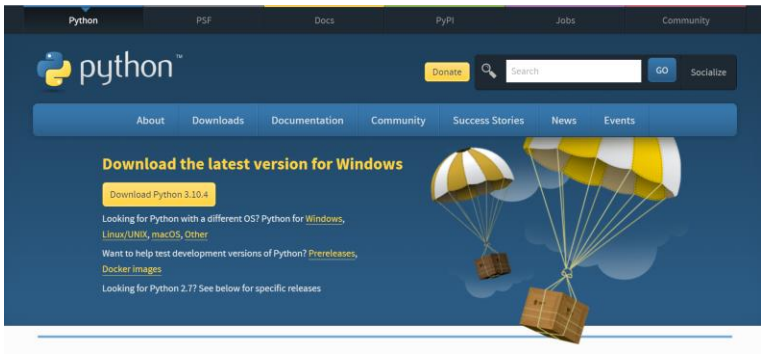
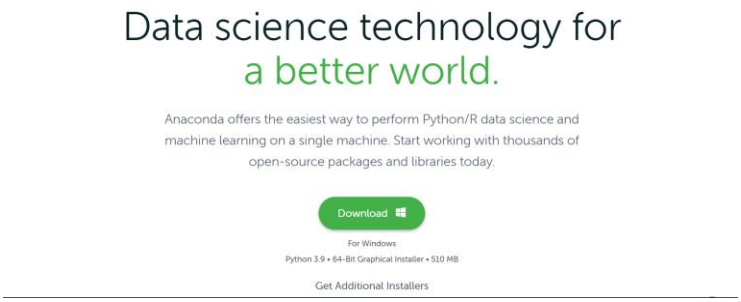
Group 8: Project Application Manual

A. Installations: Python, SSMS and Neo4J:

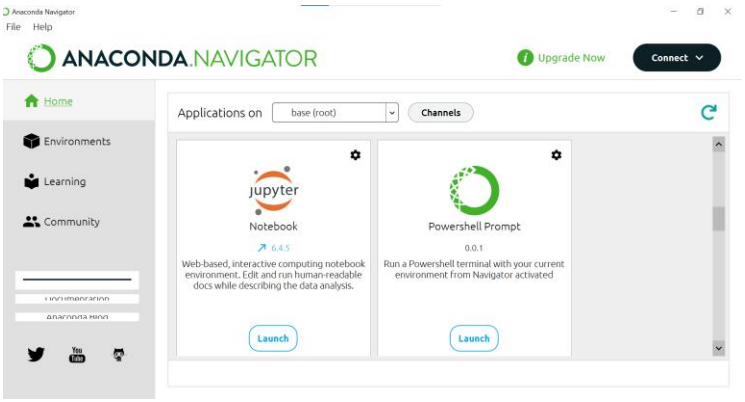
1. Download Zip of the project 10 and extract all the folders
2. Download and install python: Download Python | Python.org



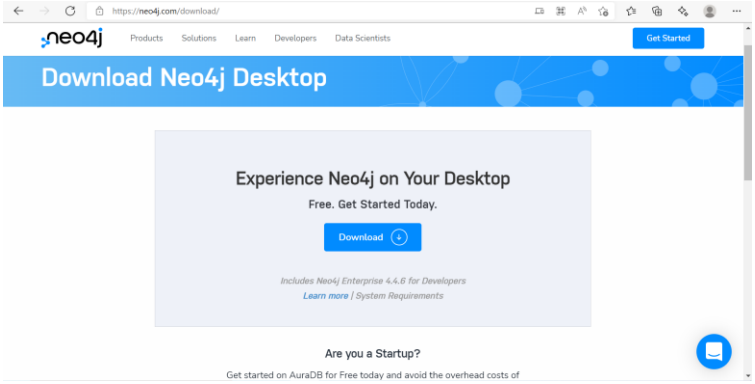
3. Download Anaconda Navigator: Anaconda | Anaconda Distribution



