセルの編集と削除のみに設定されているバックグリッドの行のホバーの問題を修正しました。

Issue #5530 -参照されたテーブルが外部キー制約に表示されることを確認します。

Issue #5554 -ランタイムテーマをサイジングの問題のないものに置き換えます。

Issue #5569 -ストレージパラメーターが指定された場合のパーティションのリバースエンジニアリングされた SQL を修正しました。

Issue #5577 -LICENSE および DEPENDENCIES[inventory] ファイルを公式パッケージに含めます。

Issue #5621 -RLS ポリシーのリバースエンジニアリング SQL から余分な角かっこを削除します。

Issue #5629 -一部のコレクションノードが選択されているときにユーザーがプロパティを編集できる問題を修正しました。

Issue #5630 -pgadmin4 のインストールが 32 ビット Windows で機能しない問題を修正しました。

Issue #5631 -RLS ポリシーの「USING」または「WITHCHECK」オプションの値を削除するときの「cantexecuteemptyquery」の問題を修正しました。

Issue #5633 -RLS ポリシーの作成メニューがカタログオブジェクトに表示されないことを確認します。

Issue #5647 -ポリシー所有者を変更するときに差分 DDL が間違った SQL を表示する問題を修正しました。

Issue #5662 -ブラウザーウィンドウを 200%ズームし、画面の解像度が低いと、ほとんどのダイアログが適切にレンダリングされないアクセシビリティの問題を修正しました。

Issue #5666 -欠落している依存関係/依存関係を追加し、誤って識別されたいくつかを修正しました。

Issue #5673 -データベースがスキーマ差分で接続されていない場合にスキーマを取得するとエラーがスローされる問題を修正しました。

Issue #5675 -safari ブラウザーの iframe で pgAdmin が開いたときの CSRF エラーを修正しました。

Issue #5677 -ダークテーマの ExplainAnalysis のテキストの色の問題を修正しました。

Issue #5686 -ポリシーがスペースで作成された場合、ユーザーがポリシーを更新できない問題を修正しました。

13.36 バージョン 4.23

発売日：2020-06-25

このリリースには、pgAdmin44.22 のリリース以降の多数のバグ修正と新機能が含まれています。

13.36.1 新機能

Issue #5468 -スキーマ差分のオブジェクトを比較するときに空白を無視するオプションを追加しました。

Issue #5500 -スキーマ差分でサーバーを選択中にサーバーグループ名を追加しました。

Issue #5516 -行セキュリティポリシーのサポートを追加しました。

Issue #5576 -storage と log のディレクトリを作成できない場合のエラーメッセージを改善します。

13.36.2 ハウスキーピング

Issue #5325 -照合順序のコードカバレッジと API テストケースを改善します。

Issue #5574 -Windows ビルドスクリプトをクリーンアップし、Windowsx64 ビルドが動作することを確認します。

Issue #5581 -行レベルのセキュリティポリシーのドキュメント。

13.36.3 バグ修正

Issue #3591 -アクティブなセッションの終了中にクエリツールが適切なエラーメッセージを表示することを確認します。

Issue #3669 -削除されたノードに適切なエラーが表示されることを確認します。

Issue #3787 -クリックした後に [プロセスの停止] ボタンを無効にし、ユーザーに通知するメッセージ「プロセスを終了しています...」を追加しました。

Issue #4226 -すべて選択チェックボックスが最初の 50 個のテーブルのみを選択する問題を修正しました。

Issue #5416 -[Don'tSave] ボタンをクリックしたときにクエリツールパネルが閉じられることを確認します。

Issue #5465 -Edge ブラウザーのバージョンが間違って表示され、警告メッセージが表示される問題を修正しました。

Issue #5492 -検索オブジェクトが継承されたテーブルを見つけられず、制約フィルターが機能しない問題を修正しました。

Issue #5507 -データベースサーバーをアップグレードするときの接続とバージョン番号の検出の問題を修正しました。

Issue #5521 -オプション「--sqlite-path」を提供して、デスクトップ pgAdmin アプリからサーバーをダンプするときの問題を修正しました。

Issue #5539 -例外キーワードのタイプミスを修正しました。

Issue #5584 -2 つの同一のテーブルがスキーマ差分ツールによって異なる問題を修正しました。

Issue #5592 -pgadmin が 1000 文字を超えるパスワードを持つサーバーに接続できることを確認します。

13.37 バージョン 4.22

発売日：2020-05-28

このリリースには、pgAdmin44.21 のリリース以降の多数のバグ修正と新機能が含まれています。

13.37.1 新機能

Issue #5452 -ログファイルに接続された pgAdmin ユーザと接続名を追加しました。

Issue #5489 -ランタイムのログビューアにサーバーログと起動ログを表示します。

13.37.2 ハウスキーピング

Issue #5255 -SeleniumGrid を実装して、さまざまなブラウザー、オペレーティングシステム、およびマシンでマルチプルのテストを実行します。

Issue #5333 -インデックスのコードカバレッジと API テストケースを改善します。

Issue #5334 -Rules モジュールのコードカバレッジと API テストケースを改善します。

Issue #5335 -トリガーと複合トリガーのコードカバレッジと API テストケースを改善します。

Issue #5443 -Python2 のサポートを削除します。

Issue #5444 -ランタイムプロジェクトファイルの Python 検出をクリーンアップします。

Issue #5455 -pgAdmin4.py をリファクタリングして、インポートして読みやすくします。

Issue #5493 -検索オブジェクトの UI を改善しました。

Issue #5525 -macOS ビルドスクリプトをクリーンアップしてリファクタリングします。

Issue #5552 -Docker コンテナの依存関係を更新します。

Issue #5553 -現在サポートが終了しているため、Docker コンテナから PG9.4 ユーティリティを削除します。

13.37.3 バグ修正

Issue #3694 -ユーザーが/"ConnectDatabase.../"をクリックしたときに、データベースが既に接続されていることを適切に通知しました。

Issue #4033 -ログインページのアラートボックスの×ボタンをクリックしても機能しない問題を修正しました。

Issue #4099 -EDBPostgresAdvancedServer の SQL ヘルプの問題を修正しました。

Issue #4223 -マテリアライズドビューのインデックスに対してメンテナンスジョブが適切に機能するようにします。

Issue #4279 -ファイルの参照/"home/"ボタンが/ではなく$HOME を指していることを確認してください。

Issue #4840 -データベースサーバーバージョン 12 以降のバックアップ中は、「WithOID」オプションを無効にする必要があります。

Issue #5001 -既存のロールの接続制限を削除するときの無効なリテラルの問題を修正しました。

Issue #5052 -パーティションの [トリガー]->[すべてを有効にする] をクリックしたときの内部サーバーエラーを修正しました。

Issue #5398 -自動バキュームオプションで生成される SQL の問題を修正しました。

Issue #5422 -依存関係タブに類義語の正しい情報が表示されていることを確認します。

Issue #5434 -新しく追加されたテーブルがアルファベット順にツリーに追加されない問題を修正しました。

Issue #5440 -スキーマ差分ツールでのリストの並べ替えの問題を修正しました。

Issue #5449 -スキーマ差分ツールを使用して 2 つの同一のスキーマを比較する際の問題を修正しました。

Issue #5450 -列の名前を変更すると適切な順序で追加されない問題を修正しました。

Issue #5466 -FrankLimpert ごとに、コンテナドキュメントの ipv4/"allinterfaces/"アドレスを修正。

Issue #5469 -サーバー作成ダイアログの SSL フィールドで select2 のホバーが一貫していない問題を修正しました。

Issue #5473 -デフォルト以外のルートでサーバーモードで実行している場合のログイン後のリダイレクトの場所を修正しました。

Issue #5480 -PATH で使用可能なバージョン固有の Python バイナリのみがある場合、バックグラウンドジョブの作成が失敗する問題を修正しました。

Issue #5481 -長さのある character 型のデータを更新するときのデータの切り捨ての問題を修正しました。

Issue #5487 -LDAP_SEARCH_BASE_DN が設定されていない場合、LDAP_BASE_DN の値が考慮される問題を修正しました。

Issue #5496 -pgAgent ジョブスケジューラのすべてのオプションを選択せずに [すべて選択] ボタンをクリックする問題を修正しました。

Issue #5503 -サーバーで pl/debugger プラグインを有効にする際のドキュメントを明確化および修正。

Issue #5510 -Unicode デコードエラー「utf-8」コーデックがバイトをデコードできない問題を修正しました。

13.38 バージョン 4.21

発売日：2020-04-30

このリリースには、pgAdmin44.20 のリリース以降の多数のバグ修正と新機能が含まれています。

13.38.1 新機能

Issue #2172 -検索オブジェクト機能を追加しました。

Issue #2186 -LDAP 認証のサポートを追加しました。

Issue #4636 -ブラウザツリー内ですばやく識別できるように、ジョブステップとジョブスケジュールを無効にするアイコンを追加しました。

Issue #5181 -テーブルのパラメータ toast_tuple_target と parallel_workers のサポートを追加しました。

Issue #5263 -スキーマ差分に外部テーブルのサポートを追加しました。

Issue #5264 -パッケージ、シーケンス、およびシノニムのサポートをスキーマ差分に追加しました。

Issue #5348 -LDAP 認証サポートのドキュメント。

Issue #5353 -起動時にブラウザタブが開かれないようにするオプションを追加しました。

Issue #5399 -サポートされていない、非推奨、または不明なブラウザーが検出された場合にユーザーに警告します。

13.38.2 ハウスキーピング

Issue #4620 -プロシージャーのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4623 -pgAgent ジョブのリバースエンジニアリングおよび変更された SQL テストを追加します。

13.38.3 バグ修正

Issue #1257 -すべてのオブジェクトタイプに/"SystemXXX?/"プロパティがあることを確認します。

Issue #2813 -データベースサーバーがトラスト認証モードの場合、パスワードプロンプトが表示されないようにします。

Issue #3495 -クエリツールが BOM マーカーを含むファイルを読み込めない問題を修正しました。

Issue #3523 -ブラウザオブジェクトを右クリックしても、右クリックが発生したオブジェクトに適用されない問題を修正しました。

Issue #3645 -pgAgent ジョブの選択をクリアするときに開始日と終了日を削除する必要があることを確認します。

Issue #3900 -テーブル制約に複数のドロップ/削除機能を追加しました。

Issue #3947 -データの表示/編集での行のコピーと貼り付けの問題を修正しました。

Issue #3972 -OSX ネイティブサポートのためにクエリツールのキーボードショートカットを変更しました。

Issue #3988 -クエリエディターで一部の文字をズームアウトするとカーソルが消える問題を修正しました。

Issue #4180 -ポインタがオブジェクト上にない限りブラウザでオブジェクトを選択しないマウスクリックの問題を修正しました。

Issue #4206 -ESC キーを押したときに付与ウィザードを閉じる必要があることを確認します。

Issue #4292 -Windows/macOS ランタイムの構成ダイアログにダークモードのサポートを追加しました。

Issue #4440 -リバースエンジニアリングされた SQL の DROP ステートメントがすべてのオブジェクトに対して適切に引用されていることを確認します。

Issue #4445 -リバースエンジニアリングされた SQL のタイトル行にあるすべてのオブジェクト名が引用符で囲まれていないことを確認します。

Issue #4504 -テーブルの作成中にリレーションが選択されていない場合、同様のオプションを無効にする必要がある問題を修正しました。

Issue #4512 -pgAgent のスケジュールタブ内の例外タブでカレンダーを開く問題を修正しました。

Issue #4545 -GrantWizard 最後のオブジェクトが選択できない問題を修正しました。

Issue #4573 -区切り文字がコンマ以外に設定されていることを確認し、ファイルを「.txt」ファイルとしてダウンロードします。

Issue #4684 -'utf-8' 以外のエンコードされた文字セットでデータを保存する際のエンコードの問題を修正しました。

Issue #4709 -重複する名前の混乱を避けるために、FTS 構成にスキーマ修飾辞書名を追加しました。

Issue #4856 -CREATE または他のスクリプトでクエリツールが開かれたときに、デフォルトで保存ボタンを有効にします。

Issue #4858 -ユーザーが CSV をダウンロードしようとしたときに接続に問題がある場合の Python 例外エラーを修正しました。

Issue #4864 -実行時に構成ウィンドウを自動サイズ変更します。

Issue #4873 -引数を持つプロシージャのコメントを変更すると、オーバーロードの場合にエラーが発生する問題を修正しました。

Issue #4946 -ユーザーが「oncommitdropas」句を使用して一時テーブルを作成する場合の問題を修正しました。

Issue #4957 -ユーザーがトリガーの EDB-SPL ファンクションを選択したときに、ConstraintTrigger,Deferrable,Deferred オプションを無効にする必要があります。

Issue #4969 -JSONB または JSON 型の列の値を NULL に変更する問題を修正しました。

Issue #5007 -一意性制約の作成中に、インデックスドロップダウンに既存のインデックスがあることを確認します。

Issue #5043 -ブラウザツリーで列名が作成順に表示される問題を修正しました。

Issue #5053 -既存のビューの列を変更するとエラーがスローされる問題を修正しました。

Issue #5157 -デフォルトの並べ替え順序がデータの表示/編集でプライマリキーを使用する必要があることを確認します。

Issue #5180 -WITH 句を使用してテーブルが作成されると、RE-SQL に autovacuum_enabled パラメーターが自動的に追加される問題を修正しました。

Issue #5210 -基になるフィールドサイズより大きいテキストが自動的に切り捨てられないようにします。

Issue #5213 -ユーザーが大きなサイズのマテリアライズドビューで更新を実行するときの問題を修正しました。

Issue #5227 -多くのユーザーが既に存在する場合、ユーザーを追加できない問題を修正しました。

Issue #5268 -FTS 構成のトークンまたは FTS 辞書のオプションが変更されたときに生成される SQL を修正しました。

Issue #5270 -シノニムのプロパティに OID が表示されることを確認します。

Issue #5275 -テーブルダイアログのパラメーターのタブキーのナビゲーションの問題を修正しました。

Issue #5302 -タイプのスキーマ差分ツールに差分 SQL が表示されない問題を修正しました。

Issue #5314 -スイッチセルがアクセシビリティのスイッチ制御と同期していることを確認します。

Issue #5315 -スキーマの差分が同一のドメイン制約の変更を示す問題を修正しました。

Issue #5350 -スキーマの差分が同一のテーブルを異なるものとしてマークする問題を修正しました。

Issue #5351 -pgAdmin のビルド中のコンパイル警告を修正しました。

Issue #5352 -Explain クエリプランの右端と下部のツールチップのトリミングの問題を修正しました。

Issue #5356 -pgAgent ジョブスケジュールに例外を追加する際の変更された SQL の問題を修正しました。

Issue #5361 -pgAdmin4GUI が IE11 で適切に表示されない問題を修正します。

Issue #5362 -スキーマ差分ツールで同一のパッケージとシーケンスが異なって表示される問題を修正しました。

Issue #5366 -クエリツールからのいずれかのパネルが読み込めなかった場合に、レイアウトをリセットにアラートメッセージを追加しました。

Issue #5371 -一部のダイアログのタブキーのナビゲーションを修正しました。

Issue #5375 -引数グリッドのモードセルが関数ダイアログに完全に表示されない問題を修正しました。

Issue #5383 -既存の類義語を更新する際の構文エラーを修正しました。

Issue #5387 -すべての引数が/"IN/"引数の場合、ファンクション/プロシージャのプロパティダイアログにモードが表示されない問題を修正しました。

Issue #5396 -オブジェクト検索モジュールがブラウザツリーでオブジェクトを見つけられない問題を修正しました。

Issue #5400 -データベースサーバーがスーパーユーザー以外でログインしている場合の内部サーバーエラーを修正しました。

Issue #5401 -オブジェクト名に特殊文字が含まれる場合の検索オブジェクトの問題を修正しました。

Issue #5402 -ダークテーマの実行時に構成ダイアログにチェックボックスが表示されない問題を修正しました。

Issue #5409 -シノニムノードの検証の問題を修正しました。

Issue #5410 -間違った変更された SQL を表示するパッケージ本体を削除する際の問題を修正しました。

Issue #5415 -クエリツールのコンテキストメニューがコレクションノードで動作することを確認します。

Issue #5419 -ユーザーが認証ソースを変更できないことを確認します。

Issue #5420 -LDAP ユーザーが同じ名前で作成された場合にエラーが適切に処理されるようにします。

Issue #5430 -ログインページにタイトルを追加しました。

Issue #5432 -認証ソースが internal および ldap に設定されている場合、内部ユーザーが作成されない問題を修正しました。

Issue #5439 -LDAP アカウントでログインすると、ユーザーがサーバーを作成できない問題を修正しました。

Issue #5441 -検索オブジェクトが pg_toast スキーマで pg_toast_*テーブルを見つけられない問題を修正しました。

Issue #5447 -更新（任意のオプション）マテリアライズドビューをクリックしたときのユーティリティエラーを取得できない問題を修正しました。

13.39 バージョン 4.20

発売日：2020-04-02

このリリースには、pgAdmin44.19 のリリース以降の多数のバグ修正と新機能が含まれています。

13.39.1 新機能

Issue #5261 -照合、FTS 構成、FTS ディクショナリ、FTS パーサー、および FTS テンプレートのサポートをスキーマ差分に追加しました。

Issue #5262 -スキーマ差分にドメイン、ドメイン制約、およびタイプのサポートを追加しました。

13.39.2 ハウスキーピング

Issue #5271 -明るいテーマと暗いテーマの両方のスイッチコントロールの色を強化します。

Issue #5284 -翻訳カバレッジを向上させるために gettext の使用を追加および修正しました。

13.39.3 バグ修正

Issue #4237 -ユーザーがキーボードを使用して DateTime ピッカーコントロールの値を変更できない問題を修正します。

Issue #4608 -ダイアログのアクセシビリティの問題を修正しました。

Issue #4942 -Ubuntu の Chrome ドライバーダウンロードユーティリティの問題を修正しました。

Issue #5128 -WCAG のカラーコントラスト基準に準拠するように、いくつかの色と不透明度を変更します。

Issue #5143 -アクセシビリティの問題を修正して、すべてのアラートダイアログのパネルを最大化します。

Issue #5221 -比較の一部として DDL ステートメントを取得するロジックを改善します。

Issue #5241 -GrantWizard のタブキーナビゲーションの問題を修正しました。

Issue #5279 -Python2 環境でエラーの原因となる Unicode 文字の問題を修正しました。

Issue #5292 -Alertify ダイアログボタンのフォーカスカラーの問題を修正しました。

13.40 バージョン 4.19

発売日：2020-03-05

このリリースには、pgAdmin44.18 のリリース以降の多数のバグ修正と新機能が含まれています。

13.40.1 新機能

Issue #5154 -AlertifyJS にアクセシビリティサポートを追加しました。

Issue #5170 -チェコ語のサポートを追加しました。

Issue #5179 -Python3.8 のサポートを追加しました。

13.40.2 ハウスキーピング

Issue #5088 -EventTrigger モジュールのコードカバレッジと API テストケースを改善します。

Issue #5133 -デフォルトテーマとダークテーマの両方の UI の改善。

Issue #5176 -ログレベルが DEBUG に設定されているときにサブプロセスの stdout と stderr を追跡することにより、ロギングを強化します。

