ADS Assignment 11

PRN: 2019BTECS00083

Name: Bhargav Kulkarni

Batch: T2

Title: Installation and configuration of Neo4J Graph Database.

Aim:

Consider the “Research Papers Database” scenario as follows :
The research papers have authors (often more than one). Most papers have a classification (what the paper is about). The classifications form a hierarchy in

several levels (for example, the classification “Databases” has the sub-classifications “Relational” and “Object-Oriented”). A paper usually has a list

of references, which are other papers. These are called citations.

1. Design/model the graph database using Neo4j for above scenario.
2. Download the raw data from Cora Research Paper Classification Project : http://people.cs.umass.edu/~mccallum/data.html The database contains approximately 25,000 authors, 37,000 papers and 220,000 relationships.
3. Load this data using Neo4j Data Browser
4. Design the python based desktop application for any kind of search on above database. The application should able to answer queries like

a) Does paper A cite paper B? If not directly, does paper A cite a paper which in its turn cites paper B? And so on, in several levels.
b) Show the full classification of a paper (for example, Databases / Relational)

Theory:

Neo4j

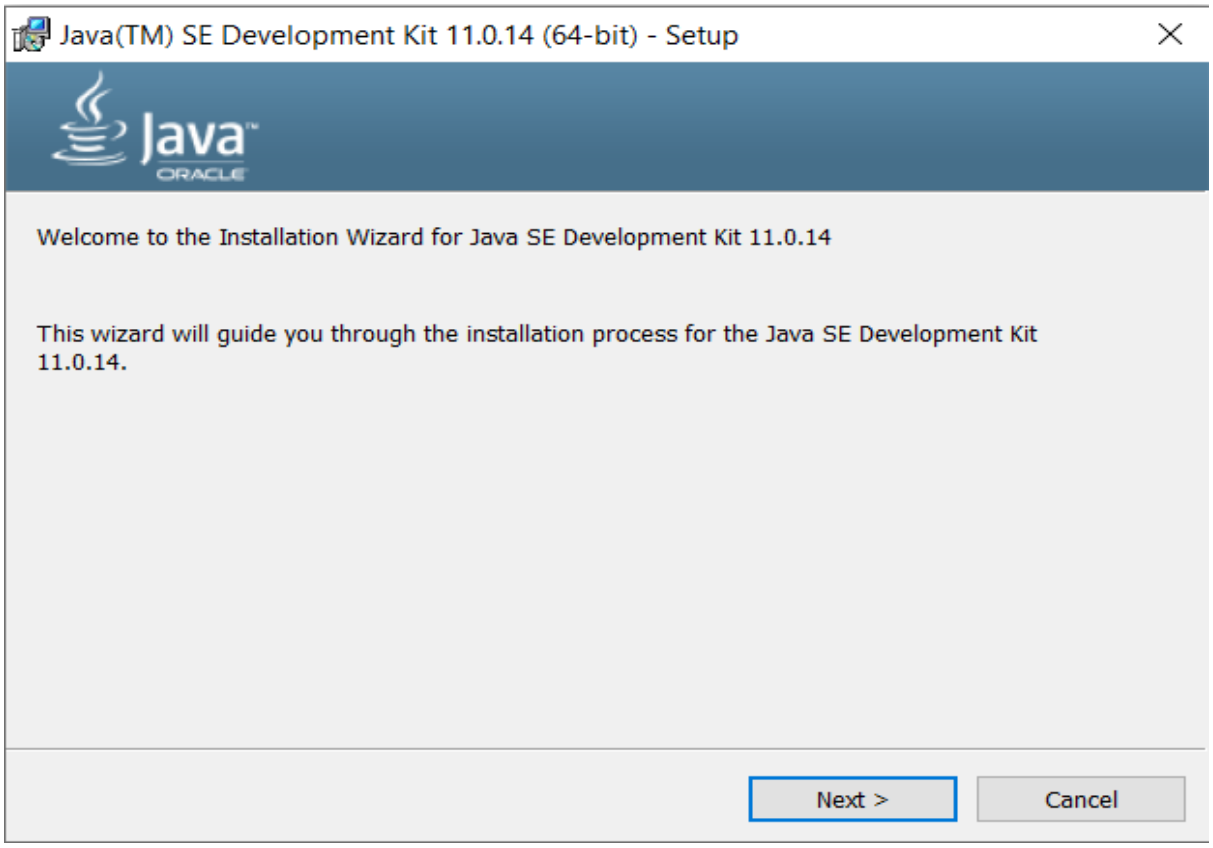
Neo4j is the world's leading graph database. The architecture is designed for optimal management, storage, and traversal of nodes and relationships. The graph database takes a property graph approach, which is beneficial for both traversal performance and operations runtime.

Cypher

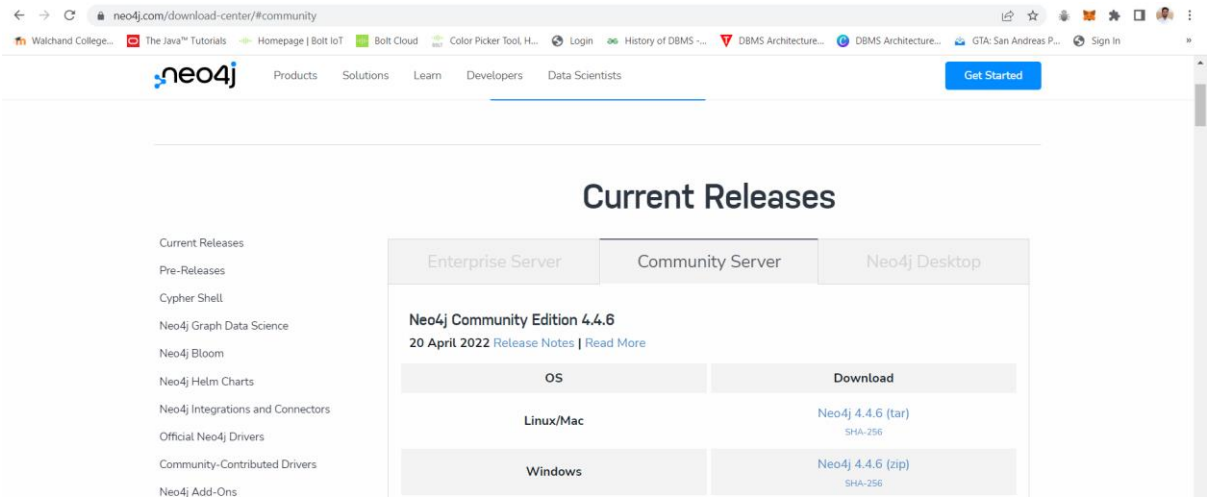
Cypher is Neo4j's graph query language that allows users to store and retrieve data from the graph database. It is a declarative, SQL-inspired language for describing visual patterns in graphs using ASCII-art syntax. The syntax provides a visual and logical way to match patterns of nodes and relationships in the graph. Cypher has been designed to be easy to learn, understand, and use for everyone, but also incorporate the power and functionality of other standard data access languages

