FIT5137
Installation Guide (Mac)

This is a guide for installing **MongoDB**, **Cassandra**, and **Neo4j** on **Mac** environment. Please ensure that you download the correct version for each software. We encourage you to install all software on your own device as the labs' devices have some issues with MongoDB and Cassandra.

Software needed throughout the semester:

- MongoDB Community Server 4.0.11
- Cassandra 3.11.4
- Neo4j Desktop 1.2.1

MongoDB

1. Go to MongoDB website:
   https://www.mongodb.com/download-center - production
   and download the correct version of MongoDB

2. After downloading MongoDB, move the .tgz file that you downloaded to the folder where you want MongoDB installed. In this case, we'll say that we want MongoDB to live in our home folder, and so the commands might look something like this:
   ```
   > cd Downloads
   > mv mongodb-osx-ssl-x86_64-4.0.11.tgz ~/
   ```
   Change your current working folder to the home folder
   ```
   > cd ~
   ```

3. Extract MongoDB from the downloaded archive
   ```
   > tar -zxvf mongodb-osx-ssl-x86_64-4.0.11.tgz
   ```
   Rename the extracted folder to **mongodb**
   ```
   > mv mongodb-osx-x86_64-4.0.11 mongodb
   ```
   Change your current working folder to the **mongodb** folder
   ```
   > cd mongodb
   ```

4. Create a directory where MongoDB will store data
   ```
   > sudo mkdir -p /data/db
   ```
   There may be a prompt to enter password
   ```
   > # Enter your password
   ```

5. Make sure this directory(/data/db) has right permissions
   ```
   > sudo chmod 777 /data/db
   ```
6. In a new terminal window, run the mongo daemon by entering:
   ```
   > ~/mongodb/bin/mongod
   ```

This will start the Mongo Server

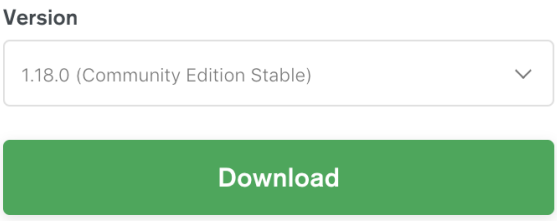
7. Open another terminal, run the mongo shell
   `> ~/mongodb/bin/mongo`

   Optional Step:
   To run mongo deamon and mongo shell by simply using the command **mongod** (instead of ~/mongodb/bin/mongod) and **mongo** (instead of ~/mongodb/bin/mongo) you will need to add 2 lines to the end of the .bash_profile file:
   `> cd ~`
   Creates a backup of the .bash_profile file
   `> cp .bash_profile .bash_profile_backup`
   In the following step it is very important to ensure there are 2 > i.e. >>
   `> echo 'export MONGO_PATH=~/mongodb' >> .bash_profile`
   `> echo 'export PATH=$PATH:$MONGO_PATH/bin' >> .bash_profile`
   Then you will need to restart the terminal

8. You can also check the databases through MongoDB Compass.
   MongoDB Compass provides a more user-friendly interface of MongoDB.

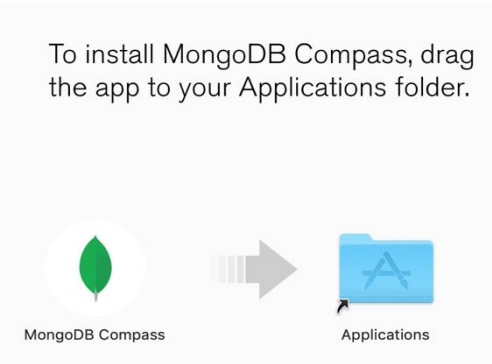
   Download MongoDB Compass from:
   https://www.mongodb.com/download-center/compass?jmp=hero

   If the Download button does not work after clicking the download button then you can change the version to 1.18.0 (Community Edition Stable)