4. Select Jupyter Notebook to run the python code



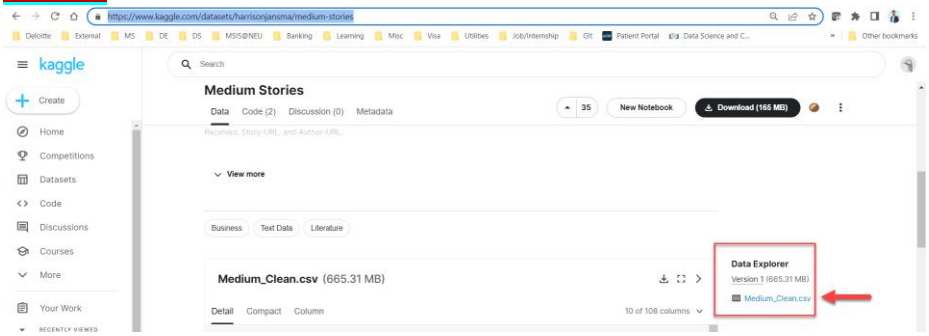
5. Install Neo4J: Neo4j Desktop Download | Free Graph Database Download



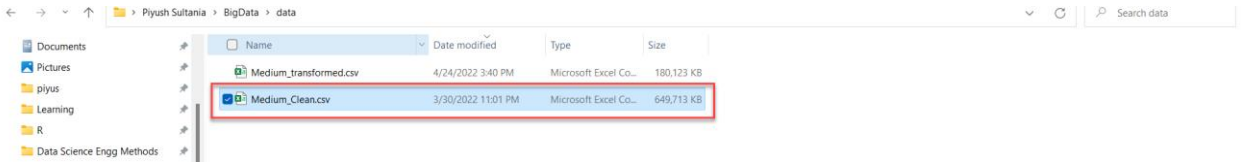
6. Download VSCode - Download Visual Studio Code - Mac, Linux, Windows
7. Download and Install SQL Server Development Edition - SQL Server Downloads
8. Download and Install SQL Server Management Studio (SSMS) - Free Download for SQL Server Management Studio (SSMS) 18.11.1

B. Data Profiling and Data Wrangling:

1) Download the source file from the "Kaggle" source path provided in the requirement document. FYR provided below:
https://www.kaggle.com/datasets/harrisonjansma/medium-stories
Note: I have also provided the source file as part of the Group08_MediumStories_ProjectDeliverables.zip file uploaded on canvas.

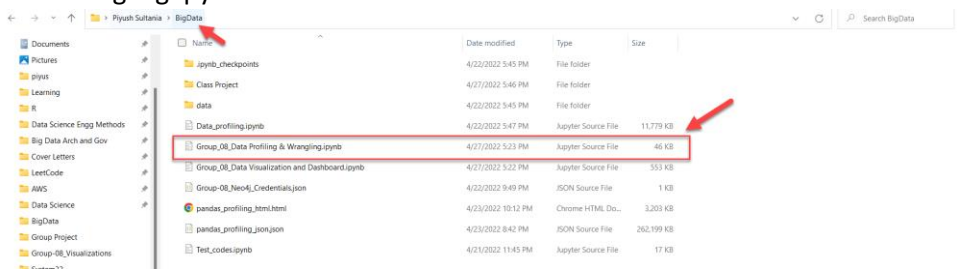


2) Create a folder under C:\Users\<user_name>\BigData\data and place the source file "**Medium_Clean.csv"** in this folder as below screenshot:

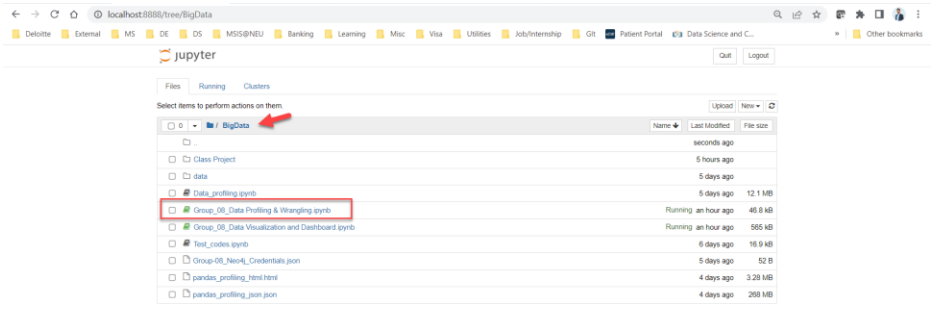


3) Place the Jupyter notebook "Group_08_Data Profiling & Wrangling.ipynb" given as part of the submission at the location mentioned and depicted below: Make sure you follow the directory hierarchy correctly to start the data pull to Python

   Location of notebook: "C:\Users\<user_name>\BigData\Group_08_Data Profiling & Wrangling.ipynb"