Procedure:

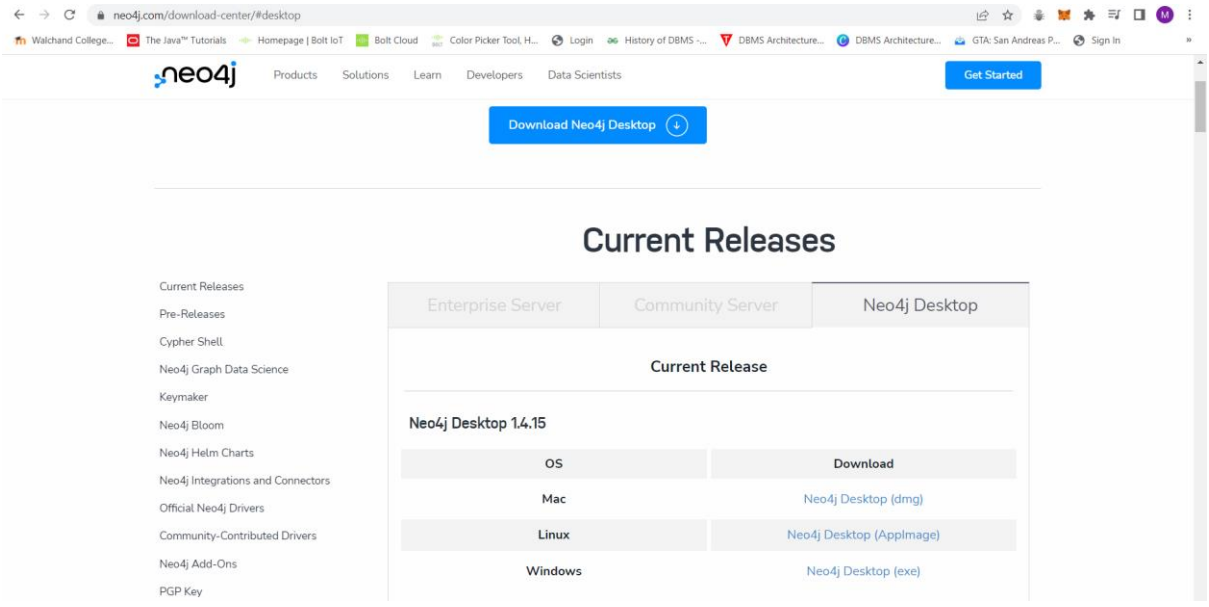
Downloaded JDK11 as per software requirements.