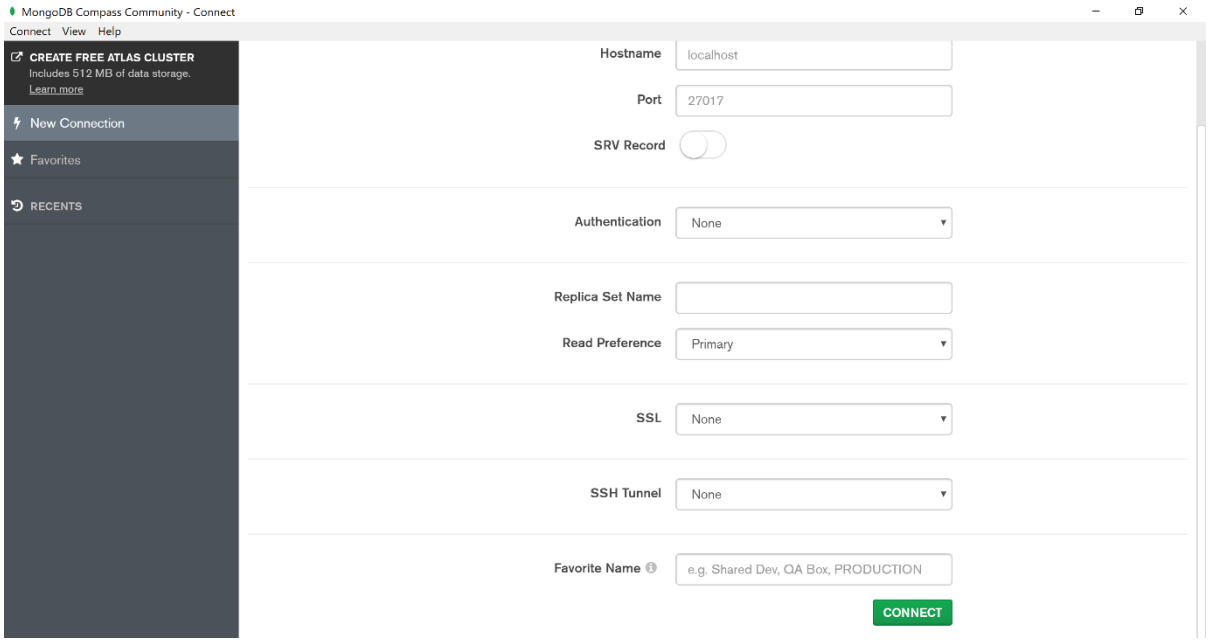
9. After you have finished downloading MongoDB Compass, run the installer

