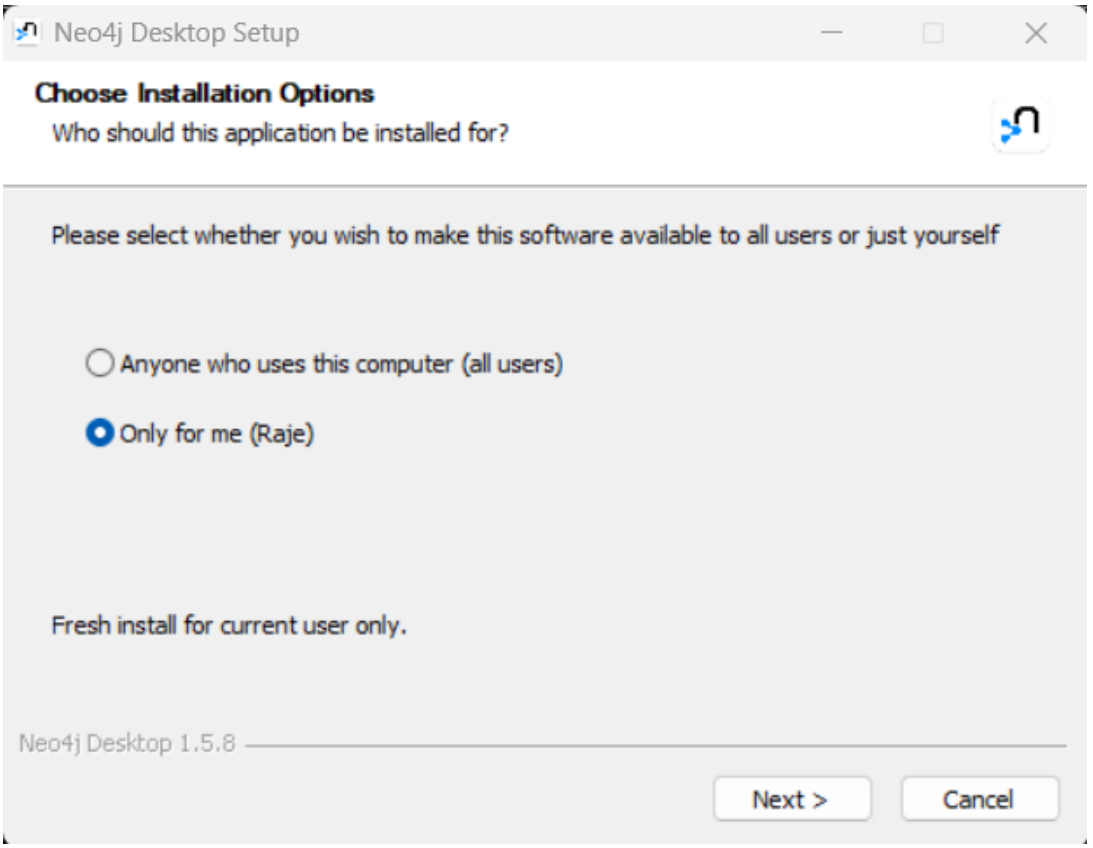
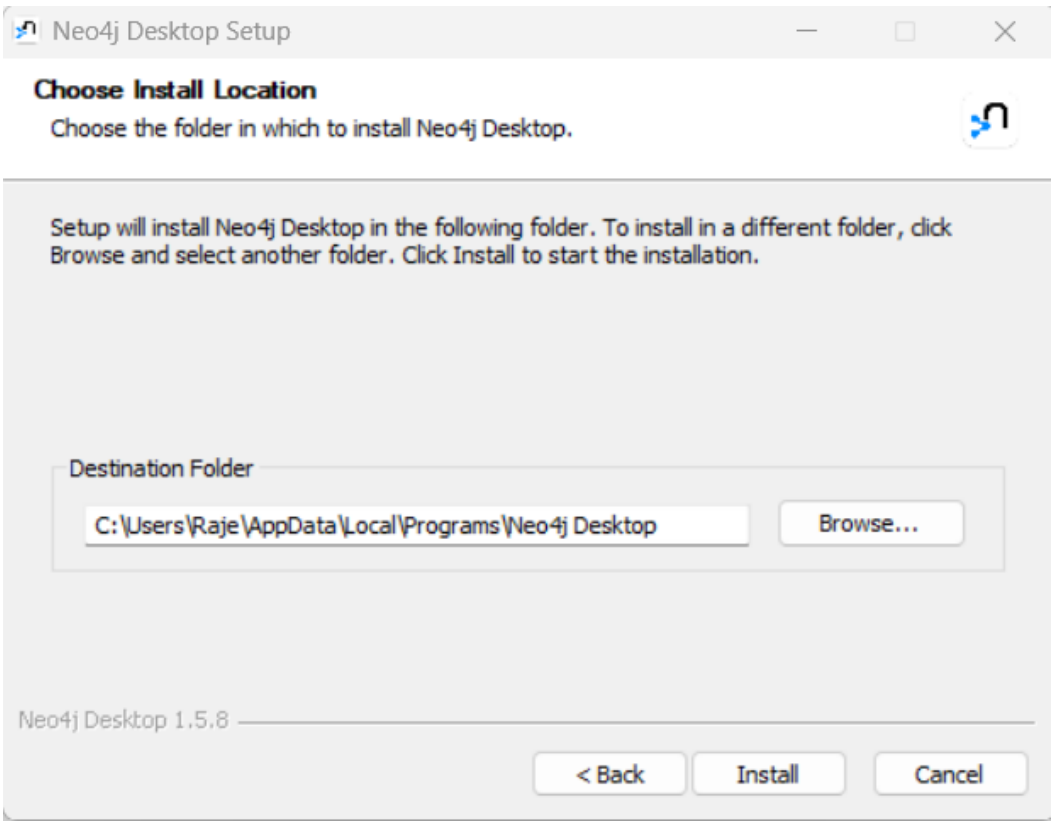
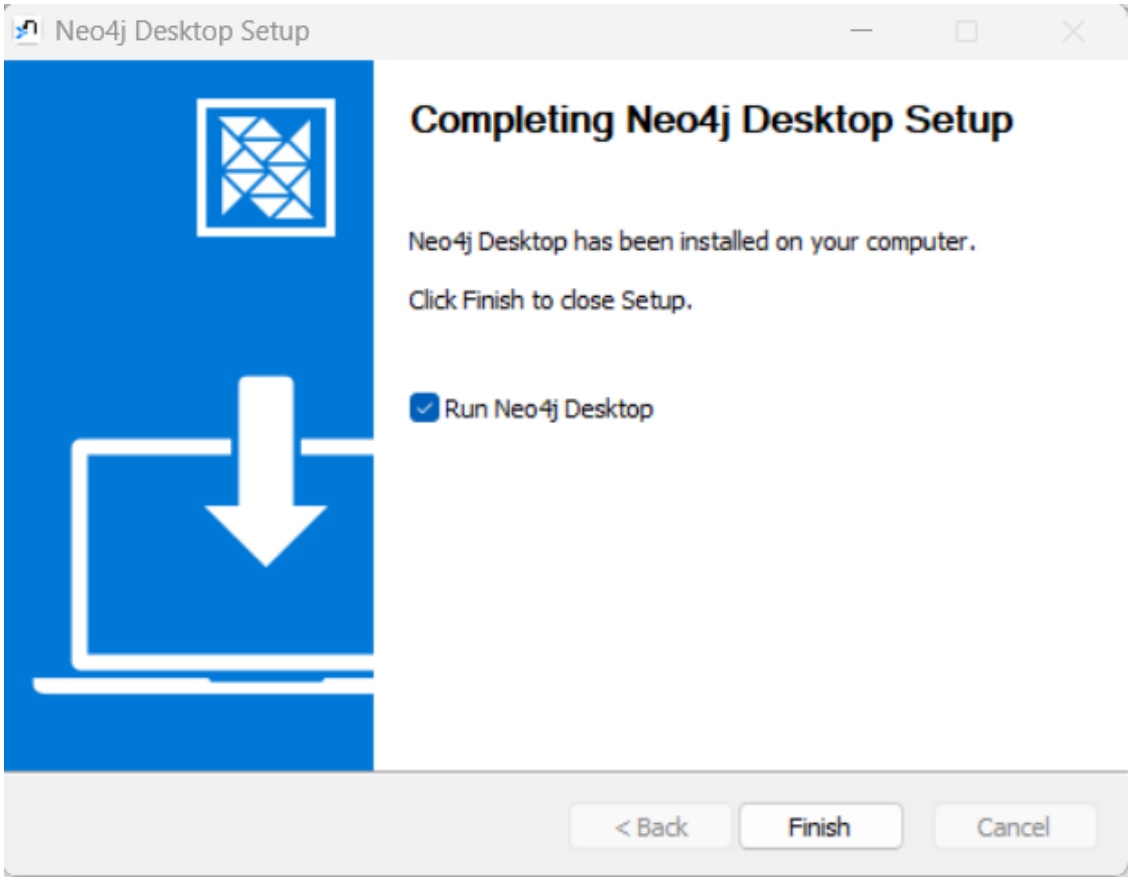
TY B.Tech. (CSE) – II [2022-23]
5CS372 : Advanced Database System Lab.
Assignment No. 11