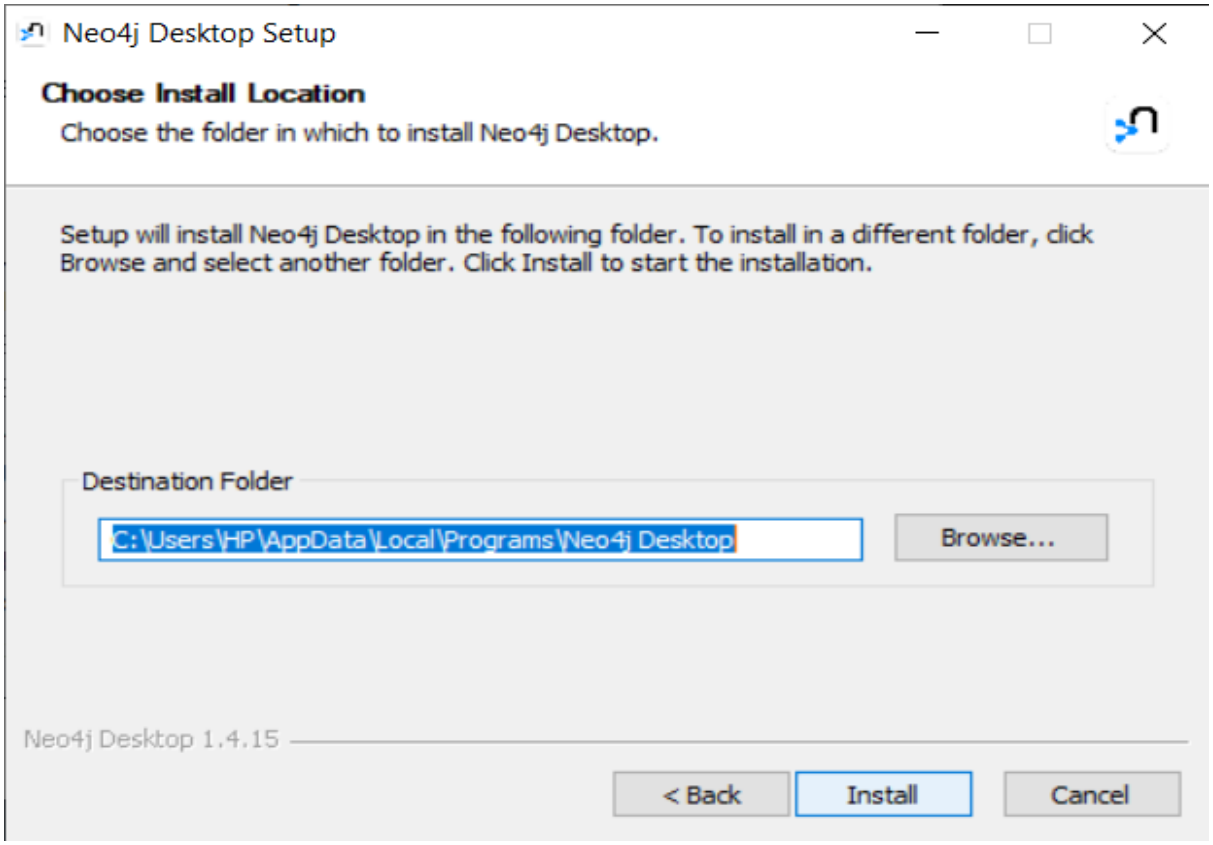


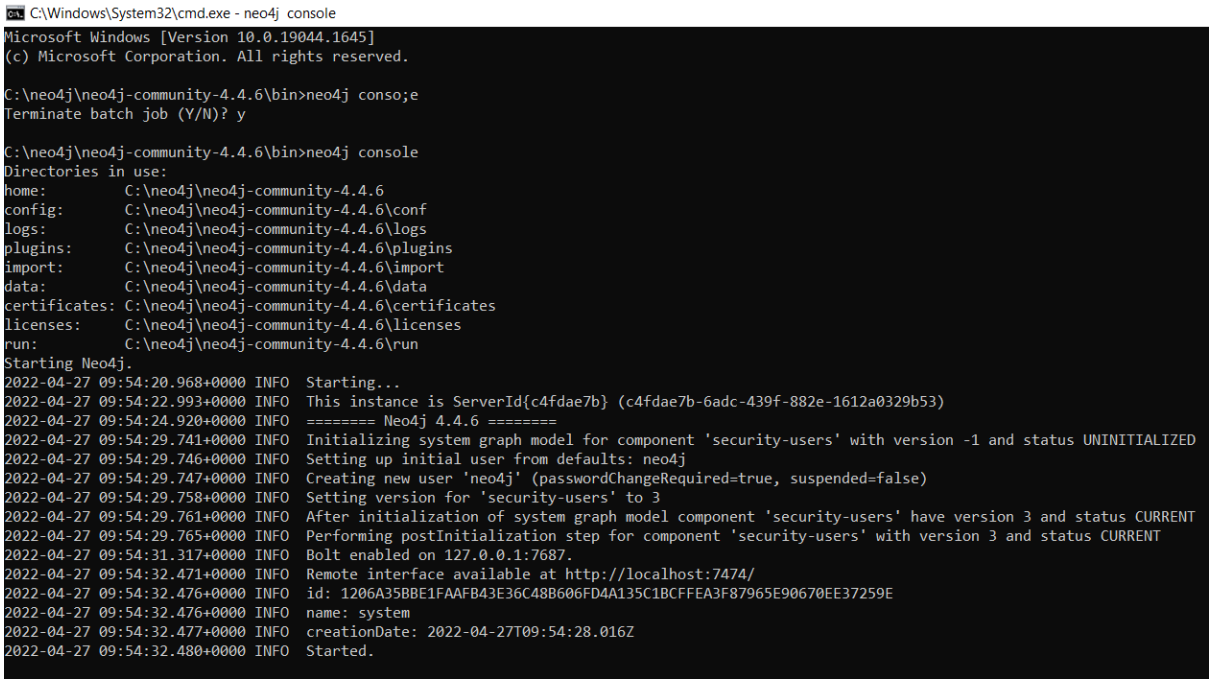
Downloaded neo4j community server edition

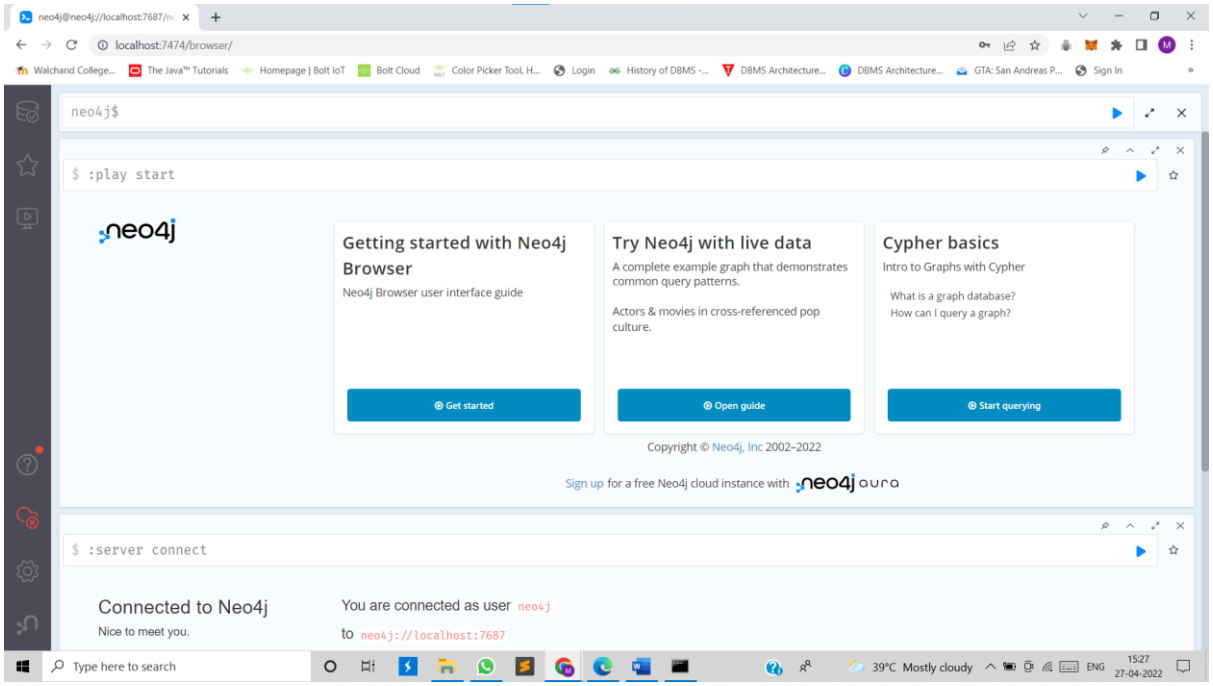


Downloaded neo4j Desktop




Neo4j Desktop Setup
Choose Install Location
Choose the folder in which to install Neo4j Desktop.
Setup will install Neo4j Desktop in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.
Destination Folder
C:\Users\HP\AppData\Local\Programs\Neo4j Desktop
Browse...
Neo4j Desktop 1.4.15
< Back
Install
Cancel


