

Introduction to SQL

Part-1



Hitesh Kumar Sharma
Instructor, Pluralsight





DR. HITESH KUMAR SHARMA

*Technical Instructor &
Consultant*

ABOUT ME

- Industry/Academic Experience: **15 Years (8 years in Database, DevOps & Python ,2 Years in RPA, 5 Years in Data Analytics)**
- Worked as **IBM Instructor**
- Worked as **Microsoft Instructor**
- Working as **PluralSight Instructor**
- Core Technical Domains: **DBMS, SQL, UiPath RPA, DevOps, Cloud Computing, Data Analytics**
- Academic Qualifications: **Ph.D. (CSE), M.Tech (CSE)**
- Certifications:
 - **UiPath RPA Certified Associate**
 - **Docker Certified Associate**
 - **Neo4J Certified Associate**
 - **Maven Certified Professional**
- 5 Books Published
- 35 Patents Published
- 02 Copyright Published

DELIVERY MODE

- ❑ 8 Hours Daily of Instructor-Led Training
- ❑ Slides Based Content
- ❑ Hands-on Live Demo
- ❑ Solved/Unsolved Lab Exercises
- ❑ Whiteboard/ Pen Tab based Interaction
- ❑ Daily Recap
- ❑ Daily Session Slides
- ❑ Lesson-End Assignment
- ❑ Course-End Projects

Introduction to SQL



PLURALSIGHT

What is Data?

Data refers to any distinct piece of information that is collected, stored, and processed for use in various contexts. It can be in the form of numbers, text, images, audio, or any other format that conveys meaning. Data, in its raw form, lacks structure and context and is often unorganized. It can exist in various states, including raw data, processed data, or information derived from the data through analysis and interpretation.



What is Database?

A database is a structured collection of data that is organized and stored in a way that facilitates efficient retrieval, management, and modification. It is designed to store and manage large volumes of information in a manner that supports data manipulation and retrieval. Databases provide a systematic and organized approach to storing data, allowing for efficient management and retrieval when needed. A Database Management System (DBMS) is used to manage and control databases, enabling users to define, create, retrieve, update, and manage data effectively.



What is DBMS?

DBMS stands for Database Management System. It is a software application that enables users to define, create, maintain, and control access to a database. DBMS serves as an interface between the database and the end-users or application programs, allowing users to interact with the data stored in the database.



PostgreSQL



Data vs DBMS

Databases are collections of organized data stored such that a program called a database management system (DBMS) can interact with it

- There are many DBMS systems out there, including:
 - PostgreSQL
 - MySQL
 - SQL Server
 - Oracle
 - IBM Db2
 - Access
 - MongoDB

Modern DBMS

Modern DBMS fall into two broad categories:

- *SQL based databases* (ex: MySQL) are called relational databases and manage their data in collections of related tables
- *NoSQL databases* (ex: MongoDB) have an entirely different way of manage data (often resembling JSON documents)

In this course, we will examine only relational databases and the PostgreSQL DBMS.

Database Operations

There are four basic types of operations we perform with data using DBLS

- Add new data
- Query the data for answers to questions
- Change data
- Delete data

There is a common acronym for these four types of operation: CRUD

- Create
- Read
- Update
- Delete

Relational DBMS

A Relational Database is a type of database that stores and provides access to data points that are related to one another. It organizes data into one or more tables, where each table consists of rows and columns. These tables are related to each other through the use of shared data, allowing the establishment of connections and relationships between different data sets.



Characteristics Relational DBMS

- **Tables:** Data is stored in tables, where each table has a predefined set of columns and rows.
- **Rows/Column:** Each row in a table represents a record, instance, or entity, and each column represents a specific attribute or field of that record.

Customers				
	CustomerId	FirstName	LastName	DateCreated
+	1	Homer	Simpson	13/06/2014 3:33:37 PM
+	2	Peter	Griffin	13/06/2014 9:09:56 PM
+	3	Stewie	Griffin	13/06/2014 9:16:07 PM
+	4	Brian	Griffin	13/06/2014 9:16:36 PM
+	5	Cosmo	Kramer	13/06/2014 9:16:41 PM
+	6	Philip	Fry	13/06/2014 9:17:02 PM
+	7	Amy	Wong	13/06/2014 9:22:05 PM
+	8	Hubert J.	Farnsworth	13/06/2014 9:22:19 PM
+	9	Marge	Simpson	13/06/2014 9:22:37 PM
+	10	Bender	Rodríguez	13/06/2014 9:22:52 PM
+	11	Turanga	Leela	13/06/2014 9:23:37 PM
*	(New)			15/06/2014 9:00:01 PM

