

ACCA7002 Database Systems

Tutorial Tasks

Tutorial 1: Installing PostgreSQL and pgAdmin4, Create a Scott Database, Create tables with Primary and Foreign Keys

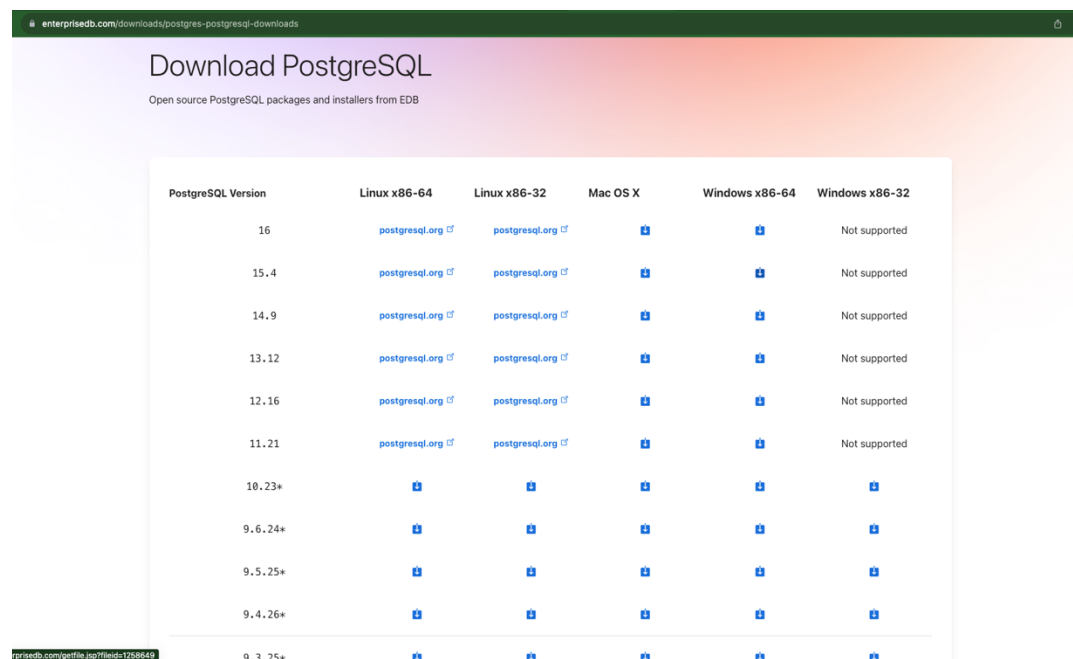
1.1: What is PostgreSQL?

1.2 List some features of PostgreSQL.

1.3: Install PostgreSQL on Windows

There are three steps to complete the PostgreSQL installation:

- Download PostgreSQL installer for Windows
 - Install PostgreSQL
 - Verify the installation
- Download PostgreSQL Installer for Windows from <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>



PostgreSQL Version	Linux x86-64	Linux x86-32	Mac OS X	Windows x86-64	Windows x86-32
16	postgresql.org	postgresql.org			Not supported
15.4	postgresql.org	postgresql.org			Not supported
14.9	postgresql.org	postgresql.org			Not supported
13.12	postgresql.org	postgresql.org			Not supported
12.16	postgresql.org	postgresql.org			Not supported
11.21	postgresql.org	postgresql.org			Not supported
10.23*					
9.6.24*					
9.5.25*					
9.4.26*					
9.3.25*					

- Install PostgreSQL on Windows step by step

To install PostgreSQL on Windows, you need to have administrator privileges.

Step 1. Double click on the installer file, an installation wizard will appear and guide you through multiple steps where you can choose different options that you would like to have in PostgreSQL.

Step 2. Click the Next button

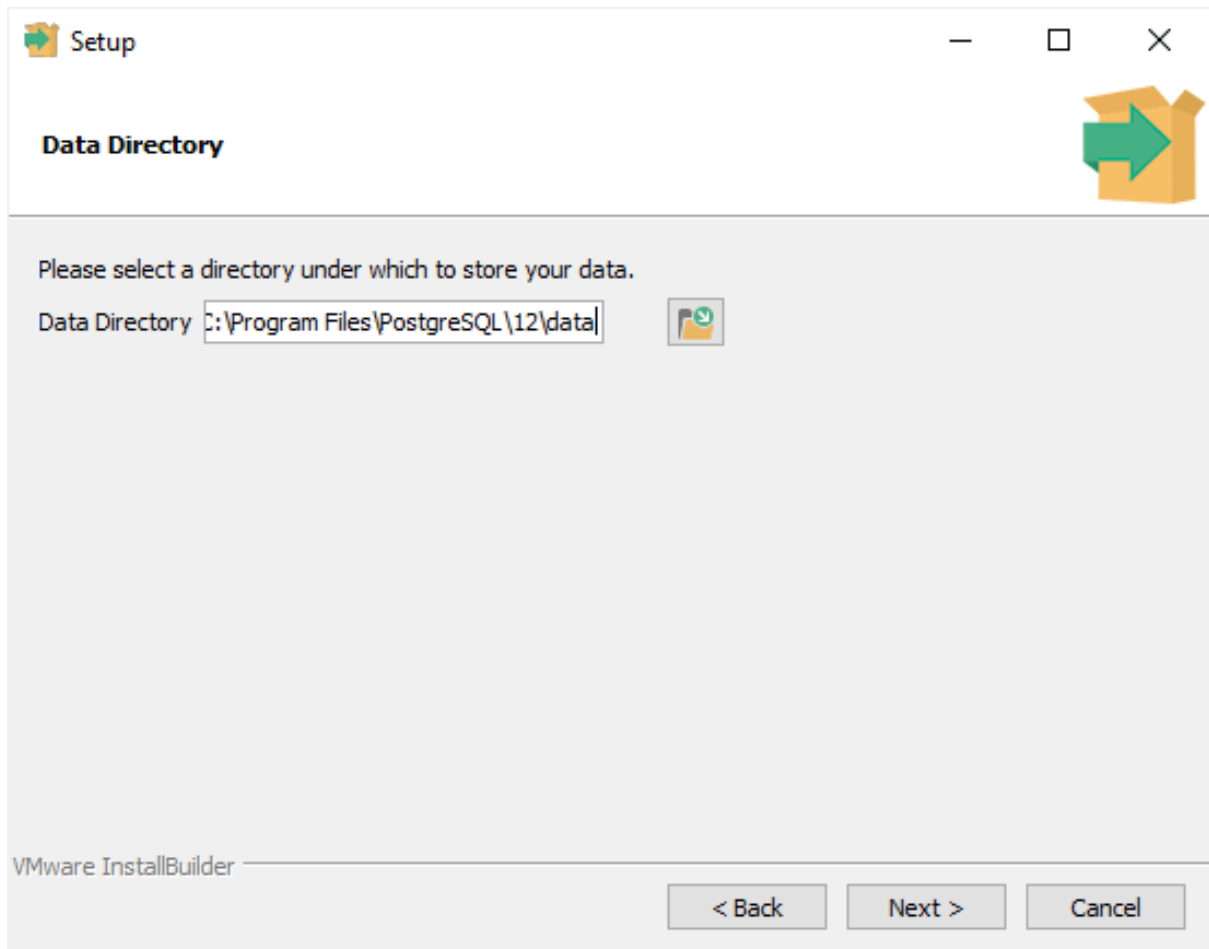
Step 3. Specify installation folder, choose your own or keep the default folder suggested by PostgreSQL installer and click the Next button

Step 4. Select software components to install:

- The PostgreSQL Server to install the PostgreSQL database server
- pgAdmin 4 to install the PostgreSQL database GUI management tool.

- Command Line Tools to install command-line tools such as psql, pg_restore, etc. These tools allow you to interact with the PostgreSQL database server using the command-line interface.
- Stack Builder provides a GUI that allows you to download and install drivers that work with PostgreSQL.