C:\Windows\System32\cmd.exe - neo4j console
Microsoft Windows [Version 10.0.19044.1645]
(c) Microsoft Corporation. All rights reserved.
C:\neo4j\neo4j-community-4.4.6\bin>neo4j conso;e
Terminate batch job (Y/N)? y
C:\neo4j\neo4j-community-4.4.6\bin>neo4j console
Directories in use:
home: C:\neo4j\neo4j-community-4.4.6
config: C:\neo4j\neo4j-community-4.4.6\conf
logs: C:\neo4j\neo4j-community-4.4.6\logs
plugins: C:\neo4j\neo4j-community-4.4.6\plugins
import: C:\neo4j\neo4j-community-4.4.6\import
data: C:\neo4j\neo4j-community-4.4.6\data
certificates: C:\neo4j\neo4j-community-4.4.6\certificates
licenses: C:\neo4j\neo4j-community-4.4.6\licenses
run: C:\neo4j\neo4j-community-4.4.6\run
Starting Neo4j.
2022-04-27 09:54:20.968+0000 INFO Starting...
2022-04-27 09:54:22.993+0000 INFO This instance is ServerId{c4fdae7b} (c4fdae7b-6adc-439f-882e-1612a0329b53)
2022-04-27 09:54:24.920+0000 INFO ======== Neo4j 4.4.6 ========
2022-04-27 09:54:29.741+0000 INFO Initializing system graph model for component 'security-users' with version -1 and status UNINITIALIZED
2022-04-27 09:54:29.746+0000 INFO Setting up initial user from defaults: neo4j
2022-04-27 09:54:29.747+0000 INFO Creating new user 'neo4j' (passwordChangeRequired=true, suspended=false)
2022-04-27 09:54:29.758+0000 INFO Setting version for 'security-users' to 3
2022-04-27 09:54:29.761+0000 INFO After initialization of system graph model component 'security-users' have version 3 and status CURRENT
2022-04-27 09:54:29.765+0000 INFO Performing postInitialization step for component 'security-users' with version 3 and status CURRENT
2022-04-27 09:54:31.317+0000 INFO Bolt enabled on 127.0.0.1:7687.
2022-04-27 09:54:32.471+0000 INFO Remote interface available at http://localhost:7474/
2022-04-27 09:54:32.476+0000 INFO id: 1206A35BBE1FAAFB43E36C48B606FD4A135C1BCFFEA3F87965E90670EE37259E
2022-04-27 09:54:32.476+0000 INFO name: system
2022-04-27 09:54:32.477+0000 INFO creationDate: 2022-04-27T09:54:28.016Z
2022-04-27 09:54:32.480+0000 INFO Started.