Issue #5185 -select2 コントロールのラベルタグのクラス名をオーバーライドするオプションを追加しました。

13.40.3 バグ修正

Issue #4955 -Select2 ドロップダウンの選択されてホバーされたアイテムの色を変更しました。

Issue #4996 -ダークモードでのクエリ実行後の強調表示されたコードのスタイルを改善します。

Issue #5058 -AlertifyJS がアラートダイアログのタイトルとして表示されないようにします。

Issue #5077 -ジオメトリビューアの背景パターンをすべてのテーマで#fff を使用するように変更しました。

Issue #5101 -デバッグ中に SQL エラーが発生した後、デバッガーがすべての引数を表示しない問題を修正します。

Issue #5107 -ファイルマネージャーダイアログのタブナビゲーションに適切なフォーカスを設定。

Issue #5115 -ルールのコマンドとステートメントが誤って解析される問題を修正します。

Issue #5142 -pgAdmin を使用してサーバーが切断されたときに、接続を閉じる前にすべてのトランザクションをキャンセルする必要があることを確認します。

Issue #5184 -フォントを最新バージョンに更新して、Firefox の等幅の問題を修正しました。

Issue #5214 -Windows で Python3.8 が動作していない Flask-SQLAlchemy および SQLAlchemy パッケージを更新します。

Issue #5215 -既存のルールのイベントタイプを変更するときの構文エラーを修正しました。

13.41 バージョン 4.18

発売日：2020-02-06

このリリースには、pgAdmin44.17 のリリース以降の多数のバグ修正と新機能が含まれています。

13.41.1 新機能

Issue #2554 -マルチレベルパーティションテーブルのサポートを追加しました。

Issue #3452 -2 つのスキーマを比較して差分スクリプトを生成する SchemaDiff ツールを追加しました。

Issue #4762 -スクリーンリーダーが非テキスト可能要素のラベルと説明を読み取れるようにします。

Issue #4763 -スクリーンリーダーがアラートエラーを識別できるようにします。

Issue #4770 -アクセシビリティのためにすべての pgAdmin4Web ページを解析および検証した後、ラベルとタイトルを追加しました。

Issue #4993 -入力コントロールを無効ではなく読み取り専用に設定すると、プロパティタブでのタブナビゲーションが許可され、スクリーンリーダーで読み取ることもできます。

13.41.2 ハウスキーピング

Issue #5049 -CAST モジュールのコードカバレッジと API テストケースを改善します。

Issue #5050 -LANGUAGE モジュールのコードカバレッジと API テストケースを改善します。

Issue #5071 -単一のファイルで定義された複数のクラスに対して実行するようにテストフレームワークを改善します。

Issue #5072 -aria-label アクセシビリティの改善を含む wcDocker パッケージを更新しました。

Issue #5096 -SCSS コンパイルのために node-sass を sass に置き換えます.

13.41.3 バグ修正

Issue #3812 -ファイルエクスプローラーダイアログのドロップダウンから ext を変更するときにパスファイル名が消えないことを確認します。

Issue #4410 -データの表示/編集から char[] または charactervary[] 列を編集するとエラーが発生する問題を修正しました。

Issue #4511 -クエリ文字列パラメーターが制限を超えると、Grant ウィザードがマルチプルのオブジェクトを処理できない問題を修正しました。

Issue #4601 -すべてのパネルと作成/プロパティダイアログの閉じるボタンにタブナビゲーションを追加しました。

Issue #4827 -ファイルエクスプローラーダイアログの列のサイズ変更可能な問題を修正しました。

Issue #5000 -マウスの移動、クリック、またはキー入力のユーザアクティビティがないときに pgAdmin セッションをログアウトします。

Issue #5025 -STORAGE_DIR を空に設定すると、Windows 上のすべてのボリュームがサーバーモードで表示される問題を修正します。

Issue #5065 -リフレッシュ時にキャストノードに使用される誤ったアイコンを更新しました。

Issue #5066 -パッケージを更新するとオブジェクトが完全に変更される問題を修正します。

Issue #5074 -テーブルでスクリプトを選択、挿入、および更新するとエラーが発生する問題を修正します。

Issue #5076 -Postfix がコンテナで起動することを確認します。デフォルトで非 root として実行されるようになりました。

Issue #5116 -サーバーの作成中にパスワードの保存コントロールをクリックすると消える問題を修正しました。

13.42 バージョン 4.17

発売日：2020-01-09

このリリースには、pgAdmin44.16 のリリース以降の多数のバグ修正と新機能が含まれています。

13.42.1 新機能

Issue #4764 -スクリーンリーダーは、ネストされた要素のリレーションシップ属性を読み取ることができます。

Issue #5060 -すべてのバイナリが安全に署名され、macOS バンドルの強化されたランタイムにリンクされていることを確認します

13.42.2 ハウスキーピング

Issue #4988 -テーブルとその子ノードのリファクタリングされた SQL.

Issue #5023 -ビューとマテリアライズドビューのリファクタリングされた SQL.

Issue #5024 -関数とプロシージャーのリファクタリングされた SQL.

Issue #5038 -Select2 でのアイテムのオンデマンドロードのサポートを追加しました。

Issue #5048 -pgAdmin 用のコードカバレッジツールを追加しました。

13.42.3 バグ修正

Issue #4198 -コードミラーでのバックスラッシュとエスケープ定数のシンタックスハイライトを修正しました。

Issue #4506 -FILLFACTOR やコメントなどの空のテキストボックスをクリックすると、変更とみなされて保存ボタンが有効になる問題を修正します。

Issue #4633 -マルチレベルパーティションテーブルを表示するサポートを追加しました。

Issue #4842 -制約、インデックス、ルール、トリガー、および複合トリガーをパーティションに作成する必要があることを確認します。

Issue #4943 -「データベースが接続されました/切断されました」メッセージに情報を追加しました。

Issue #4950 -ユーザーが v12 以降のパーティションテーブルのテーブルスペースを選択/変更できることを確認します。

Issue #4999 -Kubernetes と競合する可能性のあるいくつかの内部環境変数の名前を変更します。

Issue #5004 -「yarnaudit」によって報告された脆弱性の問題を修正します。非推奨の uglifyjs-webpack-plugin を terser-webpack-plugin に置き換えます。

Issue #5008 -インデックスの作成中にテーブルスペースが選択されていない場合にエラーメッセージが表示されないようにします。

Issue #5009 -オペレーター、アクセスメソッド、およびオペレータークラスが除外制約で表示されない問題を修正します。

Issue #5013 -コンテナの実行時に拡張属性をサポートしていないファイルシステムでの非特権ポートの使用に関するメモをドキュメントに追加します。

Issue #5047 -クエリツールの説明パネルの下のタブにタブナビゲーションを追加しました。

Issue #5068 -PG/EPAS12 の autovacuum_enabled および toast.autovacuum_enabled でテーブルが作成されない問題を修正します。

13.43 バージョン 4.16

発売日：2019-12-12

このリリースには、pgAdmin44.15 のリリース以降の多数のバグ修正と新機能が含まれています。

警告: 警告：このリリースには、非 root ユーザーとして pgAdmin を実行するためのコンテナディストリビューションへの変更が含まれています。マップされたストレージディレクトリで実行しているコンテナのユーザーは、ホストマシンで所有権を変更する必要があります。

```
sudo chown -R 5050:5050 <host_directory>
```

13.43.1 新機能

Issue #4396 -マテリアライズドビューの定義を変更するときに、データとその依存オブジェクトの損失について警告します。

Issue #4435 -コレクションノードを除く、データベースノードの下のすべてのノードでドラッグアンドドロップ機能を許可します。

Issue #4711 -他のアプリケーションとの一貫性を高めるために、クエリツールの [ExecuteQuery] ボタンに「play」アイコンを使用します。

Issue #4772 -可視ラベルを使用できない場所で不可視ラベルを提供するために aria-label を追加しました。

Issue #4773 -アクセシビリティのためのすべてのステータスメッセージに role=/"status/"属性を追加しました。

Issue #4939 -コンテナで pgAdmin を非 root ユーザーとして実行します（pgadmin、UID：5050）

Issue #4944 -コンテナ内の Gunicorn ログを GUNICORN_ACCESS_LOGFILE で指定されたファイルに転送できます。

Issue #4990 -開いているクエリツールとデータフィルターのアイコンを変更しました。

13.43.2 ハウスキーピング

Issue #4696 -マテリアライズドビューのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4807 -テーブルとその子ノードのリファクタリングされたコード。

Issue #4938 -列ノードのリファクタリングされたコード。

13.43.3 バグ修正

Issue #3538 -ダイアログですべての必須フィールドが提供されるまでリセットボタンが有効にならない問題を修正します。

Issue #4220 -「ユーザー」ダイアログのスクロールの問題を修正しました。

Issue #4516 -ラベルのないテーブルヘッダーの並べ替えを削除します。

Issue #4659 -付与者についてより明確にするために、デフォルトの権限のドキュメントを更新しました。

Issue #4674 -ユーザー名に HTML 文字が含まれている場合のクエリツールの起動エラーを修正しました。退行です。

Issue #4724 -SSH トンネルを介して接続を確立しているときにネットワークが切断され、サーバーノードを展開できない問題を修正しました。

Issue #4761 -データ型をタイムゾーン付きのタイムスタンプからタイムゾーンなしのタイムスタンプに変更すると、間違った型が表示される問題を修正します。

Issue #4792 -スーパーユーザーはすべてのアクセス制限をオーバーライドするため、スーパーユーザーがデータベースを作成できることを確認します。

Issue #4818 -クエリツールでサーバー接続がドロップアウトする問題を修正しました。

Issue #4836 -ドキュメントのコンテナ展開セクションで json ファイル名を「servers.json」から「pgadmin4/servers.json」に更新しました。

Issue #4878 -スーパーユーザーはすべてのアクセス制限をオーバーライドするため、スーパーユーザーがロールを作成できることを確認します。

Issue #4893 -コメントとして数字を指定するときのパーティションのリバースエンジニアリング SQL の問題を修正します。

Issue #4923 -サーバーとサーバーグループノードのラベルを「Delete/Drop」から「Remove」に変更するロジックを拡張します。

Issue #4925 -初期ログがロードされるまでプロセスウォッチャーにテキストを表示しました。

Issue #4926 -pgAdmin4 がサーバーノードの展開でハングする VPN ネットワークの切断の問題を修正します。

Issue #4930 -メインウィンドウのタブナビゲーションのアクセシビリティの問題を修正しました。

Issue #4933 -サーバーコレクションノードがサーバー接続とは無関係に展開されることを確認します。

Issue #4934 -[UserManagement] ダイアログのヘルプボタンのリンクを修正しました。

Issue #4935 -アクセシビリティの問題を修正します。

Issue #4947 -HTML を含むテーブルとタイプの Explain と Explainanalyze の XSS の問題を修正しました。

Issue #4952 -複合トリガーのプロパティを取得する問題を修正します。#4006 のリグレッションです。

Issue #4953 -トリガーが既に無効になっているときにユーザーが [すべて有効にする] をクリックした場合、pgAdmin4 がテーブルノードを取得できない問題を修正します。

Issue #4958 -トリガー関数に単一の引数が渡されたときのトリガーのリバースエンジニアリング SQL の問題を修正しました。

Issue #4964 -長さと精度がテーブル/列のダイアログから削除されない問題を修正します。

Issue #4965 -テーブル/列のプロパティダイアログに Interval データ型が表示されない問題を修正します。

Issue #4966 -「サーバー上でオブジェクトが見つかりませんでした」を修正チェック制約の更新中にエラーが発生しました。

Issue #4975 -ユーザーが UI 言語を切り替えることができない問題を修正します。#4348 のリグレッションです。

Issue #4976 -PG/EPAS12 で when 句が表示されない SQL のリバースエンジニアリングの問題を修正します。

Issue #4978 -バージョン 4.15 にアップグレードした後に pgAdmin4 が開始できない問題を修正しました。

Issue #4982 -テーブル/列の SQL のリバースエンジニアリングに統計とストレージ情報を追加しました。

Issue #4985 -引用符で囲まれた継承されたテーブル名が正しくエスケープされなかった問題を修正します。

Issue #4991 -コンテキストメニューがサブメニューとともに開かれ、フォーカスがコンテキストメニューまたはサブメニューにない問題を修正します。

13.44 バージョン 4.15

発売日：2019-11-14

このリリースには、pgAdmin44.14 のリリース以降の多数のバグ修正と新機能が含まれています。

13.44.1 新機能

Issue #1974 -ロールのリバースエンジニアリングされた SQL に暗号化されたパスワードを追加しました。

Issue #3741 -Dark(Beta)UITheme オプションを追加しました。

Issue #4006 -トリガーで EnableAlways と EnableReplica をサポートします。

Issue #4351 -[プロパティ] ダイアログで変更をキャンセル/リセットする前に確認を要求するオプションを追加します。

Issue #4348 -カスタムテーマの作成と選択のサポートを追加しました。

13.44.2 ハウスキーピング

13.44.3 バグ修正

Issue #3130 -コレクションノードで alt+shift+n キーが押されたときに新しいオブジェクトの作成ダイアログが開かれるようにします。

Issue #3279 -ドロップと切断の接続メニューポイントが互いに近すぎる問題を修正しました。

Issue #3789 -コンテキストメニューがメニューバーの下に非表示にならないようにします。

Issue #3859 -コンテキストメニューの名前を「DropServer」から「RemoveServer」に変更します。

Issue #3913 -pgAgent ジョブの実行時に、正しい/"runningat/"エージェントが表示されていることを確認します。

Issue #3915 -クエリエラーの後にショートカットキーが無視される可能性があるクエリツールの問題を修正します。

Issue #3999 -トグルケースのショートカットキーの組み合わせを修正しました。

Issue #4173 -ブラウザーツリー画像の背景に黒い矢印のような画像が表示される問題を修正します。

Issue #4191 -テーブルパーティションのリバースエンジニアリングされた SQL にコメントが表示されるようにします。

Issue #4242 -バックグリッドテーブルをソートするときに NULL 値を適切に処理する.

Issue #4341 -ユーザーが空白のマスターパスワードを使用しようとしたときに適切なエラーメッセージを表示します。

Issue #4451 -複合型には少なくとも 2 つのメンバーが必要という任意の（および誤った）要件を削除します。

Issue #4459 -クエリツールの結果グリッドからコピーするときに bigint を引用しないでください.

Issue #4482 -ディレクトリ形式でバックアップするときに、圧縮レベルが pg_dump に渡されるようにします。

Issue #4483 ・ディレクトリ形式でバックアップする場合にジョブ数を指定できることを確認する。

Issue #4564 -クエリツールの実行中の Javascript エラーが Ajax エラーとしてではなく、レポートされるようにします。

Issue #4610 -Select2 コントロールから来たときに Alertify ダイアログで Enter キーを押して、Enter で項目を選択できるようにします。

Issue #4647 -[ファイル] ダイアログでファイルサイズで並べ替えるときに単位が尊重されるようにします。

Issue #4730 -長時間実行されるクエリからのすべてのメッセージがクエリツールに保持されるようにします。

Issue #4734 -行を削除するのではなく、取り消し線のみにする行の削除ボタンのドキュメントを更新しました。

Issue #4779 -クエリツールのツールバーボタンのドキュメントを更新しました。

Issue #4835 -メンテナンスおよびインポート/エクスポートの実行中に v12 の psql が「シンボルが見つかりません」エラーをスローする問題を修正しました。

Issue #4845 -[コード] タブのプロパティダイアログの潜在的なエラーを修正しました。

Issue #4850 -ラベルをクリックすると Datetimepicker コントロールが開く問題を修正しました。

Issue #4895 -ネストされたオブジェクトのリセット関数の潜在的な問題を修正しました。

Issue #4896 -ファイルを開く/保存するダイアログを閉じるためにエスケープキーが機能しない問題を修正しました。

Issue #4906 -CentOS7 で Firefox を使用しているときにコンテキストメニューのキーボードショートカットが機能しない問題を修正しました。

Issue #4924 -Gunicorn の最新バージョンでの変更が原因で発生する docker コンテナの終了の問題を修正しました。

13.45 バージョン 4.14

発売日：2019-10-17

このリリースには、pgAdmin44.13 のリリース以降の多数のバグ修正と新機能が含まれています。

13.45.1 新機能

Issue #3009 -クエリツール/ビューデータからデータをコピーするときにヘッダー付きコピー機能を追加しました。

Issue #4778 -クエリプランアナライザーを実装しました。

Issue #4823 -コンテナに PostgreSQL12 バイナリを含めます。

13.45.2 ハウスキーピング

Issue #4472 -シノニムのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4628 -UniqueConstraint のリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4701 -Webpack を最適化して全体的なパフォーマンスを向上させます.