Characteristics Relational DBMS

- **Keys:** Relational databases use keys to uniquely identify each record in a table. Primary keys are used to uniquely identify each row, and foreign keys are used to establish relationships between tables.
- **Relationships:** Relationships between tables are established through the use of foreign keys that link records in one table to those in another.
- **Structured Query Language (SQL):** Relational databases are managed using SQL, a specialized language for managing, querying, and manipulating data in the database.

Structured Query Language

Structured Query Language (SQL) is a domain-specific programming language used for managing and manipulating data in relational database management systems (RDBMS). It provides a standardized way to interact with databases, allowing users to perform various operations such as querying, updating, inserting, and deleting data.



Structured Query Language (Example)

A command to fetch data written using SQL is called a "query" and might look like this:

```
SELECT FirstName, LastName FROM Customers  
WHERE LastName = 'Griffin'
```

The information returned from a query is called the "result" or a "result set".

Peter	Griffin
Stewie	Griffin
Brian	Griffin

Types of Structured Query Language

Data Definition Language (DDL):

SQL includes commands for defining and managing the structure of a database, such as creating and altering tables (CREATE TABLE, ALTER TABLE), defining constraints, and managing indexes.

Data Manipulation Language:

SQL provides commands for manipulating data in a database, including INSERT, UPDATE, and DELETE, allowing users to add, modify, or remove data as needed.

Data Querying Language:

SQL allows users to retrieve specific data from a database using queries. The SELECT statement is commonly used for this purpose.

Types of Structured Query Language

Data Control Language (DCL):

SQL offers commands for managing user permissions and access control, allowing administrators to control who can access or modify specific data in the database. Ex. (Grant, Revoke etc.)

Transaction Control Language (TCL):

SQL includes commands for managing transactions, ensuring that multiple operations can be performed as a single, atomic unit of work. Commands such as COMMIT, ROLLBACK, and SAVEPOINT are used for this purpose.

Downloading and Installing PostgreSQL



PLURALSIGHT


Downloading PostgreSQL

You can download the latest stable PostgreSQL Installer specific to your Windows by clicking [here](#)

Direct Link:


<https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

Downloading PostgreSQL











Upcoming Webinar: Why the Most Productive and Secure Teams Use EDB's Oracle Compatible Postgres • Register Now

[Products](#) [Solutions](#) [Services & Support](#) [Developers](#) [Resources](#) [Company](#) [Contact Sales](#)

 [Sign in](#) [Get Started](#)