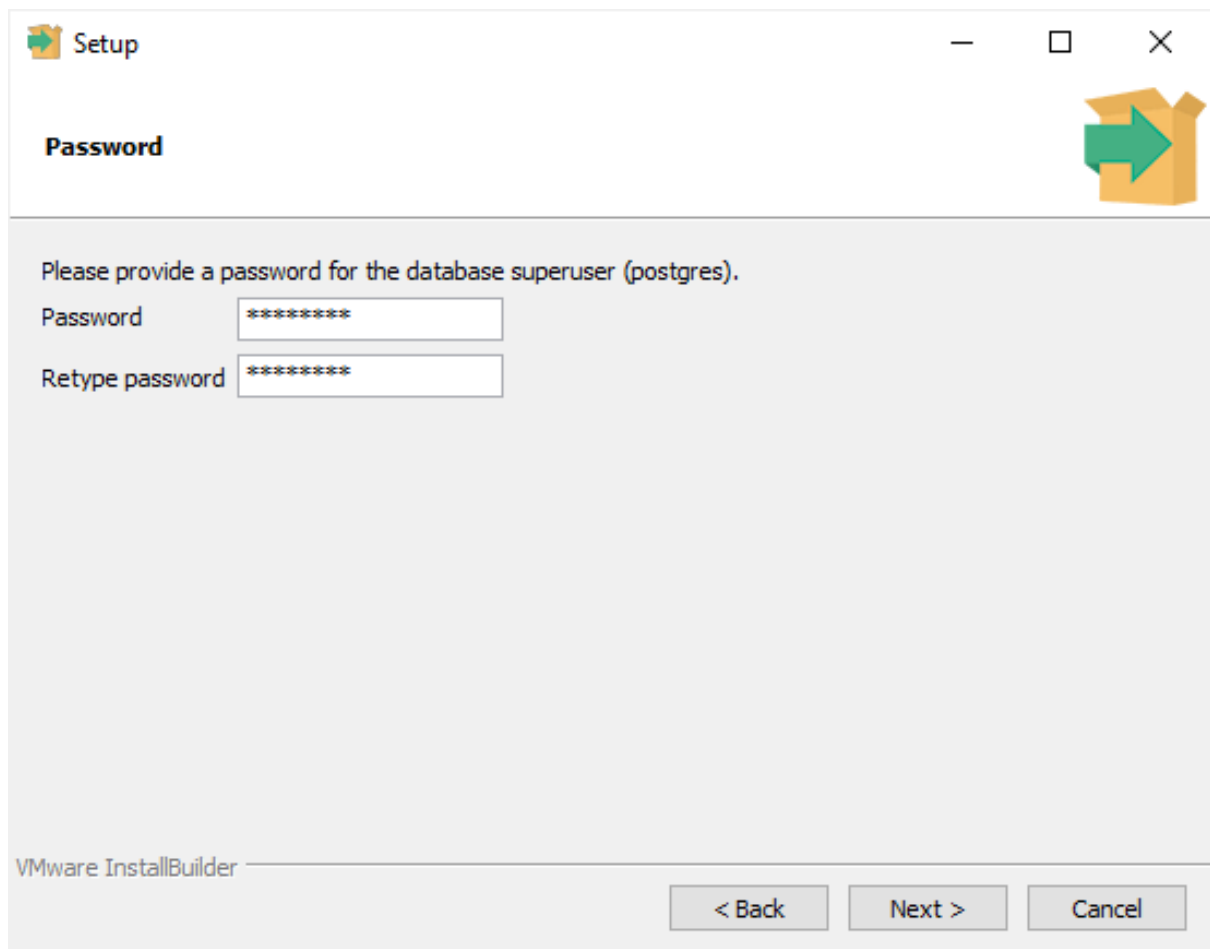
Step 5. Select the database directory to store the data or accept the default folder. And click the Next button to go to the next step:



Step 6. Enter the password for the database superuser (postgres)

PostgreSQL runs as a service in the background under a service account named postgres. If you already created a service account with the name postgres, you need to provide the password of that account in the following window.

After entering the password, you need to retype it to confirm and click the Next button:



The screenshot shows a window titled "Setup" with a standard Windows-style title bar (minimize, maximize, close buttons). In the top right corner, there is a large green arrow pointing right, superimposed on a yellow box. Below the title bar, the word "Password" is displayed in bold. The main area of the window contains the text "Please provide a password for the database superuser (postgres)." followed by two input fields. The first field is labeled "Password" and the second is labeled "Retype password". Both fields contain a series of asterisks (*****). At the bottom left, the text "VMware InstallBuilder" is visible. At the bottom right, there are three buttons: "< Back", "Next >", and "Cancel".

Setup

Password

Please provide a password for the database superuser (postgres).