13.45.3 バグ修正

Issue #3386 -パーティションテーブルのバックアップでデータベース全体をバックアップしないようにします。

Issue #4199 -データフィルターの「ENTER」キーがクエリを実行しないことを確認します。

Issue #4590 -引用符が必要なスキーマ名のバックアップが失敗する問題を修正します。

Issue #4728 -ユーザーが CodeMirror で閉じ括弧または開き括弧の色を強調表示しました。

Issue #4751 -すべての列の選択を解除するとエクスポートジョブが失敗する問題を修正します。

Issue #4753 -[パラメーター] タブに新しいパラメーターを追加したときに「false」の文字列が表示されるエラーを修正し、ユーザーがパラメーター名を変更したときに古い値もクリアします。

Issue #4755 -組み込みサーバーをそのまま使用する場合、pgAdmin がリバースプロキシの背後で動作することを確認します。

Issue #4756 -iframe にロードされた場合に pgAdmin が完全にロードされない問題を修正します。

Issue #4760 -データベースの検索パスを引用符で囲まないでください。

Issue #4768 -非標準ポートのリバースプロキシの背後で pgAdmin が動作することを確認します。

Issue #4769 -InternetExplorer でクエリツールが開く問題を修正しました。

Issue #4777 -クエリ履歴がクエリ履歴タブに表示されない問題を修正します。

Issue #4780 -ファンクション、プロシージャ、およびトリガーファンクションの検索パスを引用符で囲まないでください。

Issue #4791 -テーブルの [SQL] タブで VALID 外部キーが NOTVALID と表示される問題を修正します。

Issue #4817 -MACOSX アプリが Catalina 用に公証される必要があることを確認します。

13.46 バージョン 4.13

発売日：2019-09-19

このリリースには、pgAdmin44.12 のリリース以降の多数のバグ修正と新機能が含まれています。

13.46.1 新機能

Issue #2828 -Explain モジュールの GatherMerge、NamedTupleStoreScan、および TableFunctionScan アイコンを追加しました。

Issue #4553 -UX を向上させるために、クエリツールの UI をレンダリングする前にデータベース接続を待機しません。

Issue #4651 -コンテナディストリビューションの環境から構成オプションを設定できるようにします。

Issue #4667 -クエリツールの編集可能で読み取り専用の列は、列ヘッダーのアイコンとツールチップで識別される必要があります。

Issue #4691 -イタリア語の翻訳を追加します。

Issue #4752 -Dockerfile をリファクタリングして、サポートスクリプトを実行する必要がなくなり（「dockerbuild.」が機能します）、レイヤーを最小限に抑えます。

13.46.2 ハウスキーピング

Issue #4575 -スキーマのリバースエンジニアリングされた SQL テストを追加します。

Issue #4576 -ビューのリバースエンジニアリングされた SQL テストを追加します。

Issue #4600 -ルールのリバースエンジニアリング SQL テストを追加します。

Issue #4616 -外部キーのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4617 -外部サーバー用にリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4618 -外部テーブルのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4619 -FTS テンプレートのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4621 -インデックスのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4624 -主キーのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4627 -ユーザーマッピングのリバースエンジニアリングおよび変更された SQL テストを追加します。

Issue #4690 -リソースグループの変更された SQL テストを追加します。

13.46.3 バグ修正

Issue #2706 -Explain モジュールの ProjectSet アイコンを追加しました。

Issue #3778 -ブール列がキーボードのキーを使用して編集可能であることを確認します。

Issue #3936 -機能テストを安定させるためのさらなるコードリファクタリング。

Issue #4381 -oid 列を含むクエリ出力で行のコピー/貼り付けを使用すると、oid 列を貼り付けない問題を修正します。

Issue #4408 -SlickGrid セルの検証エラーメッセージの表示を修正しました。

Issue #4412 -外部キーの検証済みスイッチオプションが反転する問題を修正します。

Issue #4419 -Python2.7 使用時のデバッガーエラーを修正しました。

Issue #4461 -インポート/エクスポートダイアログを使用し、/"Notnullcolumns/"オプションを提供する際のエラーを修正しました。

Issue #4486 -ビューは特殊文字で作成する必要があります。

Issue #4487 -ブール列がデータの表示/編集とクエリツールで編集可能であることを確認します。

Issue #4577 -テーブルのシステム列をクリックしたときに表示されるエラーを修正しました。

Issue #4584 -クエリツールのタイトルバーにあるデータベース名の HTML エンティティをアンエスケープします.

Issue #4631 -プレーンテキストモードのエディターオプションを追加し、ブロックの折りたたみを無効にして、非常に大きなスクリプトを使用する CodeMirror のレンダリング速度の問題を回避します。

Issue #4642 -サービスが提供されるときにポートとユーザー名が必須でないことを確認します.

Issue #4643 -複合トリガーの Truncate オプションの選択解除の問題を修正しました。

Issue #4644 -ドメインノードのデータ型を変更するときの長さと精度の有効化/無効化の問題を修正しました。

Issue #4650 -ビューの SQL タブの問題を修正します。複合トリガーの回帰です。

Issue #4657 -コンテナでの PGADMIN_SERVER_JSON_FILE 環境変数のサポートを修正しました。

Issue #4663 -python2.7 のクエリ履歴の例外を修正しました。

Issue #4674 -ユーザー名に html 文字が含まれている場合のクエリツールの起動エラーを修正しました。

Issue #4681 -静的ファイルのキャッシュ制御の最大保存期間を増やして、長時間の実行でパフォーマンスを向上させます。

Issue #4698 -Column のデータ型を変更するときの SQL の長さと精度の問題を修正します。

Issue #4702 -Fillfactor と Clustered?の値をリセットするときのインデックスの変更された SQL を修正しました。

Issue #4703 -ソート順と NULL が提供されている場合の btreeIndex のリバースエンジニアリングされた SQL を修正しました。

Issue #4726 -負の値を持つシーケンスを作成する必要があります。

Issue #4727 -EXEC スクリプトがプロシージャの完全なスクリプトを書き込まない問題を修正します。

Issue #4736 -イタリア語でのクエリツールとビューデータの問題を修正します。

Issue #4742 -主キーがインデックスで作成される必要があることを確認します。

Issue #4750 -Python3.6 のクエリ履歴例外を修正しました。

13.47 バージョン 4.12

発売日：2019-08-22

このリリースには、pgAdmin44.11 のリリース以降の多数のバグ修正と新機能が含まれています。

13.47.1 新機能

Issue #4144 -EPAS12+の複合トリガーのサポートを追加します。

Issue #4333 -PostgreSQL12+関数にプランナーサポート関数のサポートを追加します。

Issue #4334 -Postgres12+で生成された列のサポートを追加します。

Issue #4540 -必要に応じて、ダイアログの CodeMirror インスタンスに完全なタブスペースを使用します。

Issue #4549 -ログインおよびカスタムテキストを表示するその他の関連ページにバナーを表示できるようにします。

Issue #4566 -動的にアドレス指定されるホスティング環境との互換性のために、拡張された Cookie 保護を無効にできます。

Issue #4570 -ツリービューノードの検索を担当する内部コードに最適化を追加します。

Issue #4574 -プロパティリストだけでなく、テーブルの行を数えるときにポップアップメッセージに行数を表示します。

Issue #4612 -クエリ履歴にサポートを追加して、データの保存操作中に pgAdmin によって生成された内部クエリを表示します。

13.47.2 ハウスキーピング

Issue #4546 -列のリバースエンジニアリング SQL テストを追加します。

Issue #4554 -トリガー関数のリバースエンジニアリングされた SQL テストを追加します。

Issue #4555 -除外制約のリバースエンジニアリング SQL テストを追加します。

Issue #4560 -RE-SQL テストスイートに--modules オプションを追加して、特定のオブジェクトタイプのテストを許可します。

13.47.3 バグ修正

Issue #3605 -N 行を削除すると最初の N 行が無効になる問題を修正します。

Issue #4179 -Greenplum5.x でのテーブルのリバースエンジニアリングされた SQL の生成を修正しました。

Issue #4229 -wcDocker を更新して、タブストリップとパネルヘッダー以外でブラウザーのコンテキストメニューを使用できるようにします。

Issue #4401 -結果グリッドで型名が適切にエンコードされていることを確認します。

Issue #4414 -パーティションテーブルのリバースエンジニアリングされた SQL の生成を修正しました。パーティションはインデックスの子として表示されました。

Issue #4489 -wcDocker を更新して、ウィンドウ状態のロードが空白のダイアログを作成しないようにします。

Issue #4490 -IE11 のチェックボックスのアクセシビリティの問題を修正します。

Issue #4492 -列のないテーブルのコンテンツを表示するときにクエリツールがエラーをスローしないことを確認します。

Issue #4496 -CYCLE オプションが有効になっている IDENTITY フィールドである場合に列を作成できることを確認します。

Issue #4497 -純粋な数値のコメントを新しい列に保存できるようにします。

Issue #4508 -バックグリッドの日時セルのアクセシビリティの問題を修正しました。

Issue #4520 -更新可能な結果セットはありませんが、クエリツールが公式にサポートされているよりも古いバージョンの psycopg2 で動作することを確認します。

Issue #4525 -クエリツールのメッセージタブにコマンドタグが表示されていることを確認します。

Issue #4536 -データの表示/編集モードでのロードオンデマンドを修正しました。

Issue #4552 -クエリツールでテキストをドラッグするときに JS コンソールでスローされるいくつかのエラーを修正します。

Issue #4559 -EPAS サーバーのトリガーが適切に更新されるようにします。

Issue #4565 -WINDOW オプションが選択されたトリガー関数のリバースエンジニアリングされた SQL を修正しました。

Issue #4578 -トリガーが無効になっているときにトリガーを有効にするメニューが表示されるようにします。

Issue #4581 -主キー制約のコメントがテーブルノードの下で編集できることを確認します。

Issue #4582 -pgAgent ジョブステップの種類（SQL/BATCH）を変更するときのコンソールエラーを修正しました。

Issue #4585 -結果セットが編集可能でない場合、ダブルクリックしてセルの内容を展開する問題を修正しました。

Issue #4586 -Rules のリバースエンジニアリングされた SQL の生成を修正しました。

Issue #4635 -イベントの複合トリガーが適切に更新されていることを確認します。

Issue #4638 -複合トリガーがビューの下に表示されるようにします。

Issue #4641 -複合トリガーで切り捨てオプションが使用可能であることを確認します。

13.48 バージョン 4.11

発売日：2019-07-25

このリリースには、pgAdmin44.10 のリリース以降の多数のバグ修正と新機能が含まれています。

13.48.1 新機能

Issue #1760 -データが更新可能として識別できる場合、クエリツールでの結果セットの編集のサポートを追加します。
Issue #4318 -レイアウトのロック状態に基づいてマウスカーソルを適切に設定します。
Issue #4335 -SETTINGS と SUMMARY の EXPLAIN オプションを追加します。

13.48.2 ハウスキーピング

Issue #4415 -ロールとリソースグループのリバースエンジニアリングされた SQL テストを追加します。
Issue #4441 -FDW のリバースエンジニアリング SQL テストを追加します。
Issue #4452 -言語のリバースエンジニアリングされた SQL テストを追加します。
Issue #4453 -拡張機能のリバースエンジニアリングされた SQL テストを追加します。
Issue #4454 -FTS 構成用にリバースエンジニアリングされた SQL テストを追加します。
Issue #4456 -パッケージのリバースエンジニアリングされた SQL テストを追加します。
Issue #4460 -FTS 辞書のリバースエンジニアリングされた SQL テストを追加します。
Issue #4463 -ドメインのリバースエンジニアリングされた SQL テストを追加します。
Issue #4464 -照合順序のリバースエンジニアリング SQL テストを追加します。
Issue #4468 -タイプのリバースエンジニアリング SQL テストを追加します。
Issue #4469 -シーケンスのリバースエンジニアリングされた SQL テストを追加します。
Issue #4471 -FTS パーサー用のリバースエンジニアリングされた SQL テストを追加します。
Issue #4475 -制約のリバースエンジニアリング SQL テストを追加します。

13.48.3 バグ修正

Issue #3919 -サブノードグリッド上のすべてのコントロールのキーボードナビゲーションを許可します。
Issue #3996 -ジョブプロパティを介した pgAgent スケジュールの削除を修正しました。
Issue #4224 -グラフィカル EXPLAIN キャンバスでの大きなツールチップのちらつきを防ぎます。
Issue #4389 -列の特権を編集するときに表示されるエラーを修正しました。
Issue #4393 -ロールを編集するときに必要に応じてパラメーター値が引用符で囲まれていることを確認します。
Issue #4395 -EXPLAIN オプションはクエリツールインスタンス固有である必要があります。
Issue #4427 -テーブルから json データを取得する際のエラーを修正しました。
Issue #4428 -pgAgent ジョブを更新するときの「malformedarrayliteral」エラーを修正しました。
Issue #4429 -ツリービューからのドラッグアンドドロップが Firefox で期待どおりに機能することを確認します。
Issue #4437 -既存のフィールドを更新するときのテーブルアイコンの問題を修正しました。
Issue #4442 -機能テストがテスト実行から除外されている場合、Selenium によってブラウザーが起動されないようにします。
Issue #4446 -ロールの RE-SQL を生成するときに、USER ではなく一貫して ROLE を使用します。

Issue #4448 -pgAgent ジョブステップで接続文字列を更新するときに発生するエラーを修正しました。

Issue #4450 -レッドウッドモードの EPAS サーバーで作成された ForeignDataWrapper のリバースエンジニアリングされた SQL を修正しました。

Issue #4462 -IE11 のいくつかのマイナーな UI の問題を修正しました。

Issue #4470 -PG/EPAS10 以降での引用符で囲まれた名前を使用したシーケンスリバースエンジニアリングされた SQL 生成を修正しました。

Issue #4484 -Explain と ExplainAnalyze が機能しない問題を修正します。これは#1760 のリグレッションです。

Issue #4485 -フィルターツールバーボタンがデータの表示/編集で機能しない問題を修正します。これはキーボードナビゲーションのリグレッションです。

13.49 バージョン 4.10

発売日：2019-07-04

このリリースには、pgAdmin44.9 のリリース以降の多数のバグ修正と新機能が含まれています。

13.49.1 新機能

Issue #4139 -一部のオブジェクトをクエリツールにドラッグ/ドロップして、署名をクエリテキストに挿入できるようにします。

Issue #4400 -コンテナディストリビューションで/pgadmin4/servers.json へのパスをオーバーライドできます。

13.49.2 バグ修正

Issue #4403 -別のブラウザータブで実行されているクエリツールを閉じるときにのみブラウザーを閉じる確認が表示されるようにします。

Issue #4404 -整数の主キーでデータを編集するときに発生する可能性のあるエラーを防止します。

Issue #4407 -フランス語で実行すると空白の UI が表示される原因となった引用の問題を修正しました。

Issue #4421 -Windows インストーラーのバージョン比較が正しいことを確認します。

13.50 バージョン 4.9

発売日：2019-06-27

このリリースには、pgAdmin44.8 のリリース以降の多数のバグ修正と新機能が含まれています。

13.50.1 新機能

Issue #3174 -単純なテーブルを、継承されたテーブルおよび他のテーブルが継承されたものと視覚的に区別します。

13.50.2 ハウスキーピング

Issue #4202 -リバースエンジニアリングされた SQL および CRUDAPI エンドポイントをテストするためのフレームワークを追加します。

13.50.3 バグ修正

Issue #3994 -継承されたテーブル/外部キーの依存関係タブに部分的なテキストが表示される問題を修正します。

Issue #4036 -主キー列の値に%記号が含まれるデータの編集を許可します。

Issue #4171 -オプションに/"=/"がある場合、リバースエンジニアリングされた SQL が外部テーブルで失敗する問題を修正します。

Issue #4195 -クエリツールやデバッガーなどの/"inner/"タブセットでのキーボードナビゲーションを修正しました。

Issue #4228 -フィルターされた行を表示するときに、パネルヘッダーで正しいラベルが使用されていることを確認します。

Issue #4253 -新しい列をデフォルト値で作成する必要がある問題を修正します。

Issue #4283 -PostgreSQL12 の初期サポート。

Issue #4288 -PostgreSQL12 の初期サポート。

Issue #4290 -PostgreSQL12 の初期サポート。

Issue #4255 -Firefox、IE、および Edge でフォーカスがないときに、ジオメトリビューアがキーを取得しないようにします。

Issue #4306 -「このページにアクセスするにはログインしてください」というメッセージが複数表示されるのを防ぎます。

Issue #4310 -Return キーを使用してマスターパスワードダイアログを送信できることを確認します。

Issue #4317 -ブラウザの自動入力により、ヘルプページが予期せず開かれないことを確認します。

Issue #4320 -パスワードを使用した SSH トンネル接続が失敗する問題を修正します。これはマスターパスワードのリグレッションです。

Issue #4329 -パラメーターを持つ 2 つの関数を並列でデバッグするときの初期化エラーを修正します。

Issue #4343 -EPASv12 で列のプロパティダイアログが適切に開く問題を修正

Issue #4345 -Import/Export モジュールで使用される単語「export」を大文字にします。

Issue #4349 -クエリ履歴で文字列が適切にエンコードされていることを確認します。

Issue #4350 -ファイルをアップロードするときに CSRF トークンを含めるようにします。

Issue #4357 -pgAdmin サーバーを再起動してページを更新したときの接続復元の問題を修正します。

Issue #4360 -初期化が完了した場合にのみデバッガーコントロールボタンが有効になるようにします。

Issue #4362 -トリガー関数の CREATE スクリプトを生成するときに含まれる追加の/"SETOF/"を削除します。

Issue #4365 -バックアップグローバルとバックアップサーバーのヘルプリンクを修正しました。

Issue #4367 -列名に HTML が含まれる場合、データの表示/編集モードで見られる XSS の問題を修正します。

Issue #4378 -Python エスケープが JS エスケープと一致することを確認し、トリガーにスーパーユーザアクセスが必要だったクエリツールのマイナーな XSS の問題を修正します。

