Book Recommendation And Sharing
Steps to execute project

Open folder Book-Recommendation-and-Sharing

1. Installing required packages :

- **Installing python3 :**
- **Installing pip :**
- **Installing the required modules :**

  Open terminal in folder and execute following commands :-

  ```
  $ sudo apt-get install libpq-dev
  $ pip install psycopg2
  $ pip install -r requirements.txt
  ```

---

2. Connecting databases :

Graph database

Installing Neo4j in ubuntu:

- sudo apt update
- sudo apt install apt-transport-https ca-certificates curl software-properties-common
- curl -fsSL https://debian.neo4j.com/neotechnology.gpg.key | sudo apt-key add -
- sudo add-apt-repository "deb https://debian.neo4j.com stable 4.1"

- sudo apt install neo4j
- sudo systemctl enable neo4j.service
- Go to browser and open link : http://localhost:7474/browser/
  Write username : neo4j
  password : neo4j
  and set new password : 123

- Close browser and close the terminal

Installing Neo4j desktop in ubuntu:

-> Go to link and download neo4j desktop

-> Go to Downloads folder

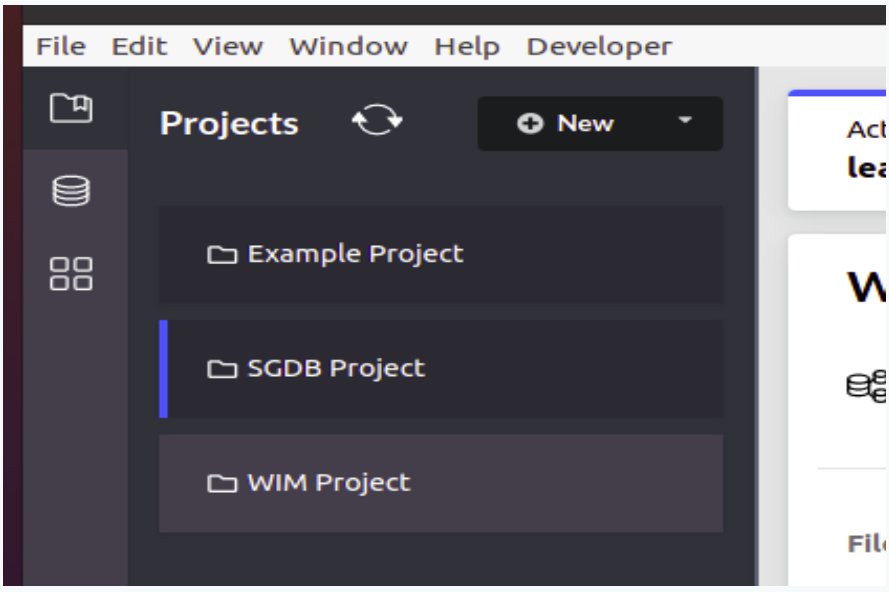
-> Open terminal and execute the following commands :