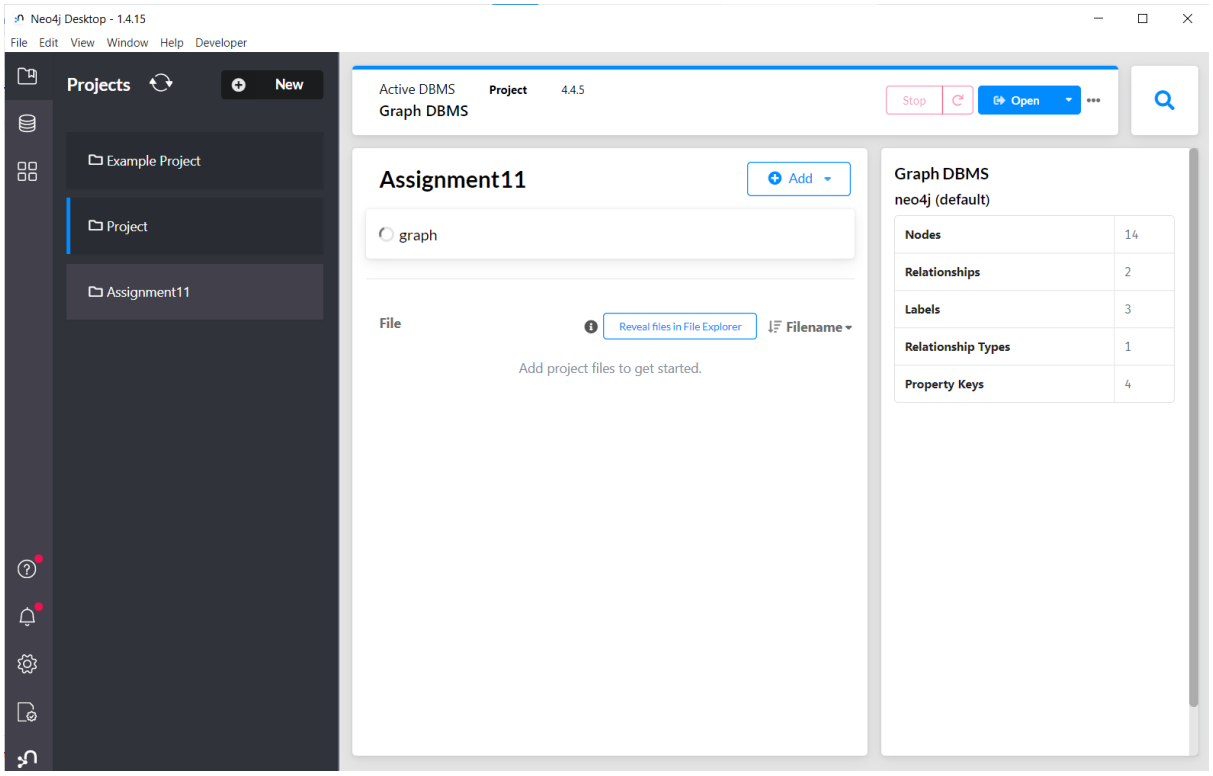


README - Notepad

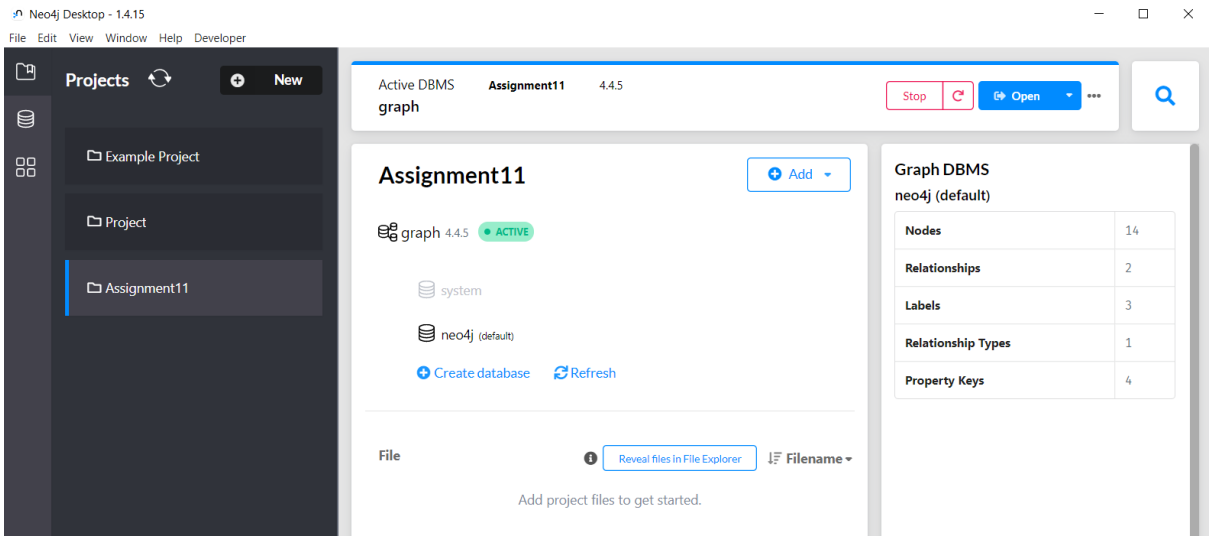
File Edit Format View Help

```
This directory contains the Cora dataset, version 1.0.  Feel free to
use the data for any research.  To cite this dataset please use the
following:

@article{McCallumIRJ,
 author = "Andrew McCallum and Kamal Nigam and Jason Rennie and Kristie Seymore",
 title = "Automating the Construction of Internet Portals with Machine Learning",
 journal = "Information Retrieval Journal",
 volume = 3,
 pages = "127--163",
 publisher = "Kluwer",
 year = 2000,
 note = "www.research.whizbang.com/data"
}


Note that in Cora there are two types of papers: those we found on the
Web, and those that are referenced in bibliography sections.  It is
possible that a paper we found on the Web is also referenced by other
papers.


FILE SUMMARY:

* The file `papers' contains limited information on the papers we found
on the Web.

* The file `citations' contains the citation.

* The file `classifications' contains class labels

* The directory `extractions' contains the extracted authors, title,
abstract, etc, plus the references (and in some cases surrounding
text). from the postscript papers we found on the Web.
```

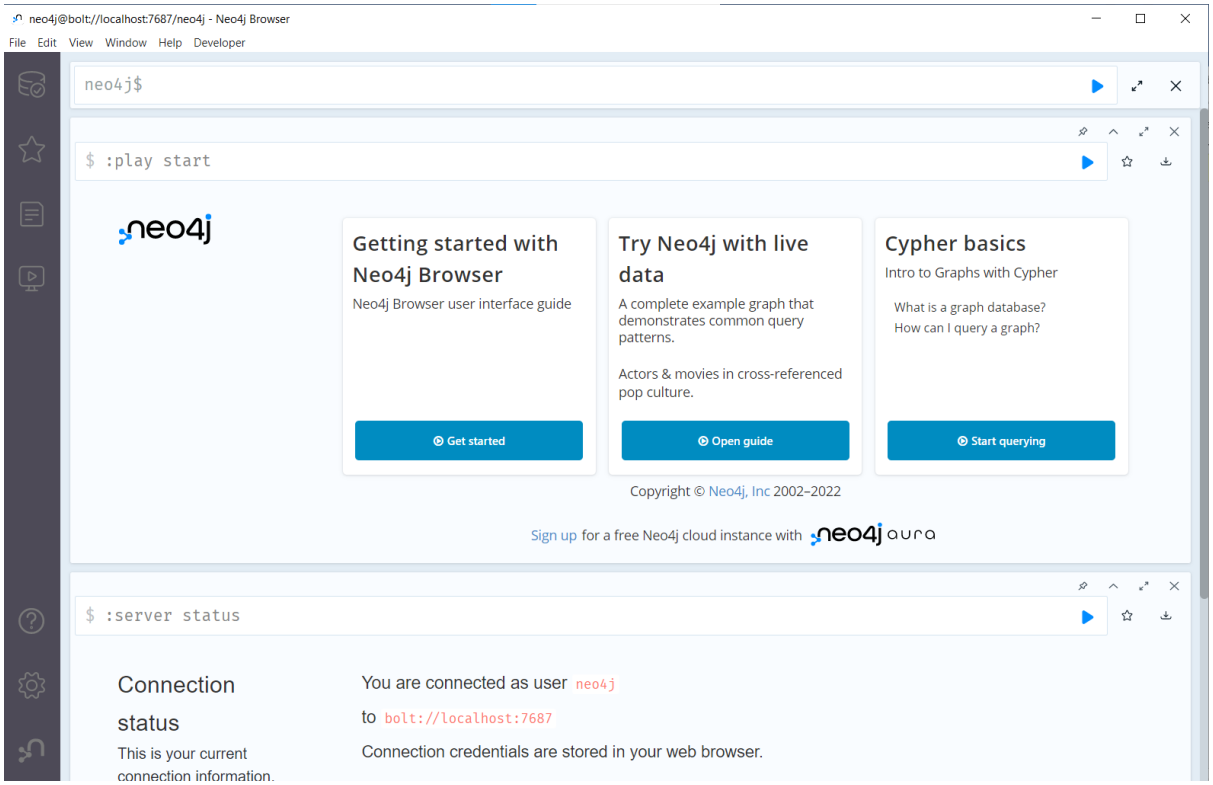
Creating a new project named Assignment11