Issue #4380 -テーブルダイアログで列とパーティションの両方を同時に編集できることを確認します。

Issue #4386 -ユーザー名に XSS の脆弱なテキストが含まれる場合の XSS の問題を修正します。

13.51 バージョン 4.8

発売日：2019-06-03

このリリースには、pgAdmin44.7 のリリース以降の多数のバグ修正と新機能が含まれています。

13.51.1 新機能

Issue #2653 -UI レイアウトを完全にロックするか、ドッキングの変更を防止します。

13.51.2 バグ修正

Issue #4169 -クエリツールのジオメトリビューアをレイアウトの保存から除外します。

Issue #4307 -ダウンロード時にのみ画像を埋め込むことにより、説明計画のパフォーマンスを向上させます。

Issue #4308 -ビューとマテリアライズドビューの SQL へのアクセスの問題を修正します。フォルダー名の複数形化の回帰。

13.52 バージョン 4.7

発売日：2019-05-30

このリリースには、pgAdmin44.6 のリリース以降の多くのバグ修正が含まれています。

13.52.1 バグ修正

Issue #3377 -サーバーモードでは、ユーザーパスワードが変更されたときに保存されたすべてのサーバー資格情報を更新します。

Issue #3885 -メインメニューバーのレスポンシブなレイアウトを修正しました。

Issue #4162 -既存のテーブルに複数の列を追加するときの構文エラーを修正しました。

Issue #4164 -クライアントが Windows にあり、サーバーが Mac/Linux 上にある場合に発生するファイルブラウザーパスの問題を修正します。

Issue #4184 -保存されたパスワードのセキュリティを強化するためにマスターパスワードを追加しました。

Issue #4194 -メニューナビゲーションのアクセシビリティの問題を修正しました。

Issue #4208 -UI ロゴを更新します。

Issue #4217 -CSRF セキュリティの脆弱性の問題を修正しました、ごとに

Issue #4218 -Select2 コントロールに dropdownParent を適切に割り当てます。

Issue #4219 -必要に応じて popper.js がインストールされていることを確認します。

Issue #4227 -メンテナンスダイアログの Tab キーナビゲーションを修正しました。

Issue #4244 -Safari ブラウザーのダイアログフッターのトグルスイッチコントロールとボタンの Tab キーの問題を修正しました。

Issue #4245 -Tab キーを使用してフォーカスを取得したときに要素が強調表示されるようにします。

Issue #4246 -パネルでサブノード制御を使用するときのコンソールエラーを修正しました。

Issue #4261 -MIME タイプとしての application/x-javascript の使用を停止し、代わりに RFC 準拠の application/javascript を使用します。

Issue #4262 -デフォルトのパーティションを持つリストでパーティション化されたテーブルのテーブルプロパティを表示する際のエラーを修正しました。

Issue #4263 -クエリツールでの NULL 要素の JSON の処理を修正しました。

Issue #4269 -グリッド内のスイッチセルのナビゲーションを修正しました。

Issue #4275 -MichelFeinstein ごとに、制約の NOINHERIT オプションの表現を明確化。

Issue #4276 -OpenShift などの一部の環境で失敗する可能性があるため、構成データベースを含むディレクトリの権限チェックを緩和します。

Issue #4278 -ブラウザーがオートコンプリートリストを開いたときに、バックグリッドパスワードのセルがフォーカスを失うのを防ぎます。

Issue #4284 -シリアル列を持つテーブルを作成するときの構文エラーを修正しました。

13.53 バージョン 4.6

発売日：2019-05-02

このリリースには、pgAdmin44.5 のリリース以降に報告された多くの新機能と修正が含まれています

13.53.1 特徴

Issue #4165 -psycopg2 ではなく、Pythonwheel の psycopg2-binary に依存します。

13.53.2 バグ修正

Issue #2392 -[削除] ボタンをクリックしても、データベースサーバから行がすぐに削除されないようにしてください。[保存] ボタンがクリックされたときに削除する必要があります。

Issue #3327 -バックグリッドに新しく追加された行が表示されることを確認します。

Issue #3582 -コメントとしての JSON 文字列がすべてのオブジェクトに適切に追加されることを確認します。

Issue #3605 -N 行を削除すると最初の N 行が無効になる問題を修正します。

Issue #3938 -デフォルトパーティションのサポートを追加しました。

Issue #4087 -デフォルトのシーケンス権限に対して「GRANTUPDATE」SQL が表示される問題を修正します。

Issue #4101 -クエリツールまたはデバッガーが新しいブラウザータブで開かれた場合、リロードする前に確認ダイアログがポップアップするようにします。

Issue #4104 -プライマリキーを持つルートパーティションテーブルのレコードを追加/編集する必要があることを確認します。

Issue #4121 -Index ノードの定義セクションの列のアラインメントの問題を修正しました。

Issue #4134 -表領域のプロパティを開くときの「場所を空にできません」エラーを修正しました。

Issue #4138 -ドロップダウンがずれる/ずれる問題を修正します。

Issue #4154 -IDENTITY 列（列の一部として編集可能）の実装に使用されるシーケンスを除くすべてのシーケンスがツリービューに表示されることを確認します。ShowSystemObjects が有効になっている場合はすべて表示します。

Issue #4160 -既存のテーブルの「インクリメント値を空にすることはできません」エラーを修正しました。

Issue #4161 -EPAS サーバー 10 以下のプロシージャのパラメーターを適切に設定/リセットする必要があることを確認します。

Issue #4163 -テーブルのリバースエンジニアリングされた SQL に重複した列が含まれないようにします。

Issue #4182 -ストレージとセッションのディレクトリ、および構成データベースに対する権限の健全性を確認します。

13.54 バージョン 4.5

発売日：2019-04-10

このリリースには、pgAdmin44.4 のリリース以降に報告された多くの新機能と修正が含まれています。

13.54.1 バグ修正

Issue #2214 -SCRAM 認証の「ChangePassword」の問題を修正しました。

Issue #3656 -2 つの連続する SELECT ステートメントが適切に動作することを確認します。

Issue #4131 -データベースを更新する実際の [保存] ボタンとの混乱を避けるために、データグリッドテキストエディターの [保存] ボタンのラベルを変更します。

Issue #4142 -推奨される ESLinter チェックを追加しました。

Issue #4143 -pgAdmin4 が psycopg2v2.8 で適切に動作することを確認します

13.55 バージョン 4.4

発売日：2019-04-04

このリリースには、pgAdmin44.3 のリリース以降に報告された多くの新機能と修正が含まれています。

警告: このリリースには、サーバーモードで実行しているときに既存のユーザーのユーザーごとのストレージディレクトリの名前を変更するバグ修正（ Issue #3887 ）が含まれています。以前は、保存された SQL クエリは、ユーザーのメールアドレスのユーザー名部分にちなんで名前付けられたサブディレクトリの *STORAGE_DIR* に保存されていました。このバージョン以降、@をアンダースコアに置き換えた完全なメールアドレスが使用されます。たとえば、 *STORAGE_DIR* を */var/lib/pgadmin4* に設定した v.4.3 では、ユーザーファイルは次の場所に保存できます。

```
/var/lib/pgadmin4/storage/username/
```

修正により、そのディレクトリは次のように名前変更（または新しいユーザー用に作成）されます。

```
/var/lib/pgadmin4/storage/username_example.com/
```

13.55.1 特徴

Issue #2001 -Gunicorn を使用したリバースプロキシ設定のサポートを追加し、Gunicorn、uWSGI&NGINX 構成を文書化します。

Issue #4017 -クエリツールの履歴をセッション間で持続させる.

Issue #4018 -React および 87 他の関連ライブラリへの大きくて不要な依存関係を削除します。

Issue #4030 -IDENTITY 列のサポートを追加します。

Issue #4075 -ePubdoc ビルドターゲットを追加します。

13.55.2 バグ修正

Issue #1269 -列と FTS パーサーモジュールの名前付けの矛盾を修正しました。

Issue #2627 -継承された列のコメントとデフォルトをリバースエンジニアリングされたテーブル SQL に含めます。

Issue #3104 -いくつかのドイツ語の翻訳を改善します。

Issue #3887 -ストレージディレクトリ名のベースとして、ユーザーの完全なメールアドレス（ユーザー名の部分だけでなく）を使用します。

Issue #3968 -wcDocker を更新して、結果パネルのサイズを変更するとスクラッチパッドのサイズが大きくなる問題を修正します。

Issue #3995 -トリガーを持つテーブルの SQL を表示するときに、Postgres からの「偽の varno」メッセージを回避します。

Issue #4019 -Python と JavaScript のすべての依存関係を更新します。

Issue #4037 -リバースエンジニアリングされたテーブル SQL に継承された列のコメント SQL を含めます。

Issue #4050 -EventTrigger ダイアログの WHEN フィールドを CodeMirror コントロールにします。

Issue #4052 -リソースグループダイアログのオンラインヘルプボタンを修正しました。

Issue #4053 -索引ダイアログのオンラインヘルプボタンを有効にします。

Issue #4054 -クエリツールで列がゼロの結果セットを正しく処理する.

Issue #4058 -継承された列を SELECT スクリプトに含めます。

Issue #4060 -latexpdfdoc ビルドを修正しました。

Issue #4062 -データの表示/編集での数値配列の処理を修正しました。

Issue #4063 -ダイアログなどのサイズを変更するためのグラブハンドルを拡大する.

Issue #4069 -FileCreate ダイアログで必要に応じて、ファイル名にファイルサフィックスを追加します。

Issue #4071 -クエリ結果を CSV ファイルとしてダウンロードするときに、Firefox がファイル名/場所の入力を求めるようにします。

Issue #4073 -CodeMirror のアクティブラインの背景色を$color-danger-lighter に変更して、選択色と競合しないようにします。

Issue #4081 -VALIDUNTIL 句を持つロールの RE-SQL 構文を修正しました。

Issue #4082 -CSV ダウンロードを使用して CREATE スクリプトを「ダウンロード」するときに空のエラーメッセージが表示されるのを防ぎます。

Issue #4084 -レイアウト保存コードをオーバーホールして、クエリツールとデバッガーを含み、終了時ではなく（信頼性が低い）変更イベントが検出されたときにレイアウトを保存します。

Issue #4085 -クエリツールからの CSV ダウンロード中にエラーを CSV ファイルに配置するのではなく、UI に表示します。

Issue #4090 -BackupServer のドイツ語翻訳を改善しました。

Issue #4096 -クエリツールで CSV をダウンロードした後、ツールバーボタンが適切にリセットされることを確認します。

Issue #4099 -EPAS10+の SQL ヘルプを修正し、URL 生成コードをテスト可能なファンクションにリファクタリングします。

Issue #4100 -インクリメント、スタート、最小、および最大のオプションを設定してシーケンスを作成できることを確認します。

Issue #4105 -クエリツールで JSON データがレンダリングされない問題を修正します。

Issue #4109 -プロパティを更新した後に View/MaterializedView ノードが表示されることを確認します。
Issue #4110 -マテリアライズドビューのカスタム自動バキューム構成を修正しました。

13.56 バージョン 4.3

発売日：2019-03-07

このリリースには、pgAdmin44.2 のリリース以降に報告された多くの新機能と修正が含まれています

13.56.1 特徴

Issue #1825 -Pythonwheel からインストールする場合にコマンドラインから pgAdmin（pgadmin4）を起動するスクリプトをインストールします。
Issue #2233 -クエリツールに/"スクラッチパッド/"を追加して、編集中にテキストスニペットを保持します。
Issue #2418 -クエリツールにコミットボタンとロールバックボタンを追加します。
Issue #3439 -セキュリティのために X-FRAME-OPTIONS を設定できるようにします。デフォルトは SAMEORIGIN です。
Issue #3559 -ツリービューで選択されたノードだけでなく、子ノードも自動的に展開します。
Issue #3886 -コンテナに PG ユーティリティの複数のバージョンを含めます。
Issue #3991 -docker コンテナの AlpineLinux バージョンを更新します。
Issue #4034 -クエリツールの結果グリッド列のサイズ変更ハンドルのダブルクリックをサポートして、コンテンツに合わせてサイズ変更します。

13.56.2 バグ修正

Bug #3096 -UI 言語が英語でない場合、サイズの統計が [統計] タブに表示されるようにします。
Bug #3352 -有効期限が無限に設定されたロールの表示を正しく処理します。
Bug #3418 -実際の行 OID ではない oid データ型の列の値を編集できるようにします。
Bug #3544 -クエリツールのタブのタイトルをより簡潔で使いやすいものにします。
Bug #3587 -JSONB データの bigint のサポートを修正しました。
Bug #3583 -CodeMirror を 5.43.0 に更新して、自動インデントの問題を解決。
Bug #3600 -データの表示/編集を使用するときに、psycopg2 によって処理中に JSON データが変更されないことを確認します。
Bug #3673 -[CSV としてダウンロード] オプションを変更して、実行されているクエリツールと同じ接続を使用して、一時テーブルなどを使用します。
Bug #3873 -Safari でのコンテキストサブメニューの配置を修正しました。
Bug #3890 -新しいデザインに従ってドキュメントのスクリーンショットを更新。
Bug #3906 -GrantWizard の [閉じる] と [最大化] ボタンの配置を修正しました。
Bug #3911 -すべての PG サーバーサイドエンコーディングの完全なサポートとテストを追加します。
Bug #3912 -JSON 主キーを使用したテーブルデータの編集を修正しました。

Bug #3933 -ロガーの例外を無視します。

Bug #3942 -ユーザが pgAdmin からログアウトしたときに接続を正常に閉じる.

Bug #3946 -pgAgent ジョブの複数のスケジュールを削除するチェックボックスの配置を修正しました。

Bug #3958 -データ変更関数を呼び出す場合に、クエリツールのトランザクション管理から SELECT ステートメントを除外しないでください。

Bug #3959 -依存関係と依存関係の表示を最適化し、100 行のバッチで行のオンデマンドロードを使用します。

Bug #3963 -インポート/エクスポートのトグルスイッチの配置を修正しました。

Bug #3970 -空のフィルター文字列で Sort/Filter ダイアログを閉じるときのエラーを防ぎます。

Bug #3974 -pgagent の接続タイプのトグルスイッチのアライメントを修正しました。

Bug #3981 -読み取り専用スタンバイが書き込みクエリと見なしないように、setbytea_output へのクエリを修正します。

Bug #3982 -すべての PG サーバーサイドエンコーディングの完全なサポートとテストを追加します。

Bug #3985 -外部サイトがセキュリティのために X-FRAME-OPTIONS=DENY を設定できるように、ドキュメントと外部サイトを iframes に埋め込まないでください。

Bug #3992 -すべての PG サーバーサイドエンコーディングの完全なサポートとテストを追加します。

Bug #3998 -ApacheHTTPD では許可されない場合があるため、URL パラメーターのスラッシュをカスタムエンコードします。

Bug #4000 -CodeMirror を 5.43.0 に更新して、スペースを有効にしたタブのインデントに関する問題を解決。

Bug #4013 -クエリツールで CSV をダウンロードするときに長いクエリでエラーが発生しないようにします。

Bug #4021 -エディターを無効にし、クエリの実行中に関数を実行します。

Bug #4022 -別のユーザーのグループが既に存在する場合、サーバーのインポートが失敗する問題を修正します。

13.57 バージョン 4.2

発売日：2019-02-07

このリリースには、pgAdmin44.1 のリリース以降に報告された多くの修正が含まれています

13.57.1 バグ修正

Issue #3051 -Bootstrap スイッチを Bootstrap4 トグルに置き換えて、パフォーマンスを向上させます。

Issue #3272 -PyCrypto モジュールを暗号化モジュールに置き換えます。

Issue #3453 -外部テーブルオプションの SQL を修正しました。

Issue #3475 -クエリツールで長時間実行されるクエリの時間部分を表示するように実行時間を修正しました。

Issue #3608 -クエリツールの [メッセージ] タブは、データの表示/編集を使用したテーブル/ビューのその後の実行でクリアされる必要があります。

Issue #3609 -データの表示/編集では、クリアドロップダウンメニューを無効にする必要があります。

Issue #3664 -1000 以上のテーブルの統計パネルがハングする問題を修正しました。

Issue #3693 -サーバーグループが既存の名前で作成されると、適切なエラーがスローされる必要があります。

Issue #3695 -長い文字列はアラートダイアログでラップする必要があります。

Issue #3697 -データ出力ウィンドウがクエリツールから切り離されている場合でも、クエリの出力が表示されるようにします。

Issue #3740 -インライン edbspl トリガー関数は、GrantWizard で表示されません。

Issue #3774 -カスタム型引数として戻り値の型を使用して関数を作成するときに、適切な SQL を生成する必要があります。

Issue #3800 -サーバーダイアログのデータベース制限が特殊文字で機能することを確認します。

Issue #3811 -[バックアップ/復元] ボタンがシングルクリックで動作することを確認します。

Issue #3837 -トリガー作成中の when 句の SQL を修正しました。

Issue #3838 -カスタムタイプ引数で列を作成/変更するときに適切な SQL を生成する必要があります。

Issue #3840 -隠しファイルのチェックボックスが切り替えられたときにファイル形式コンボボックスの値が保持されるようにします。

Issue #3846 -カスタム型引数でプロシージャを作成するときに適切な SQL を生成する必要があります。

Issue #3849 -ブラウザを閉じるか更新する前に警告する。

Issue #3850 -プロシージャの EXEC スクリプトを修正しました。

Issue #3853 -interval タイプのドメインを高精度に作成するときに、適切な SQL を生成する必要があります。

Issue #3858 -他のツールバーボタンをクリックしたときにドロップダウンを閉じる必要があります。

Issue #3862 -ダイアログタブのキーボードナビゲーションを修正しました。

Issue #3865 -フレームスプリッターのマウスホバー領域を増やして、サイズ変更しやすくします。

Issue #3871 -InternetExplorer のツリー矢印アイコンの配置を修正しました。

Issue #3872 -外部プロセスダイアログのオブジェクト名が適切にエスケープされていることを確認します。

Issue #3891 -json/jsonb 編集の保存とキャンセルボタンの順序を修正しました。

Issue #3897 -FTS 構成、FTS 辞書、FTS パーサー、および FTS テンプレートのデータを適切に更新する必要があります。

Issue #3899 -プロパティタブから複数のルールと外部テーブルを削除できない問題を修正しました。

Issue #3903 -クエリツールの初期化エラーを修正しました。

Issue #3908 -バックグリッドの Select2 と Privilege セルのキーボードナビゲーションを修正しました。

Issue #3916 -マテリアライズドビューダイアログに正しいスキーマが表示されるはずです。

Issue #3927 -EPAS サーバーのパッケージ内のプロシージャのデバッガーの問題を修正しました。

Issue #3929 -プロパティパネルのヘルプメッセージの配置を修正しました。

Issue #3932 -InternetExplorer のサブメニューの配置を修正しました。

Issue #3935 -GrantWizard が no-redwood-compat モードで実行されている EPAS サーバーの機能をリストすることを確認します。