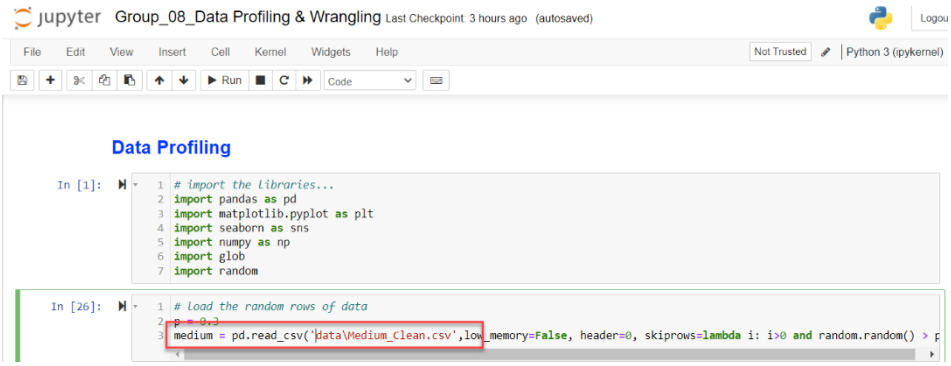


4) Launch Jupyter notebook from Start menu and open the notebook "Group_08_Data Profiling & Wrangling.ipynb" by navigating to path as "/BigData"

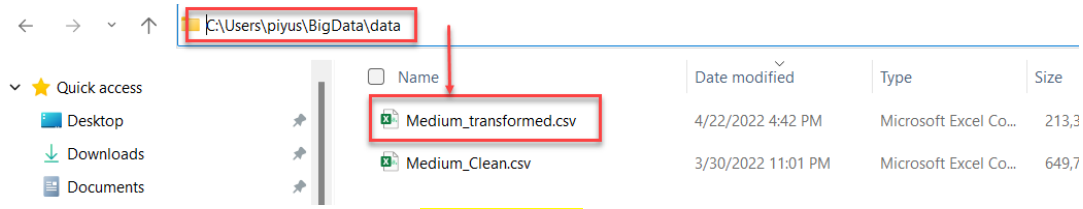


5) First let's install all the required libraries and dependencies (packages) using below commands in the notebook opened earlier
    i. pip install pandas
    ii. pip install numpy
    iii. pip install matplotlib
    iv. pip install seaborn
    v. pip install ipywidgets
    vi. pip install pandas-profiling
6) Once the data profiling and cleansing notebook in open insert the path of source file in the code that is provided in the project zip and run the code for data cleansing and data profiling (It acts as current working directory in Jupyter notebook)
Note: If you are using some other source file path, then use that location where you place your Medium_Clean.csv" file by mentioning the correct path

Caution! : As mentioned, our data set uses Melting/Pivot transformation so it is recommended not to use high volume of data to load in read_csv as it may give "Memory Overflow" error during melt command execution in the process!



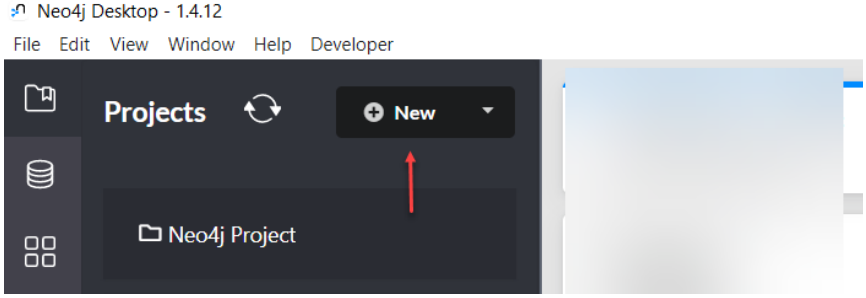
7) Once all the cells are executed in the python notebook, the transformed file will get saved at the location "TargetFile_Destination" mentioned below with the name "Medium_transformed.csv". Once verify the downloaded file at the location in your C:\ drive by navigating as below