10. Drag the icon into Applications and double click the icon.

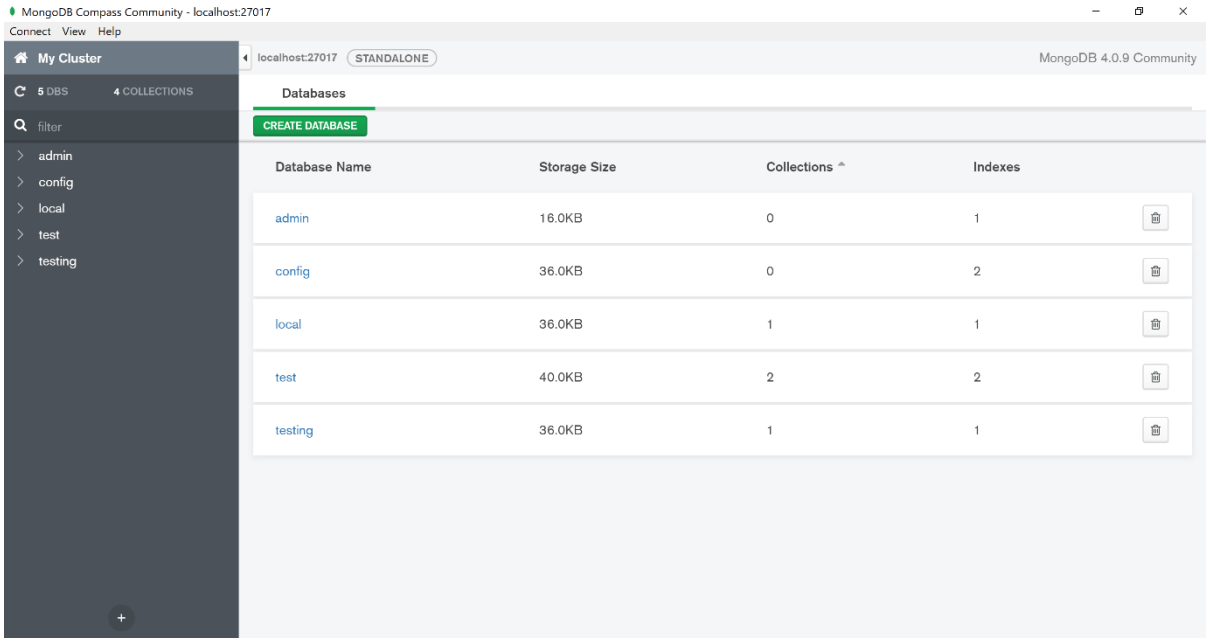


11. Click next and finish the installation.

12. Open MongoDB Compass, then click **CONNECT** button.



13. The list of databases will be shown in MongoDB Compass, similar to the figure below.

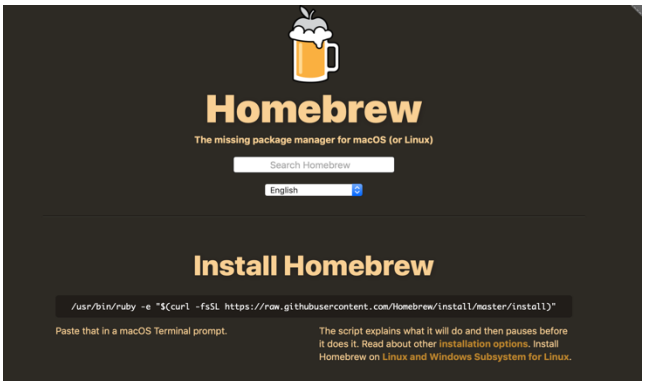


Cassandra (Using Homebrew)

Warning: Before installing Cassandra, you must ensure that you have homebrew installed in your machine.

Set Homebrew Path:

1. Open a browser, and type brew.sh in the search bar.



2. Copy the command in the website.

3. Open terminal and paste the command.

```
(base) Huashuns-MacBook-Pro:~ huashun$ /usr/bin/ruby -e "$(curl -fsSL https://raw.githubuser
content.com/Homebrew/install/master/install)"
==> This script will install:
/usr/local/bin/brew
/usr/local/share/doc/homebrew
/usr/local/share/man/man1/brew.1
/usr/local/share/zsh/site-functions/_brew
/usr/local/etc/bash_completion.d/brew
/usr/local/Homebrew
==> The following existing directories will be made group writable:
/usr/local/bin
/usr/local/include
/usr/local/lib
/usr/local/lib/pkgconfig
==> The following existing directories will have their owner set to huashun:
/usr/local/bin
/usr/local/include
/usr/local/lib
/usr/local/lib/pkgconfig
==> The following existing directories will have their group set to admin:
/usr/local/bin
/usr/local/include
/usr/local/lib/pkgconfig
==> The following new directories will be created:
/usr/local/etc
/usr/local/sbin
/usr/local/share
/usr/local/var
/usr/local/opt
/usr/local/share/zsh
/usr/local/share/zsh/site-functions
```

4. Then we can test if homebrew is installed successfully or not. We can type brew in the terminal and the Command Prompt screen should look like the Figure below.

   $ brew

```
(base) Huashuns-MacBook-Pro:~ huashun$ brew
Example usage:
  brew search [TEXT|/REGEX/]
  brew info [FORMULA...]
  brew install FORMULA...
  brew update
  brew upgrade [FORMULA...]
  brew uninstall FORMULA...
  brew list [FORMULA...]

Troubleshooting:
  brew config
  brew doctor
  brew install --verbose --debug FORMULA

Contributing:
  brew create [URL [--no-fetch]]
  brew edit [FORMULA...]

Further help:
  brew commands
  brew help [COMMAND]
  man brew
  https://docs.brew.sh
(base) Huashuns-MacBook-Pro:~ huashun$ brew install
Usage: brew install [options] formula

Install formula.

formula is usually the name of the formula to install, but it can be specified
in several different ways.

Unless HOMEBREW_NO_INSTALL_CLEANUP is set, brew cleanup will be run for the
installed formulae or, every 30 days, for all formulae.

    -d, --debug                    If brewing fails, open an interactive
                                   debugging session with access to IRB or a
```