Issue #3941 -ダッシュボードグラフの最適化。

Issue #3954 -廃止された Python2.6 コードを削除します。

Issue #3955 -PGADMIN_BIND_ADDRESS を介して Docker コンテナのバインドアドレスを公開します。

Issue #3961 -HTTPExceptions をそのまま返す必要があるため、all_exception_handler から除外します。

13.58 バージョン 4.1

発売日：2019-01-15

このリリースには、pgAdmin44.0 のリリース以降に報告された多くの修正が含まれています

13.58.1 バグ修正

Issue #3505 -継承された列を持つテーブルに対して生成された SQL を修正しました。
Issue #3575 -サーバーの名前が変更された後にコンテキストメニューが機能することを確認します。
Issue #3836 -PG11 で変更された VACUUM オプションの順序付けを修正しました。
Issue #3842 -システムオブジェクトの表示が有効になっていない限り、スキーマプロパティリストにシステムカタログを表示しない.
Issue #3861 -バックアップ/復元ダイアログのヘルプを修正しました。
Issue #3866 -テーブルデータの最後の行が表示され、ユーザーが新しい行を追加できることを確認します。
Issue #3877 -SQL_ASCII データベースのマルチバイト文字に対して、ブラウザーをより堅牢にします。

13.59 バージョン 4.0

発売日：2019-01-10

このリリースには、pgAdmin43.6 のリリース以降に報告された多くの機能と修正が含まれています

13.59.1 特徴

Issue #3589 -クエリプランを SVG ファイルとしてダウンロードできるようにします。
Issue #3692 -新しい UI デザイン。
Issue #3801 -サーバーをコンテナデプロイメントにプリロードできます。

13.59.2 バグ修正

Issue #3083 -エディットグリッドテキストのポップアウトのサイズ変更ハンドルのサイズを大きくします。
Issue #3354 -デバッガーへの入力としての配列型の処理を修正しました。
Issue #3433 -クエリツールのレンダリングに失敗する可能性がある問題を修正します。
Issue #3549 -EPAS でイベントトリガー機能を正しく表示します。
Issue #3559 -ツリービューの復元のさらなる改善。
Issue #3599 -コンテナビルドで Postfix を実行して、パスワードなどをリセットできます

Issue #3619 -クエリツールとデバッガーのコード領域にタイトルを追加して、パネルをその中に再ドッキングできるようにします。

Issue #3679 -クエリツールのレンダリングに失敗する可能性のある Webpack の問題を修正します。

Issue #3702 -可能な限りロックテーブルにリレーション名（OID ではなく）を表示します。

Issue #3711 -クエリツールのエンコーディングの問題を修正します。

Issue #3726 -RE-SQL の EXCLUDE 制約に WHERE 句を含めます。

Issue #3753 -ユーザー定義の Castfromsmallint->text が作成されるときの問題を修正します。

Issue #3757 -メンテナンスダイアログに表示しないラジオボタンを非表示にします。

Issue #3780 -CSV ダウンロードで null 値が適切に処理されるようにします。

Issue #3796 -GrantWizard の言葉遣いを微調整。

Issue #3797 -スキーマオブジェクトの一括削除を防止します。

Issue #3798 -ブラウザーのツールバーボタンが英語以外の言語で機能することを確認します。

Issue #3805 -エディットグリッドテキストのポップアウトの水平方向のサイズ変更を許可します。

Issue #3809 -FROM 句の最初の識別子を引用する必要があるときにオートコンプリートが機能することを確認します。

Issue #3810 -スキーマ修飾されたテーブルの列でオートコンプリートが機能することを確認します。

Issue #3821 -識別子がプランビューアーに適切に表示されていることを確認します。

Issue #3830 -セットアッププロセスを中止された実行に対してより堅牢にします。

Issue #3856 -エクスポートジョブの作成中の問題を修正しました。

13.60 バージョン 3.6

発売日：2018-11-29

このリリースには、pgAdmin43.5 のリリース以降に報告された多くの機能と修正が含まれています

13.60.1 特徴

Issue #1513 -コレクションの [Properties] パネルから一度に複数のオブジェクトをドロップするサポートを追加します。

Issue #3772 -構成データベースからサーバー定義をインポートおよびエクスポートする機能を追加します。

13.60.2 バグ修正

Issue #3016 -クエリの実行中にエラーが発生した場合、クエリツールの [メッセージ] タブから以前の通知が削除されないことを確認します。

Issue #3029 -Select2 コントロールで選択順序を保持して、データのインポート/エクスポートで列の順序を修正できるようにします。

Issue #3629 -デバッガーでパラメーターとして 0（整数）と空の文字列を使用できるようにします。

Issue #3723 -デバッグを開始できないときにエラーを適切に報告します。

Issue #3734 -前の処理が完了する前にデバッガーコントロールが再度押されないようにします。

Issue #3736 -デバッガーのトグルブレークポイントボタンを修正しました。

Issue #3742 -TableDialogue の NOTNULL およびデフォルト値オプションの変更を修正しました。

Issue #3746 -複数のファンクション/プロシージャを一度に削除する問題を修正しました。

13.61 バージョン 3.5

発売日：2018-11-01

このリリースには、pgAdmin43.4 のリリース以降に報告された多くの機能と修正が含まれています

13.61.1 特徴

Issue #1253 -ツリービューの状態を定期的に保存し、再接続時に自動的に復元します。

Issue #3562 -Bootstrap3 から Bootstrap4 に移行します。

13.61.2 バグ修正

Issue #3232 -バイナリパスが間違っている場合にユーティリティ（バックアップ/復元/メンテナンス/インポート-エクスポート）が開始されないようにし、プロセスをキャンセルするための「プロセスの停止」ボタンも追加しました。

Issue #3638 -開始日時と例外を使用して新しい pgAgent スケジュールを作成するときの構文エラーを修正しました。

Issue #3674 -セッションファイルを定期的にクリーンアップします。

Issue #3660 -エディターだけでなくツール全体に適用されるため、設定の「SQLEditor」セクションの名前を「QueryTool」に変更します。

Issue #3676 -EDB でラップされた関数の CREATEScript 機能を修正しました。

Issue #3700 -接続のガベージコレクターを修正しました。

Issue #3703 -ログアウト時にキャッシュから接続を削除します。

Issue #3722 -バックアップ/復元中に、スキーマおよび他の子オブジェクトに対してユーティリティの存在チェックが機能することを確認します。

Issue #3730 -pgAdmin43.5 の起動中の致命的なエラーを修正しました。Flask のバージョンを 0.12.4 に更新してリリースします。

13.62 バージョン 3.4

発売日：2018-10-04

このリリースには、pgAdmin43.3 のリリース以降に報告された多くの機能と修正が含まれています

13.62.1 特徴

Issue #2927 -一貫性と色のメンテナンスなどを容易にするために、すべての CSS を SCSS ファイルに移動します.

Issue #3514 -オプションのデータポイントマーカーとマウスオーバーのツールチップを追加して、グラフに値を表示します。

Issue #3564 -データの表示とクエリツールのショートカットをブラウザーのヘッダーバーに追加します。

13.62.2 バグ修正

Issue #3464 -Windows のログファイルパスにワイド文字がある場合にランタイムが正常に起動できることを確認します。

Issue #3551 -編集グリッドでのバックスラッシュの処理を修正しました。

Issue #3576 -ダッシュボードを閉じたときにクエリが実行されないようにします。

Issue #3596 -EPAS の CLOB データ型のサポートを修正しました。

Issue #3607 -クエリツールでの並べ替え/フィルターの検証と強調表示に関するロジックを修正しました。

Issue #3630 -public および pg_catalog 以外のスキーマのオブジェクトでオートコンプリートが機能することを確認します。

Issue #3657 -クエリを実行する前に、[設定] ダイアログからのクエリツール設定の変更が適用されていることを確認します。

Issue #3658 -間違った方法で使用されていたスキーマとスキーマのアイコンとカタログとカタログのアイコンを入れ替えます。

13.63 バージョン 3.3

発売日：2018-09-06

このリリースには、pgAdmin43.2 のリリース以降に報告された多くの機能と修正が含まれています

13.63.1 特徴

Issue #1407 -空白のキャンバスまたはさまざまなマップソースに PostGIS データをレンダリングできるジオメトリビューアを追加します.

Issue #3503 -PostgreSQL11 の新しいバックアップ/復元オプションを追加しました。「pg_dumpall」のダンプオプションを追加しました。

Issue #3553 -スペイン語の翻訳を追加します。

13.63.2 バグ修正

Issue #3136 -CI システムでの継続的な実行の機能テストを安定させます。

Issue #3191 -デバッガーの実行の問題を修正しました。

Issue #3313 -pgAgent スケジュールで「selectall」および「unselectall」が適切に動作することを確認します。

Issue #3325 -誤って ASC/DESC を必要とするソート/フィルターダイアログの問題を修正しました。

Issue #3347 -バックアップが任意の形式の「--data-only」および「--schema-only」で動作することを確認します。

Issue #3407 -設定パネルのキーボードショートカットのレイアウトを修正しました。

Issue #3420 -オートコンプリート機能に使用される pgcli コードをバージョン 1.10.3 とマージします。

Issue #3461 -ノードを更新するとプロパティリストも更新されるようにします。

Issue #3525 -ダッシュボードの更新ボタンでテーブルが更新されるようにします。

Issue #3528 -クエリツールで接続エラーを適切に処理する.

Issue #3547 -セッション実装をスレッドセーフにする

Issue #3548 -外部テーブルノードが GPDB でのみ可視であることを確認します。

Issue #3554 -ステップインおよびステップアウト時のデバッガーの自動スクロールの問題を修正しました。

Issue #3558 -ソート/フィルタダイアログの編集の問題を修正しました。

Issue #3561 -データベース接続が失われた後、並べ替え/フィルタダイアログに適切なメッセージが表示されるようにします。

Issue #3578 -ロールの sql が GPDB の SQL パネルに表示されることを確認します。

Issue #3579 -Windows インストーラーをビルドするとき、依存関係をインストールする前にシステムの Python パッケージをコピーして、意図したよりも古いバージョンで終わらないようにします。

Issue #3604 -データの表示/編集のドキュメントを修正しました。

13.64 バージョン 3.2

発売日：2018-08-09

このリリースには、pgAdmin43.1 のリリース以降に報告された多くの機能と修正が含まれています

13.64.1 特徴

Issue #2136 -ファイルがバージョンごとにのみキャッシュされるように URL のバージョン番号を追加しました。

Issue #2214 -SCRAM パスワード変更のサポートを追加します（psycopg2>=2.8 が必要）。

Issue #3074 -保存されたパスワードのリセットのサポートを追加します。

Issue #3397 -グラフィカルクエリプランビューアーにトリガーと JIT 統計のサポートを追加します。

Issue #3412 -PG/EPAS11 のパーティションテーブルにプライマリキー、外部キー、ユニークキー、インデックス、およびトリガーのサポートを追加します。

Issue #3506 -ユーザーがランタイムで固定ポート番号を指定して、Cookie のホワイトリストなどを支援できるようにします。

Issue #3510 -ランタイムにメニューオプションを追加して、アプリケーションサーバーの URL をクリップボードにコピーします。

13.64.2 バグ修正

Issue #3185 -macOS でのアップグレードチェックを修正しました。

Issue #3191 -多数のデバッガー実行の問題を修正しました。

Issue #3294 -リアルタイム設定を処理するためのインフラストラクチャ (およびクエリツール、ダッシュボード、デバッガーの変更)。

Issue #3309 -バックアップのディレクトリ形式のサポートを修正しました。

Issue #3316 -システムトレイのないシステムでの実行をサポート。

Issue #3319 -クエリツールの [キャンセル] ボタンのステータスの処理をクリーンアップして修正しました。

Issue #3363 -セクションの復元オプションの復元を修正しました。

Issue #3371 -/misc/ping テストエンドポイントが呼び出されたときにセッションを作成しないでください.

Issue #3446 -EPAS/PG11 のさまざまなプロシージャ/関数関連の修正。

Issue #3448 -インポート/エクスポートでシステム列を除外します。

Issue #3457 -EPAS パッケージのプロシージャのデバッグを修正しました。

Issue #3458 -pgAdmin4 は python3.7 で動作するはずです。

Issue #3468 -パスフレーズのないキーを使用した SSH トンネリングをサポートします。

Issue #3471 -SSH トンネルのポート番号が尊重されていることを確認します。

Issue #3511 -SSH トンネルのパスワードを保存およびクリアするサポートを追加します。

Issue #3526 -トリガー関数を作成した後、COST ステートメントを自動的に複製しないでください。

Issue #3527 -データの表示->フィルター処理された行ダイアログが表示されます。

13.65 バージョン 3.1

発売日：2018-06-28

このリリースには、pgAdmin43.0 のリリース以降に報告された多くの機能と修正が含まれています

13.65.1 特徴

Issue #1447 -SSH トンネル接続のサポートを追加する

Issue #2686 -キーワードを大文字でオートコンプリートするオプションを追加

Issue #3204 -クエリツールに LISTEN/NOTIFY のサポートを追加する

Issue #3273 -ファイルダイアログでのソートを許可する

Issue #3362 -PG11 の関数と手続きのサポート

Issue #3388 -サーバーごとに接続タイムアウトを構成できるようにする

13.65.2 バグ修正

Issue #1220 -データベース名に/"=/"記号が含まれている場合は、バックアップと復元を開始しないでください

Issue #1221 -データベース名に/"=/"記号が含まれている場合は、メンテナンスを開始しないでください

Issue #3179 -トリガー関数の SQL 生成エラーを修正する

Issue #3238 -サーバーからの JSON 応答メッセージを解析するためのエラー処理を標準化する

Issue #3250 -クエリツールでの SQL_ASCII データの処理を修正しました

Issue #3257 -無効なクエリを EXPLAIN しようとしたときにエラーをキャッチする

Issue #3277 -終了時のサーバーのクリーンアップは、サーバーが実際に起動した場合にのみ発生するようにします

Issue #3284 -F5 キーはブラウザツリーを更新するはずです

Issue #3289 -クエリツールでの SQL_ASCII データの処理を修正しました

Issue #3290 -alertify メッセージボックスに追加された閉じるボタン。バックエンドエラーが発生した場合に表示されます

Issue #3295 -ロード時にデバッガーがフォーカスを取得して、ショートカットキーが期待どおりに動作することを確認する

Issue #3298 -実行オプションのドロップダウンで矢印キーが期待どおりに動作しないクエリツールのキーボードの問題を修正しました

Issue #3303 -サーバーの起動を妨げる可能性のある日本語の翻訳エラーを修正

Issue #3306 -Greenplum データベースのインデックスを持つテーブルの表示 SQL を修正

Issue #3307 -コンテナ環境で見られる 1024 未満のポート番号を持つサーバーへの接続を許可します

Issue #3308 -パーティションテーブルのアイコンが Greenplum データベースの非パーティションテーブルと同じだった問題を修正

Issue #3310 -クエリツールの alertify エラーメッセージのレイアウトを修正

Issue #3324 -Windows で確実に動作するようにテンプレートローダーを修正しました（Greenplum での外部テーブルを修正）

Issue #3333 -設定にアクセスする前にランタイムコアアプリケーションが設定されていることを確認する

Issue #3342 -Flask の推奨事項ごとに SESSION_COOKIE_SAMESITE='Lax' を設定して、フォームの送信など、外部サイトからの CSRF を起こしやすい要求で Cookie を送信しないようにします

Issue #3353 -データベースの名前を変更するときにエラーが発生した場合、適切に処理する

Issue #3356 -パッケージの RE-SQL にスキーマ名を含める

Issue #3374 -オートコンプリートを修正する

Issue #3392 -コンテナビルドでの IPv6 サポートを修正

Issue #3409 -テーブルで RE-SQL を取得するときに GreenPlum で例外を回避する

Issue #3411 -サーバーの起動を妨げる可能性のあるフランス語の翻訳エラーを修正

Issue #3431 -GreenPlum 外部テーブルの RE-SQL 生成を修正

13.66 バージョン 3.0

発売日：2018-03-22

このリリースには、pgAdmin42.1 のリリース以降に報告された多くの機能と修正が含まれています

13.66.1 特徴

Issue #1894 -データの表示/編集時にソートを許可する

Issue #1978 -UI アニメーションを有効/無効にする機能を追加

Issue #2895 -メインブラウザーウィンドウにキーボードナビゲーションオプションを追加する

Issue #2896 -Tab/Shift-Tab キーを介してクエリツールモジュールにキーボードナビゲーションを追加します

Issue #2897 -デバッガーでのキーボードナビゲーションのサポート

Issue #2898 -ダイアログでのタブナビゲーションをサポート

Issue #2899 -メインウィンドウにさまざまな一般的なオプションの構成可能なショートカットキーを追加します

Issue #2901 -デバッガーで設定可能なショートカット

Issue #2904 -クリック可能な画像/ボタンにスクリーンリーダー用の適切なツールチップがあることを確認する

Issue #2950 -マーカー（*/ pga4dash /*）をダッシュボードのクエリに追加して、サーバーログからより簡単にフィルタリングできるようにします

Issue #2951 -ダッシュボードのテーブルとチャートを有効/無効にする

Issue #3004 -Greenplum のサーバーとデータベースの統計をサポート

Issue #3036 -Greenplum でパーティションを表示する

Issue #3044 -Greenplum で関数を表示する

Issue #3086 -ランタイムを Web ブラウザを起動できるトレイベースのサーバーに書き換える

Issue #3097 -Greenplum の EXPLAIN をサポート

Issue #3098 -REACT のベンダーを解除するため、ソースツリーで必要なくなりました

Issue #3107 -GPDB でテーブルスペースノードを非表示にする
Issue #3140 -pg_service.conf ファイルを使用した接続のサポートを追加する
Issue #3168 -GPDB の外部テーブルのサポート
Issue #3182 -Jasmine を v3 に更新する
Issue #3184 -フランス語の翻訳を追加する
Issue #3195 -サービス名を外部プロセスに渡す
Issue #3246 -CentOS/Apache の代わりに AlpineLinux と Gunicorn を使用するようにコンテナビルドを更新します