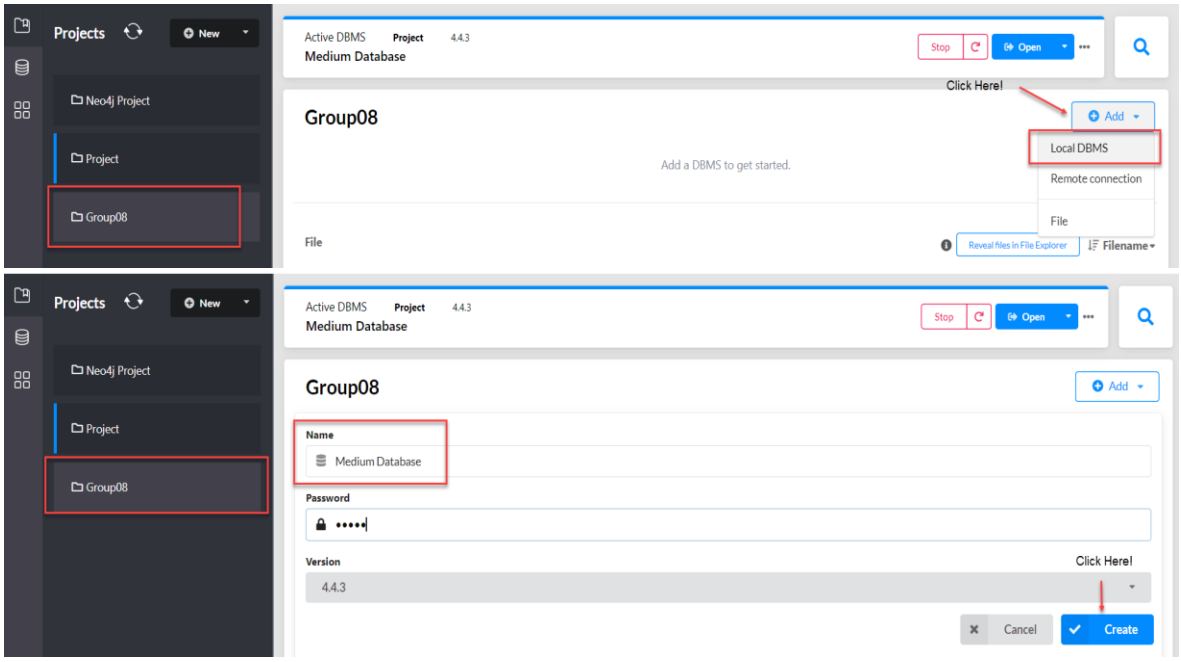
Sourcefile_Destination - "C:\Users\<user_name>\BigData\data\Medium_Clean.csv"
TargetFile_Destination - "C:\Users\<user_name>\BigData\data\Medium_transformed.csv"

C. Configuration of the source Neo4J database and project:

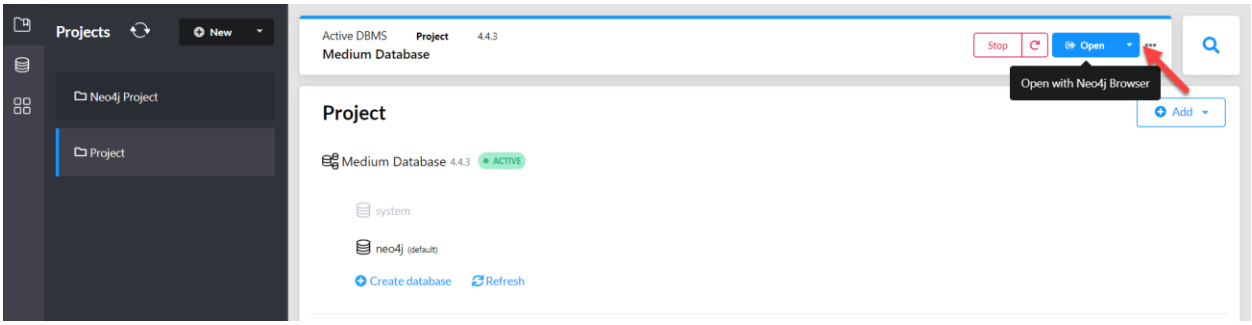
1. Open Neo4j Desktop from Start menu.
2. Create a new project by clicking on +New as shown below and give Project Name as "**Group08**" (or anything as wish), here I have created the project as name "Project"



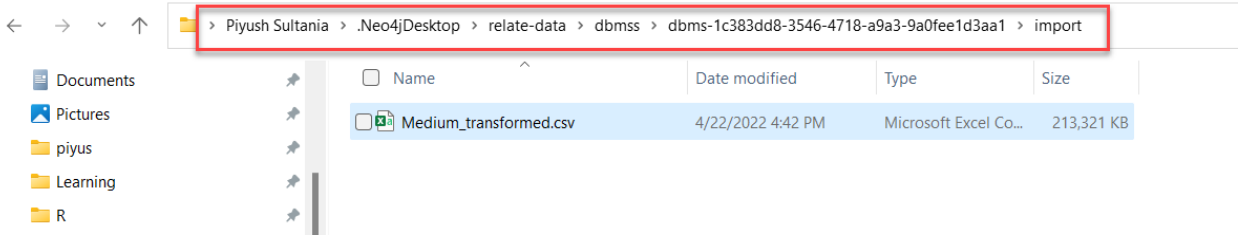
3. Click on +Add button to create a local DBMS named "**Medium Database**" and set password and click on "Create"



4. Once you set the Project and Local Database, Start the "**Medium Database**" and open the database by clicking on "Open" as shown in image below.



