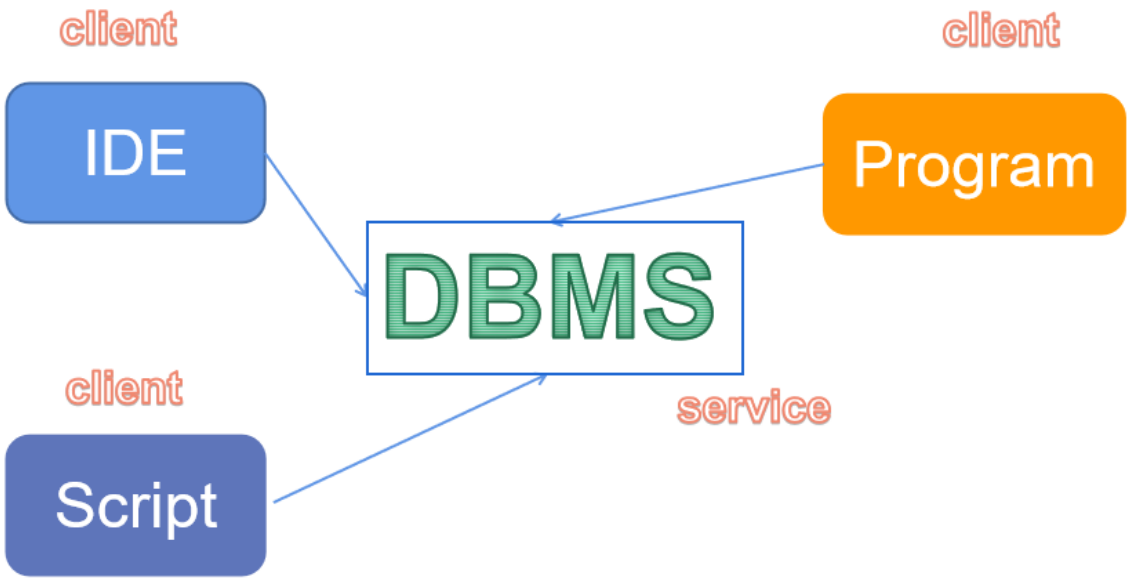
Tutorial Software Installing

by Zhu Yueming, Wu Yechang, Wang Weiyu

- Database: PostgresSQL
- Client: Datagrip (here we only use Datagrip as example, you can choose any other one)
- Take Postgres 12 as example, now upgrade to stable version 16.

When we talk about using a database, we usually mean the services that the database provides.



Here are some official tutorials for you, including common Operating Systems. You may also use our tutorial as follow.

Part 1. Environment Configuration

For MacOS users

Installation & Usage

Here are alternative ways to install postgresql. You can choose **one**.

By Homebrew (Recommended)

1. Install Homebrew. (If you already have it, skip this step.)

   1-1 Prerequisites according to Requirements:

   (1) 64-bit Intel CPU 1

   (2) macOS High Sierra (10.13) (or higher) 2

   (3) Command Line Tools (CLT) for Xcode: `xcode-select --install` , developer.apple.com/downloads or Xcode 3

   (4) A Bourne-compatible shell for installation (e.g. bash or zsh)