Cassandra setup:

1. Update the repository index of homebrew in the terminal

   $ brew update

2. Install Cassandra

   $ brew install cassandra

   This installs Cassandra on location **/usr/local/cellar/Cassandra**.

```
(base) Huashuns-MacBook-Pro:~ huashun$ brew install cassandra
==> Installing dependencies for cassandra: gdbm, openssl, readline, sqlite, xz, python and cython
==> Installing cassandra dependency: gdbm
==> Downloading https://homebrew.bintray.com/bottles/gdbm-1.18.1.mojave.bottle.1.tar.gz
######################################################################## 100.0%
==> Pouring gdbm-1.18.1.mojave.bottle.1.tar.gz
🍺  /usr/local/Cellar/gdbm/1.18.1: 20 files, 586.8KB
==> Installing cassandra dependency: openssl
==> Downloading https://homebrew.bintray.com/bottles/openssl-1.0.2s.mojave.bottle.tar.gz
==> Downloading from https://akamai.bintray.com/c4/c4a762d719c2be74ac686f1aafabb32f3c5d5ff3a98935c4925a1ddb9c
######################################################################## 100.0%
==> Pouring openssl-1.0.2s.mojave.bottle.tar.gz
==> Caveats
A CA file has been bootstrapped using certificates from the SystemRoots
keychain. To add additional certificates (e.g. the certificates added in
the System keychain), place .pem files in
  /usr/local/etc/openssl/certs

and run
  /usr/local/opt/openssl/bin/c_rehash

openssl is keg-only, which means it was not symlinked into /usr/local,
because Apple has deprecated use of OpenSSL in favor of its own TLS and crypto libraries.

If you need to have openssl first in your PATH run:
  echo 'export PATH="/usr/local/opt/openssl/bin:$PATH"' >> ~/.bash_profile

For compilers to find openssl you may need to set:
  export LDFLAGS="-L/usr/local/opt/openssl/lib"
  export CPPFLAGS="-I/usr/local/opt/openssl/include"
```

3. Start Cassandra

   $ brew services start cassandra

```
(base) Huashuns-MacBook-Pro:~ huashun$ brew services start cassandra
==> Successfully started `cassandra` (label: homebrew.mxcl.cassandra)
```

4. Go to /usr/local/Cellar/cassandra/3.11.4/bin

   $ cd /usr/local/Cellar/cassandra/3.11.4/bin

5. Initialize cassandra

   $ cassandra

Note if there is a Java VM error then
Step 1: Download **Java 8** from:
http://www.oracle.com/technetwork/java/javase/downloads/jdk8-downloads-2133151.html
and install it

Step 2: After installation of Java 8, to confirm installation of all versions on your MAC, type the following command in the terminal:

$ /usr/libexec/java_home -V

Step 3: Edit .bash_profile and add 1.8 as default. (Add the below line to bash_profile file).

export JAVA_HOME=$(/usr/libexec/java_home -v 1.8)
You can do this in 2 ways
(i) open **~/.bash_profile**

paste the export JAVA_HOME=$(/usr/libexec/java_home -v 1.8) to the end of the file and save it

source **~/.bash_profile**

OR

(ii) Use the command:
$ echo 'export JAVA_HOME=$(/usr/libexec/java_home -v 1.8)' >> ~/.bash_profile

Step 4: Reload bash_profile

$ source ~/.bash_profile

Step 5: Confirm current version of Java is Java 8

$ java -version

6. Open another terminal, then start CQL

   $ cqlsh

```
(base) Huashuns-MacBook-Pro:~ huashun$ cqlsh
Connected to Test Cluster at 127.0.0.1:9042.
[cqlsh 5.0.1 | Cassandra 3.11.4 | CQL spec 3.4.4 | Native protocol v4]
Use HELP for help.
cqlsh>
```