さらに、*PEP8* 準拠のためにさまざまな変更が加えられました

13.66.2 バグ修正

Issue #1173 -既存のノードにコメントを追加する
Issue #1925 -最初のクエリ後にクエリツールで列幅をサイズ変更できない問題を修正
Issue #2104 -インストーラーのプロパティの下のランタイム更新表示ファイルのバージョンと著作権
Issue #2249 -実行時にリロードした後にアプリケーションがハングしなくなりました
Issue #2251 -MacOS のセットアップで OSX の html スクロール方向が無視されるように修正
Issue #2309 -無効になっている CodeMirror インスタンスからのテキストの選択/コピーを許可する
Issue #2480 -Mac で動作しないコンテキストメニューのランタイムアップデートを修正
Issue #2578 -動作しない HTML アクセスキーの実行時更新を修正
Issue #2581 -テキスト選択のキーボードショートカットを修正
Issue #2677 -Windows での pgAdmin4 の象アイコンを更新する
Issue #2776 -リモートデスクトップで読めないフォントを修正する
Issue #2777 -サーバーツリーのスペーシングの問題を修正
Issue #2783 -ランタイムの更新で Windows デスクトップの空白の画面が修正されました
Issue #2906 -HiDPI 画面の表示の問題を修正
Issue #2961 -pgAgent スケジュールを作成するときの問題
Issue #2963 -外部プロセスツールでのユニコード処理を修正し、プロセスビューアーに完全なコマンドを表示する
Issue #2980 -クエリツールからクリップボードにテキストをコピーして、非表示の文字を追加する
Issue #2981 -デバッガーでのキーボードナビゲーションのサポート
Issue #2983 -再起動時に断続的に指定された version_numberValueError の問題を修正
Issue #2985 -ドラッグアンドドロップの問題を修正する
Issue #2998 -コンテナ起動時に TLS が有効になっていない場合、ポート 443 でリッスンしない
Issue #3001 -MacpgAdmin4.2.1 でのマウスホイールによるスクロールを修正
Issue #3002 -設定可能な幅でブロックのインデント/アウトデントを修正する
Issue #3003 -実行時更新の修正をクリップボードにコピー
Issue #3005 -pgAdmin4.2.1 でタブを選択できないランタイム更新の修正

Issue #3013 -バックグリッドにサブノードコントロールを表示する際のダッシュボードのマイナーな UI の問題を修正

Issue #3014 -シーケンスパラメーターの検証を修正

Issue #3015 -Greenplum データベースのサポートプロパティ

Issue #3016 -エラーが発生したときに/"messages/"ウィンドウでデバッグメッセージを利用できることを確認する

Issue #3021 -より明確にするためにスキャンとインデックススキャンの EXPLAIN アイコンを更新

Issue #3027 -クエリによって発生した通知を確実にキャプチャする

Issue #3031 -二重引用符と一重引用符が機能しない実行時の問題

Issue #3039 -count 列で間違った行カウントを引き起こす実行時の問題

Issue #3042 -更新時に空のダイアログボックスが発生する実行時の問題

Issue #3043 -macOSHighSierra でワードサイジングを引き起こすランタイムの問題

Issue #3045 -キーバインドのためにセルをコピーする際の問題を引き起こす実行時の問題

Issue #3046 -IE/FF の接続ステータスインジケーターを修正

Issue #3050 -Greenplum のパーティションテーブルの RE-SQL の表示を修正

Issue #3052 -列を変更するときに持ってはいけないプリミティブデータ型にサイズを含めないでください

Issue #3054 -ユーザーがキャンセル、開く、保存などのボタンコントロールを使用した後にキーボードショートカットを使用できることを確認します

Issue #3057 -リグレッションテストを更新して、Python3.5 および PG9.2 の問題を修正します

Issue #3058 -一部の状況で SQL/Dependencies/Dependents を更新していなかったツリービューノードのクリック時の処理を修正

Issue #3059 -Greenplum のテーブル統計を修正

Issue #3060 -RE-SQL の関数名の引用を修正

Issue #3066 -ビューのインデックスの列名が RE-SQL で適切に引用されていることを確認する

Issue #3067 -フィルターダイアログ CodeMirror がダイアログのボタンバーにオーバーフローしないようにする

Issue #3072 -[統計] タブに表示される pgAgent ジョブ履歴の行数に（構成可能な）制限を追加する

Issue #3073 -pgAgent ジョブの開始/終了時間グリッドフィールドがサブノードコントロールと同期していることを確認し、正しく検証します

Issue #3075 -テーブルの Select、Update、および Insert スクリプト生成の原因となる実行時の問題がロードに失敗する

Issue #3077 -有効になっている standards_conforming_strings への依存関係を削除する

Issue #3079 -テーブルに列を追加するときの tie/datetime 配列型の処理を修正しました

Issue #3080 -キーボードショートカットオプションの配置の問題を修正

Issue #3081 -不足しているリバースエンジニアリングされた SQL ヘッダーを追加し、シーケンスのステートメントを削除

Issue #3090 -ドライバーが必要に応じてメッセージの重大度をデコードする

Issue #3094 -クエリツールですべてのメッセージがサーバーから取得されていることを確認する

Issue #3099 -GPDB でのテーブルと列の作成を修正

Issue #3105 -大文字の主キー列で行を適切に更新できることを確認する

Issue #3135 -テーブルに大文字の OID と主キーがある場合に行を正しく挿入する

Issue #3122 -SSL オプションが pg_dump などの外部ツールにプッシュされていることを確認します

Issue #3129 -非 UTF8 互換ファイルのオープンを処理する

Issue #3137 -ダッシュボードテーブルからの SQL のコピーを許可する

Issue #3138 -Python3.x のテーブルスペーステストを修正

Issue #3150 -GPDB のファンクションリザーブ SQL を修正

Issue #3157 -外部プロセスツールでのユニコード処理を修正し、プロセスビューアーに完全なコマンドを表示する

Issue #3171 -OSX のトラックパッドを使用してファイルセレクターでスクロールできないランタイムの問題

Issue #3176 -Greenplum の関数統計を無効にする

Issue #3180 -インデックスが PG10 テーブルに表示されるようにする

Issue #3190 -GPDB で必要に応じてテストをスキップする

Issue #3196 -ファイルマネージャーがファイルとディレクトリ名を適切にエスケープすることを確認する

Issue #3197 -クッキーパスを適切に設定する

Issue #3200 -host パラメーターがサービスファイルから正しく取得されていることを確認します

Issue #3219 -Chrome の現在のバージョンに必要な ChromeDriver バージョンを更新する

Issue #3226 -影響を受けるフィールドの前にフィールドエラーインジケーターを移動して、スピナーやドロップダウンなどを不明瞭にしないようにします

Issue #3244 -クエリツールの履歴パネルに詳細なタイミング情報を表示する

Issue #3248 -Alertify ダイアログがモーダルであることを確認して、誤ってクリックして閉じないようにします

13.67 バージョン 2.1

発売日：2018-01-11

このリリースには、pgAdmin42.0 のリリース以降に報告された多くの機能と修正が含まれています

13.67.1 特徴

Issue #1383 -ツリービューとクエリツールで接続を色付けできるようにする

Issue #1489 -データフィルターウィンドウの選択クエリのユーザーインターフェイスを改善

Issue #2368 -クエリツールのデータ入力を改善する

Issue #2781 -クエリ結果の CSV とクリップボードのフォーマットを許可する

Issue #2802 -ツリービューとクエリツールで接続を色付けできるようにします。

Issue #2810 -クエリツール内でファイルをダブルクリックして開くことを許可する

Issue #2845 -クエリツールの/"変更を保存/"プロンプトをオプションにする

Issue #2849 -OID はあるが主キーがないテーブルのデータを編集するためのサポートを追加し、エディターを更新して保存時にすべての行値を取得するため、すぐにデフォルト値が表示され、その後の編集が可能

13.67.2 バグ修正

Issue #1365 -Windows インストーラーが無効な文字を含むパスを受け入れないようにする

Issue #1366 -Windows インストーラーの/NOICONS スイッチを修正

Issue #1436 -PPAS9.6 のサブプロシージャーで失敗するデバッガーの問題を修正します

Issue #1749 -以下を含む pgAgent モジュールの修正。1）ジョブをスケジュールするときに終了日よりも前の開始日を許可する、2）日時ピッカーがグリッドに表示されない、3）日時コントロールに検証エラーが適切に表示されない

Issue #2094 -スキーマの作成中にアクセスが拒否されたときに関連するエラーメッセージを表示する

Issue #2098 -一貫性のないエラーダイアログのタイトルをクリーンアップ

Issue #2258 -DATERANGE[] 型の処理を修正

Issue #2278 -ダイアログのヘッダーに長い名前を適切に表示する

Issue #2443 -ランタイムを終了する前にユーザーに確認する

Issue #2524 -自己参照関数のデバッグを修正

Issue #2566 -PostgreSQL10 の WAL ファイルの一時停止/再開のリプレイを修正

Issue #2624 -テーブルダイアログでスイッチのアニメーションが一貫していることを確認し、エラーが誤って表示されないようにします

Issue #2651 -推定行が関数の CREATE スクリプトに正しく含まれていることを確認する

Issue #2679 -ユーザーが URL を開き、クロスバーの付いた閉じるボタンをクリックすると、はじめにリンクが 2 回開かない

Issue #2705 -ユーザーは Windows で有効期限を追加できます

Issue #2715 -大規模なファイルをダウンロードし、進行状況をユーザーに通知できるようにする

Issue #2720 -pgpass ファイルを使用して認証する場合にパスワードの変更が成功することを確認する

Issue #2726 -オートコンプリート選択リストに長い名前を表示できるようにする

Issue #2738 -CodeMirror の行番号がメニューの上部に表示されないようにする

Issue #2748 -グリッドエディターポップアップに表示するときに JSON/JSONB を適切にフォーマットする

Issue #2760 -SSL 証明書またはキーを選択するときに、UI で予想されるパスのみを更新します。それらのすべてではありません

Issue #2765 -パスワードが「None」の場合、パスワードを復号化しません。これにより、一般的だが無害な例外/"ValueError：パスワードの復号化中は IV の長さを 16 バイトにする必要があります。/"

Issue #2768 -libpq>=10 の場合にのみ pgpass ファイルの指定を許可します

Issue #2769 -正しいキーボードショートカット。alt+でコードのコメントを外さないでください。クエリツールで。ctrl/cmd+にのみ応答することになっています

Issue #2772 -Panel のコンテキストメニューから外部リンクを削除する

Issue #2778 -ドメインが追加されたときにデータ型キャッシュが更新されるようにする

Issue #2779 -フィールドサイズを変更するときに列の照合が失われないようにする

Issue #2780 -自動インデントがスペース/タブの構成設定を尊重するようにする

Issue #2782 -配列などの特殊な型のレンダリングを処理する方法を再ハッシュ

Issue #2787 -型を作成するときに所有者名を引用する

Issue #2806 -lc_messages に基づいてデータベースエラーをデコードしようとする

Issue #2811 -プロセスの出力をそのまま表示する

Issue #2820 -バックアップと復元の実行時に利用できるログ

Issue #2821 -lc_messages に基づいてデータベースエラーをデコードしようとする

Issue #2822 -配列などの特殊な型のレンダリングを処理する方法を再ハッシュします。

Issue #2824 -多数のグラフィカル Explain レンダリングの問題を修正

Issue #2836 -テーブルプロパティのカウントされた行の表示を修正

Issue #2842 -多数のグラフィカル Explain レンダリングの問題を修正

Issue #2846 -テーブルの行を手動でカウントしてプロパティをレンダリングするオプションを追加

Issue #2854 -ユーティリティの出力キャプチャのエンコーディングを修正

Issue #2859 -フォーム上で何かを日食した場合にフォーム検証メッセージを閉じることを許可します

Issue #2866 -SSL 証明書にサーバーモードで仮想ファイルシステムルートを使用するときにサーバーにフルパスを表示しないようにする

Issue #2875 -ユーザーがタブを変更して戻った場合に、スクロールの場所がクエリツールのデータグリッドに保持されるようにします

Issue #2877 -テキストエリアから 4000 文字の人為的な制限を削除する

Issue #2880 -データグリッドで空白を適切に尊重する

Issue #2881 -タイムゾーンのない時間のサポートを修正

Issue #2886 -デフォルトの主キー値で挿入しようとすると失敗する問題を解決する

Issue #2891 -アプリを終了せずにユーザーのパスワードを変更できるようにする

Issue #2892 -通知メールを送信できない場合、パスワードの変更を拒否する（およびユーザーに通知する）

Issue #2908 -コードミラーの/r/n 行末が原因で失敗していた Windows でのバンドル作成を修正

Issue #2918 -MSVCWindows ビルドをビルドするときに不足している init.py を backports.csv に追加します

Issue #2920 -HTTPD ログを必要に応じてコンテナ stdout/stderr にプッシュする

Issue #2921 -以下を含む pgAgent モジュールの修正。1）ジョブをスケジュールするときに終了日よりも前の開始日を許可する、2）日時ピッカーがグリッドに表示されない、3）日時コントロールに検証エラーが適切に表示されない

Issue #2922 -デスクトップモードですべての要求でユーザーをログインしないでください。1 回だけ

Issue #2923 -無効になっているときにユーザーがファイルマネージャーの選択ボタンを押さないようにする

Issue #2924 -フィルターデータダイアログのレイアウトをクリーンアップ

Issue #2928 -応答が遅い新しいサーバーへの複数の接続が誤って開始されるのを防ぐ

Issue #2934 -ファイルダイアログの割り当て前の参照エラーを修正

Issue #2937 -ファイルダイアログでディレクトリをファイルとして選択しようとするのを防ぐ

Issue #2945 -ビューのトリガーで無効なオプションを選択できないようにする

Issue #2949 -FTS 辞書の完全な SQL を表示する

Issue #2952 -デスクトップモードでセキュリティ URL をレンダリングしようとしないでください

Issue #2954 -検証エラーテキストを選択できるようにする

Issue #2974 -EXPLAIN/EXPLAINANALYZE 実行時にメッセージタブをクリアする

Issue #2993 -ビュー/マットビューのビューデータを修正する

13.68 バージョン 2.0

発売日：2017-10-05

このリリースには、pgAdmin41.6 のリリース以降に報告された多くの機能と修正が含まれています

13.68.1 特徴

Issue #1918 -ServerDialogue にフィールドを追加して、ユーザーがツリービューに表示するデータベースのサブセットを指定できるようにします

Issue #2135 -アプリケーションのロードを大幅に高速化

Issue #2556 -低速または高速の接続障害を許可する

Issue #2579 -ファイルブラウザビューをデフォルトでリストし、構成可能にする

Issue #2597 -クエリをダッシュボードからキャンセルし、サブノードコントロールに追加情報を表示できるようにします

Issue #2649 -認証のための SSL 証明書の使用のサポート

Issue #2650 -pgpass ファイルの使用をサポート

Issue #2662 -サーバーモードとデスクトップモードの両方ですぐに動作する事前構成されたパスが付属しています

Issue #2689 -新しいデザインでアイコンを更新し、メニューから削除して UI を整理します

13.68.2 バグ修正

Issue #1165 -サーバーが切断されている場合、ダッシュボード上のグラフデータの継続的なポーリングを防止する

Issue #1697 -CodeMirror のバージョンを更新する

Issue #2043 -パラメーターを持つトリガー関数を適切に処理する

Issue #2074 -$の引用符を一貫させる

Issue #2080 -大きな BLOB データを含むテーブルからデータを（SQL エディターを使用して）ロードするとブラウザーがハング/クラッシュする問題を修正

Issue #2153 -大きなファイルのアップロードのハンドラインを修正し、発生する可能性のあるエラーを適切に表示する

Issue #2168 -CodeMirror のバージョンを更新する

Issue #2170 -リグレッションテストで SSL をサポート

Issue #2324 -SQL タブの PostGIS データタイプを修正、テーブル、列、外部テーブル、およびタイプノードのダイアログを作成/更新

Issue #2447 -CodeMirror のバージョンを更新する

Issue #2452 -Centos7 に pgadmin4-v11.5 をインストールする

Issue #2501 -Windows での照合テストを修正し、テストのためにデフォルトの「POSIX」照合の使用を「C」照合に置き換えます

Issue #2541 -MacOS での特殊キーの使用に関する問題を修正する

Issue #2544 -カスタムタイプを使用するときに生成される不正な形式のクエリを修正する

Issue #2551 -パーティションの表領域を表示する

Issue #2555 -関数/プロシージャからメッセージが適切に表示されなかったクエリツールの問題を修正

Issue #2557 -タブスタイルを整理する

Issue #2558 -デタッチされたタブを閉じるときにタブバーが非表示になるのを防ぐ

Issue #2559 -ツールボタンのスタイルが予期せず変更されないようにする

Issue #2560 -ビュー「CREATEScript」の問題を修正

Issue #2562 -CodeMirror のバージョンを更新する

Issue #2563 -非標準の仮想ディレクトリの下のパスを修正する

Issue #2566 -PostgreSQL10 の WAL ファイルの再生の一時停止/再開を修正

Issue #2567 -適切なデータベース接続を使用して、データベースの [プロパティ] タブでデフォルトの権限を取得します

Issue #2582 -バックアップモジュールの空の文字列の場合、圧縮率の設定を解除します

Issue #2586 -機能テストのクリーンアップ

Issue #2590 -矢印キーを使用してクエリ履歴を移動できるようにする

Issue #2592 -デバッグモードで Flask がサービスを 2 回初期化するのを止める

Issue #2593 -babel-polyfill が古い qWebKits にロードされていることを確認する

Issue #2594 -新しいデータベースの切断を修正する

Issue #2596 -Webpack の実行中に適切な NODE_ENV 環境を定義する

Issue #2606 -メンバーシップ制御でロール名がエスケープされていることを確認する

Issue #2616 -ドメイン作成ダイアログが開かず、セキュリティラベルコントロールのフォントサイズの問題

Issue #2617 -webpack.config.js に欠落している pgagent ファイルを追加する

Issue #2619 -テーブルのインデックス列名の引用を修正

Issue #2620 -pgAgent ジョブの作成中に、ジョブタイプがバッチに設定されている場合、データベース名を空白（''）に設定します

Issue #2631 -数値セルがコードから削除されたため、整数制御のためにセルのマッピングを「数値」から「整数」に変更

Issue #2633 -pgAgent ジョブステップの問題を修正する

Issue #2634 -クイックリンクから新しいサーバーを追加する

Issue #2637 -クエリ結果がコピーされた後も機能するように Copy を修正

Issue #2641 -ユーザー管理の問題-スタイル設定とユーザーを適切に編集できない

Issue #2644 -チェックマークボックスがフレームから切断されたアラート通知メッセージを修正

Issue #2646 -vendor ディレクトリを参照していた load-node.gif のパス参照を修正

Issue #2654 -日時ピッカーを更新する

Issue #2655 -pgAgent ジョブの接続文字列の検証を修正

Issue #2656 -pgAgent で Datetimepicker を下から展開してカレンダーが非表示にならないように変更

Issue #2657 -pgAgent ジョブの更新中の開始/終了時間、平日、月日、月、時、分の変更を保存する際の構文エラーを修正

Issue #2659 -テーブルの列の変数を追加/更新できない問題を修正する

Issue #2660 -[履歴] タブで行を選択できません

Issue #2668 -単一の引数を持つトリガーの RE-SQL を修正

Issue #2670 -ユーザー（開発者）がデータモデルに検証関数を実装していない場合のデフォルトのバリデーターの検証を改善

Issue #2671 -bigint、real、float、double 精度の配列データ型のフォーマットを修正

Issue #2681 -テストを実行する前にクエリツールのオプションをリセットする

Issue #2684 -パスワードプロンプトダイアログのレイアウトを修正

Issue #2691 -pgAdmin42.0 バージョンにデータの表示オプションがありません

Issue #2692 -pgAdmin4 のドメインのベースタイプがありません

Issue #2693 -ユーザーリストはユーザーマッピング pgAdmin4 で利用できません

Issue #2698 -戻り値の型がないため、ユーザーは関数を作成できません

Issue #2699 -pgAdmin4 でのフィルタされた行の問題

Issue #2700 -クエリが正常に実行された後にキャンセルボタンが表示される

Issue #2707 -pgAdmin4 でトリガーボタンを無効にしない

Issue #2708 -MoveObjectstoanothertablespace で%s(new_tablespace)s の代わりに表領域名を表示する必要があります