-> $ chmod a+x <downloaded filename>

-> $ ./<downloaded filename>

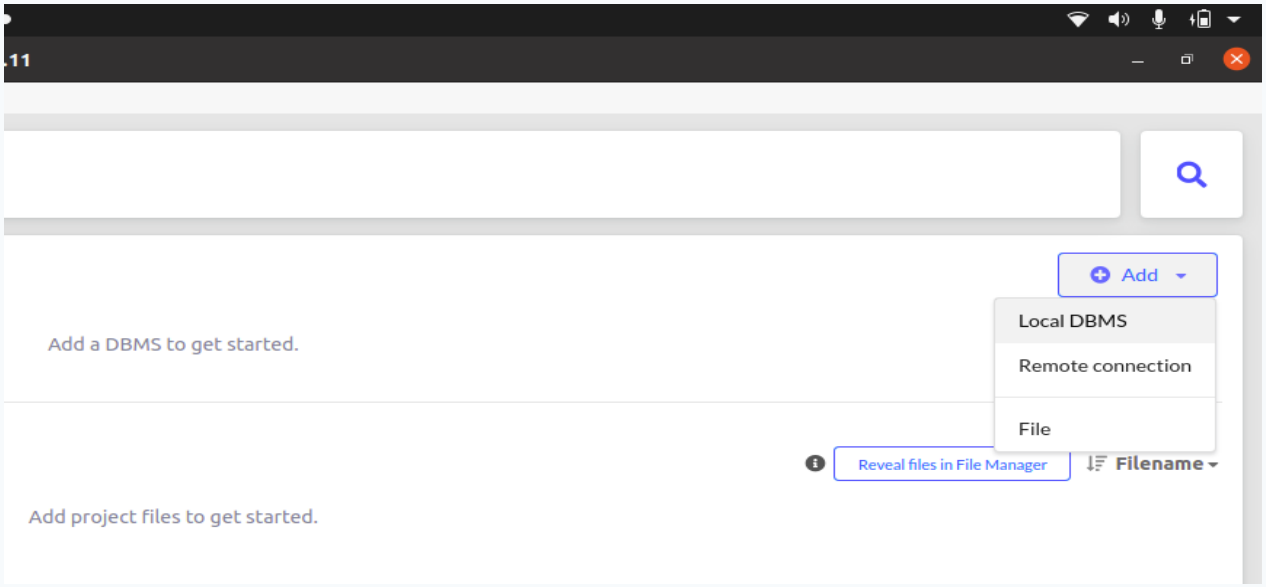
-> Follow the steps as asked in window

-> Create new project by clicking on “New”