PRN: 2020BTECS00033
Name: Prathamesh Raje

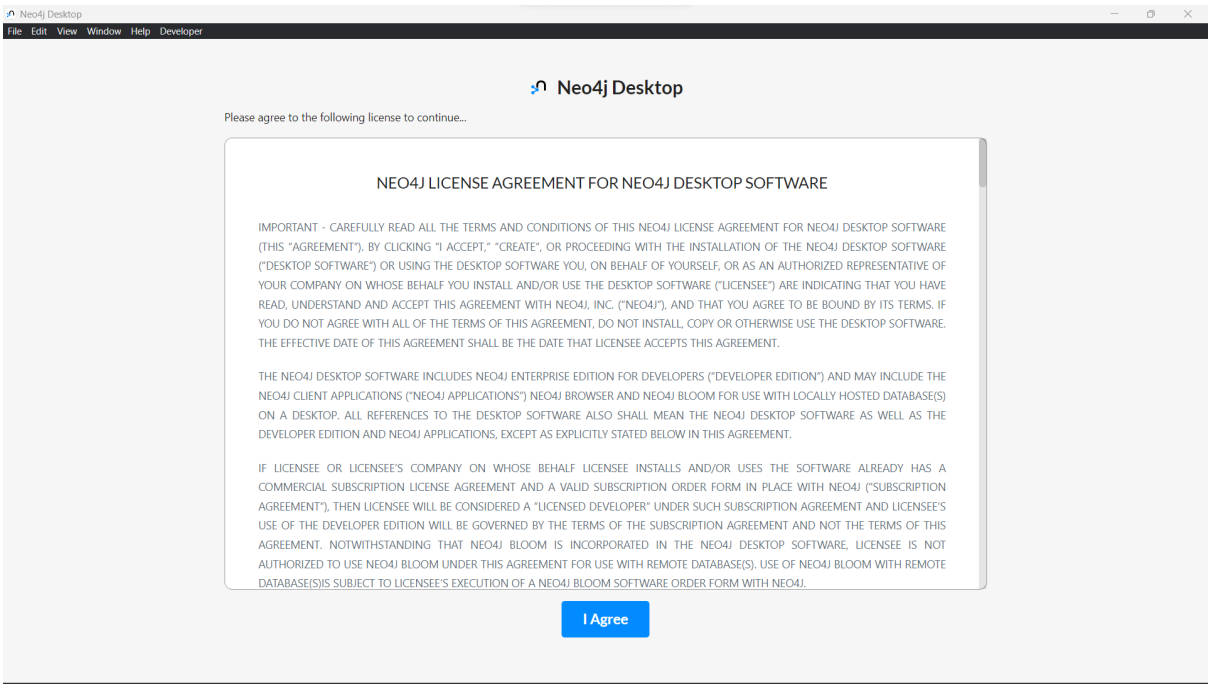
Title: Installation of Neo4j Desktop

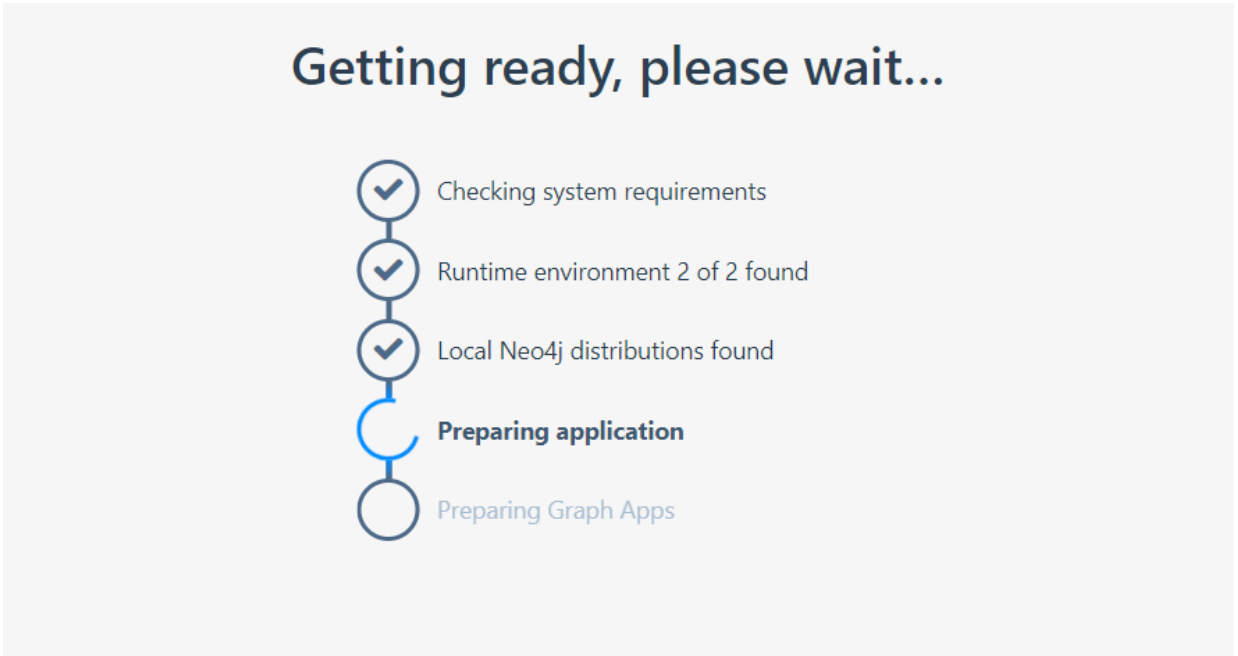


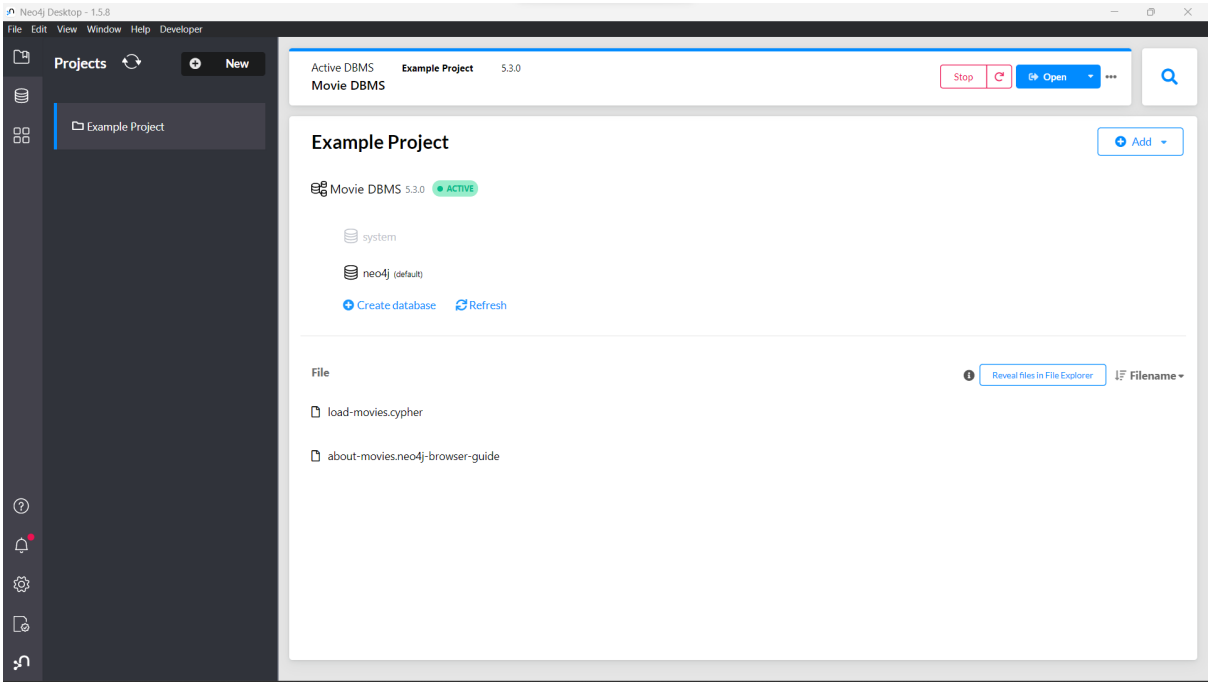

Neo4j Desktop Setup
Choose Install Location
Choose the folder in which to install Neo4j Desktop.
Setup will install Neo4j Desktop in the following folder. To install in a different folder, click Browse and select another folder. Click Install to start the installation.
Destination Folder
C:\Users\Raje\AppData\Local\Programs\Neo4j Desktop
Browse...
Neo4j Desktop 1.5.8
< Back
Install
Cancel


Neo4j Desktop Setup
Completing Neo4j Desktop Setup
Neo4j Desktop has been installed on your computer.
Click Finish to close Setup.
Run Neo4j Desktop
< Back
Finish
Cancel

Creating the New Database






Neo4j Desktop - 1.5.8
File Edit View Window Help Developer
Projects
New
Example Project
Active DBMS
Example Project
5.3.0
Movie DBMS
Stop
Open
Example Project
Add
Movie DBMS 5.3.0
ACTIVE
system
neo4j (default)
Create database
Refresh
File
Reveal files in File Explorer
Filename
load-movies.cypher
about-movies.neo4j-browser-guide


Edit settings
#server.directories.transaction.logs.root=data/transactions