Issue #2709 -'pg' で始まるスキーマのユーザーリレーションを表示する

Issue #2713 -サーバー再起動時に時々発生する例外を修正

Issue #2742 -代替ロールを使用してデータベースに接続しても、リカバリ状態を確認するときにエラーが発生しないことを確認します。

13.69 バージョン 1.6

発売日：2017-07-13

このリリースには、pgAdmin41.5 のリリース以降に報告された多くの機能と修正が含まれています

13.69.1 特徴

Issue #1344 -クエリツール、デバッガー、Web ブラウザーのタブを必要に応じて別のモニターに移動できるようにします

Issue #1533 -ダイアログが開かれたときに最初に有効になったフィールドにフォーカスを設定する

Issue #1535 -Escape でキャンセル、Enter で保存/OK、F1 でヘルプを教える

Issue #1971 -同じクエリが連続して複数回実行された場合、クエリツールの結果グリッドで列のサイズを保持する

Issue #1972 -より自然なワークフローのために、ダーティクエリを破棄するのではなく保存するように求めます

Issue #2137 -クエリツールの結果のオンデマンド読み込み

Issue #2191 -hostaddr 接続パラメーターのサポートを追加します。これは、Kerberos/SSPI や友達とうまくやるのに役立ちます

Issue #2282 -クエリ履歴タブを見直して、履歴と完全なクエリテキストを参照できるようにします

Issue #2379 -各行の [保存] をクリックせずにテーブルに複数の新しい行を挿入することをサポート

Issue #2485 -実行時にズームレベルをリセットするショートカットを追加

Issue #2506 -ユーザーがダッシュボードパネルを閉じることを許可する

Issue #2513 -SQL エディターで中括弧のマッチングと中括弧を閉じるための設定を追加

13.69.2 バグ修正

Issue #1126 -さまざまな FTS 辞書のクリーンアップ

Issue #1229 -イベントトリガーのデフォルト値と SQL フォーマットを修正

Issue #1466 -可変個引数を持つプロシージャーをデバッグしようとするのを防ぐ

Issue #1525 -$の引用符を一貫させる

Issue #1575 -EPAS9.2+でセキュリティラベルを適切に表示する

Issue #1795 -外部および範囲型の検証を修正

Issue #1813 -シノニムを作成するときに PPAS9.2-9.4 のパッケージを一覧表示する

Issue #1831 -inet が連結のためにテキストにキャストする必要がある EPAS9.2 のサーバー統計表示を修正

Issue #1851 -テーブルを返す関数の SQL を正しくリバースエンジニアリング

Issue #1860 -列を追加/編集するときにデフォルト値が尊重されるようにする

Issue #1888 -pgAgent のジョブステップとスケジュールに関するさまざまな問題を修正する

Issue #1889 -pgAgent のジョブステップとスケジュールに関するさまざまな問題を修正する

Issue #1890 -pgAgent のジョブステップとスケジュールに関するさまざまな問題を修正する

Issue #1920 -最初に保存したときに再起動した後ではなく、保存したパスワードがすぐに有効になるようにする

Issue #1928 -倍精度 [] 型の処理を修正

Issue #1934 -TSV データで正常に動作するようにインポート/エクスポートを修正

Issue #1999 -クエリ結果をアンマウントされた USB ドライブに保存するときの警告を正しく処理する

Issue #2013 -GrantWizard のデフォルトサイズを大きくして、小さなディスプレイにデフォルトサイズで権限を適切に表示できるようにします

Issue #2014 -ユーザーがパッケージ/プロシージャー fire_emp でデバッグを停止した場合に表示される予期しない動作を修正するには

Issue #2043 -パラメーターを持つトリガー関数を適切に処理する

Issue #2078 -サイズ変更時に SQL エディタービューを更新して、新しいビューポート用にコンテンツが再レンダリングされるようにします

Issue #2086 -ロールメンバーシップの WITHADMIN オプションの編集を許可する

Issue #2113 -インデックス/除外制約を変更する際の検証ロジックを修正

Issue #2116 -言語と外国語のテーブルのダイアログでダイアログのヘルプボタンを有効にする

Issue #2142 -Windows での GrantWizard のキャンセルを修正

Issue #2155 -列定義からのサイズの削除を修正

Issue #2162 -非スーパーユーザーが自分の関数をデバッグし、グローバルブレークポイントを設定できないようにする

Issue #2242 -cache-node フィールドを使用した NodeAjaxControl キャッシングの問題を修正し、トリガーとイベントトリガーノードに cache-node フィールドを追加して、ユーザーが新しいトリガーファンクションを作成するたびに NodeAjaxControl のサーバーから新しいデータを取得します

Issue #2280 -EPAS の RE-SQL でプロシージャフラグ

（IMMUTABLESTRICTSECURITYDEFINERPARALLELRESTRICTED）を適切に処理する

Issue #2324 -SQL タブの PostGIS データタイプ、テーブル、列、外部テーブル、およびタイプノードの作成/更新ダイアログを修正

Issue #2344 -クエリツールで ctrl-c/ctrl-v が機能しない問題を修正

Issue #2348 -すべての行/列が選択/選択解除される QueryToo/ViewData で列のサイズを変更するときの問題を修正します

Issue #2355 -子の名前を変更するときに親ノードを適切に更新する

Issue #2357 -統計をより確実にキャッシュする

Issue #2381 -ビューの RE-SQL を修正して、トリガー関数名を適切に修飾

Issue #2386 -トリガーダイアログからトリガーの有効化/無効化ステータスを表示し、トグルできるようにします

Issue #2398 -Windows でローカルアドレスのプロキシサーバーをバイパスする

Issue #2400 -データ編集時のデフォルト/ヌル値のクリーンアップ

Issue #2414 -ユーザーが既存のグループを複製するサーバーグループの名前を変更または作成しようとした場合のエラー処理を改善

Issue #2417 -マルチカラム pkey のカラムを正しく並べる

Issue #2422 -ヘッダーと DROP ステートメントでテーブル名が間違っているルールの RE-SQL を修正

Issue #2425 -エディットグリッドで行を削除するときに複合主キーを正しく処理する

Issue #2426 -メンバーのない ENUM 型の作成を許可する

Issue #2427 -オブジェクトが実際に存在すると思われるときに存在することを確認するための多数の欠落チェックを追加します

Issue #2435 -スクリプトオプションを使用するときにデータベース ID をクエリツールに渡す

Issue #2436 -テーブルの UPDATE スクリプトを生成するときに最後のプレースホルダーが含まれていることを確認する

Issue #2448 -ブールチェックボックスが値を正しい順序で循環することを確認する

Issue #2450 -PG10 の統計タブのエラーを修正しました。また、新しいバージョン管理に合わせて、10.0_plus テンプレートディレクトリの名前を 10_plus に変更します。

Issue #2461 -ユーザーが列からデフォルト値を適切に削除できるようにする

Issue #2468 -関数作成スクリプトがコンパイルされない問題を修正

Issue #2470 -結果のポーリング中に発生する断続的なエラーを修正

Issue #2476 -UI の改善を含むクエリ結果グリッドの改善、行、列、または任意のセルブロックからのコピー/貼り付け

Issue #2477 -テキストエディターが結果グリッドの適切な場所にレンダリングされるようにする

Issue #2479 -pgAdmin は SPA であるため、ホームページにリンクするメニューアイコンは必要ありません

Issue #2482 -クエリツールのタブにもっとわかりやすい名前を使用する

Issue #2486 -機能テストが正しいテスト設定データベースを使用していることを確認する

Issue #2487 -非同期呼び出しを使用して入力された設定値のクライアント側キャッシュを維持する

Issue #2489 -大規模なデータセットでのクリップボードの処理を修正

Issue #2492 -Web モードでのセットアップ中に初期パスワードが適切にハッシュされていることを確認します

Issue #2498 -bytea[]、および'infinity'::real/real[] を適切に扱う

Issue #2502 -bytea[]、および'infinity'::real/real[] を適切に扱う

Issue #2503 -欠落した/削除された同義語を適切に処理する

Issue #2504 -最近の整数/数値の変更を処理するように MatView と pgAgent モジュールを更新します

Issue #2507 -取り消されたパブリック特権が関数の RE-SQL に表示されるようにする

Issue #2518 -サーバー保存時のエンコーディングの問題を修正

Issue #2522 -結果グリッドの [すべて選択] の速度を向上させます

Issue #2527 -列定義が NOTNULLTRUE であり、デフォルト値がない場合のテーブル行の削除を修正

Issue #2528 -ビューのトリガーにブレークポイントを設定できるようにする

Issue #2529 -ドメインとドメインの制約に関する多くの問題を解決する

Issue #2532 -更新される子が 1 つある場合にノードを正しく更新する

Issue #2534 -CREATETABLEOF<type>の処理を修正しました

Issue #2535 -履歴クリア機能を修正

Issue #2540 -サーバー作成時にパスワードの保存オプションが有効になっていることを確認する

13.70 バージョン 1.5

発売日：2017-05-19

このリリースには、pgAdmin41.4 のリリース以降に報告された多くの機能と修正が含まれています。

13.70.1 特徴

Issue #2216 -クエリツールで列または行を選択できるようにする

13.70.2 バグ修正

Issue #2225 -オブジェクトタイプが hidden に設定されている場合、オブジェクトを作成するためのメニューオプションを非表示にします。ジャスミンテストが含まれています

Issue #2253 -CSV ファイルのダウンロード機能のさまざまな問題を修正する

Issue #2257 -データエディターでの null とデフォルト値の処理を改善する

Issue #2271 -トリガーが無効になっているときに更新されたときに、トリガーアイコンを/"有効/"に戻さないでください

Issue #2284 -純粋な数値名でテーブルを作成できるようにする

Issue #2292 -以前に接続したデータベースにのみ再接続する

Issue #2314 -CSV ファイルのダウンロード機能のさまざまな問題を修正する

Issue #2315 -生の値を並べ替えてクライアント側で整形することにより、統計ビューでのサイズの並べ替えを修正しました。prettyfying 関数の Jasmine テストが含まれています

Issue #2318 -外部テーブルの列を正しく並べる

Issue #2331 -新しいツリービューノードが正しい位置に追加されるように二分探索アルゴリズムを修正

Issue #2336 -ツリービューノードを更新するときに inode 情報を更新します。

Issue #2339 -ツリービューを水平にスクロールできるようにする

Issue #2350 -関数内のデフォルトのパラメーターの順序付けの処理を修正しました

Issue #2354 -ユーザーが英語以外の設定言語を変更した場合に動作していなかったバックアップモジュールを修正しました

Issue #2356 -クエリツールで編集モードで行を削除するとスローされるエラーが適切に表示されるようにする

Issue #2360 -CSV ファイルのダウンロード機能のさまざまな問題を修正する

Issue #2369 -UnicodeBOM を含むファイルのロードをサポート

Issue #2377 -PostgreSQL10 互換性のために psycopg2 バージョンを更新する

Issue #2379 -データエディターでの NULL/DEFAULT の処理をさまざまに改善

Issue #2405 -外部プロセス管理のためにオブジェクト名が適切にエスケープされていることを確認する

Issue #2410 -PostgreSQL10.0 の互換性問題を修正する

13.71 バージョン 1.4

発売日：2017-04-13

このリリースには、pgAdmin41.3 のリリース以降に報告された多くの機能と修正が含まれています。

13.71.1 特徴

Issue #2232 -サーバーへの接続を作成するときに/"SavePassword/"オプションをグレーアウト/無効にする機能を追加

Issue #2261 -SQL タブに Greenplum のテーブル DDL を表示する

Issue #2320 -ドイツ語の翻訳を追加

13.71.2 バグ修正

Issue #2077 -pgAdmin ジョブに欠落している/"RunNow/"オプションを追加する

Issue #2105 -テーブルダイアログの検証を修正して、名前が削除され、autovac のカスタム設定が有効になっている場合、[保存] ボタンが有効になりません

Issue #2145 -バックアップからテーブルを復元する問題を解決する

Issue #2187 -Windows インストールをアップグレードする前に web/ディレクトリがクリアされていることを確認してください

Issue #2190 -ユーザーがブラウザーからの予測できない動作ではなく、ログイン時または設定から UI 言語を選択できるようにします

Issue #2226 -ユーザーの学習に役立つ無効なボタンのツールチップを表示する

Issue #2241 -ネストされたスキーマの数値制御の検証を修正

Issue #2243 -Unicode 名のデータベースのドロップを修正

Issue #2244 -ユーザーが列のないテーブルでデータを表示するときにエラーが表示されるのを防ぐ

Issue #2246 -不足している中かっこをリバースエンジニアリングされたファンクションの SQL ヘッダーブロックに追加する

Issue #2258 -DATERANGE[] 型の処理を修正

Issue #2264 -ubuntu14.04Linux64 での Python3.4 の新しい pgAdmin4 セットアップに表示されるエラーメッセージ *ExtDeprecationWarning* を解決します

Issue #2265 -テーブルの解決されたインポート/エクスポートの問題

Issue #2274 -切り捨てられたテーブル名を適切に処理する

Issue #2277 -さまざまなファイルシステムのエンコード/デコード関連のケースを解決

Issue #2281 -サーバーを切断した後にメニューが更新されるようにする

Issue #2283 -複数の値のデフォルト選択を設定する前に、セルが複数選択モードかどうかを確認する

Issue #2287 -クエリツールでユーザーが直接入力した EXPLAIN クエリを適切に処理する

Issue #2291 -クエリツールでのエラーの強調表示を修正

Issue #2299 -QString の使用を修正

Issue #2303 -ヘッダーをクリックしている間のバックグリッドの昇順/降順の並べ替えを修正しました

Issue #2304 -バックアップからテーブルを復元する問題を解決する

Issue #2305 -ジェネリックファンクション qtLiteral が非 ASCII 文字を含む場合に値を適切に適応しない問題を解決

Issue #2310 -クエリツール/デバッガーが新しいブラウザータブで開くダイアログヘルプを修正

Issue #2319 -pgAdmin4 ロゴをクリックすると不正なエラーが発生する問題を解決する

Issue #2321 -親コレクションノードがロードされていない場合に新しいノードを追加するときのブラウザーツリーの機能の改善

Issue #2330 -クエリツールは表示しますが、サーバーから返された HTML を結果グリッドにレンダリングしないことを確認します

13.72 バージョン 1.3

発売日：2017-03-10

このリリースには、pgAdmin41.2 のリリース以降に報告された多くの機能と修正が含まれています。

13.72.1 特徴

Issue #2036 -クエリツールの効率-SlickGrid 結果セットフォーマットの効率

Issue #2038 -クエリツールの効率-ポーリング時のインクリメンタルバックオフ

Issue #2163 -シンタックスハイライトをより見やすくする

Issue #2210 -Python バージョンごとの代わりにユニバーサル Python ホイールをビルドする

Issue #2215 -シンタックスハイライトの色の視認性を向上させる

13.72.2 バグ修正

Issue #1796 -pgAdmin ジョブに欠落している/"RunNow/"オプションを追加する

Issue #1797 -DATA_DIR のエンコーディングの問題を解決する

Issue #1914 -解決されたエラー utf8' コーデックはバイトをデコードできません

Issue #1983 -Sql クエリにアラビア語の文字が含まれるバグを修正

Issue #2089 -PARALLELSAFE|UNSAFE|RESTRICTED のサポートを追加する

Issue #2115 -除外制約のリバースエンジニアリングされた SQL を修正する

Issue #2119 -長整数と小数の表示を修正

Issue #2126 -EDB 関数とプロシージャーの設定のノードラベルを修正

Issue #2151 -クエリツールでサイズのない varlen 列タイプを正しく表示する

Issue #2154 -長整数と小数の表示を修正

Issue #2159 -クエリエディターが Python2.6 で動作しない問題を解決する

Issue #2160 -「ascii」コーデックが位置 66 のバイト 0xc3 をデコードできるようにするさまざまなエンコーディングの修正：序数が範囲にありません（128）