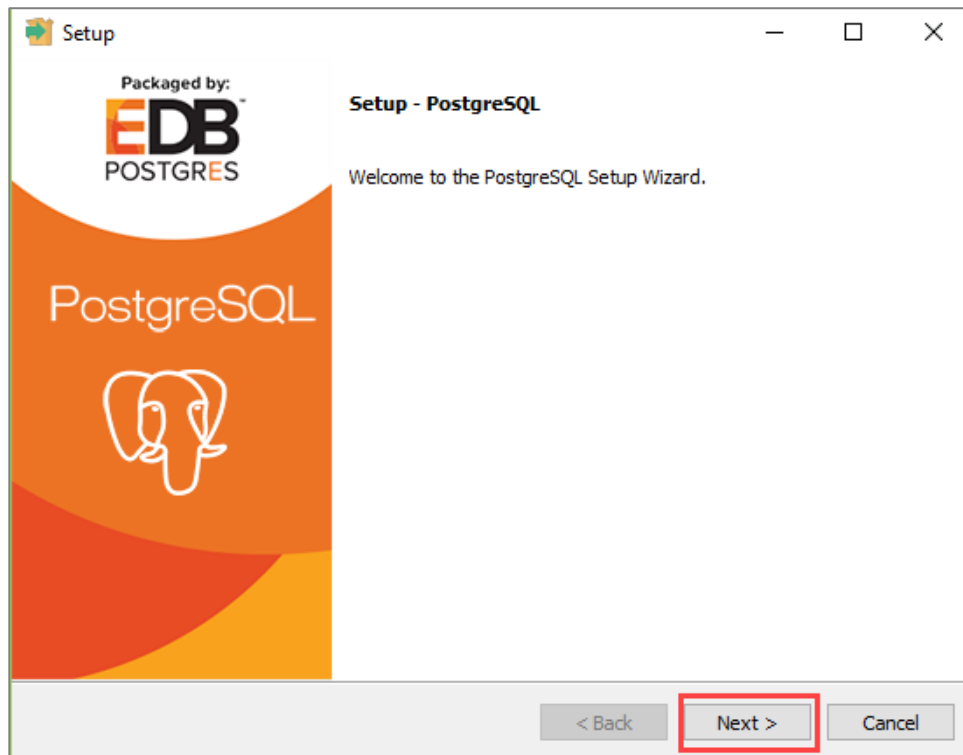
Download PostgreSQL

Open source PostgreSQL packages and installers from EDB

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

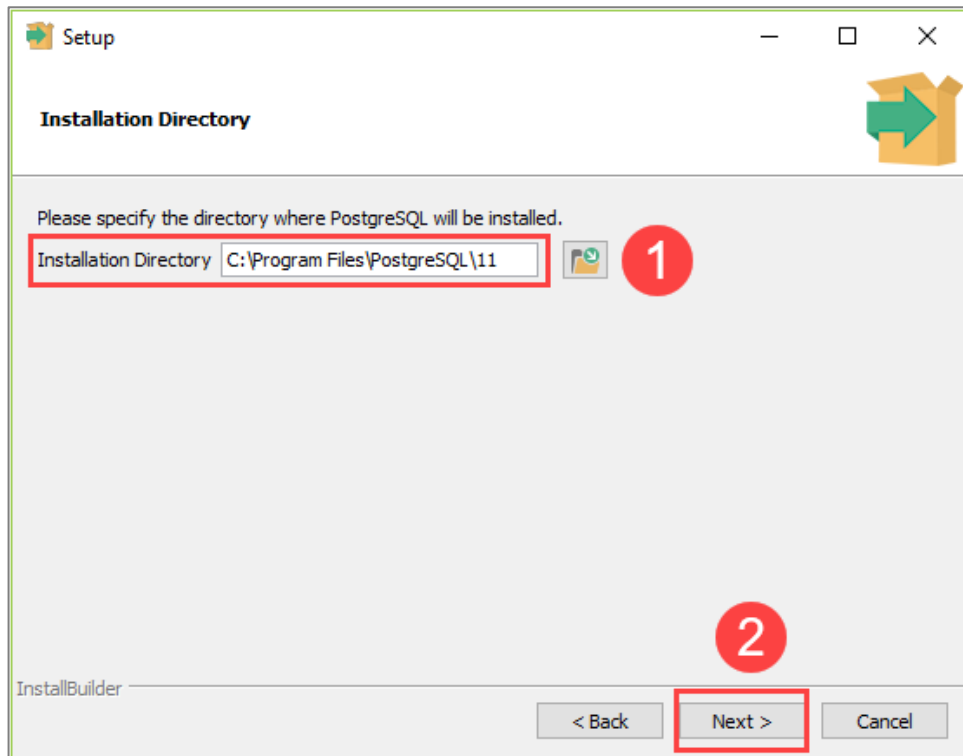
Installation of PostgreSQL

Step 1: Click the Next button



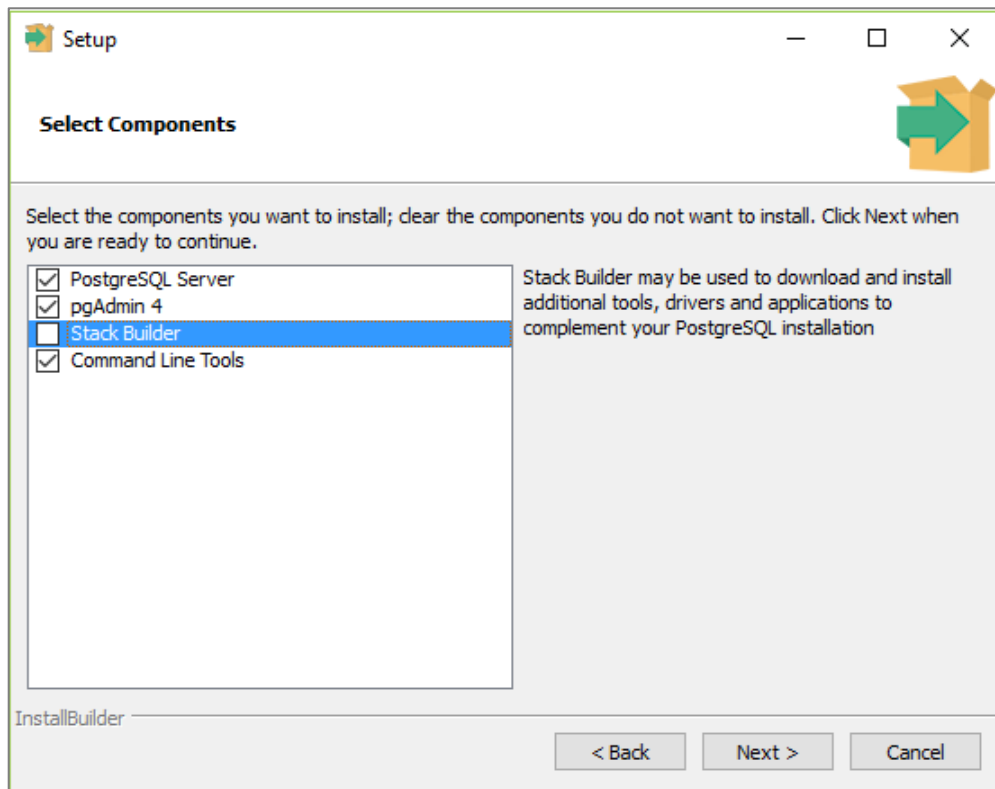