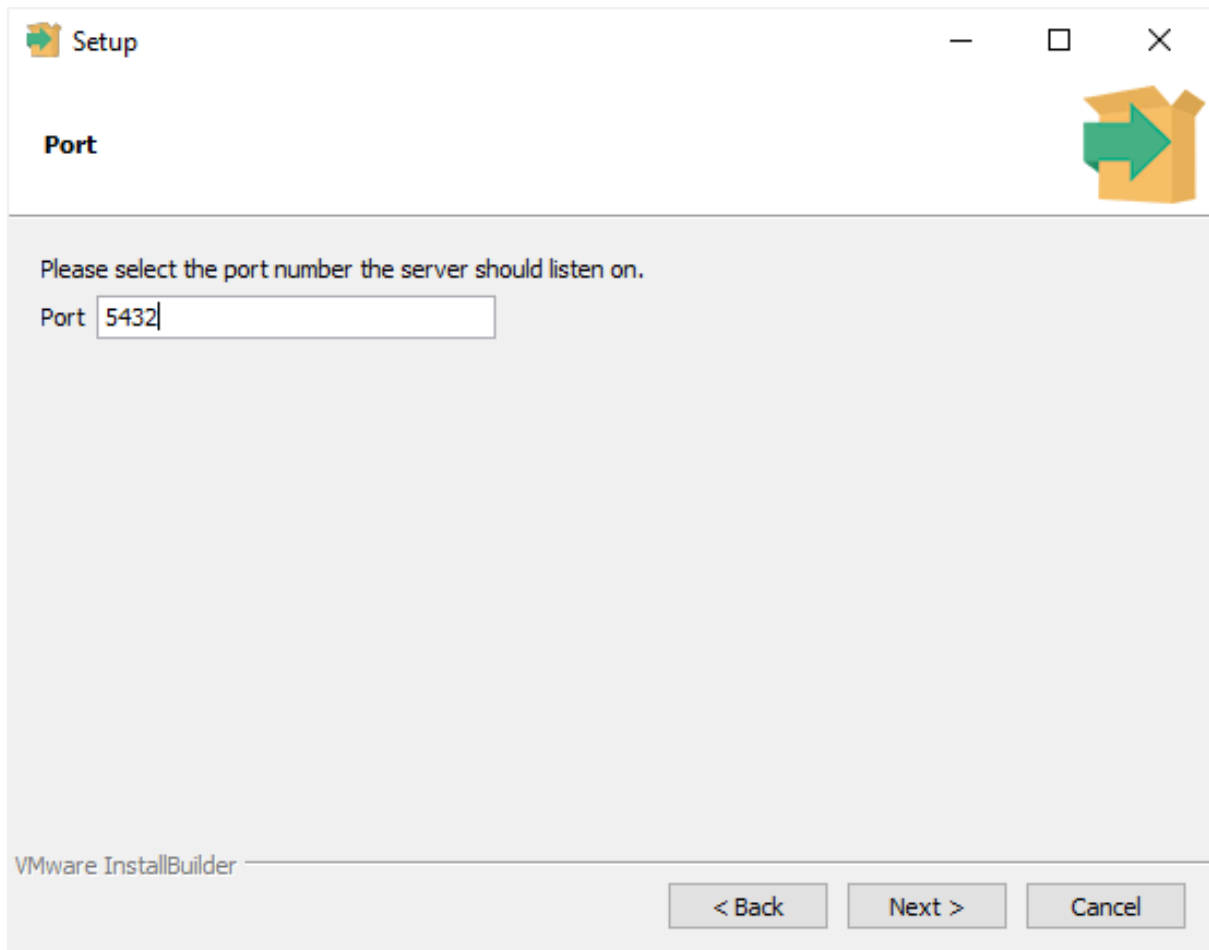
Password *****

Retype password *****

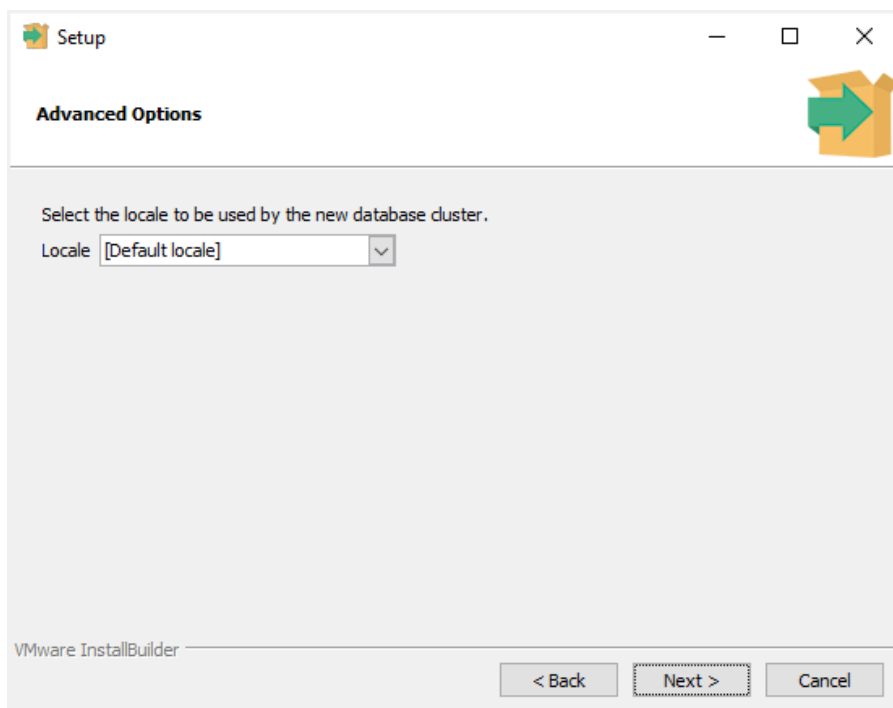
VMware InstallBuilder

< Back Next > Cancel

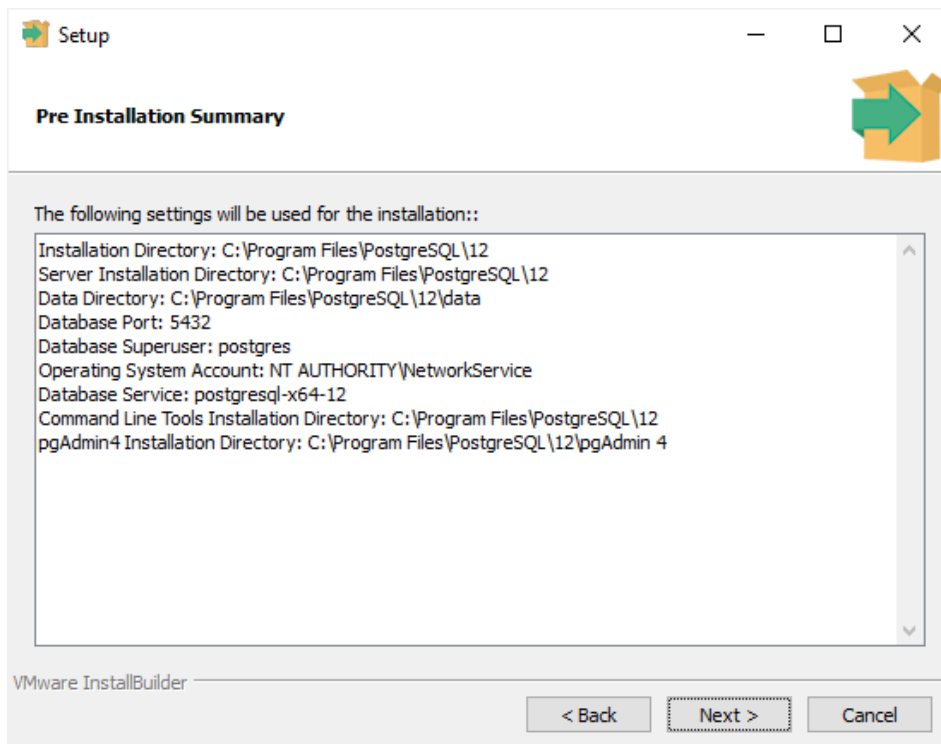
Step 7. Enter a port number on which the PostgreSQL database server will listen. The default port of PostgreSQL is 5432. You need to make sure that no other applications are using this port.



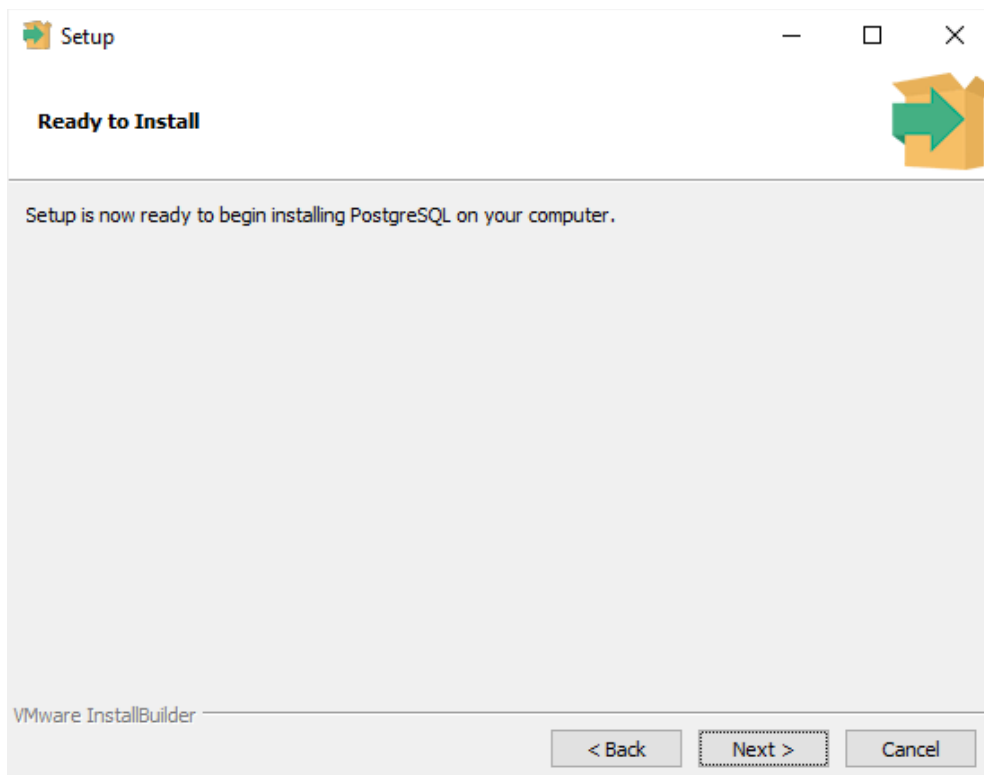
Step 8. Choose the default locale used by the PostgreSQL database. If you leave it as default locale, PostgreSQL will use the operating system locale. After that click the Next button.



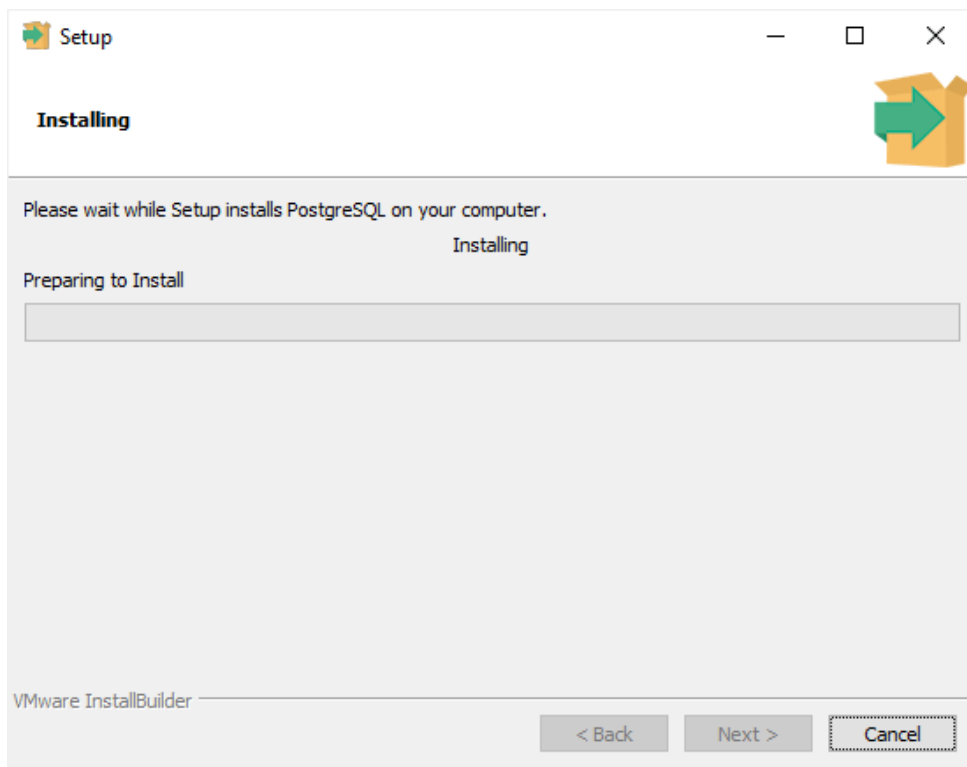
Step 9. The setup wizard will show the summary information of PostgreSQL. You need to review it and click the Next button if everything is correct. Otherwise, you need to click the Back button to change the configuration accordingly.