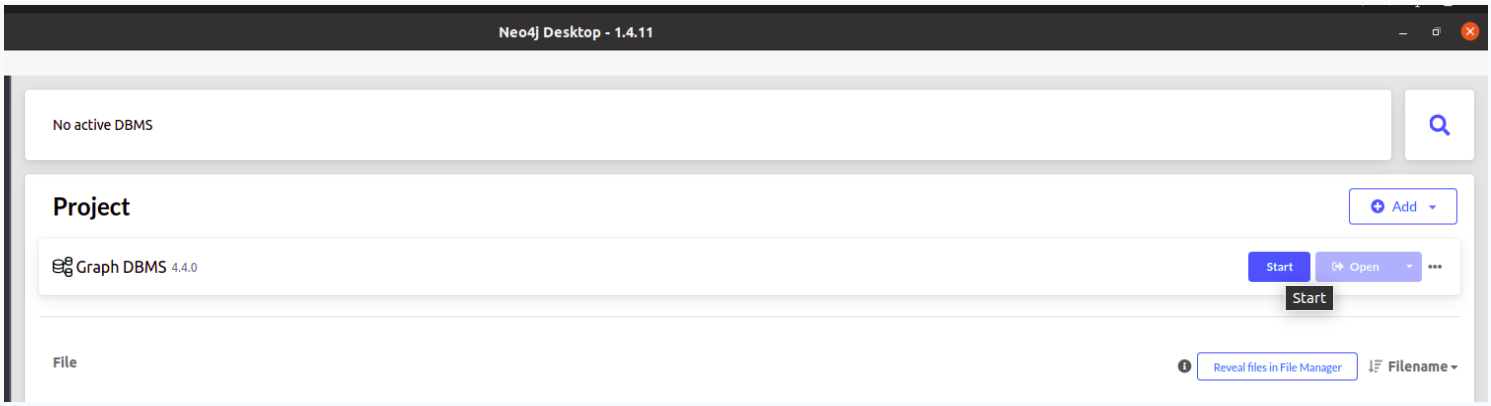
-> Select the new Project created

-> Press on Add -> Local DBMS

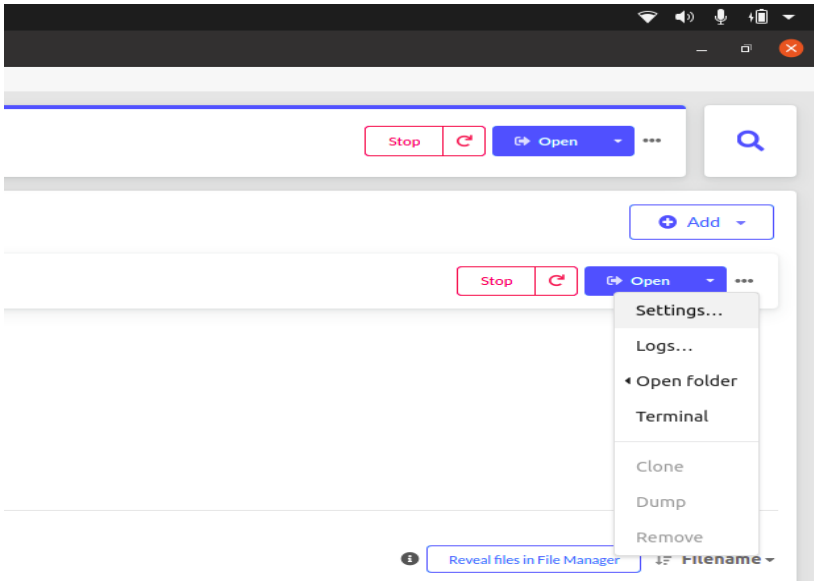


-> Enter password as 123

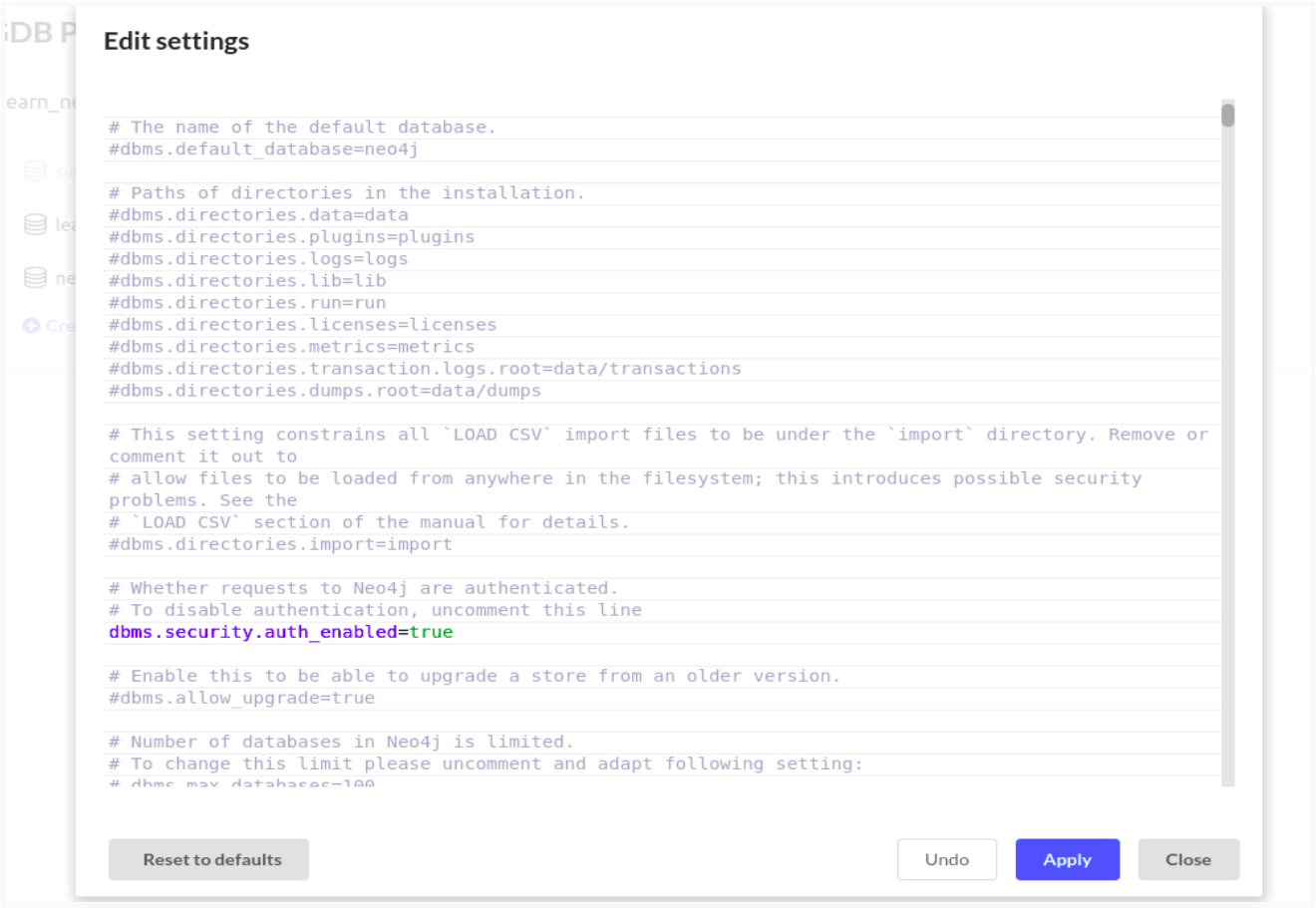
-> Press on Start



-> Press on 3 dots to right of Open, Got to Settings

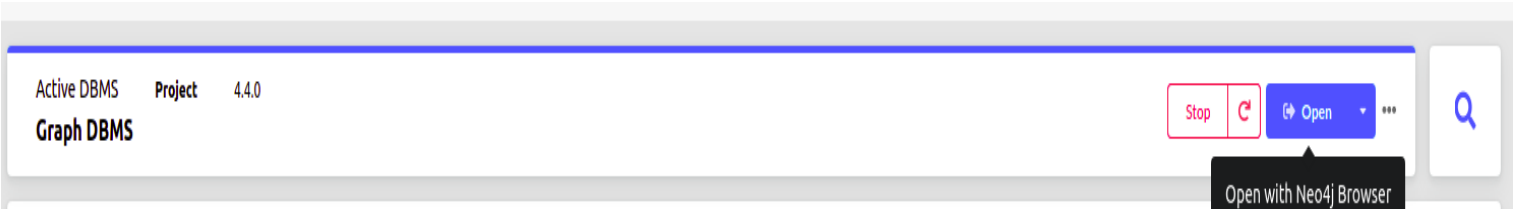


-> comment the line dbms.directories.import=import using #



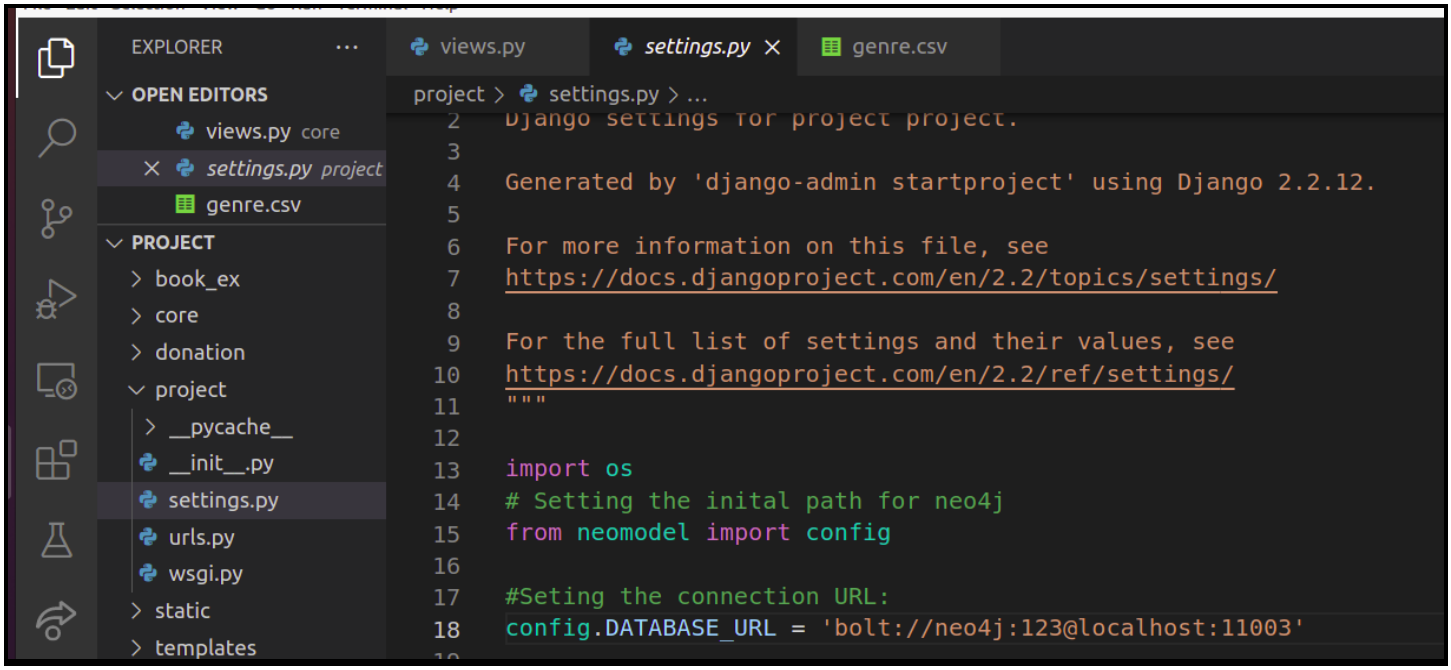
->Press Apply

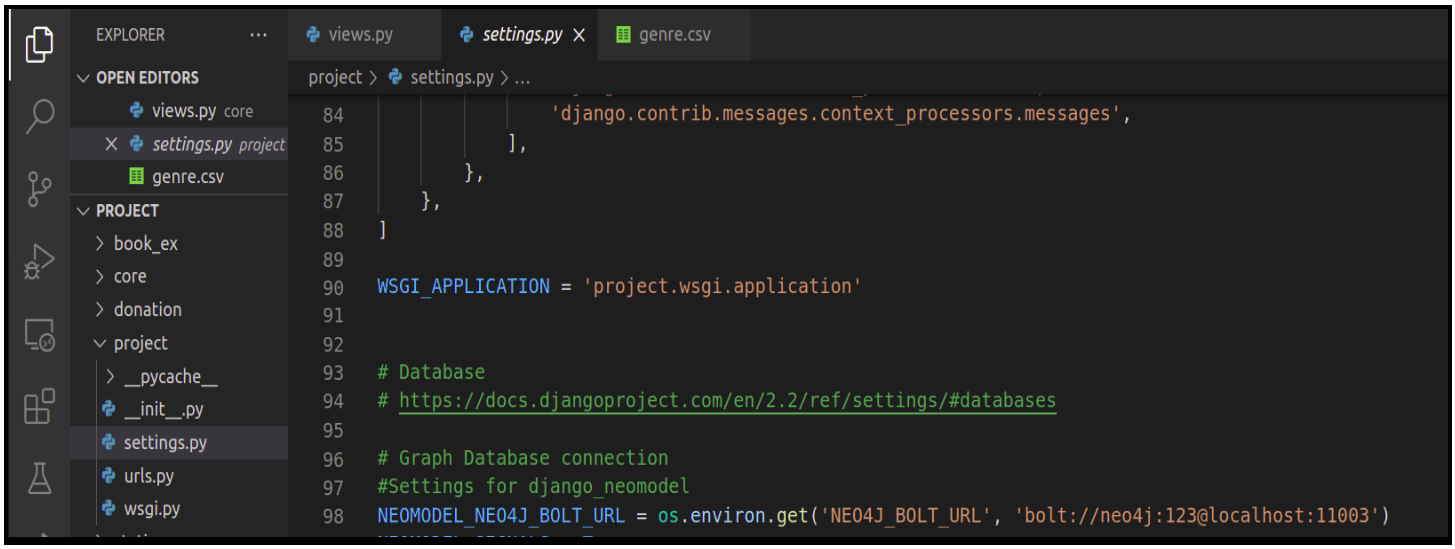
-> Click on Open to open the database in Neo4j browser