Installation of PostgreSQL

- **Step 2:** Choose the installation folder, where you want PostgreSQL to be installed, and click on Next.



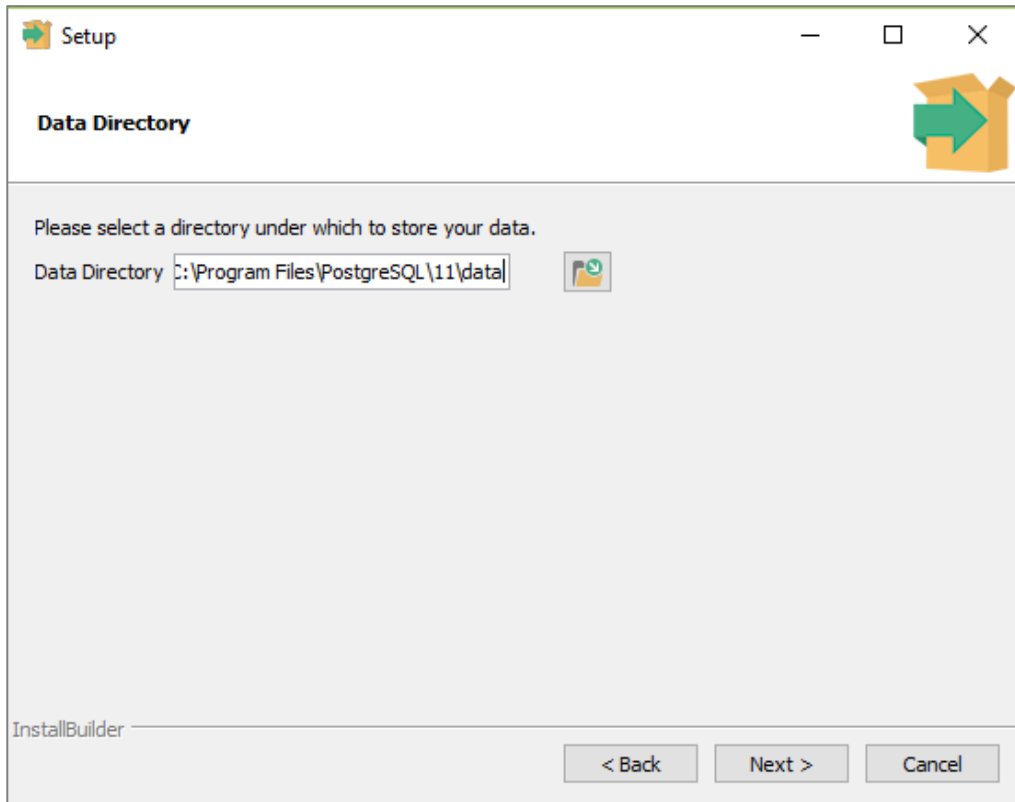
Installation of PostgreSQL

- Step 3:** Select the components as per your requirement to install and click the Next button.