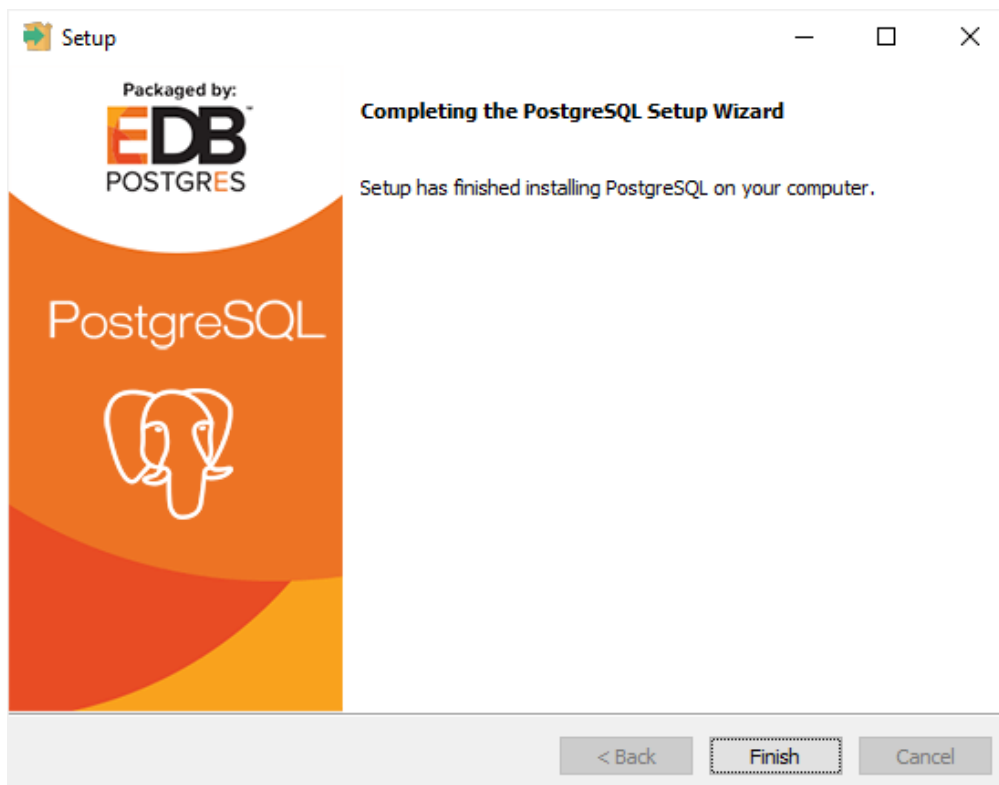
Now, you're ready to install PostgreSQL on your computer. Click the **Next** button to begin installing PostgreSQL.



The installation may take a few minutes to complete.



Step 10. Click the **Finish** button to complete the PostgreSQL installation.

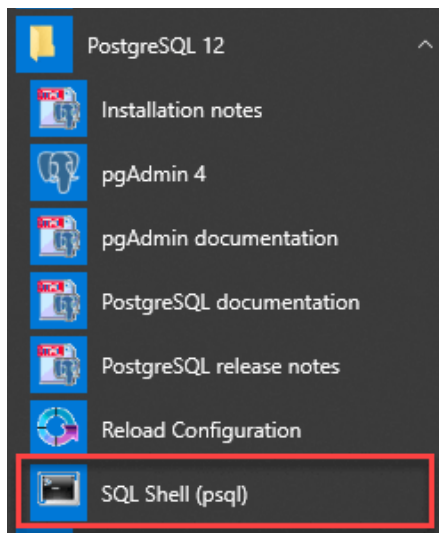


3) Verify the Installation

There are several ways to verify the PostgreSQL installation. You can try to [connect to the PostgreSQL](#) database server from any client application e.g., psql and pgAdmin.

The quick way to verify the installation is through the psql program.

First, click the psql application to launch it. The psql command-line program will display.

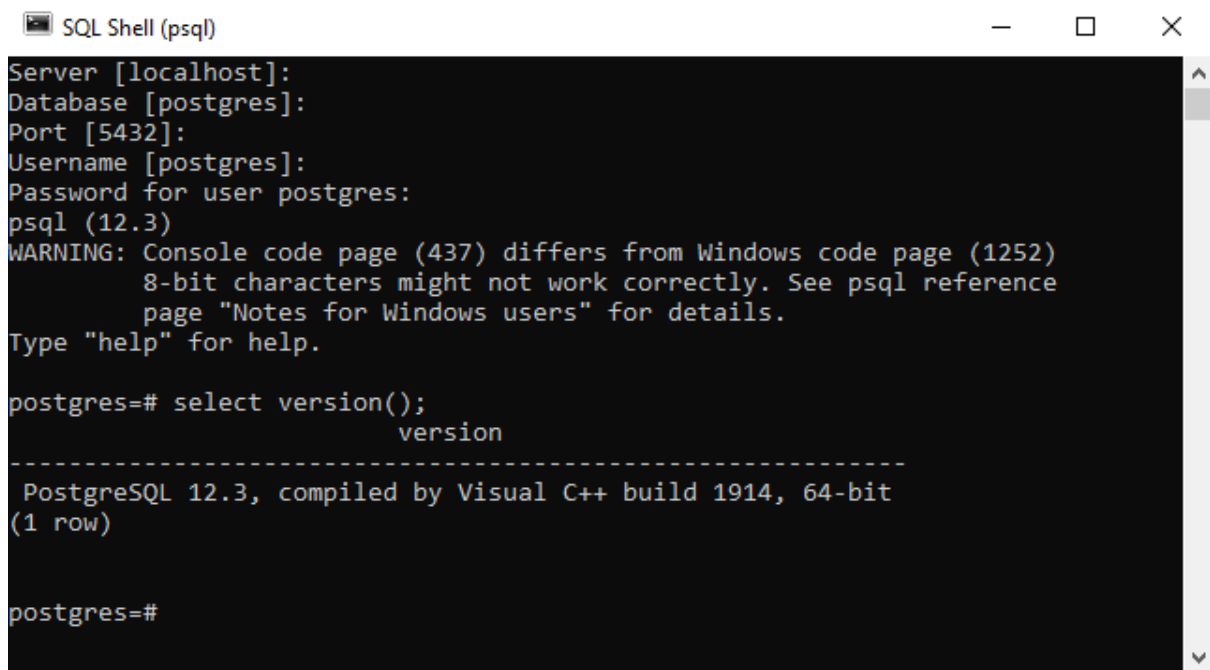


Second, enter all the necessary information such as the server, database, port, username, and password. To accept the default, you can press **Enter**. Note that you should provide the password that you entered during installing the PostgreSQL.

```
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (12.3)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.
```

```
postgres=#Code language: Shell Session (shell)
```

Third, issue the command `SELECT version();` you will see the following output:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (12.3)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=# select version();
              version
-----
PostgreSQL 12.3, compiled by Visual C++ build 1914, 64-bit
(1 row)

