

## Quick Links



[Online Demo](#)



[Download](#)



[FAQ](#)



[Latest Docs](#)



[Ask a  
Question](#)



[Report a Bug](#)



[Screenshots](#)

This page in other versions: [Latest \(4.19\)](#) | [4.18](#) | [4.17](#) | [3.6](#) | [2.1](#) | [1.6](#) | [Development](#)

[pgAdmin 4 4.19 documentation](#) » [Managing Database Objects](#) » [previous](#) | [next](#) | [index](#)

**Warning:** This documentation is for a pre-release version of pgAdmin 4

### Contents

- [Extension Dialog](#)
  - [Example](#)
- [Getting Started](#)
- [Connecting To A Server](#)
- [Managing Cluster Objects](#)
- [Managing Database Objects](#)
  - [Cast Dialog](#)
  - [Collation Dialog](#)
  - [Domain Dialog](#)
  - [Domain Constraints Dialog](#)
  - [Event Trigger Dialog](#)
  - [Extension Dialog](#)
  - [Foreign Data Wrapper Dialog](#)
  - [Foreign Server Dialog](#)
  - [Foreign Table Dialog](#)
  - [FTS Configuration Dialog](#)
  - [FTS Dictionary Dialog](#)
  - [FTS Parser Dialog](#)
  - [FTS Template Dialog](#)
  - [Function Dialog](#)
  - [Language Dialog](#)
  - [Materialized View Dialog](#)
  - [Package Dialog](#)
  - [Procedure Dialog](#)
  - [Schema Dialog](#)
  - [Sequence Dialog](#)
  - [Synonym Dialog](#)
  - [Trigger Function Dialog](#)
  - [Type Dialog](#)
  - [User Mapping Dialog](#)
  - [View Dialog](#)
- [Creating or Modifying a Table](#)
- [Management Basics](#)
- [Backup and Restore](#)
- [Developer Tools](#)
- [pgAgent](#)
- [pgAdmin Project Contributions](#)
- [Release Notes](#)
- [Licence](#)

Use the *Extension* dialog to install a new extension into the current database. An extension is a collection of SQL objects that add targeted functionality to your Postgres installation. The *Extension* dialog adds the functionality of an extension to the current database only; you must register the extension in each database that use the extension. Before you load an extension into a database, you should confirm that any pre-requisite files are installed.

The *Extension* dialog allows you to implement options of the CREATE EXTENSION command through the following dialog tabs: *General* and *Definition*. The *SQL* tab displays the SQL code generated by dialog selections.



The screenshot shows a dialog box titled "Create - Extension". It has three tabs: "General", "Definition", and "SQL". The "General" tab is selected. In the "General" tab, there is a "Name" field with a drop-down listbox containing the text "adminpack". Below the "Name" field is a "Comment" field, which is a large text area. At the bottom of the dialog, there are five buttons: an information icon (i), a question mark icon (?), a "Cancel" button, a "Reset" button, and a "Save" button.

Use the fields in the *General* tab to identify an extension:

- Use the drop-down listbox in the *Name* field to select the extension. Each extension must have a unique name.
- Store notes about the extension in the *Comment* field.

Click the *Definition* tab to continue.

The screenshot shows a dialog box titled "Create - Extension" with a close button (X) in the top right corner. It has three tabs: "General", "Definition" (which is selected and underlined), and "SQL". In the "Definition" tab, there are two labels: "Schema" and "Version". Next to "Schema" is an empty drop-down listbox. Next to "Version" is a text input field containing "1.0" with a small "x" and a drop-down arrow on its right side. At the bottom of the dialog, there are four buttons: an information icon (i), a help icon (?), a "Cancel" button with an "x" icon, a "Reset" button with a circular arrow icon, and a "Save" button with a floppy disk icon.

Use the *Definition* tab to select the *Schema* and *Version*:

- Use the drop-down listbox next to *Schema* to select the name of the schema in which to install the extension's objects.
- Use the drop-down listbox next to *Version* to select the version of the extension to install.

Click the *SQL* tab to continue.

Your entries in the *Extension* dialog generate a SQL command (see an example below). Use the *SQL* tab for review; revisit or switch tabs to make any changes to the SQL command.

#### Example ¶

The following is an example of the sql command generated by user selections in the *Extension* dialog:



Create - Extension

General Definition SQL

```
1 CREATE EXTENSION adminpack
2     VERSION "1.0";
```

i ? x Cancel Reset Save

The command creates the *chkpass* extension in the *public* schema. It is version *1.0* of *chkpass*.

- Click the *Info* button (i) to access online help.
- Click the *Save* button to save work.
- Click the *Cancel* button to exit without saving work.
- Click the *Reset* button to restore configuration parameters.