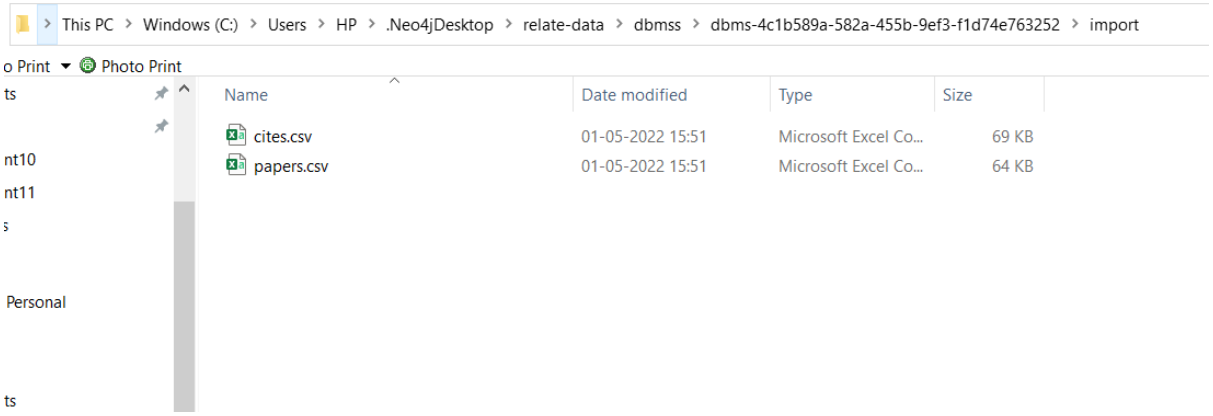


Started database graph

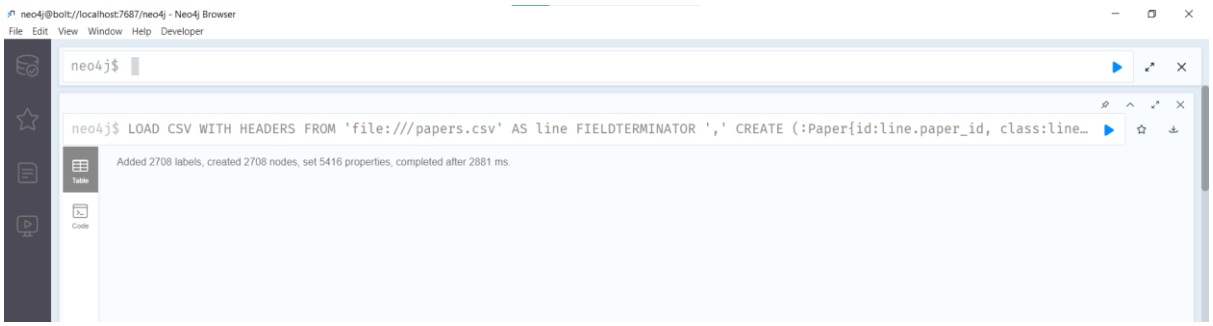


Opened using Neo4j browser



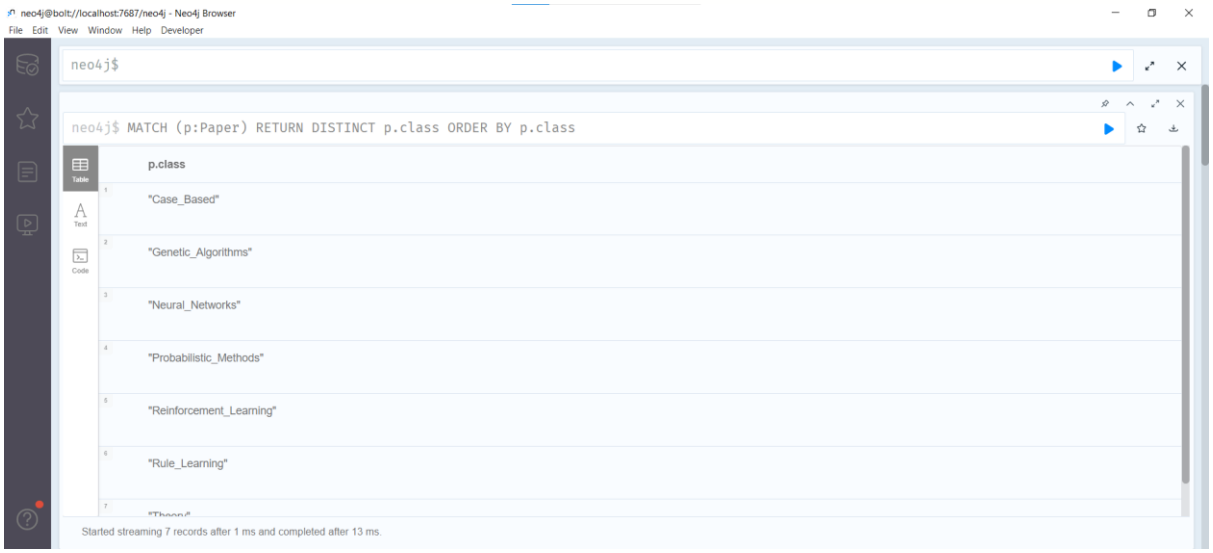



```
LOAD CSV WITH HEADERS FROM 'file:///papers.csv' AS row FIELDTERMINATOR ','

CREATE (:Paper {id: row.paper_id, class: row.label})
```