postgres=#
```

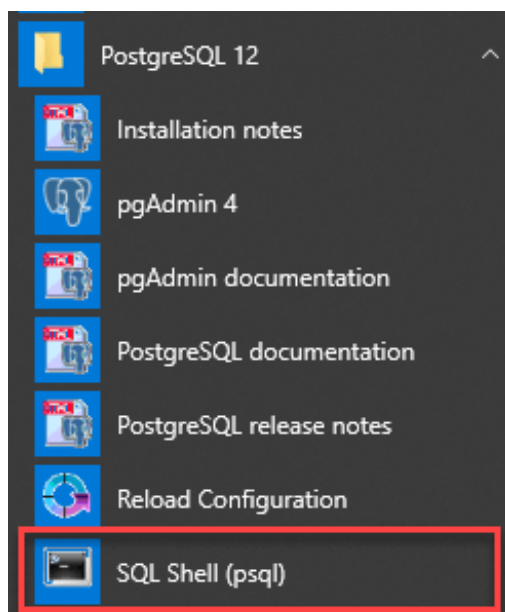
Well done you've successfully installed PostgreSQL database server on your local system.

1.4 Connect to a PostgreSQL Database Server

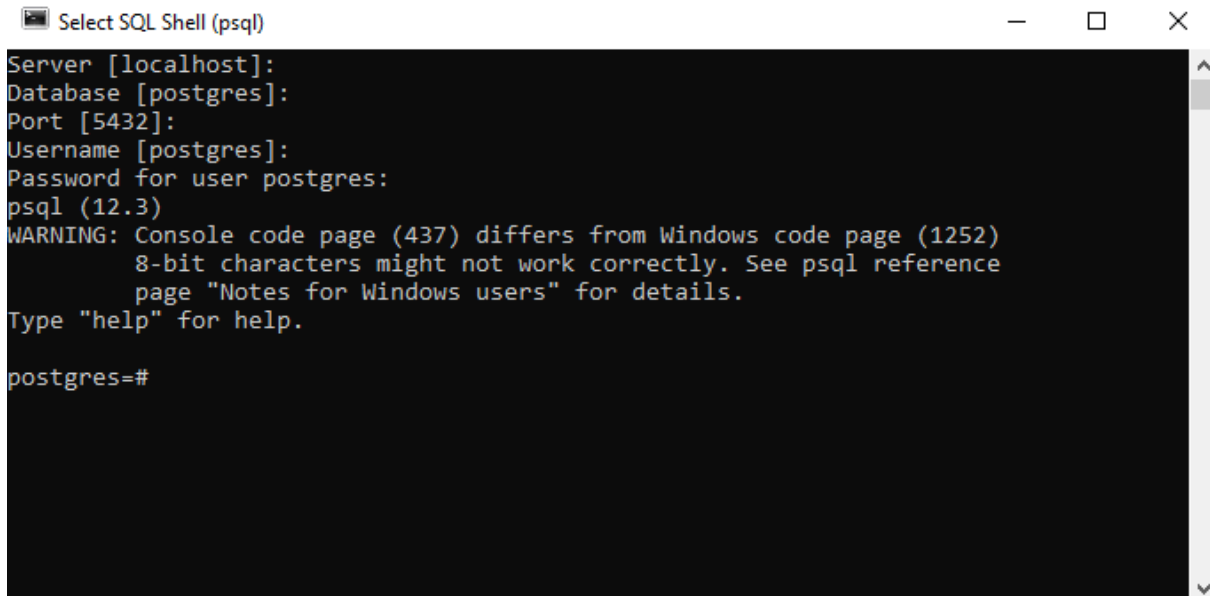
Steps to connect to the PostgreSQL database server via the following tools:

- psql – a terminal-based front-end to PostgreSQL database server.
- pgAdmin – a web-based front-end to PostgreSQL database server.

First, launch the **psql** program and connect to the PostgreSQL Database Server using the **postgres** user:



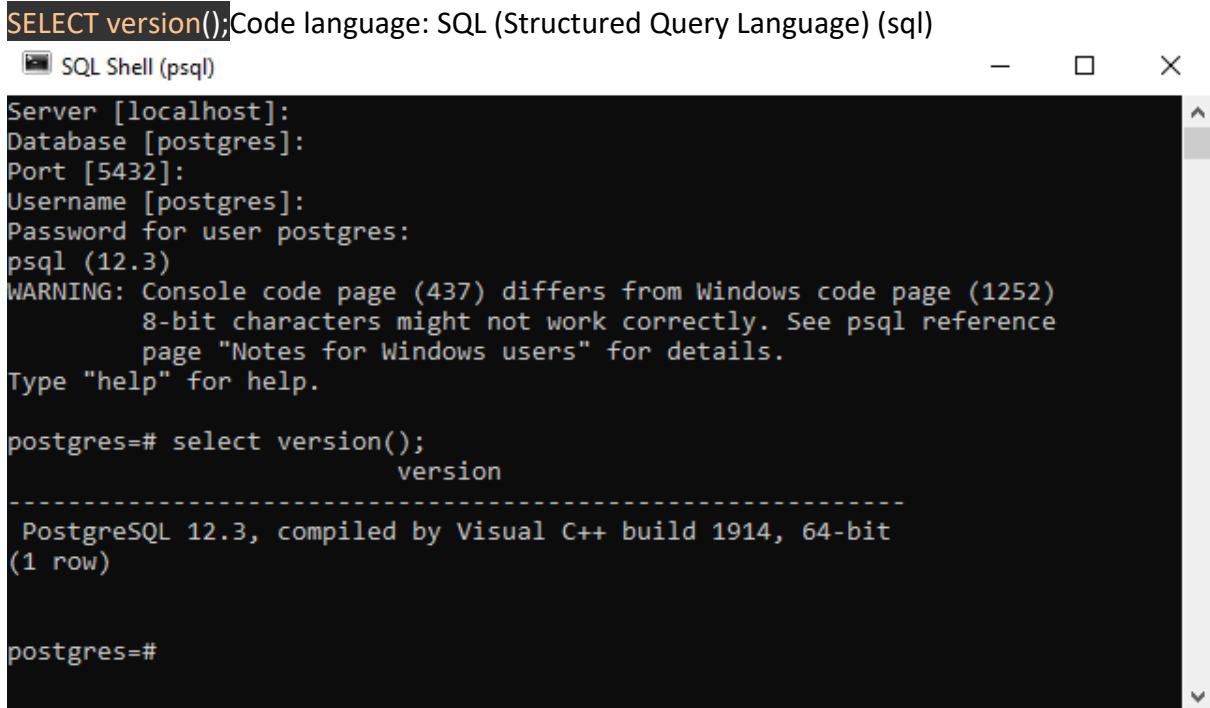
Second, enter all the information such as Server, Database, Port, Username, and Password. If you press Enter, the program will use the default value specified in the square bracket [] and move the cursor to the new line. For example, localhost is the default database server. In the step for entering the password for user postgres, you need to enter the password the user postgres that you chose during the [PostgreSQL installation](#).

A screenshot of a Windows command prompt window titled "Select SQL Shell (psql)". The window shows the following text: "Server [localhost]:", "Database [postgres]:", "Port [5432]:", "Username [postgres]:", "Password for user postgres:", "psql (12.3)", "WARNING: Console code page (437) differs from Windows code page (1252) 8-bit characters might not work correctly. See psql reference page 'Notes for Windows users' for details.", "Type 'help' for help.", and "postgres=#". The cursor is positioned at the end of the "postgres=#" line.

```
Select SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (12.3)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=#
```

Third, interact with the PostgreSQL Database Server by issuing an SQL statement. The following statement returns the current version of PostgreSQL:

A screenshot of a Windows command prompt window titled "SQL Shell (psql)". The window shows the same initial connection text as the previous screenshot, followed by the command "select version();" and its output. The output is displayed in a table with one row: "PostgreSQL 12.3, compiled by Visual C++ build 1914, 64-bit". The cursor is positioned at the end of the "postgres=#" line.

```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (12.3)
WARNING: Console code page (437) differs from Windows code page (1252)
         8-bit characters might not work correctly. See psql reference
         page "Notes for Windows users" for details.
Type "help" for help.

postgres=# select version();
              version
-----
PostgreSQL 12.3, compiled by Visual C++ build 1914, 64-bit
(1 row)