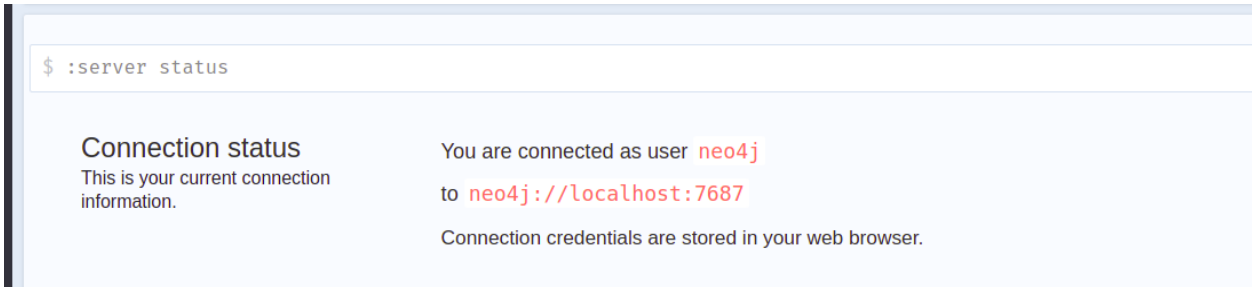
Connecting backend to graph data model :

1. Open folder Book-Recommendation-and-Sharing
2. Open folder project
3. Open file settings.py
4. On line 18 and on line 98 in settings.py





5. On line 18 and 98, change “11003” in link to connection credentials as shown in neo4j browser
   For example, using the below neo4j credentials, 11003 in settings.py should be changed to 7687



Setting up graph data model :

-> Download book.csv and genre.csv from Book-Recommendation-and-Sharing folder

-> Go to neo4j browser

-> Run command :

:use neo4j

-> Creating genre node from csv file, run query:

(replace <path to genre.csv file> with absolute path of genre.csv e.g. 'file:///home/bhumika/genre.csv')

```
LOAD CSV WITH HEADERS FROM 'file:///<path to genre.csv file>' AS row
WITH row
CREATE (b:Genre{genre_id:row.genre_id, name:row.name})
```

-> Creating index on genre node, run query :

```
CREATE INDEX genre_index FOR (g:Genre) ON (g.genre_id)
```

-> Creating book node from csv file, run query:

(replace <path to book.csv file> with absolute path of book.csv e.g. 'file:///home/bhumika/book.csv')

```
LOAD CSV WITH HEADERS FROM 'file:///<path to book.csv file>'
AS row
WITH row WHERE row.AUTHOR is NOT NULL
CREATE (b:Book{Title:row.TITLE, img_url:row.IMAGEURL})
```

-> Creating index on book node, run query :

```
CREATE INDEX book_index FOR (b:Book) ON (b.Title)
```

-> GENRE Relationship btw book and genre, run query :

(replace <path to book.csv file> with absolute path of book.csv e.g. 'file:///home/bhumika/book.csv')

```
LOAD CSV WITH HEADERS FROM 'file:///<path to book.csv file>'
AS row
WITH row WHERE row.AUTHOR is NOT NULL
MATCH (b:Book {Title:row.TITLE}), (g:Genre{genre_id:
row.CATEGORYID})
```

```
CREATE (g)-[:GENRE]->(b)
```

-> Creating author node from csv file, run query:

(replace <path to book.csv file> with absolute path of book.csv e.g. 'file:///home/bhumika/book.csv')

```
:auto USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'file:///<path to book.csv file>'
AS row
WITH row WHERE row.AUTHOR is NOT NULL
MERGE (a:Author{name:row.AUTHOR})
```

-> Creating index on author node, run query :

```
CREATE INDEX author_index FOR (a:Author) ON (a.name)
```

-> WROTE Relationship between author and book, run query :

(replace <path to book.csv file> with absolute path of book.csv e.g. 'file:///home/bhumika/book.csv')

```
:auto USING PERIODIC COMMIT
LOAD CSV WITH HEADERS FROM 'file:///<path to book.csv file>'
AS row
WITH row WHERE row.AUTHOR is NOT NULL
MATCH (a:Author{name:row.AUTHOR}), (b:Book{Title:row.TITLE})
CREATE (a)-[:WROTE]->(b)
```

-> Creating index on UserProfileInfo node, run query :

```
CREATE INDEX user_index FOR (u:UserProfileInfo) ON (u.username)
```

Spatial database

Connecting backend to postgres data model :

1. Installing PostgreSQL in Ubuntu 20.04

Open terminal and run the following commands:

$ sudo apt update

$ sudo apt install postgresql postgresql-contrib

Steps To Change Postgres User Password :

1. Login into the psql:
   $ sudo -u postgres psql
2. Then in the psql console change the password to 123 and quit:
   postgres=# \password postgres
   Enter new password: 123
   postgres=# \q

   Close terminal

2. Installing pgAdmin4 in Ubuntu 20.04

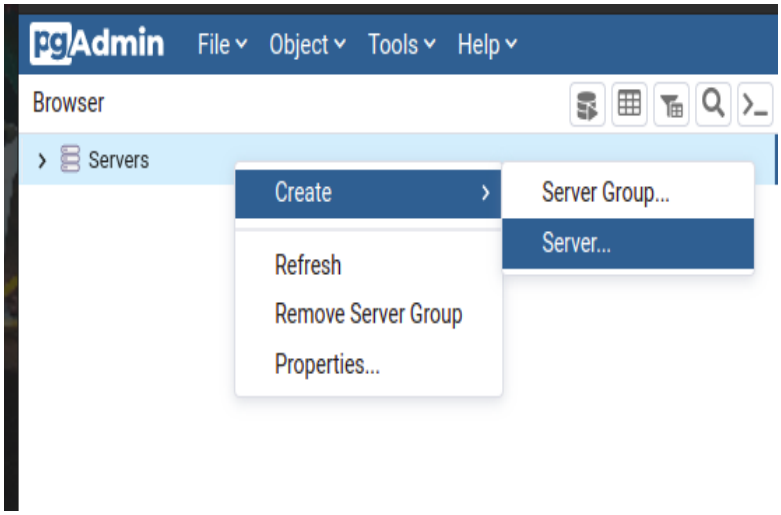
Open terminal and run the following commands:

1. sudo sh -c 'echo "deb http://apt.postgresql.org/pub/repos/apt/ $(lsb_release -cs)-pgdg main" >> /etc/apt/sources.list.d/pgdg.list'
2. sudo apt install wget ca-certificates
3. wget --quiet -O - https://www.postgresql.org/media/keys/ACCC4CF8.asc | sudo apt-key add
4. sudo apt update
5. sudo apt install pgadmin4