5. Go to C drive > Users > Select your User > Select .Neo4jDesktop > Select relate-data > select dbmss > select the recently created database(remember to select the correct database, as this database does not have its name on file) > select import > Paste the "**Medium_transformed.csv**" excel sheet in this folder (refer below screenshot)
Pick the file from :: "C:\Users\<user_name>\BigData\data\Medium_transformed.csv"



6. Open the settings of Group2_Project, by selecting three dots on the extreme right hand side and paste the command at the end of the settings - dbms.security.procedures.unrestricted=apoc.*

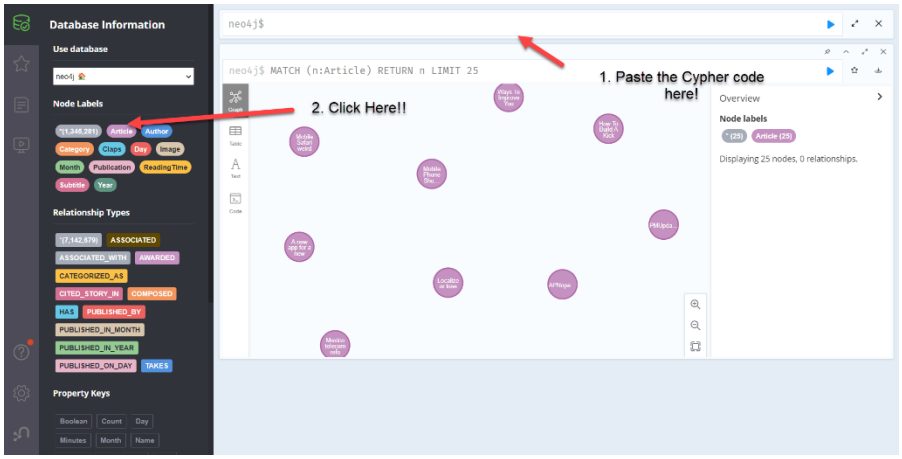
```
#********************************************************************
# Other Neo4j system properties
#********************************************************************

dbms.security.procedures.unrestricted=apoc.*
```

7. Copy the file apoc-4.4.0.3-all and navigate to -
   C drive > Users > Select your User > Select .Neo4jDesktop > Select relate-data > select dbmss > select the recently created project > plugins and paste the apoc-4.4.0.3-all folder here.

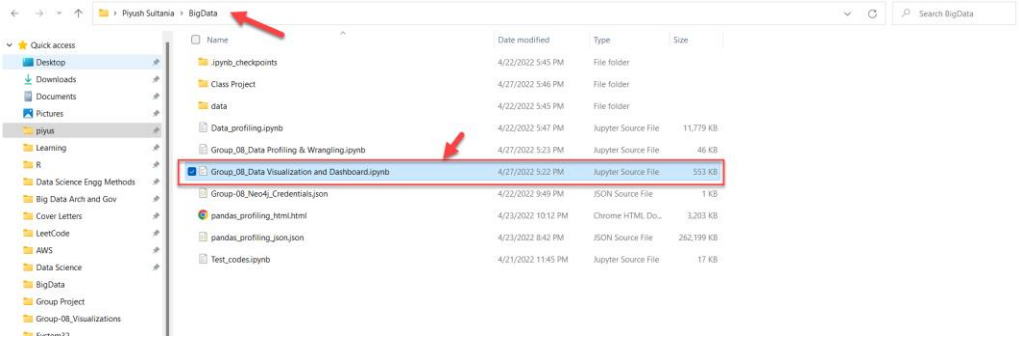
D. Loading Data into Neo4j

1. Start the "**Medium Database**" (that we setup in previous step) by pressing start (if not already)
2. Open "**Group-08_Neo4j_DataIngestionScript.txt**" file provided in the submission .zip and copy the code
3. Once, the database is active, open the database, paste the query in front of the $ sign and execute it. Wait for all data to be loaded into Neo4j (execution time can vary based on volume of data)
4. Once successful completion, verify the database creation and load by clicking on any node as shown below:

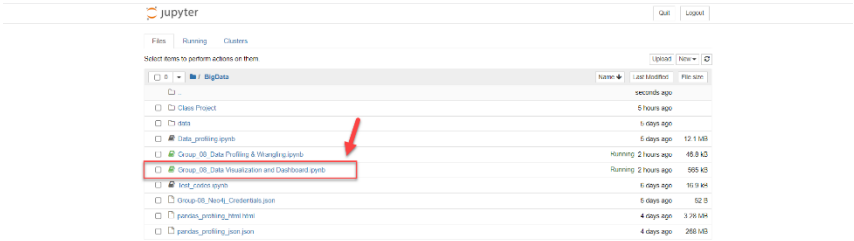


E. Data Visualization and Dashboard:

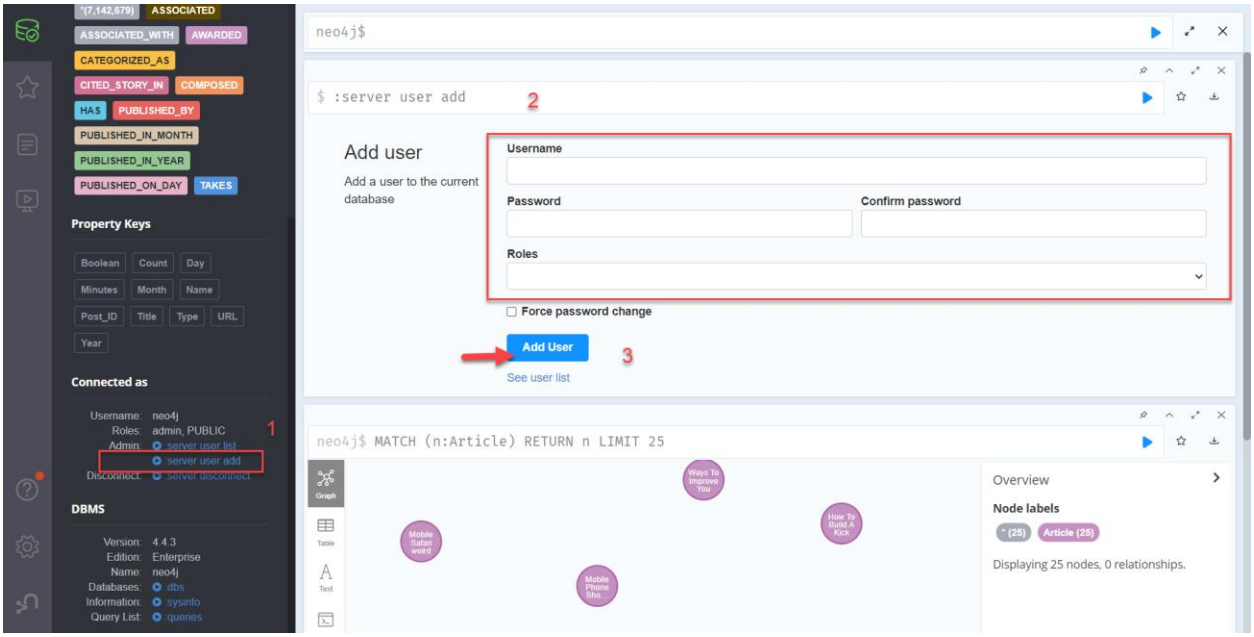
1. Place the visualization notebook named "**Group_08_Data Visualization and Dashboard.ipynb**" at the location "C:\Users\<user_name>\BigData\" (similar to data cleaning notebook directory)



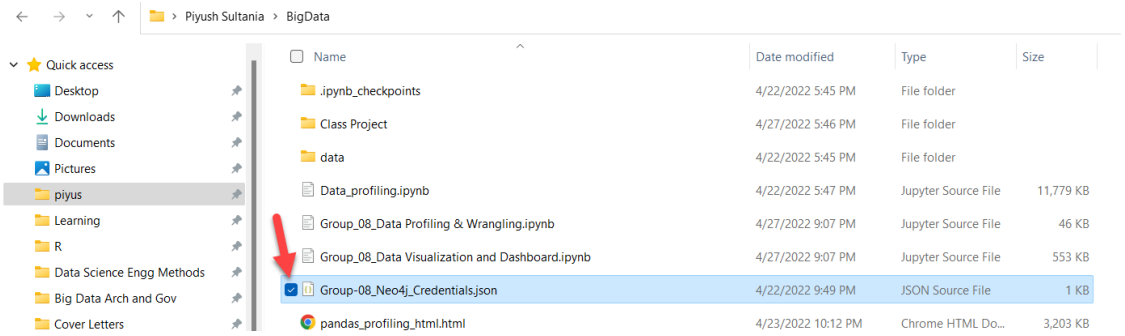
2. Launch Jupyter notebook from Start menu and open the notebook "**Group_08_Data Visualization and Dashboard.ipynb**" by navigating to path as /BigData