postgres=#
```

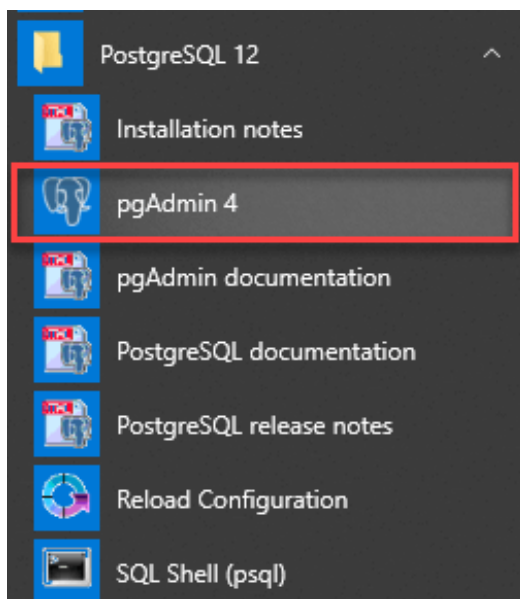
Please do not forget to end the statement with a semicolon (;). After pressing **Enter**, psql will return the current PostgreSQL version on your system.

2) Connect to PostgreSQL database server using pgAdmin

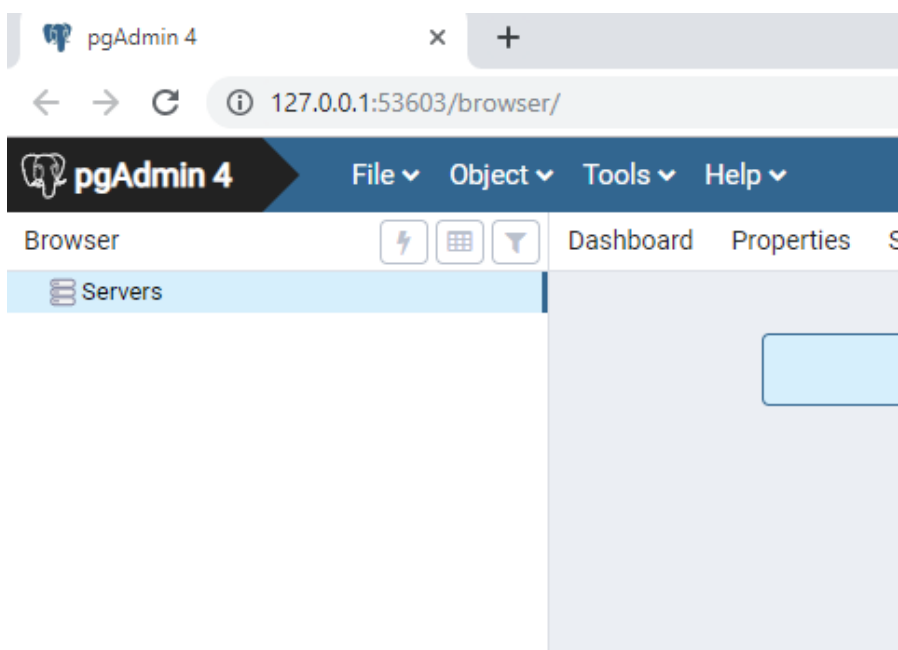
The second way to connect to a database is by using a pgAdmin application. The pgAdmin application allows you to interact with the PostgreSQL database server via an intuitive user interface.

The following illustrates how to connect to a database using pgAdmin GUI application:

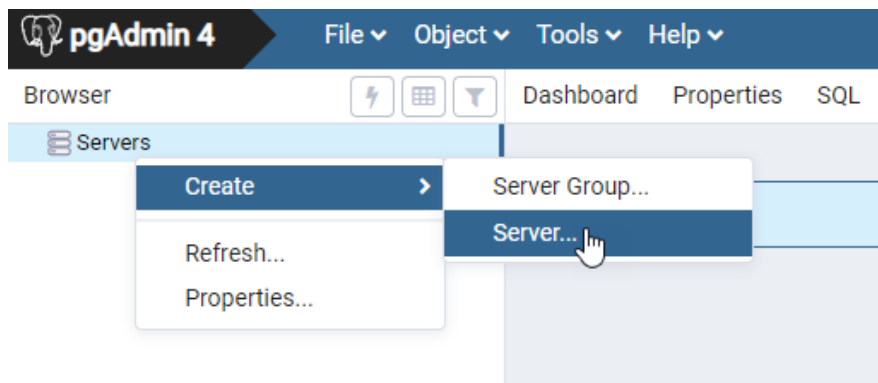
First, launch the pgAdmin application.



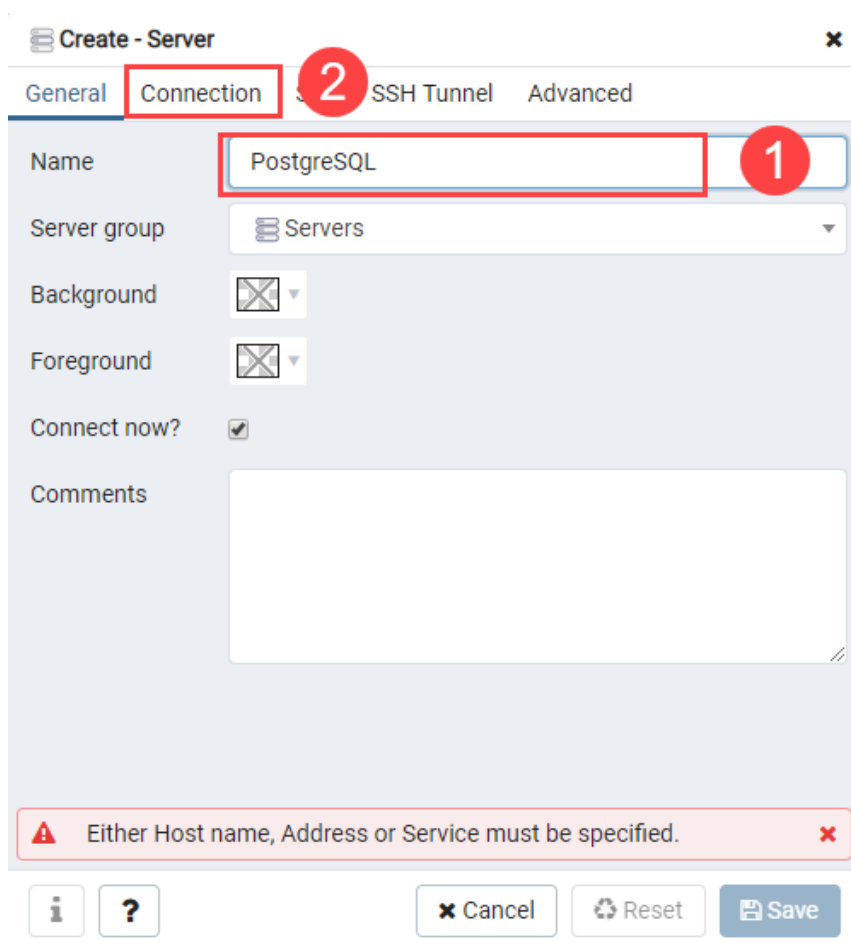
The pgAdmin application will launch on the web browser as shown in the following picture:



Second, right-click the Servers node and select **Create > Server...** menu to create a server



Third, enter the server name e.g., **PostgreSQL** and click the **Connection** tab:



Fourth, enter the host and password for the **postgres** user and click the **Save** button:

Create - Server

General

Connection

SSL

SSH Tunnel

Advanced

Host name/address

localhost

1

Port

5432

Maintenance database

postgres

Username

postgres

Password

.....

2

Save password?

☒

Role

Service

3

i

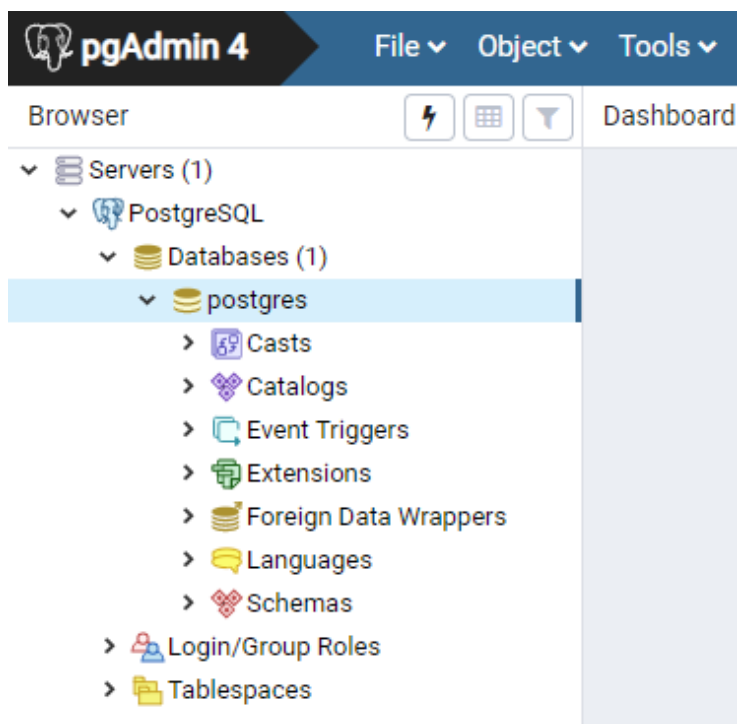
?

Cancel

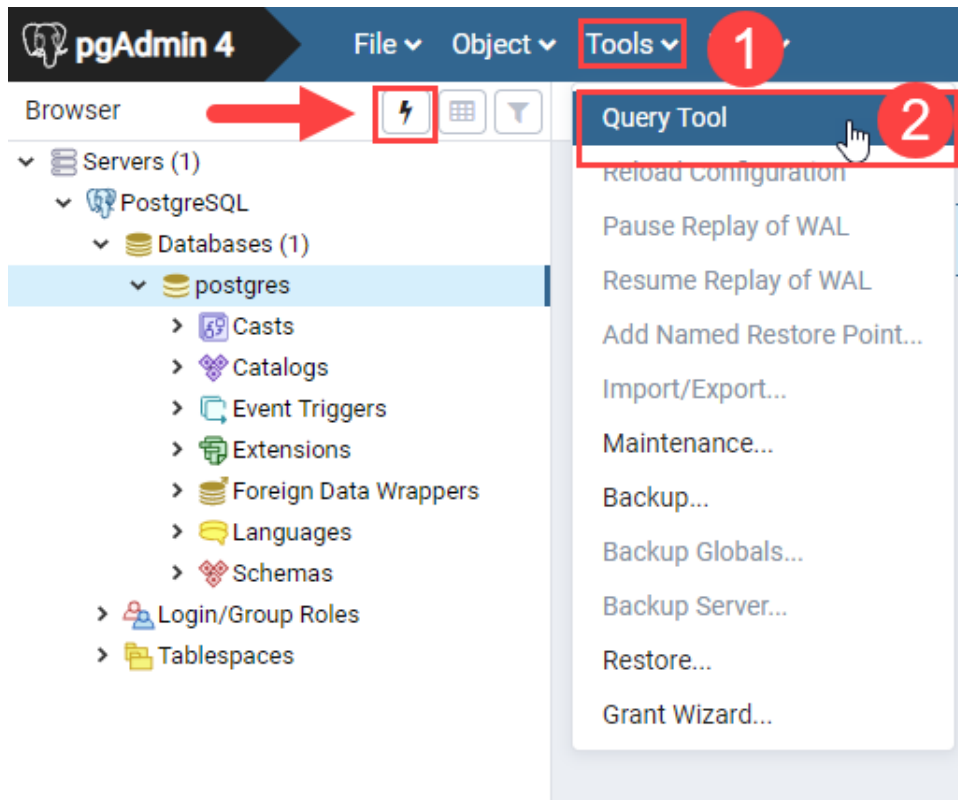
Reset

Save

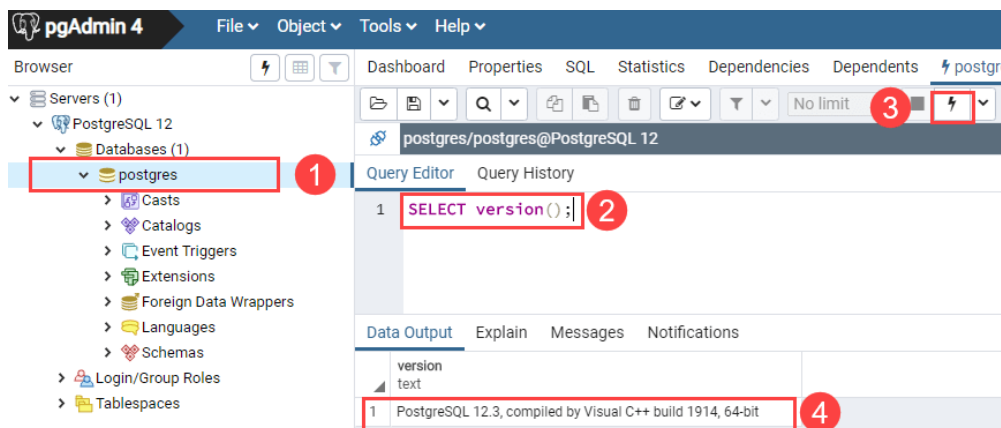
Fifth, click on the Servers node to expand the server. By default, PostgreSQL has a database named postgres as shown below:



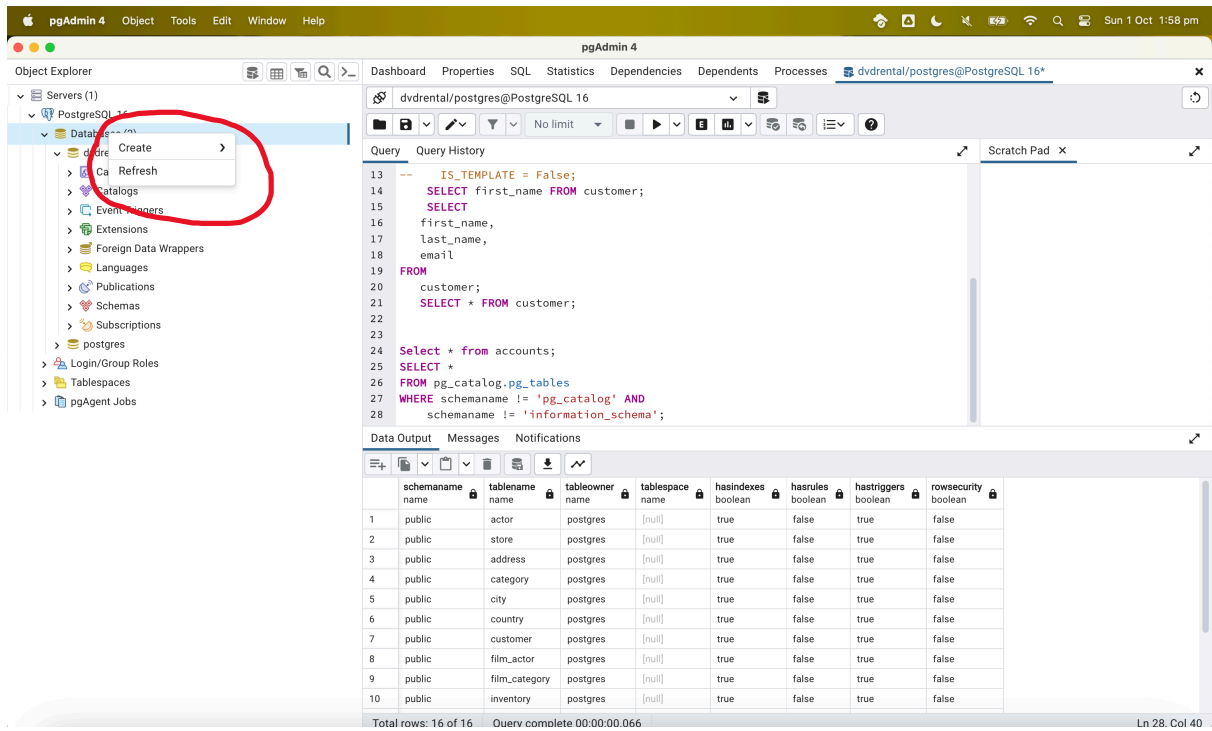
Sixth, open the query tool by choosing the menu item **Tool > Query Tool** or click the lightning icon.



Seventh, enter the query in the **Query Editor**, click the **Execute** button, you will see the result of the query displaying in the **Data Output** tab:



1.5 Create Database 'Scott'



The screenshot shows the pgAdmin 4 interface. In the Object Explorer on the left, the 'Servers' tree is expanded to show 'PostgreSQL 16'. Under 'PostgreSQL 16', the 'Databases' folder is selected, and the 'Create' menu is open. The 'Create' menu has a red circle around it. The 'Query' tab is active in the main pane, showing a SQL query. The 'Data Output' tab is also visible, showing a table of database information.

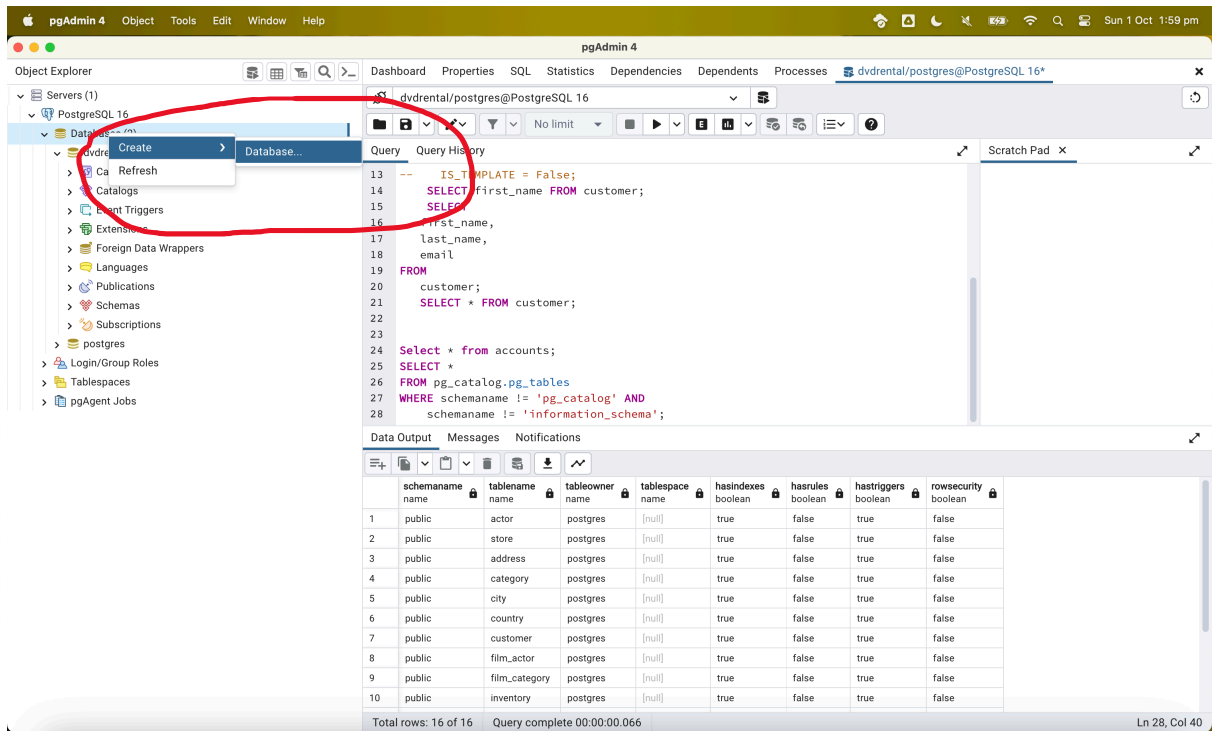
Query:

```
13 -- IS_TEMPLATE = False;  
14 SELECT first_name FROM customer;  
15 SELECT  
16 first_name,  
17 last_name,  
18 email  
19 FROM  
20 customer;  
21 SELECT * FROM customer;  
22  
23  
24 Select * from accounts;  
25 SELECT *  
26 FROM pg_catalog.pg_tables  
27 WHERE schemaname != 'pg_catalog' AND  
28 schemaname != 'information_schema';
```

Data Output:

	schemaname	tablename	tableowner	tablespace	hasindexes	hasrules	hastriggers	rowsecurity
1	public	actor	postgres	[null]	true	false	true	false
2	public	store	postgres	[null]	true	false	true	false
3	public	address	postgres	[null]	true	false	true	false
4	public	category	postgres	[null]	true	false	true	false
5	public	city	postgres	[null]	true	false	true	false
6	public	country	postgres	[null]	true	false	true	false
7	public	customer	postgres	[null]	true	false	true	false
8	public	film_actor	postgres	[null]	true	false	true	false
9	public	film_category	postgres	[null]	true	false	true	false
10	public	inventory	postgres	[null]	true	false	true	false

Total rows: 16 of 16 Query complete 00:00:00.006 Ln 28, Col 40



The screenshot shows the pgAdmin 4 interface. In the Object Explorer on the left, the 'Servers' tree is expanded to show 'PostgreSQL 16'. Under 'PostgreSQL 16', the 'Databases' folder is selected, and the 'Create' menu is open. The 'Database...' option is selected, and a red circle highlights it. The 'Query' tab is active in the main pane, showing a SQL query. The 'Data Output' tab is also visible, showing a table of database information.

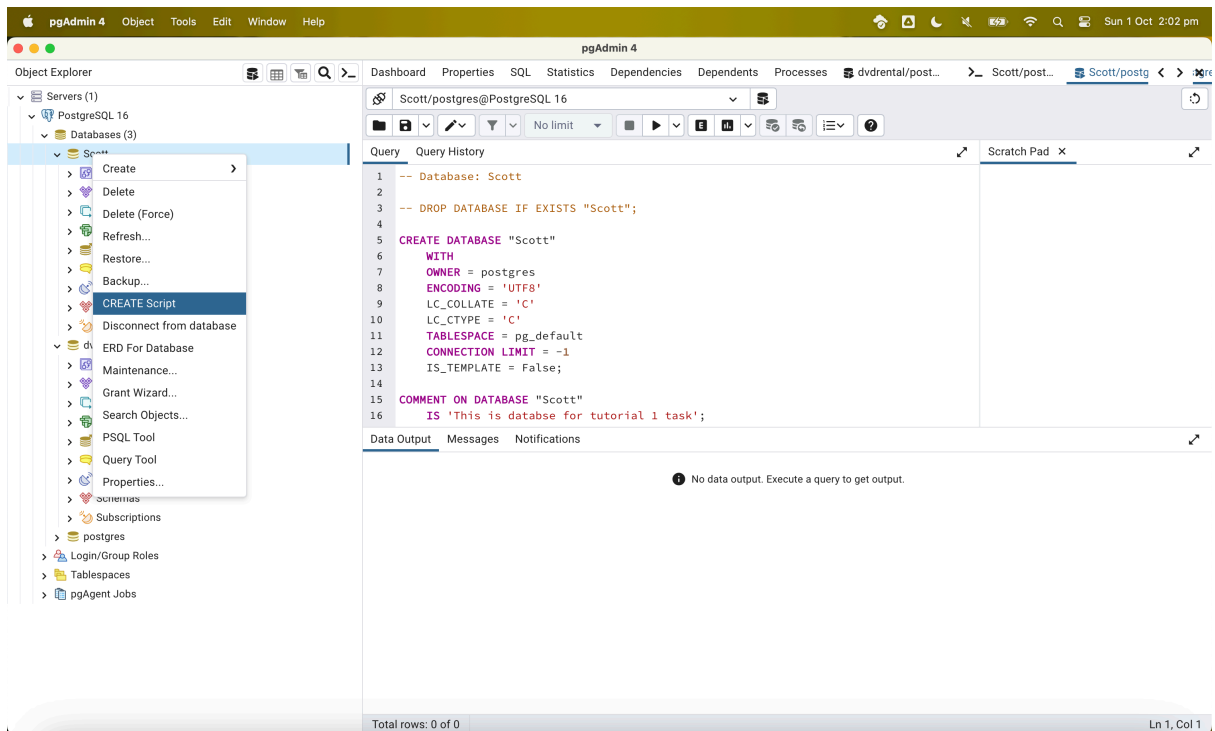
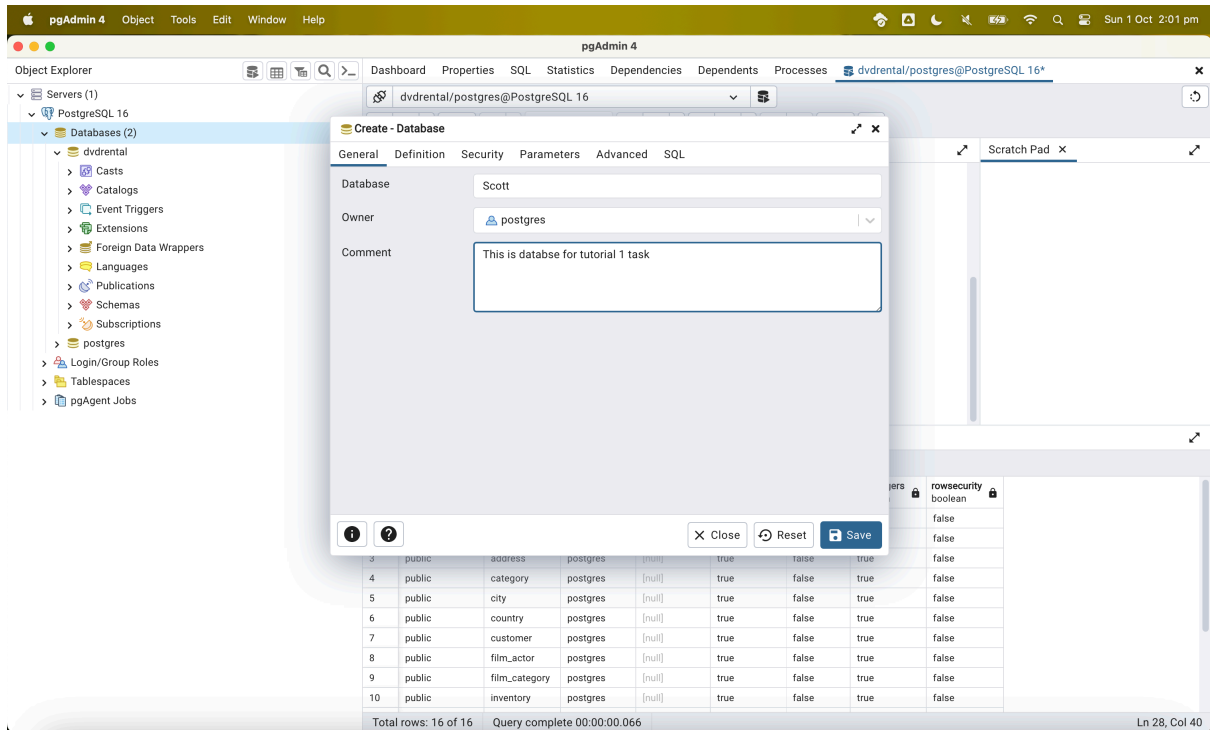
Query:

```
13 -- IS_TEMPLATE = False;  
14 SELECT first_name FROM customer;  
15 SELECT  
16 first_name,  
17 last_name,  
18 email  
19 FROM  
20 customer;  
21 SELECT * FROM customer;  
22  
23  
24 Select * from accounts;  
25 SELECT *  
26 FROM pg_catalog.pg_tables  
27 WHERE schemaname != 'pg_catalog' AND  
28 schemaname != 'information_schema';
```

Data Output:

	schemaname	tablename	tableowner	tablespace	hasindexes	hasrules	hastriggers	rowsecurity
1	public	actor	postgres	[null]	true	false	true	false
2	public	store	postgres	[null]	true	false	true	false
3	public	address	postgres	[null]	true	false	true	false
4	public	category	postgres	[null]	true	false	true	false
5	public	city	postgres	[null]	true	false	true	false
6	public	country	postgres	[null]	true	false	true	false
7	public	customer	postgres	[null]	true	false	true	false
8	public	film_actor	postgres	[null]	true	false	true	false
9	public	film_category	postgres	[null]	true	false	true	false
10	public	inventory	postgres	[null]	true	false	true	false

Total rows: 16 of 16 Query complete 00:00:00.006 Ln 28, Col 40



1.6 Create Table

PostgreSQL CREATE TABLE syntax

A relational database consists of multiple related tables. A table consists of rows and columns. Tables allow you to store structured data like customers, products, employees, etc.

To create a new table, you use the CREATE TABLE statement. The following illustrates the basic syntax of the CREATE TABLE statement:

```
CREATE TABLE [IF NOT EXISTS] table_name (  
    column1 datatype(length) column_constraint,  
    column2 datatype(length) column_constraint,  
    column3 datatype(length) column_constraint,  
    table_constraints  
);
```

PostgreSQL includes the following column constraints:

- NOT NULL – ensures that values in a column cannot be NULL.
- UNIQUE – ensures the values in a column unique across the rows within the same table.
- PRIMARY KEY – a primary key column uniquely identify rows in a table. A table can have one and only one primary key. The primary key constraint allows you to define the primary key of a table.
- CHECK – a CHECK constraint ensures the data must satisfy a boolean expression.
- FOREIGN KEY – ensures values in a column or a group of columns from a table exists in a column or group of columns in another table. Unlike the primary key, a table can have many foreign keys.

Note: Table constraints are similar to column constraints except that they are applied to more than one column.

Try the below PostgreSQL statements

1.6.1 Basic Table Creation:

```
CREATE TABLE employee (  
    employee_id serial PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    hire_date DATE,  
    is_active BOOLEAN DEFAULT true  
);
```

1.6.2 Now alter the above to include [IF NOT EXISTS] and record the output.

1.6.3 Primary Keys and Foreign Keys:

```
CREATE TABLE department (  
    department_id serial PRIMARY KEY,  
    department_name VARCHAR(100) NOT NULL  
);  
  
CREATE TABLE staff (  
    employee_id serial PRIMARY KEY,  
    first_name VARCHAR(50) NOT NULL,  
    last_name VARCHAR(50) NOT NULL,  
    department_id INT REFERENCES department(department_id)  
);
```

1.6.4 Table relationships

Create tables for Customers, Orders, and Products with appropriate primary and foreign keys to establish relationships.