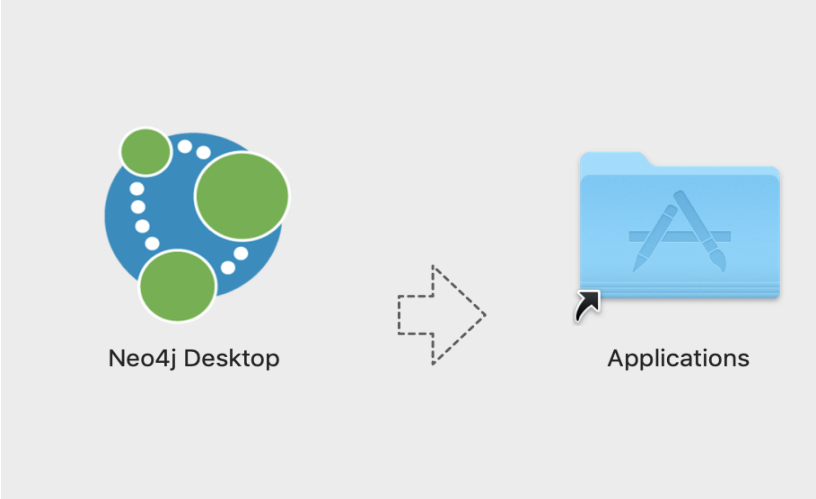
7. Stop Cassandra

$ brew services stop cassandra

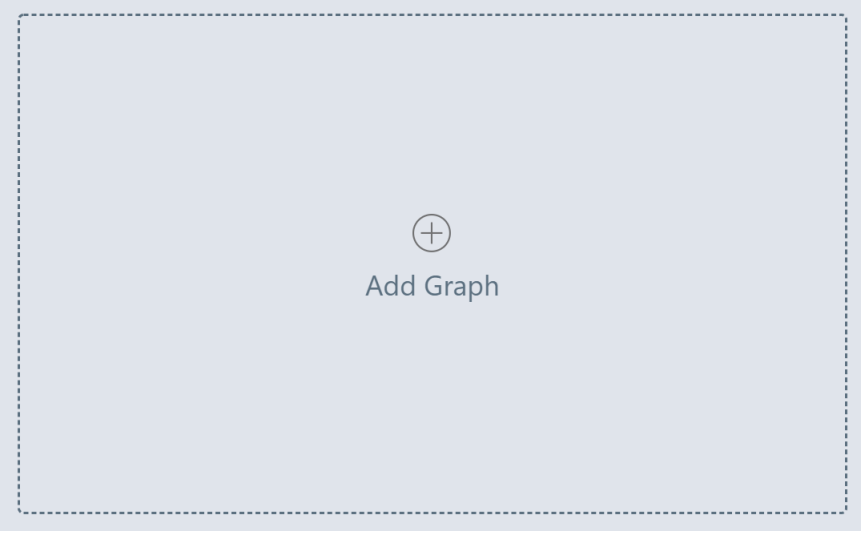
```
(base) Huashuns-MacBook-Pro:~ huashun$ brew services stop cassandra
Stopping `cassandra`... (might take a while)
==> Successfully stopped `cassandra` (label: homebrew.mxcl.cassandra)
```