   1-2 Open the “Terminal” application

1-3 Enter the following command into a single line of the terminal

```
/usr/bin/ruby -e "$(curl -fsSL
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

1-4 If you meet the problem: "Failed to connect to raw.githubusercontent.com port 443:Operation". You can solve it by the link.

2. Install PostgreSQL.

   2-1 Enter the command to update brew home.

   ```
   brew update
   ```

   2-2 Enter the command to install PostgreSQL.

   ```
   brew install postgresql
   ```

   2-3 Check your PostgreSQL version.

   ```
   postgres --version
   ```

   After the initial installation, it would generate a database named **postgres**.

3. Start & Stop.

   Run `brew info postgres` for details.

   3-1 Manually (will not start after system startup), in command line:

   ```
   pg_ctl -D /usr/local/var/postgres start # To start
   pg_ctl -D /usr/local/var/postgres stop # To stop
   ```

   3-2 Automatically (will start after system startup), in command line:

   ```
   brew services start postgresql # To start
   brew services stop postgresql # To stop
   ```

By Postgres.app

It is a brand new installation method, it's reaaaaaaly simple but there may be some potential issues. Visit PostgresApp for details and optional versions.

1. Installation

   1-1 Download the .dmg file. (Be patient...) Download Link

   1-2 Mount the file, move Postgres to Applications folder, double click Postgres in Applications folder, the dmg file is useless and can be removed.

   1-3 Click "Initialize" to create a new server.

   1-4 You can change path and port in "Server settings" when stopped.

2. Start & Stop

   2-1 Click "Start" to start server.

   2-2 When server is started, double clicking schema can open a command line client connected to this schema.

2-3 Click "Stop" to stop server.

By Installer

1. Go to Postgresql Download Page, download installer. When the wizard prompts you to choose where to install PostgreSQL, point it to the **apps** subdirectory of your i.e.

   `/Library/PostgreSQL/12.`

2. Keep track of the **database superuser** name and **password**. You'll need these to initially create the LabKey database, the LabKey database user, and grant that user the owner role.
3. Keep track of the **database port**. (5432 for default)

By Docker

See For Docker users

Uninstallation

By Homebrew

```
brew uninstall postgres
```

By Postgres.app

1. Open **Finder**.
2. Go to Applications.
3. Move Postgres.app to Trash.

By Installer

There is a uninstall-postgresql.app in your installation directory. (i.e. /Library/PostgreSQL/12)

```
open /Library/PostgreSQL/12/uninstall-postgresql.app
```

For rest files, see this sof answer.

By Docker

See For Docker users

For Linux users

Installation

By Package manager

Take **Ubuntu** as an example.

1. Following command will access each URL in the source list and read the software list and save it on the local computer.

   ```
   sudo apt update
   ```

2. Install PostgreSQL client first

   ```
   sudo apt install postgresql-client
   ```

Then Install PostgrsSQL server

```
sudo apt install postgresql
```

Generally, after the installation, the postgreSQL server will automatically open on **port 5432** of the machine.

3. Check your PostgreSQL version

```
postgres --version
```

4. Noticed that, after the initial installation, it would generate three elements:

   4-1 a *database* named **postgres** .

   4-2 a *database user* named **postgres**.

   4-3 a *Linux system user* named **postgres**.

By Docker

See For Docker users

Uninstallation

By Package manager

```
sudo apt remove postgresql postgresql-client
```

By Docker

See For Docker users

For Windows users

Installation

By Installer

1. Go to Postgresql Download Page, download installer. When the wizard prompts you to choose where to install PostgreSQL, point it to the **apps** subdirectory of your i.e.

   C:\labkey\apps\postgresql-12.6

2. Keep track of the **PostgreSQL Windows Service account name and password**. LabKey Server needs to ask for it so that we can pass it along to the PostgreSQL installer.
3. Keep track of the **database superuser** name and password. You'll need these to initially create the LabKey database, the LabKey database user, and grant that user the owner role.

By Chocolatey

If you haven't chocolatey, go and get one.

```
choco install postgresql
```

By Docker

See For Docker users

Uninstallation

Universal way (by installer or choco)

1. Click `Start Menu`, Go to `Settings` > `Apps` > `Apps & features`.
2. Select `PostgreSQL`, click `Remove`.

By choco

```
choco uninstall postgresql
```

By Docker

See For Docker users

For Docker users

If you haven't Docker environment, please choose another installation method.

Installation

In command line

```
docker run --name some-postgres -p 5432:5432 -e
POSTGRES_PASSWORD=mysecretpassword -d postgres
```

Uninstallation

If your postgres container names "some-postgres"