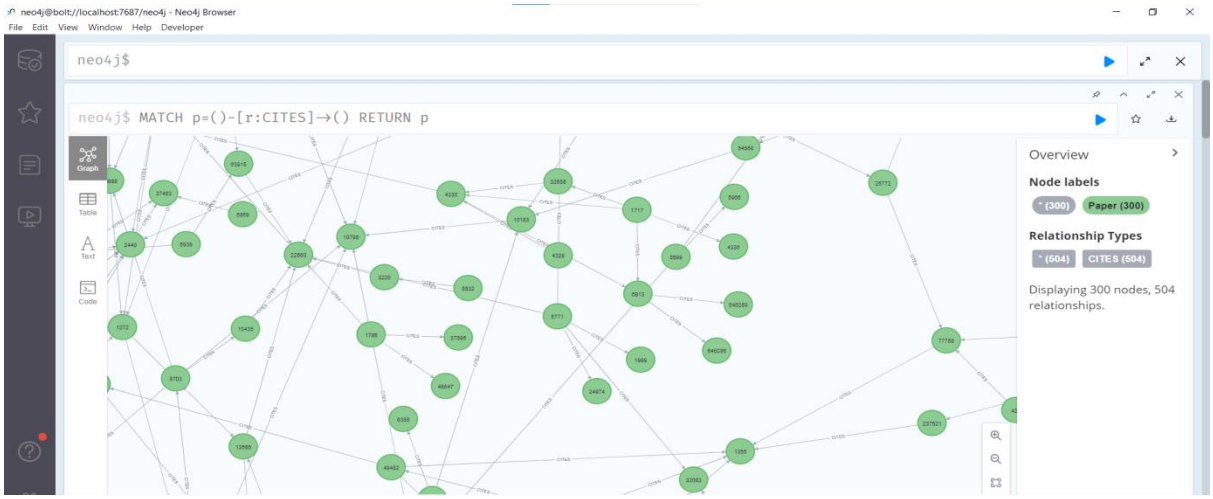



```
LOAD CSV WITH HEADERS FROM 'file:///cites.csv' AS line FIELDTERMINATOR ','
MATCH (citing_paper:Paper{id:line.citing_paper_id}), (cited_paper:Paper{id:line.cited_paper_id})
CREATE (citing_paper)-[:CITES]->(cited_paper)
```

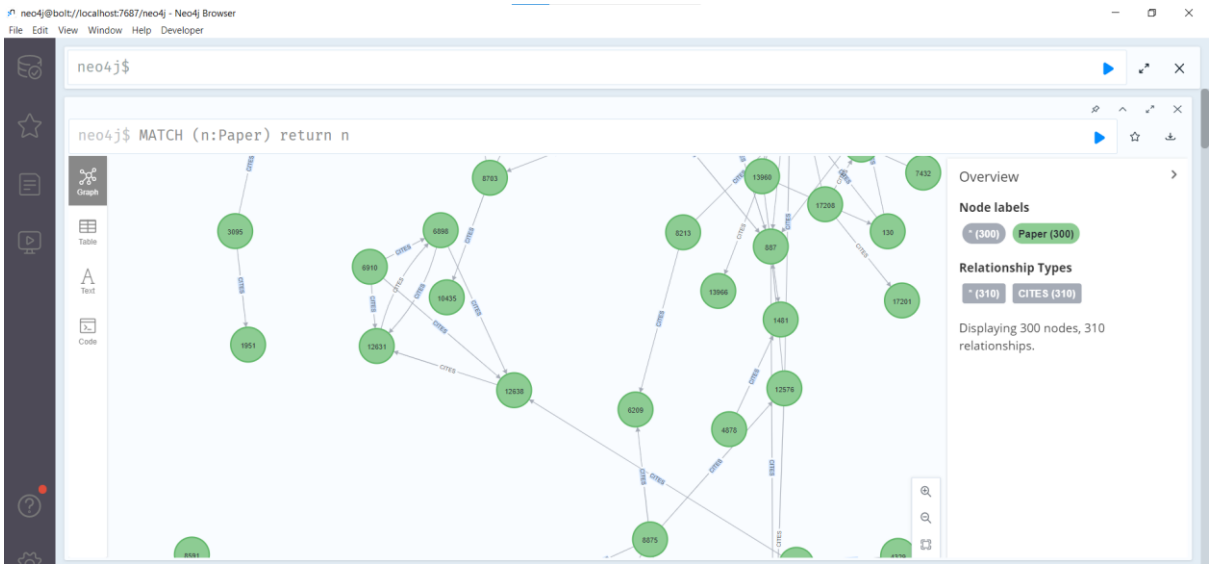
MATCH (p:Paper) RETURN DISTINCT p.class ORDER BY p.class