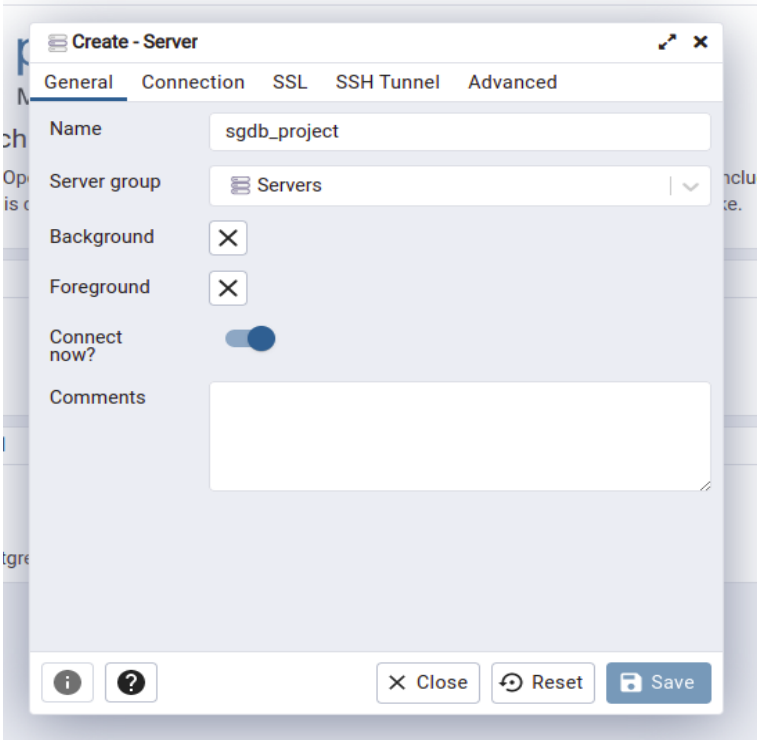
3. Creating Spatial Database

- Open pgAdmin4
- Set password as : 123

- Go to Server->Create->Server



-> name : sgdb_project



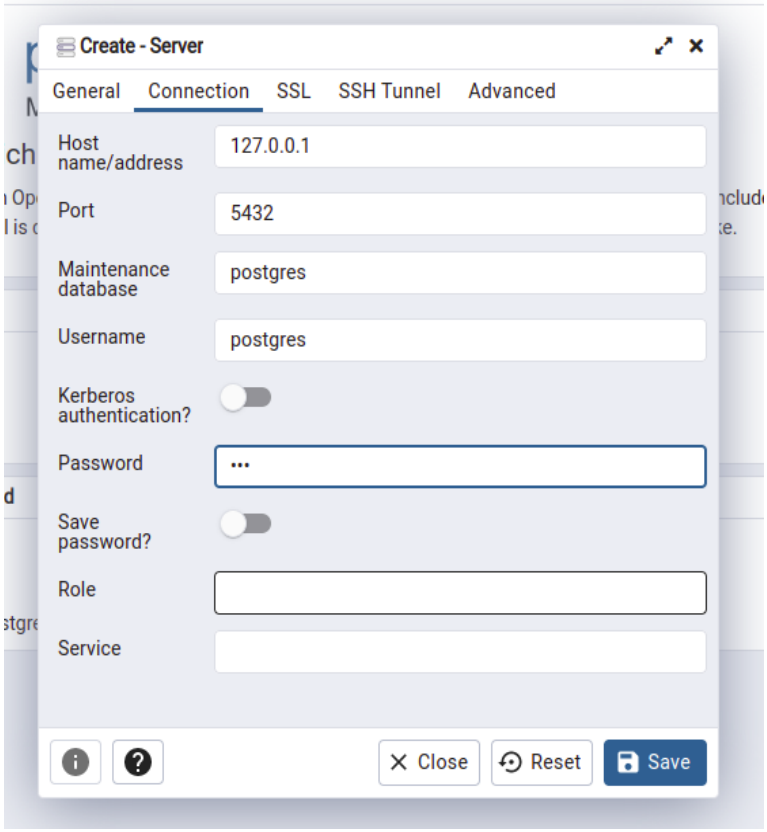
-> Host address : 127.0.0.1

Port : 5432

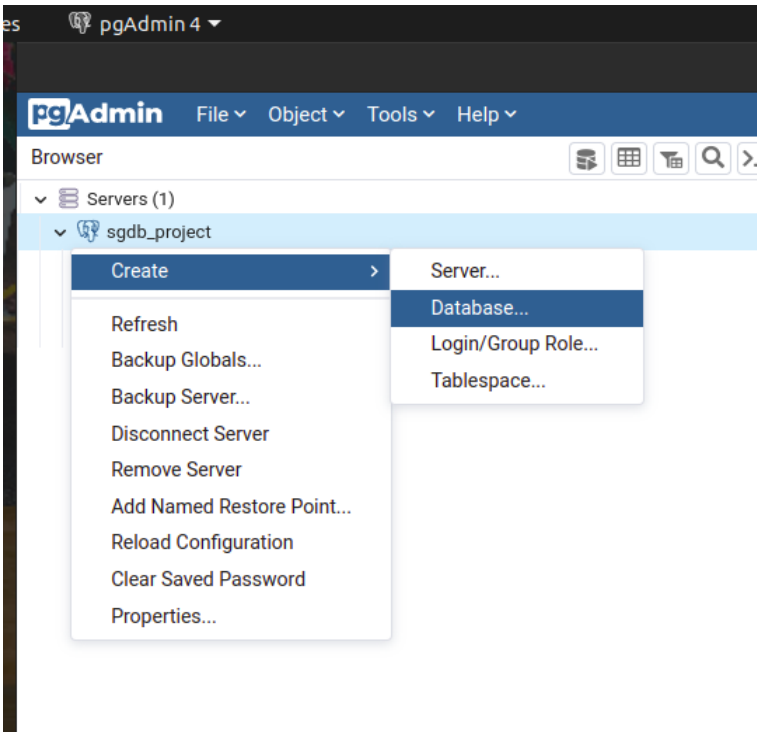
Username : postgres

Password : 123

Click **Save** after adding the details

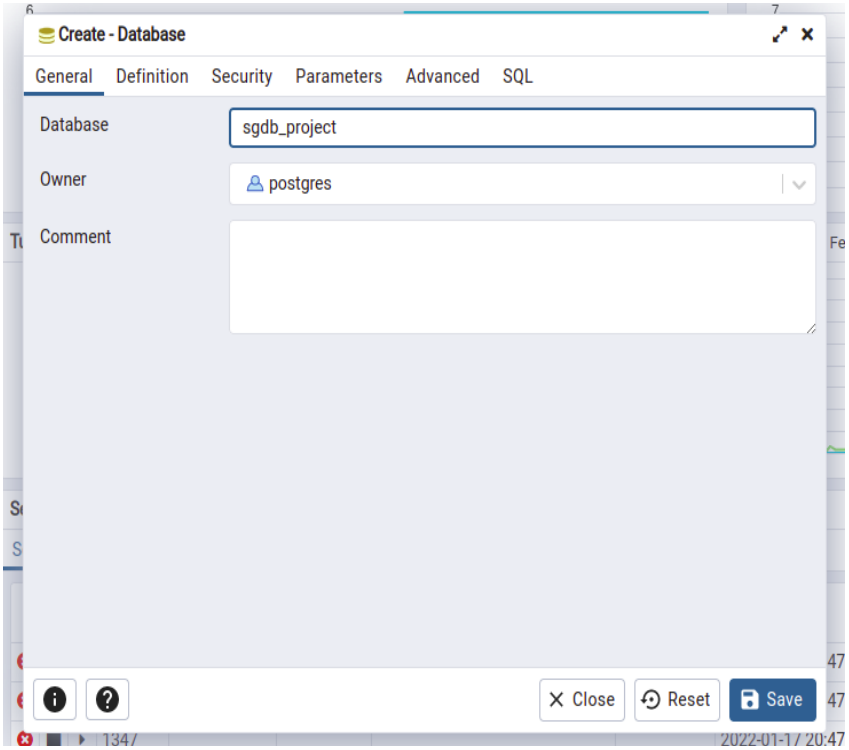

Create - Server
General
Connection
SSL
SSH Tunnel
Advanced
Host name/address
127.0.0.1
Port
5432
Maintenance database
postgres
Username
postgres
Kerberos authentication?
Password
Save password?
Role
Service
Close
Reset
Save

- Go to sgdb_project->right click on->create->database



-> Database : sgdb_project

Click **Save** after adding the details



---

3. Migrating postgres tables :

- Open Book-Recommendation-and-Sharing folder
- Open terminal
- Run the following commands :

```
$ python manage.py makemigrations
$ python manage.py migrate
```

---

4. Running the project :

- Open Book-Recommendation-and-Sharing folder
- Open terminal
- Run the following commands :

```
$ python manage.py runserver
```

- Open link in browser : http://127.0.0.1:8000/ in google chrome

---