```
docker stop some-postgres # Stop container
docker rm some-postgres # Remove container
docker rmi postgres # Remove image
```

Part 2. How to use PostgreSQL

In command line:

1. Enter command to visit postgres database:

   Mac OX user:

   ```
   psql postgres
   ```

   Linux user:

   Step1: Changed to Postgres database user

   ```
   sudo su - postgres
   ```

   Step2: connect PostgreSQL database by input psql command:

```
psql
```

Then system prompt would be `postgres=#`, which means you have connected to postgres database

Windows user:

We suggest using client, for example, Datagrip.

2. Finding all roles in postgreSQL

```
\du
```

3. Create a superuser named `checker`

```
create user checker with superuser
```

4. Change password, and then enter password `123456`

```
\password checker
```

5. Create a database named `cs307`

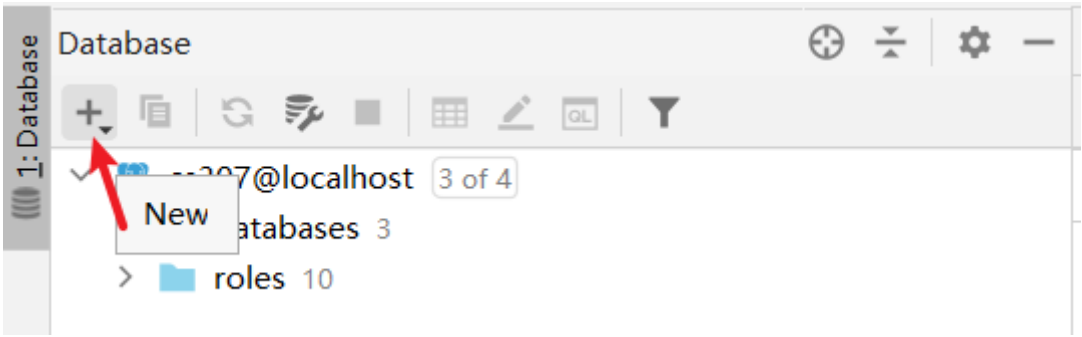
```
create database cs307;
```

6. Exit database from command line

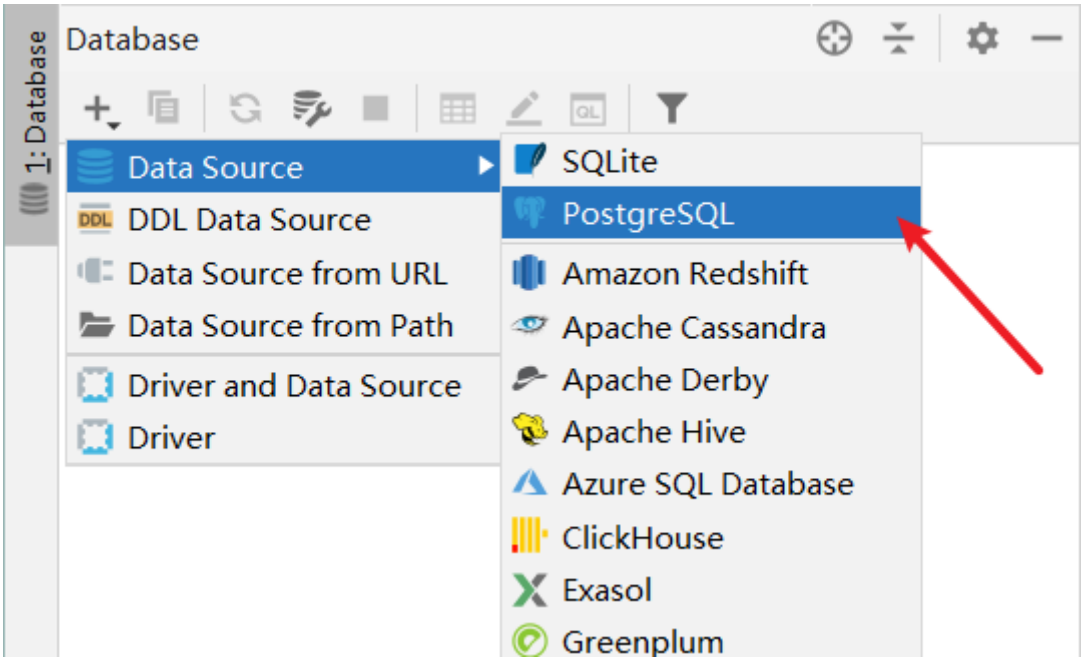
```
\q
```

Part3. Datagrip

1. Here to download Datagrip, choose your OS and then install it. If this App need an account, you could apply for one using Free Education license.
2. Add a postgreSQL client

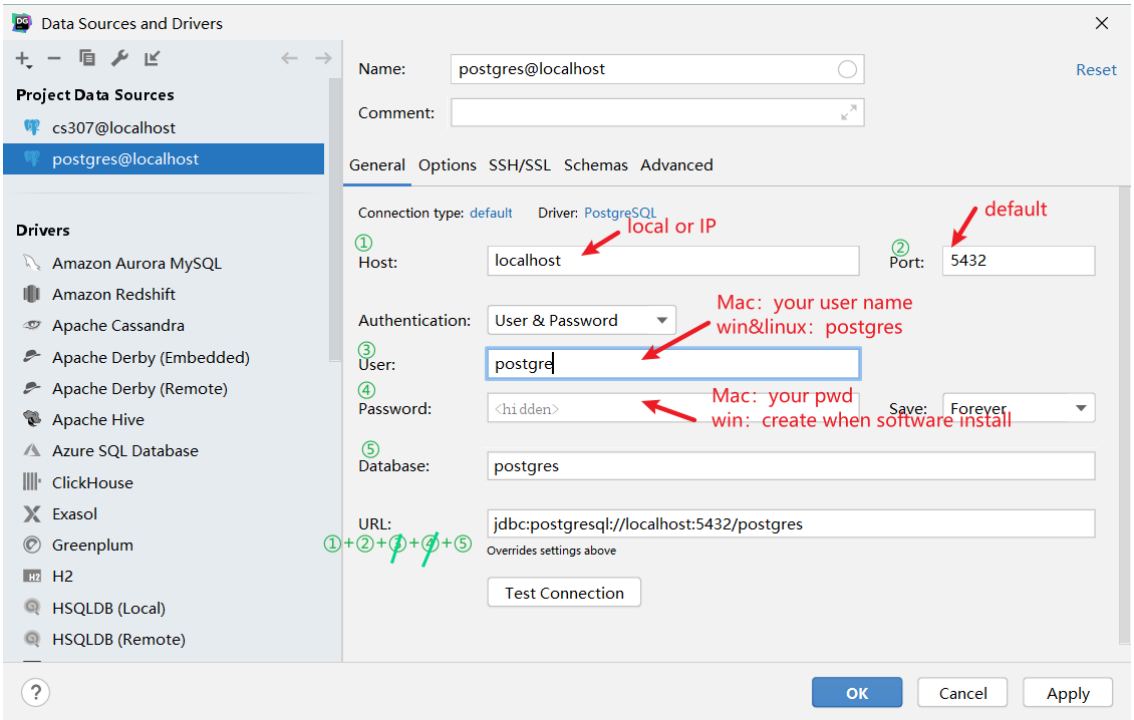


3. Choose data source Postgres

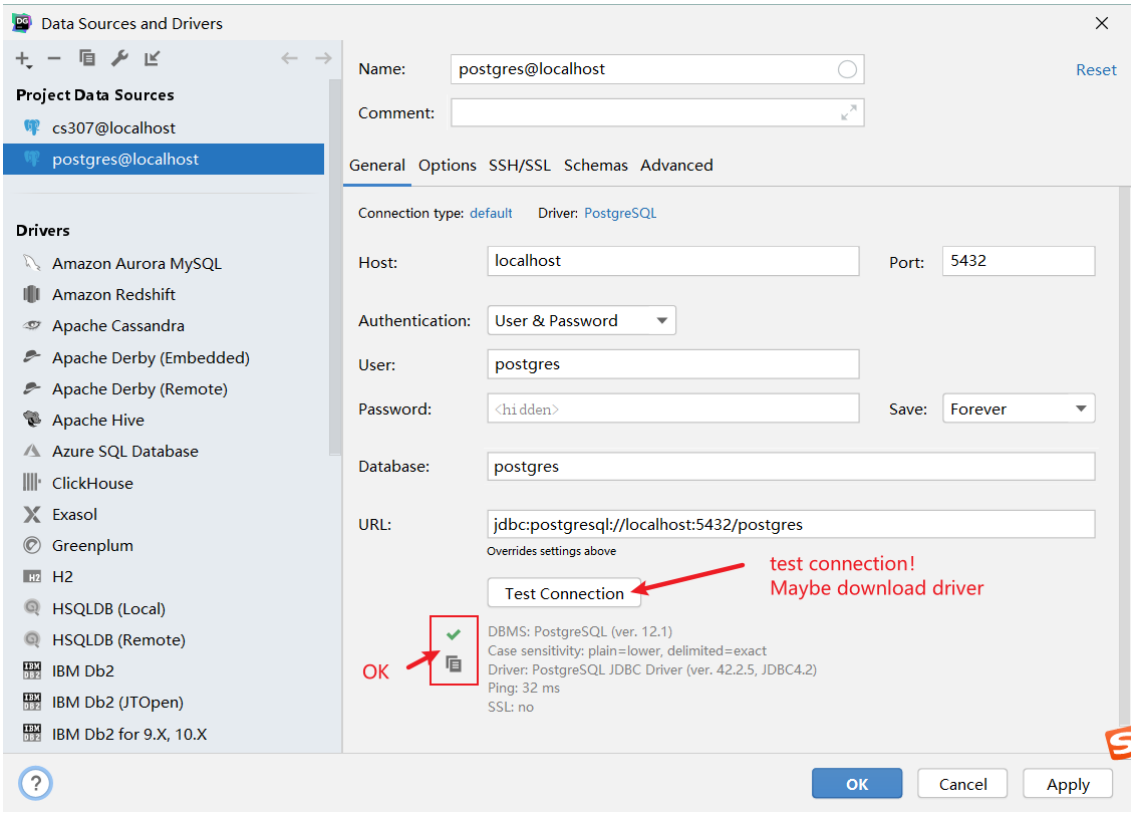


4. Fill in host,port,user,password,database as following. Some of them should be default mode. Notice your Operating System, there will be little difference.

   Tips: Here URL contain all information of host, port, user, password, database.



5. Press `Test Connection'` button, usually extra driver should be download.


Data Sources and Drivers
Project Data Sources
cs307@localhost
postgres@localhost
Drivers
Amazon Aurora MySQL
Amazon Redshift
Apache Cassandra
Apache Derby (Embedded)
Apache Derby (Remote)
Apache Hive
Azure SQL Database
ClickHouse
Exasol
Greenplum
H2
HSQLDB (Local)
HSQLDB (Remote)
IBM Db2
IBM Db2 (JTOpen)
IBM Db2 for 9.X, 10.X
Name: postgres@localhost
Reset
Comment:
General Options SSH/SSL Schemas Advanced
Connection type: default
Driver: PostgreSQL
Host: localhost
Port: 5432
Authentication: User & Password
User: postgres
Password: <hidden>
Save: Forever
Database: postgres
URL: jdbc:postgresql://localhost:5432/postgres
Overrides settings above
Test Connection
test connection!
Maybe download driver
OK
DBMS: PostgreSQL (ver. 12.1)
Case sensitivity: plain=lower, delimited=exact
Driver: PostgreSQL JDBC Driver (ver. 42.2.5, JDBC4.2)
Ping: 32 ms
SSL: no
OK
Cancel
Apply