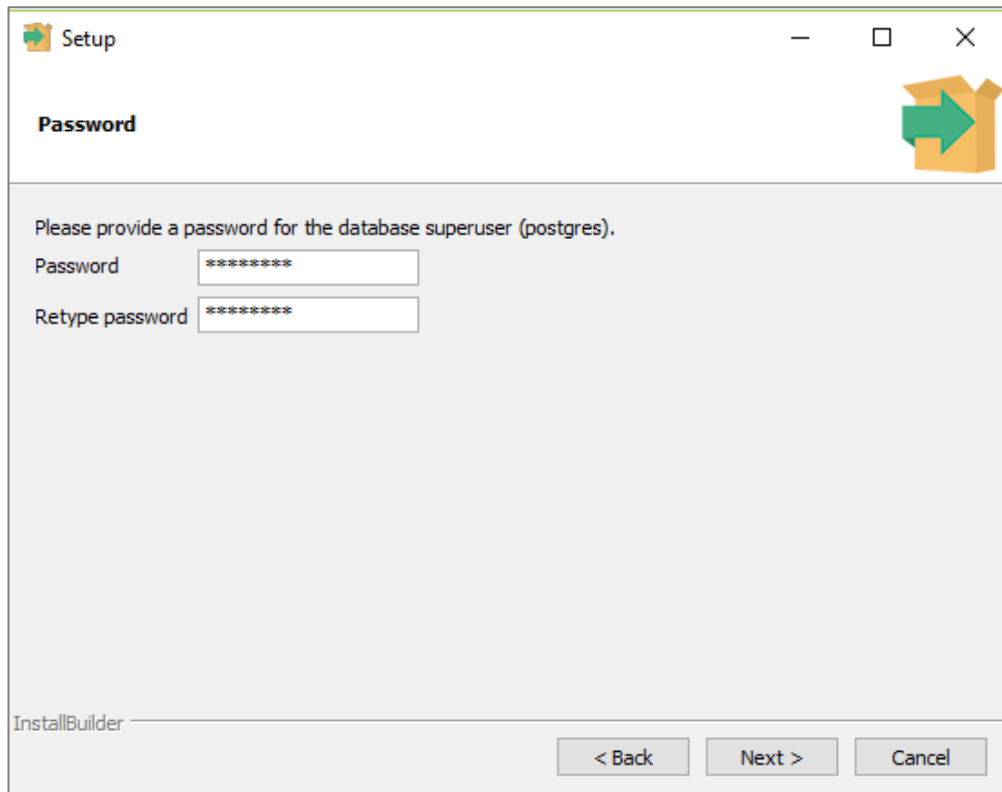
Installation of PostgreSQL

- **Step 4:** Select the database directory where you want to store the data and click on Next.



Installation of PostgreSQL

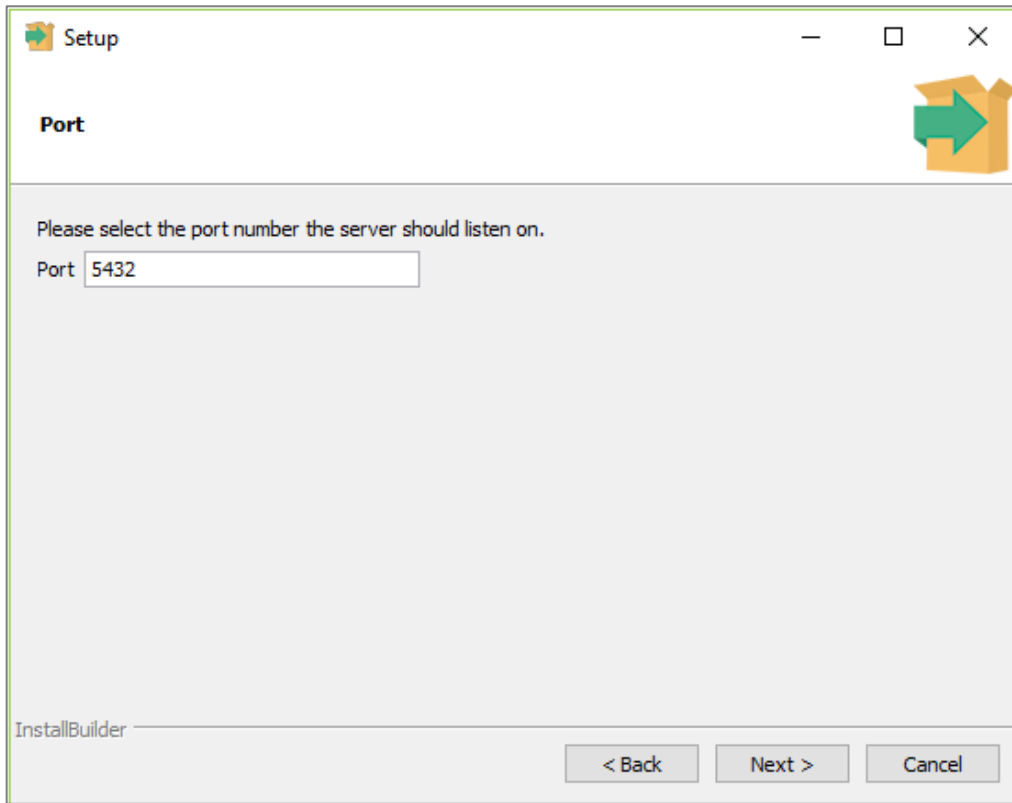
- **Step 5:** Set the password for the database superuser (Postgres)