#server.directories.dumps.root=data/dumps
# This setting constrains all `LOAD CSV` import files to be under the `import` directory. Remove or comment it out to
# allow files to be loaded from anywhere in the filesystem; this introduces possible security problems. See the
# `LOAD CSV` section of the manual for details.
server.directories.import=import
# Whether requests to Neo4j are authenticated.
# To disable authentication, uncomment this line
dbms.security.auth_enabled=false
# Number of databases in Neo4j is limited.
# To change this limit please uncomment and adapt following setting:
# server.max_databases=100
# Enable online backups to be taken from this database.
#server.backup.enabled=true
By default the backup service will only listen on localhost.
To enable remote backups you will have to bind to an external
network interface (e.g. 0.0.0.0 for all interfaces).
The protocol running varies depending on deployment. In a cluster this is the
Reset to defaults
Undo
Apply
Close

GUI Python File Code

```
from neo4j import GraphDatabase

class Neo4jConnection:

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

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

    def query(self, query, db=None):
        assert self.__conn is not None, "conn not initialized!"
        session = None
        response = None
        try:
            session = self.__conn.session(database=db) if db is not None else
self.__conn.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="", pwd="")


# res = conn.query("show databases")
# print(res)
# conn.query("CREATE OR REPLACE DATABASE coradb")


query_string = '''
CALL {
LOAD CSV WITH HEADERS FROM
'https://raw.githubusercontent.com/ngshya/datasets/master/cora/cora_content.csv'
AS line FIELDTERMINATOR ','
```

```
CREATE (:Paper {id: line.paper_id, class: line.label}) }
'''
conn.query(query_string, db='coradb')

query_string = '''
CALL{
LOAD CSV WITH HEADERS FROM
'https://raw.githubusercontent.com/ngshya/datasets/master/cora/cora_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)}
'''
conn.query(query_string, db='coradb')

# a = '1152448'
# b = '2354'

# query_string = " MATCH (p1:Paper { id: '" + a + "' }),(p2:Paper { id: '"+b+"' }), path =
shortestPath((p1)-[*..15]->(p2)) return path "

# print(query_string)

# res = conn.query(query_string,db='coradb')

# print(res)
```

View the Database in Neo4j Browser