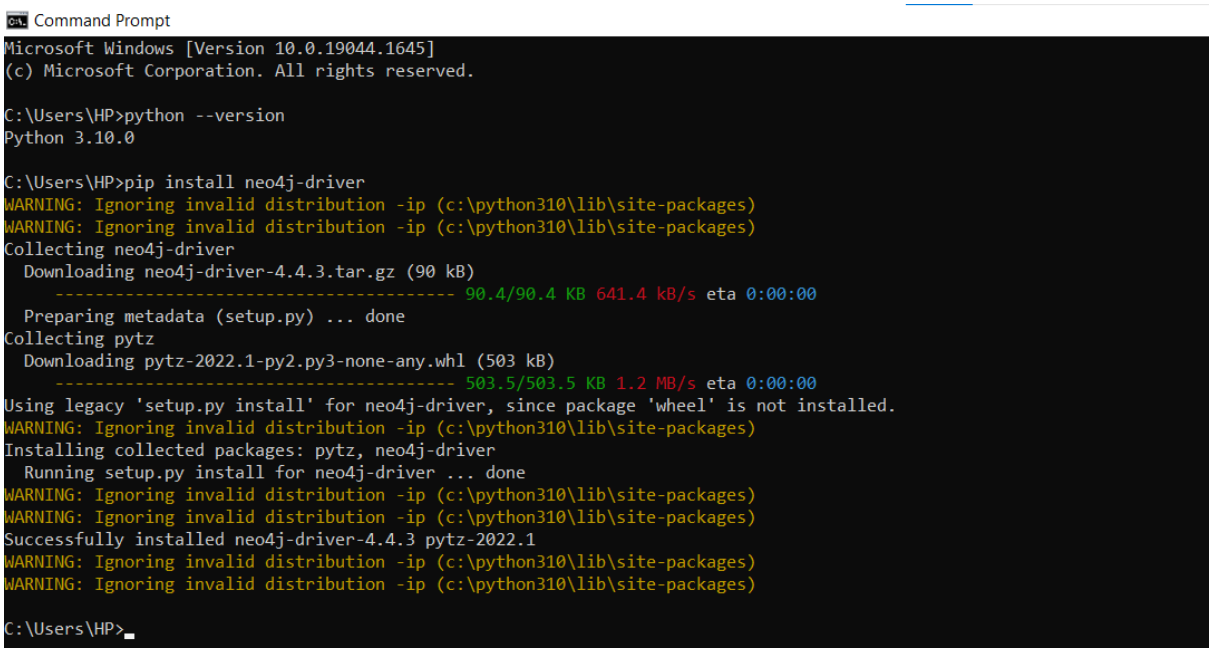
MATCH p=()-[r:CITES]->() RETURN p



MATCH (n:Paper) return n



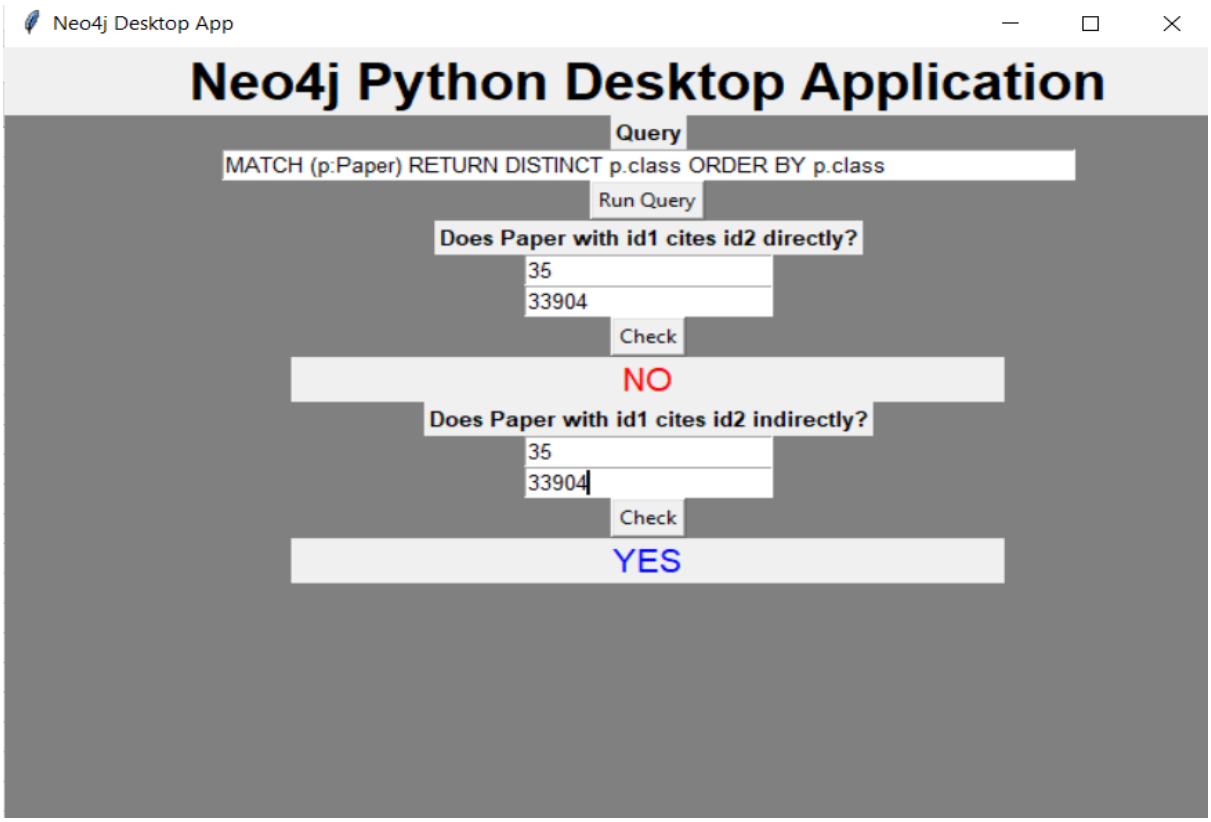
Installed neo4j-driver to run python application



Python Desktop Application:

```python
import sys
import os
import tkinter as tk
from tkinter import *
import tkinter.messagebox

# For Neo4j Connection
from neo4j import GraphDatabase
class Neo4jConnection:

    def __init__(self, uri, user, pwd):
        self.__uri = uri
        self.__user = user
        self.__pwd = pwd
        self.__driver = None
        try:
            self.__driver = GraphDatabase.driver(self.__uri, auth=(self.__user, self.__pwd))
        except Exception as e:
            print("Failed to create the driver:", e)

    def close(self):
        if self.__driver is not None:
            self.__driver.close()

    def query(self, query, db=None):
        assert self.__driver is not None, "Driver not initialized!"
        session = None
        response = None
        try:
            session = self.__driver.session(database=db) if db is not None else self.__driver.session()
            response = list(session.run(query))
        except Exception as e:
            print("Query failed:", e)
        finally:
            if session is not None:
                session.close()
        return response
conn = Neo4jConnection(uri="bolt://localhost:7687", user="neo4j", pwd="neo4j")
```

Output:



Conclusion: Successfully installed and configured neo4j graph database and run python desktop application for the given dataset to achieve required aim.

References:

https://neo4j.com/docs/cypher-manual/current/