The screenshot shows a Windows-style window titled "Setup" with a standard icon in the top-left corner. The window has a title bar with minimize, maximize, and close buttons. The main content area is titled "Password" and contains the instruction "Please provide a password for the database superuser (postgres)." Below this, there are two text input fields: "Password" and "Retype password", both containing masked text (seven asterisks). At the bottom of the window, there is a footer area with the text "InstallBuilder" on the left and three buttons: "< Back", "Next >", and "Cancel". A large green arrow icon is visible in the top-right corner of the window.

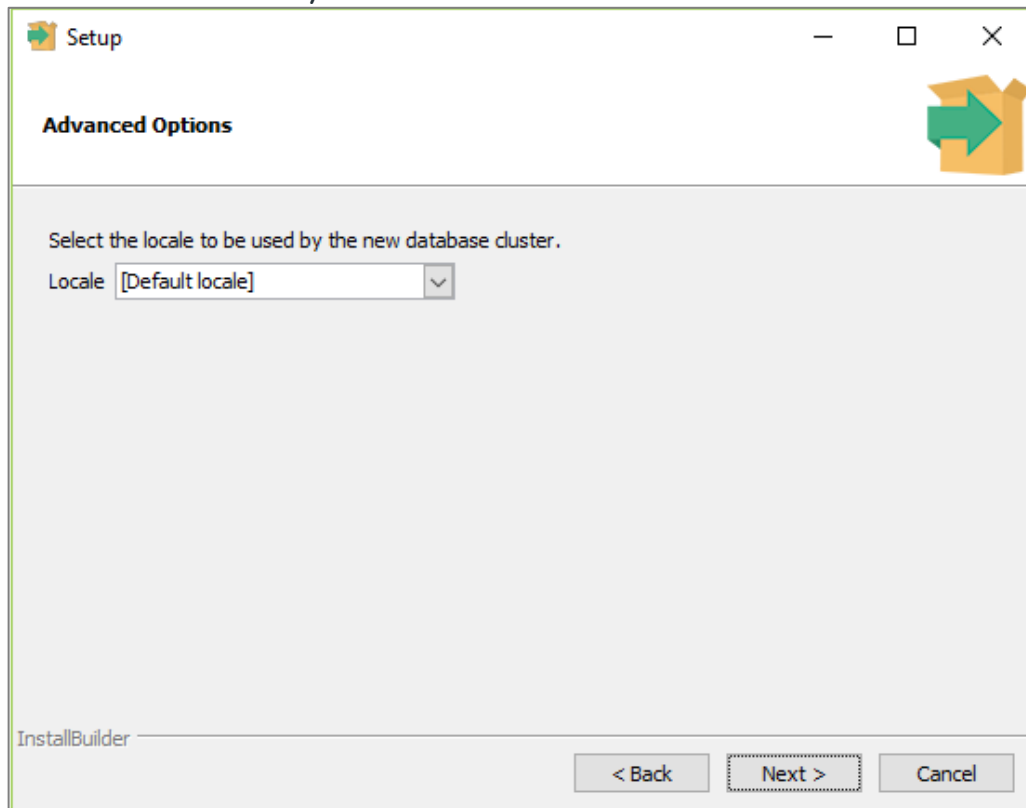
Installation of PostgreSQL

Step 6: Set the port for PostgreSQL. Make sure that no other applications are using this port. If unsure leave it to its default (5432) and click on Next.