Issue #2166 -テーブルの解決されたインポート/エクスポートの問題

Issue #2173 -SequencesAPI テストケースが PG9.2 および PPAS9.2 で機能しない問題を解決

Issue #2174 -さまざまなファイルシステムのエンコード/デコード関連のケースを解決

Issue #2185 -ツリービューの並べ替え列を削除しました

Issue #2192 -スタートアップが完了したかどうかサーバーを適切にポーリングするように、スタートアップの完了テストを修正しました

Issue #2198 -createSQL 生成時の関数引数を修正

Issue #2200 -異なるスキーマでイベントトリガー機能を適切に処理する

Issue #2201 -テーブル名が同時に変更された場合のチェック制約の名前付けを修正

Issue #2202 -非 ASCII 文字が原因で Dependents クエリが失敗する問題を修正

Issue #2204 -pgadmin4 ジョブがアクティビティを表示しない問題を修正しました

Issue #2205 -クエリツールでのブール値の表示を修正

Issue #2208 -クエリツールのデータの表示モードで主キーの列名が引用符で囲まれていることを確認する

Issue #2212 -親グループが削除されたときにサーバーが削除されるようにする

Issue #2213 -ブラウザツリーの右クリックを有効にする

Issue #2218 -新しいブール値を編集するときに正しい不定状態を表示する

Issue #2228 -サーバーに対してランタイムを認証する

Issue #2230 -サーバーモードの小型ディスプレイで Slonik のロゴがログインダイアログを覆い隠すのを防ぐ

13.73 バージョン 1.2

発売日：2017-02-10

このリリースには、pgAdmin41.1 のリリース以降に報告された多くの機能と修正が含まれています。

13.73.1 特徴

Issue #1375 -ランタイムを QtWebKit から QtWebEngine に移行する

Issue #1765 -機能を見つけて、正規表現とグループ置換で置き換える

Issue #1789 -データ出力パネルの列幅はデータに収まる必要があります（pgAdminIII として）

Issue #1790 -[Web] フィールドの値の/"null/"への設定をサポート

Issue #1848 -macOSappbundle にインポートなどのための postgresql バイナリがありません

Issue #1910 -ファイルマネージャーに最後に使用したディレクトリを記憶する

Issue #1911 -ファイルマネージャーでのダイレクトパスナビゲーション

Issue #1922 -破損した構成データベースの処理を改善する

Issue #1963 -中国語（簡体字）の翻訳を追加する

Issue #1964 -ソース tarball とともに docstarball を作成する

Issue #2025 -SQL エディターでワードラップできるようにする

Issue #2124 -SQL テンプレートの場所を簡素化するテンプレートローダーを作成し、重複するテンプレートを削除する

13.73.2 バグ修正

Issue #1227 -デバッガーリスナーの開始エラーと実行間のリセットの改善されたエラーメッセージを表示する

Issue #1267 -MINIFY_HTML がドキュメントで機能しない問題を修正する

Issue #1364 -ダイアログコントロールボタンが一貫していることを確認する

Issue #1394 -テーブルダイアログの列指定の問題を修正

Issue #1432 -拡張 OSX ファイルブラウザ

Issue #1585 -スクリプトをネットワークに保存できません

Issue #1599 -Grant ウィザードが名前に特殊文字を含むオブジェクトで動作することを確認する

Issue #1603 -外部ユーティリティのオブジェクト名の引用を修正しました。

Issue #1679 -バックグラウンドプロセスエグゼキューターをリエンジニアリングして、依存するコンポーネントの一部のビルドがフォークされた子での作業をサポートしていないため、sqlite の使用を回避

Issue #1680 -Firefox のクエリツールで列ヘッダーを正しい幅でレンダリングする

Issue #1729 -ロールオプションの表示を改善する

Issue #1730 -プロパティパネルとロールダイアログの両方でのロールメンバーシップの表示を改善

Issue #1745 -デバッガーで作業するときにブレークポイントが適切にクリアされることを確認する

Issue #1747 -新しく作成されたトリガーをツリービューに追加する

Issue #1780 -行番号の幅が増えるにつれて SQLEditor のガターのサイズを適切に設定する

Issue #1792 -ファイルとフォルダーをアルファベット順にリストする

Issue #1800 -データベースのテンプレートプロパティを適切に処理する

Issue #1801 -datallowconn==false でデータベースを処理する

Issue #1807 -クエリツールでファイルが変更されたことを適切に検出し、それに応じてフラグを設定する

Issue #1830 -EPAS サーバーで ROLESQL をリバースエンジニアリングするときの SQL エラーを修正

Issue #1832 -[依存関係] タブの空のリストにアクセスしようとするのを防ぐ

Issue #1840 -インデックス列と除外制約の NULL と ASC/DESC オプションを適切に有効/無効にします

Issue #1842 -RE-SQL でインデックス列を正しい順序で表示する

Issue #1855 -トリガーファンクションを作成するときに、ダイアログパネルにエラーが表示されないようにする

Issue #1865 -SQL をリバースエンジニアリングするときにドメインを適切にスキーマ修飾する

Issue #1874 -Windows ランタイムにファイルリソースを追加する

Issue #1893 -Unique コンストレインのリフレッシュを修正

Issue #1896 -新しく作成された除外制約のプロパティを取得するために正しい OID を使用する

Issue #1899 -関数の所有権を指定するときにロール名を適切に引用する

Issue #1909 -実行時に起動エラーをより適切に処理する

Issue #1912 -トリガーから関数に渡される引数を適切にフォーマットする

Issue #1919 -行へのすべての変更がデータエディターに保存されていることを確認します

Issue #1924 -必要に応じてビューを編集するときにのみ check_option が設定されていることを確認します

Issue #1936 -不正な形式のデータにつながるため、行の/"DownloadasCSV/"バッチから/r/n を削除しないでください

Issue #1937 -新しいスキーマの mSQL を正しく生成する

Issue #1938 -統計グリッドの数値のソートを修正

Issue #1939 -ウィンドウサイズの動的デフォルトを更新しました（90 %× 90 %）

Issue #1949 -トリガー関数名がトリガー RE-SQL でスキーマ修飾されていることを確認する

Issue #1951 -oid 値が maxint を超えたときにテーブルの列を参照できない問題を修正

Issue #1953 -クエリツールで受信した表示メッセージと通知を追加する

Issue #1961 -Python3 のアップグレードチェックを修正

Issue #1962 -ツリービューのコレクションノードが UI で翻訳されていることを確認する

Issue #1967 -調整ごとにレイアウトの変更を保存する

Issue #1976 -ユーザーが選択できない UI の要素を選択しないようにする

Issue #1979 -プロパティダイアログで関数の引数を正しく扱う

Issue #1986 -マルチバイトのパスとファイル名で空のファイルが保存されるさまざまなエンコーディングの問題を修正

Issue #1992 -オートコンプリートで識別子を正しく引用する

Issue #1994 -編集グリッドに変更を表示するように更新

Issue #2000 -9.6 以降のテーブルスペースで effective_io_concurrency を設定できるようにする

Issue #2005 -VACUUM のさまざまなスペルミスを修正

Issue #2006 -postgis ジオメトリ列を持つテーブルでテーブル名を変更するか、スキーマを設定するときのエラーを修正

Issue #2007 -最初/最後の 100 を表示するときに pkey で行を正しくソートする

Issue #2009 -エラー処理が正しく動作するようにインデックスでアクセス方法が変更された場合、列リストを適切にリセットする

Issue #2012 -名前のないサーバーグループの作成を防ぐ

Issue #2015 -ユーザーがプロパティセクションから行トリガー値を変更しようとしたときにトリガーオプションを有効にする

Issue #2024 -allowconn=False でデータベースのコメントやその他のオプションを適切に処理する

Issue #2026 -デバッグを許可する前の pldbgapi 拡張機能と関数の検出を改善

Issue #2027 -一貫性のないテーブルスタイルを修正する

Issue #2028 -付与ウィザードのダブルスクロールバーの表示を修正

Issue #2032 -ダッシュボードの時間フォーマットを修正

Issue #2033 -依存関係/依存関係パネルにユニーク制約と除外制約のアイコンを表示する

Issue #2045 -ドキュメントページの著作権年を更新する

Issue #2046 -Windows で pgadmin4 のリグレッションを設定するときのエラーを修正

Issue #2047 -ダイアログがナビゲーションバーの下に移動できないようにする

Issue #2061 -インデックス列と除外制約の NULL と ASC/DESC オプションを適切に有効/無効にします

Issue #2065 -プロパティリストの除外制約と外部キーの列の表示を改善

Issue #2069 -テーブルのプロパティに表示される正しいテーブルスペース

Issue #2076 -サイズ設定された時間/タイムスタンプ列を正しく処理する

Issue #2109 -著作権年の更新

Issue #2110 -存在しなくなった保存済みディレクトリを正常に処理する

Issue #2112 -右クリックで初期データベースのコメントを有効にする

Issue #2133 -UPDATE/DELETE クエリのグラフィカルなクエリプランの表示を修正

Issue #2138 -読み取り専用のグリッドエディターでのゼロの表示を修正

Issue #2139 -マテリアライズドビューと [SQL] タブの下のビューで表示されるメッセージ（サーバーへの接続が失われました）の問題を修正しました

Issue #2152 -/"char/"列の処理を修正

Issue #2156 -Jinja2 の新しいバージョン（2.9.5 以降など）の互換性修正を追加しました

13.74 バージョン 1.1

発売日：2016-10-27

このリリースには、pgAdmin41.0 のリリース以降に報告された多くの機能と修正が含まれています。

13.74.1 特徴

Issue #1328 -Python3.5 サポートを追加する

Issue #1859 -ダッシュボードのアクティビティタブに待機情報を含める

13.74.2 バグ修正

Issue #1155 -ユーザーがシーケンスを作成するときに開始値を表示する

Issue #1531 -/"文字列インデックスは整数である必要があります/"エラーが表示されるビューとマテリアルビューの更新権限を修正しました

Issue #1574 -PG および EPAS サーバーのセキュリティラベルの SQL を SQL ペインに表示

Issue #1576 -プロシージャーのプロパティでセキュリティラベルオプションを使用できるようにする

Issue #1577 -パッケージ関数とプロシージャーでデバッグオプションを利用できるようにする

Issue #1596 -evnt_turncate から evnt_truncate へのスペルミスを修正する

Issue #1599 -Grant ウィザードが名前に特殊文字を含むオブジェクトで動作することを確認する

Issue #1622 -シノニムを作成するときに特殊文字を使用する問題を修正

Issue #1728 -オブジェクトが編集された後に更新されるプロパティ

Issue #1739 -ユーザーが...

Issue #1785 -最初の接続時にサーバーの種類を正しく識別する

Issue #1786 -新しいサーバーを追加するときに errorModelunset プロパティが正しく設定されていることを確認する

Issue #1808 -pgAgent ジョブスケジュールで秒を有効な値に設定する

Issue #1817 -ca-cert または ca-full が追加されたサーバーの場合、メッセージ「サーバーは ssl をサポートしていません」を表示する

Issue #1821 -オプションで Mac アプリバンドルとディスクイメージの両方に署名します

Issue #1822 -接続時にサーバーからの非 ASCII 応答を処理する

Issue #1823 -Windows インストーラーに署名しようとすると、使用可能な証明書がない場合は警告が表示されます

Issue #1824 -pgAgent のドキュメントを追加する

Issue #1835 -ユーザーが付与ウィザードでテーブルとシーケンスの SELECT 権限を選択できるようにします

Issue #1837 -「サーバーへの接続が失われました」エラーの原因となっていた FTS 辞書の更新を修正しました

Issue #1838 -正しい親がロードされていない場合、ツリーブラウザで間違った親を持つ新しいオブジェクトを追加しないでください

Issue #1843 -関数定義が pg_get_functiondef() から返された値と一致する

Issue #1845 -同義語ノードを更新できるようにします。メッセージ/"この url の実装されていないメソッド（ノード）（/browser/synonym/nodes/1/7/14301/2200/test）/"を表示しません

Issue #1847 -テーブル SQL をリバースエンジニアリングするときに照合を正しく識別します。エラー: スキーマ/"default/"は存在しません

Issue #1849 -config.py/config_local.py からセキュリティキーを削除する

Issue #1857 -FTS 辞書と FTS テンプレートノードの名前を変更する際のエラーを修正

Issue #1858 -ファイルマネージャーがディレクトリをトラバースするときにファイルの種類を優先することを確認します。

Issue #1861 -除外制約 SQL を適切に生成します。

Issue #1863 -テーブルのリバースエンジニアリングされた SQL の型名を正しく引用する

Issue #1864 -DateTimePicker コントロールのヘルプメッセージのレイアウトを修正しました。

Issue #1867 -パッケージ本体を削除できます。

Issue #1868 -設定の整数型が更新されない問題を解決しました

Issue #1872 -Python3 で使用する場合のファイルマネージャーを修正しました。

Issue #1877 -設定値が適切に保存されていることを確認します。

Issue #1878 -ステップとスケジュールを空のジョブで作成できることを確認します。ProgrammingError：タイプ「Undefined」を適応できません

Issue #1880 -ツリービューの正しい親に新しいインデックスを追加します。

13.75 バージョン 1.0

発売日：2016-09-29

pgAdmin4 の最初のメジャーリリース。より現代的なルックアンドフィールを備えたこのリリースには、次の機能が含まれます。

- マルチプラットフォーム
- 複数の PostgreSQL バージョンおよび派生物向けに設計
- 豊富なドキュメント
- 複数の展開モデル
- ツール
- 定期的なメンテナンス
- すべての一般的な PostgreSQL オブジェクトを作成、表示、編集します
- マルチバイトサポート

第14章　Licence

pgAdmin は、BSD または MIT に似たリベラルなオープンソースライセンスである PostgreSQL Licence の下でリリースされ、OpenSourceInitiative によって承認されています。プロジェクトのソースコード、Web サイト、ドキュメントの著作権は、

pgAdmin4

Copyright(C)2013-2023、pgAdmin 開発チーム

上記の著作権表示、この段落および次の 2 つの段落がすべてのコピーに表示されることを条件として、このソフトウェアとそのドキュメントを目的を問わずに使用、コピー、変更、および配布することを書面による同意なしに許可します。

イベントこのような損害の可能性を通知しました。

PGADMIN 開発チームは、商品性および特定の目的への適合性の黙示の保証を含むがこれらに限定されないいかなる保証も特に否認します。本契約に基づいて提供されるソフトウェアは「現状有姿」で提供され、PGADMIN 開発チームはメンテナンス、サポート、アップデート、機能強化、または変更を提供する義務を負いません。

注釈: Postgres、PostgreSQL、および Slonik ロゴは PostgreSQL Community Association of Canada の商標または登録商標であり、許可を得て使用しています。

索引

A

Add Named Restore Point Dialog, 377
Amazon AWS RDS Cloud Deployment, 101
Azure PostgreSQL Cloud Deployment, 110

B

Backup and Restore, 391
Backup Dialog, 391
Backup Globals Dialog, 397
Backup Server Dialog, 398

C

Cast Dialog, 155
Change Ownership Dialog, 29
Change Password Dialog, 377
Change User Password Dialog, 31
Check Dialog, 297
Clear Saved Passwords, 100
Cloud Deployment, 101
Code Overview, 476
Code Review Notes, 483
Coding Standards, 480
Collation Dialog, 159
Column Dialog, 301
Compound Trigger Dialog, 310
Connect to Server, 118
Connecting To A Server, 91
Connection Error, 120
Container Deployment, 15
Creating a pgAgent Job, 464
Creating or Modifying a Table, 297

D

Database Dialog, 129
Debugger, 407
Deployment, 3
Desktop Deployment, 4
Developer Tools, 407
Domain Constraints Dialog, 169
Domain Dialog, 163

E

EDB BigAnimal Cloud Deployment, 106
Enabling Kerberos Authentication, 36
Enabling LDAP Authentication, 33
Enabling OAUTH2 Authentication, 38
Enabling two-factor authentication (*2FA*), 26
Enabling Webserver Authentication, 40
ERD Tool, 448
Event Trigger Dialog, 172
Exclusion Constraint Dialog, 316
Extension Dialog, 177
External database for pgAdmin user settings, 89

F

Foreign Data Wrapper Dialog, 181
Foreign key Dialog, 319
Foreign Server Dialog, 186
Foreign Table Dialog, 192
FTS Configuration Dialog, 199
FTS Dictionary Dialog, 203
FTS Parser Dialog, 208
FTS Template Dialog, 211
Function Dialog, 214

G

Getting Started, 3
Grant Wizard, 379

I

Import/Export Data Dialog, 381
Import/Export Servers, 122
Index Dialog, 326
Installing pgAgent, 462

K

Keyboard Shortcuts, 82

L

Language Dialog, 221
Licence, 589

Lock/Restore Account, 32
Login Dialog, 24
Login/Group Role Dialog, 139

M

Maintenance Dialog, 386
Management Basics, 377
Managing Cluster Objects, 129
Managing Database Objects, 155
Master Password, 117
Materialized View Dialog, 226
Menu Bar, 44

P

Package Dialog, 232
pgAdmin Project Contributions, 475
pgAgent, 461
Preferences Dialog, 61
Primary key Dialog, 330
Procedure Dialog, 236
Processes, 459
PSQL Tool, 457
Publication Dialog, 244

Q

Query Tool, 412
Query Tool Toolbar, 414

R

Resource Group Dialog, 136
Restore Dialog, 402
RLS Policy Dialog, 334
Role Reassign/Drop Own Dialog, 152
Rule Dialog, 338

S

Schema Dialog, 248
Schema Diff, 444
Search objects, 87
Sequence Dialog, 253
Server Deployment, 9
Server Dialog, 92
Server Group Dialog, 91
Storage Manager, 387
Submitting Pull Requests, 475
Subscription Dialog, 260
Synonym Dialog, 266

T

Tabbed Browser, 51
Table Dialog, 344
Tablespace Dialog, 146
The config.py File, 3
Toolbar, 50
Translations, 485
Tree Control, 57
Trigger Dialog, 365
Trigger Function Dialog, 269
Type Dialog, 277

U

Unique Constraint Dialog, 372
User Interface, 42
User Management Dialog, 27
User Mapping Dialog, 287
Using pgAgent, 461

V

View Dialog, 290
View/Edit Data, 438
View/Edit Data Filter, 443