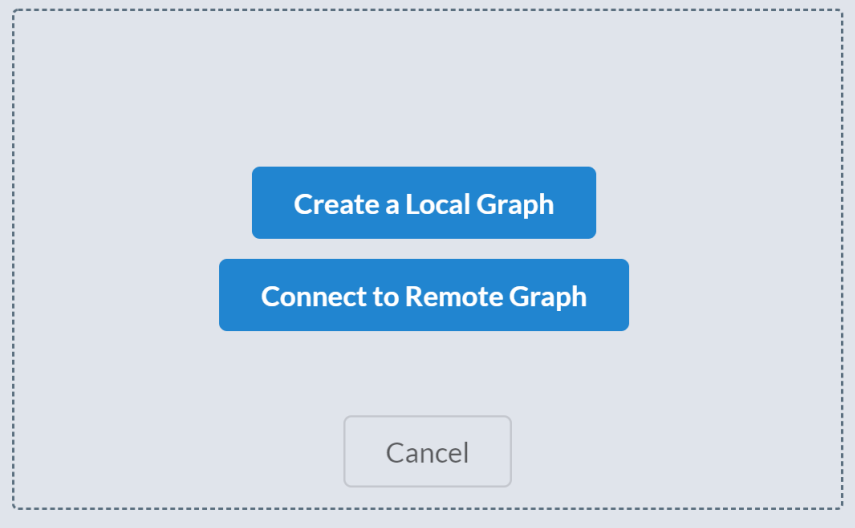
Neo4j

1. Download Neo4j from: https://neo4j.com/download-center/#desktop
2. Choose Neo4j Desktop version **1.2.1**. You may need to register to Neo4j before downloading the software.
3. Click download.
4. After you have finished downloading Neo4j, run the installer.
5. Drag the icon into Applications and double click the icon.

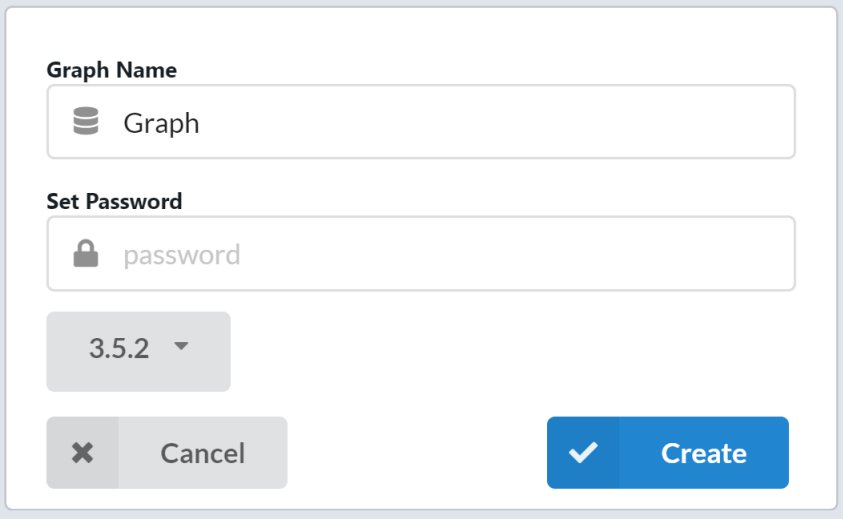


6. Once it is installed, run Neo4j.
7. You should use your **Monash account to register**, then you are in the system.
8. Click on **Add Graph → Create a Local Graph**.