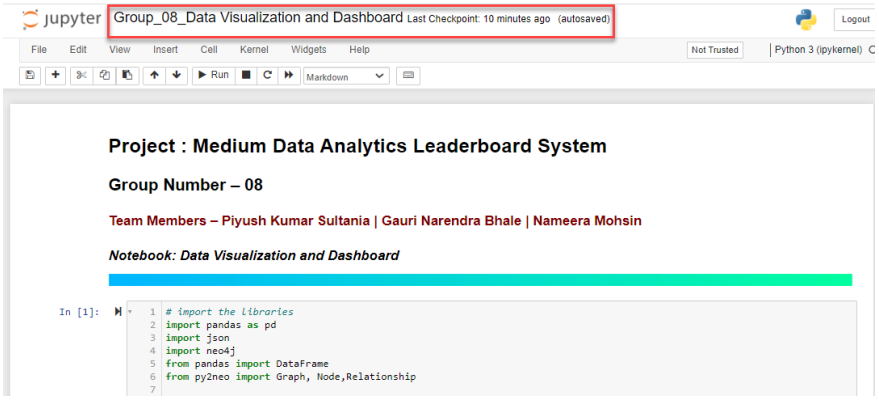
3. **Install the neo4j libraries to connect python with neo4j**
    a. pip install py2neo
    b. pip install neo4j
4. Open Neo4j desktop and Click on Database icon on extreme left corner and scrolled to the bottom on the left panel. Now click on "**:servers user add**" as depicted in image (action-1).
5. Now, in Add user panel in the right, specify the username as "admin" and password as "admin" (action-2) and finally click on "Add User" (action-3). Please refer to below screenshot.



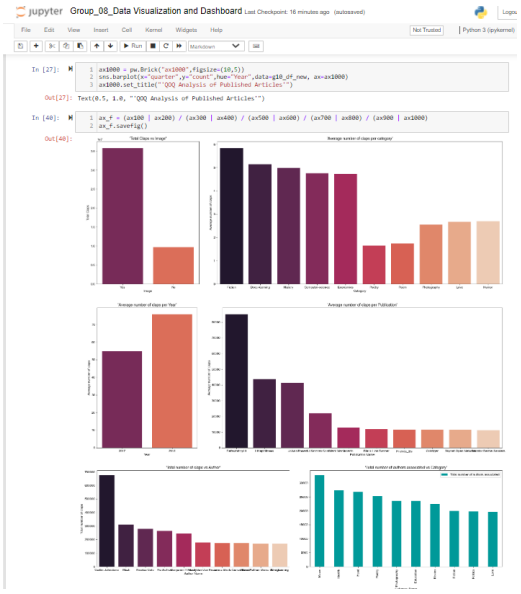
6. Now, place the ne04j user credentials json file named as "Group-08_Neo4j_Credentials.json" at the below location specified and depicted in the image.
   Location for Credentials file: C:\Users\<user_name>\BigData\

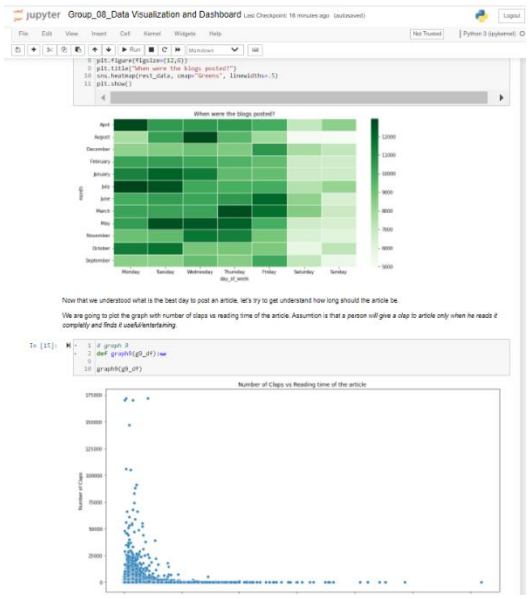


7. Now, run the notebook "Group_08_Data Visualization and Dashboard.ipynb" and it will show all the visualizations and dashboard. In case of any technical issues, restart the Kernel and rerun the notebook. *(Make sure your neo4j database is up and running, if not follow step D.1)*



8. Once, all the notebook is executed successfully, verify the graphs and dashboard by scrolling down the notebook at the bottom of the notebook.