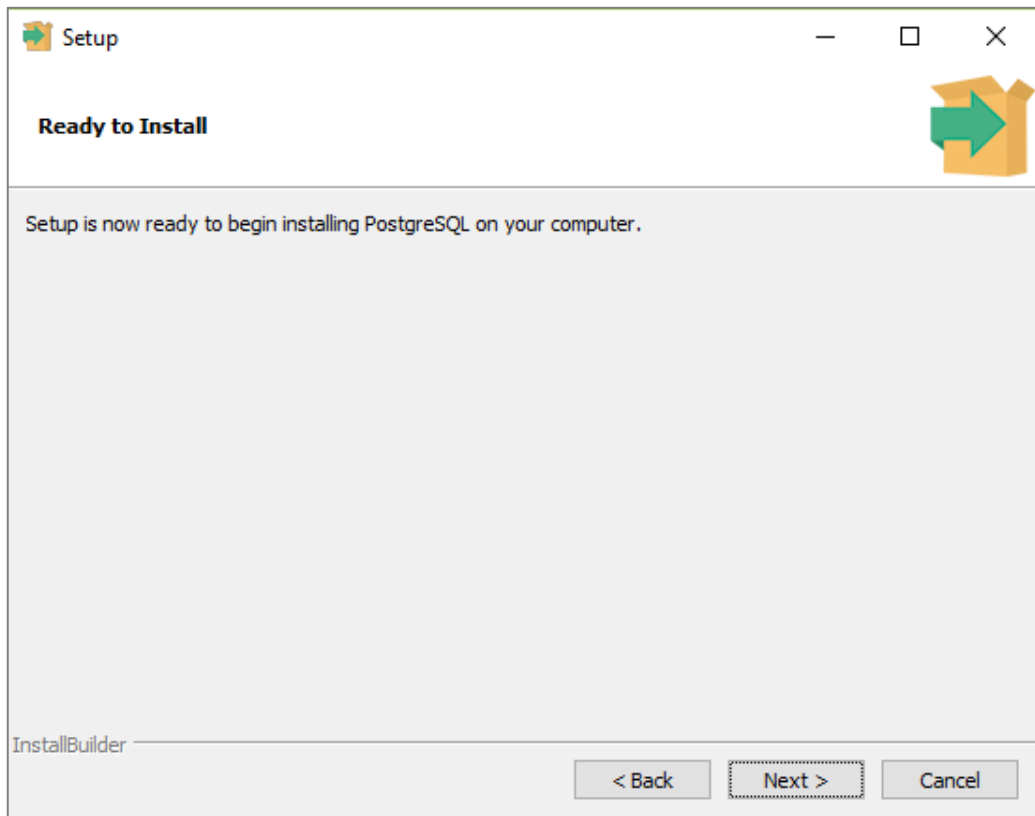
Installation of PostgreSQL

Step 7: Choose the default locale used by the database and click the Next button.