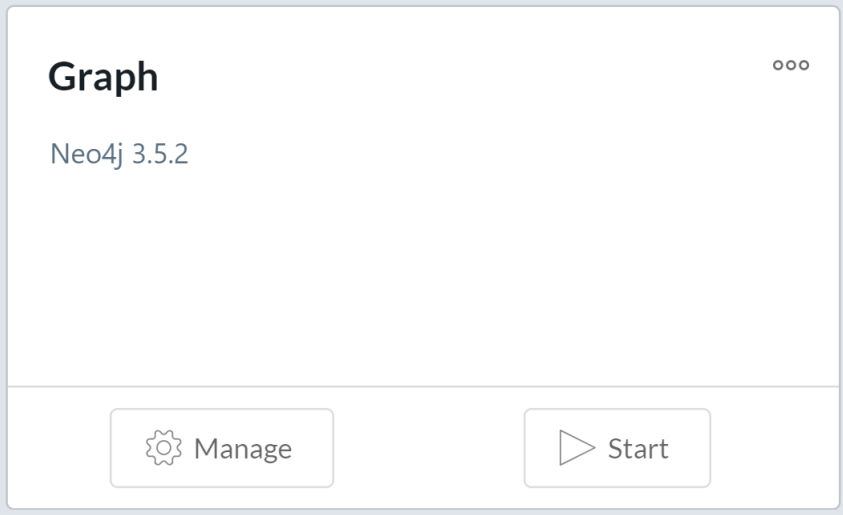




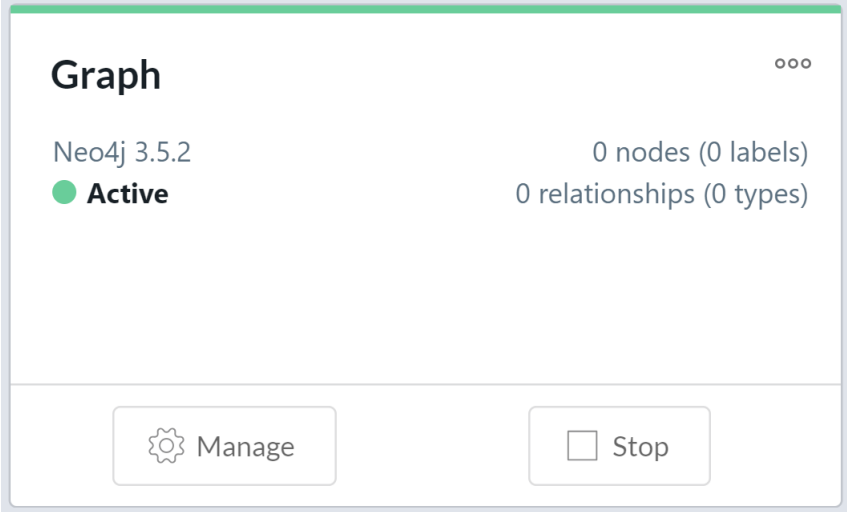
9. Fill in the **Graph name** and **Password**, choose 3.5.2, then select **Create**.



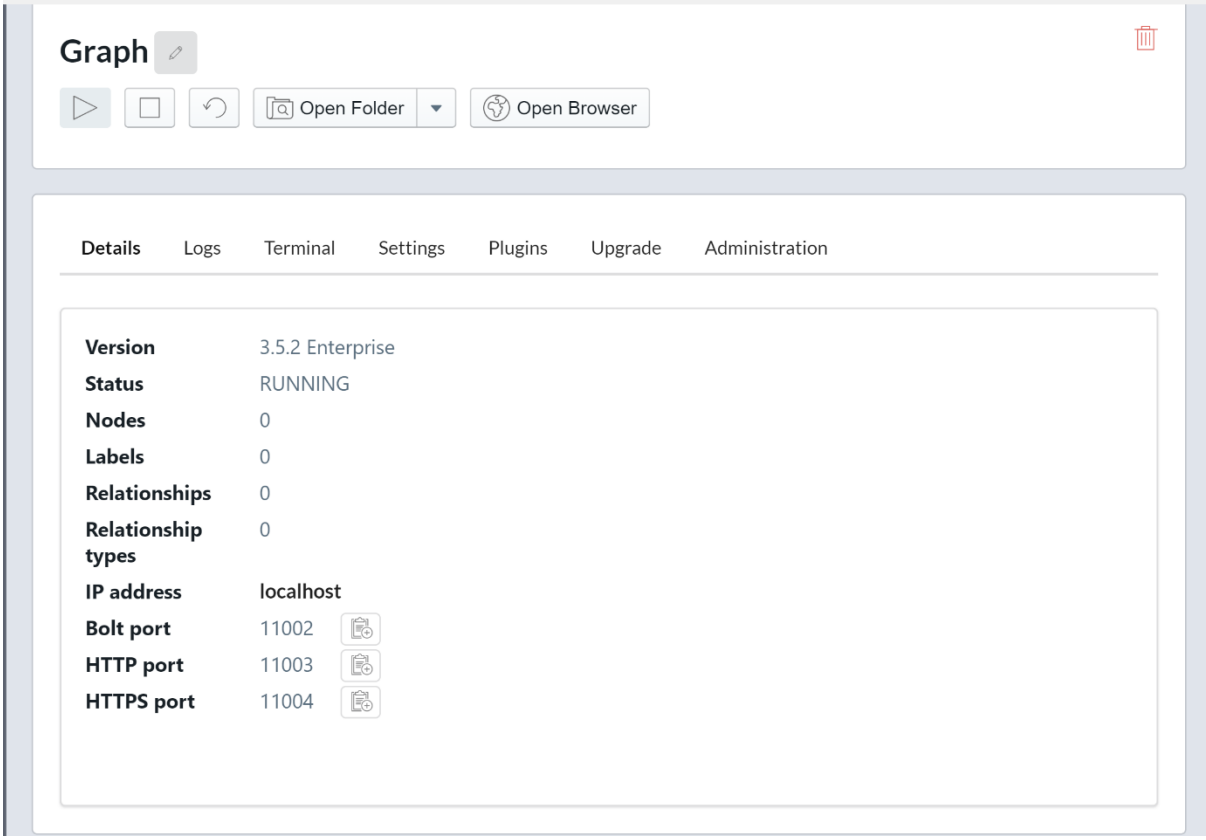
10. After your graph has been created, click **Start**.