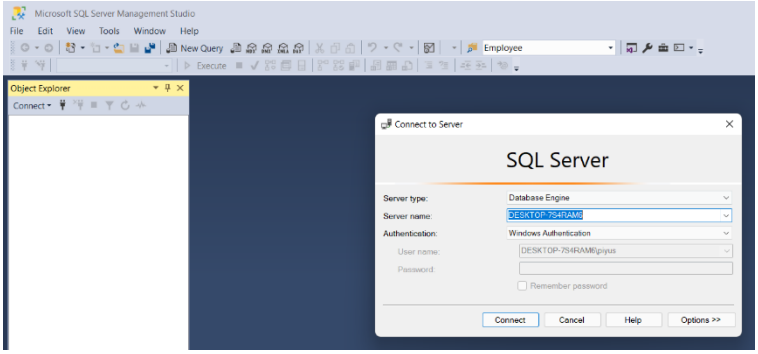


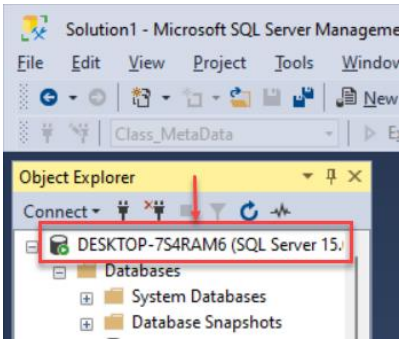


Bingo! We are able to see the insights of our data in Python using visualization libraries.

G. Technical and Business Metadata Integration:

1. Before this activity, please make sure all the above steps are executed successfully and you are connected to SQL Server instance and Neo4j localhost connection.
2. If not connected to SQL server instance, click on Start menu and open SSMS and connect to the sql server as below screenshot. Once connected you will see the instance as DESKTOP-*





3. Run the SQL script to create the tables
   a. Open the script (SSMS_Script.sql) in VSCode for better readability
   b. Copy the code into SSMS
   c. Select and Execute the first line of code :- DROP DATABASE (if same database exists) else CREATE DATABASE [Class_MetaData]
   d. If database is created successfully, Select rest of the query and execute it, these query will create all the required tables

4. Now that data is loaded into the source, lets run ETL process to dump data into SSMS
   1) First let's install all the required libraries needed to run the ETL, open cmd and run the following commands

      a. pip install neo4j (*ignore if already installed earlier*)
      b. pip install pyodbc
      c. Pip install openpyxl
5. Open MetaDataETL.ipynb in VSCode
6. Change the paths according to your system for the following files
   a. Give path of DataBases.xlsx to the databasePath
   b. Give path of AttributeDataTypes.xlsx to the attributeDatatypePath
   c. Assign the business term path of BusinessTermListG8 in the cell where Group 8 data is being loaded
7. Run the ETL
   a. As the neo4j database is running (Group08 – if you did not run any database in between this integration), execute the cell for this database connection mentioned in notebook
   b. The data has been successfully loaded into SSMS
8. Open Insert-Bridge-Table.ipynb in VSCode and execute. This code will insert all the values in the Bridge Table.

H. User-Interface via Streamlit:

1. Open User-Interface-SSMS.py in VSCode
   a. Download dependencies using
      a) Pip install enum
      b) Pip install streamlit

2. Execute the file.
3. Open CMD
4. Change the directory to the folder where all the project files are kept
5. Run the command - streamlit run User-Interface-SSMS.py
6. If using for first time it will ask the email and other fields leave them blank
7. The UI will open in a web page.
8. You can see the vital information of all the databases.
   a. Select Database to view the nodes and relationships
   b. Select a node to view its attribute and the business term description
   c. Select attribute to view its information - type, range etc.

In case of any issues, please reach out to below POCs:

Piyush Kumar Sultania – sultania.p@northeastern.edu
Gauri Narendra Bhale - bhale.g@northeastern.edu
Nameera Mohsin - mohsin.n@northeastern.edu