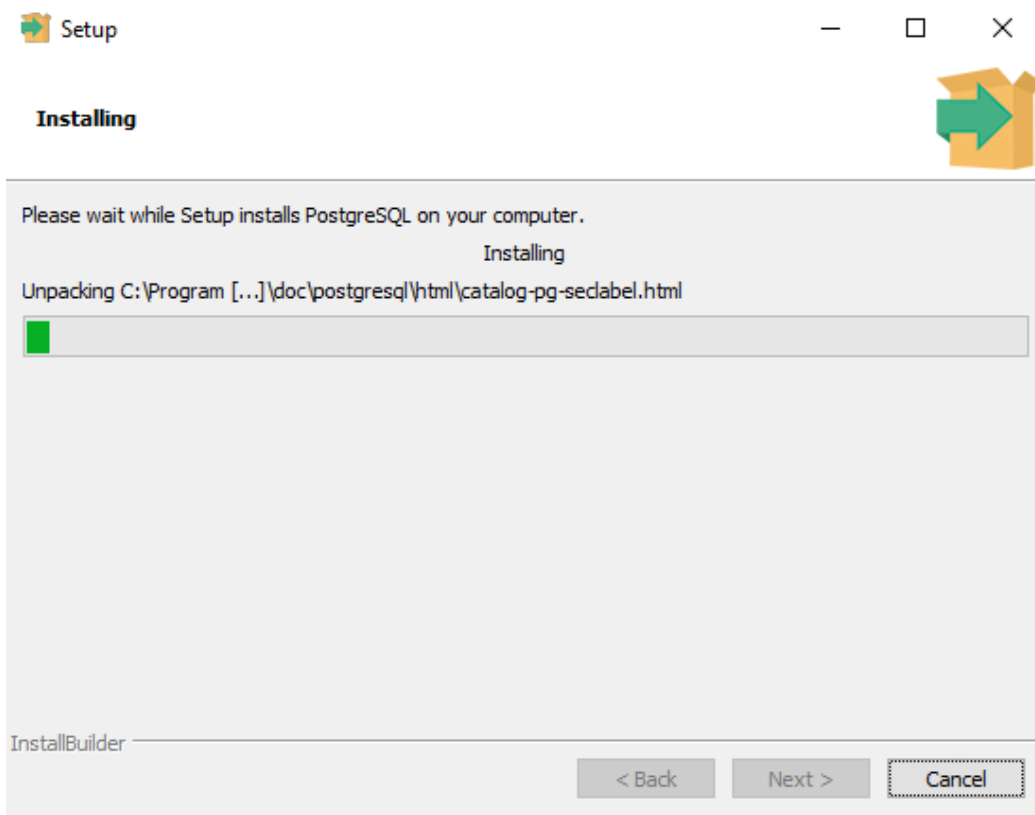
Installation of PostgreSQL

Step 8: Click the Next button to start the installation.



Installation of PostgreSQL

Wait for the installation to complete, it might take a few minutes.



Installation of PostgreSQL

Step 9: Click the Finish button to complete the PostgreSQL installation.



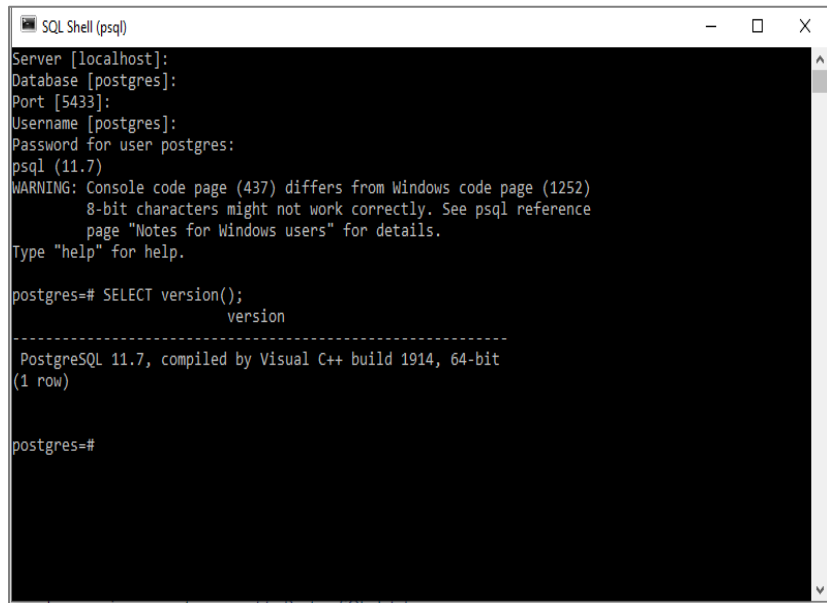
Verifying the Installation of PostgreSQL

There are couple of ways to verify the installation of PostgreSQL like connecting to the database server using some client applications like **pgAdmin** or **psql**.

Step 1: Search for the **psql shell** in the windows search bar and open it.

•**Step 2:** Enter all the necessary information like the server, database, port, username, and password and press **Enter**.

•**Step 3:** Use the command `SELECT version();` you will see the following result:



```
SQL Shell (psql)
Server [localhost]:
Database [postgres]:
Port [5433]:
Username [postgres]:
Password for user postgres:
psql (11.7)
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 11.7, compiled by Visual C++ build 1914, 64-bit
(1 row)

postgres=#
```