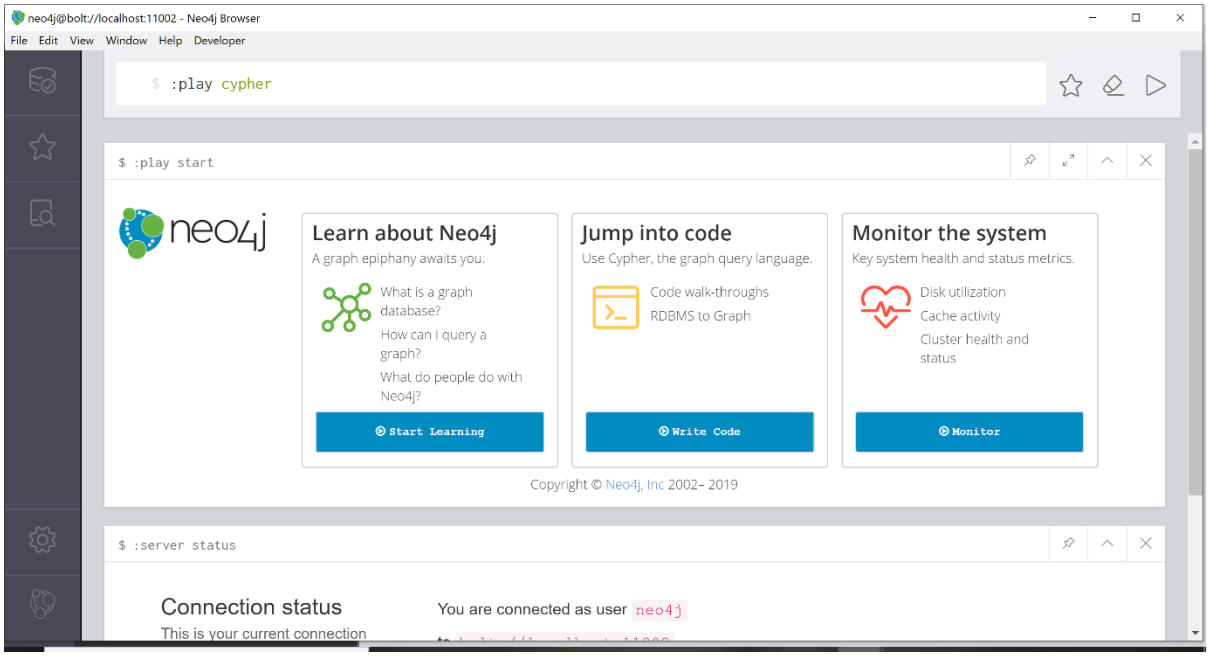
11. Once the graph is Active, select **Manage**.



12. Select **Open Browser**.



13. The browser will pop up. Type "**:play cypher**", the click the play button.


neo4j@bolt://localhost:11002 - Neo4j Browser
File Edit View Window Help Developer
$:play cypher
$:play start
neo4j
Learn about Neo4j
A graph epiphany awaits you.
What is a graph database?
How can I query a graph?
What do people do with Neo4j?
Start Learning
Jump into code
Use Cypher, the graph query language.
Code walk-throughs
RDBMS to Graph
Write Code
Monitor the system
Key system health and status metrics.
Disk utilization
Cache activity
Cluster health and status
Monitor
Copyright © Neo4j, Inc 2002- 2019
$:server status
Connection status
This is your current connection
You are connected as user neo4j