UNIVERSITEIT VAN PRETORIA
UNIVERSITY OF PRETORIA
YUNIBESITHI YA PRETORIA
Faculty of Engineering, Built Environment and Information Technology

School of Information Technology
Department of Computer Science

COS326 Database Systems
Practical 10 2023

Release Date: 17 October 2024
Submission Date: 27 October 2024 @ 23:59Hrs
Lecturer: Mr S.M Makura

Total: 50 Marks

Objectives

1. Get exposure to the Neo4j graph DBMS.
2. Learn how to create and use a graph for a Neo4j database.
3. Appreciate the differences between SQL and NoSQL databases.
4. Learn how to create a Java application that uses a Neo4j database.

Instructions

1. You must submit the following files:
   a. ***Task1Queries.txt*** with all the queries for Task 1.
   b. **All the necessary files** for your Java application for Task 2.
2. Compress the above documents into an archive and upload it to ClickUP **before** the due date/time. The file name for the archive must have your student number as part of the file name, e.g. **uxxxxxxxx-prac10.zip** or **uxxxxxxxx-prac10.tar.gz** where uxxxxxxxx is your student number.
3. Book for a demo session via Discord to demo the practical.

NO LATE submissions will be accepted after the submission date and time has lapsed. Do not wait till the last minute to submit and start giving excuses that you faced technical challenges when you tried to submit.

Task 1: IMPORTING DATA INTO A Neo4j DATABASE [16 marks]

Create a new database and import data into this database:

1. The `actors.csv` and `movies.csv` files are attached on ClickUP.
2. Create a second database called **Prac10Neo4jB.graphdb** and then click the 'Start' button.
3. In the database folder for **Prac10eo4jB.graphdb**, create a sub-folder called `import`. Copy the files `actors.csv` and `movies.csv` to the `import` folder.
4. Write a Cypher query statement to import the `actors.csv` file data into the database. (6)
5. Write a Cypher query statement to show the current database contents. (2)
6. Write a Cypher query statement which uses the `movies.csv` file data to create the `ACTED_IN` relationships between actor nodes and movie nodes. (6)
7. Write a Cypher query statement to show the current database contents. (2)

Task 2: JAVA APPLICATION TO ACCESS Neo4j DATABASE [34 marks]

In this task, you will create a Java application that accesses a Neo4j database. To do this, you will need the Neo4j Java Driver. Here is a link with instructions on setting up Neo4j and Java: Neo4j Java Driver.

Create a Java application (either a JavaFX/Swing or a Web application) using NetBeans or any Java IDE you are comfortable with, which accesses the **Prac10Neo4jB.graphdb** database you created in Task 1. The application should:

1. Report that it has successfully connected to the Neo4j database. (4)
2. Use transactions for database operations. (4)
3. Provide a JavaFX/Swing/Web interface for the actions that a user can perform (the actions are listed in question 4 below). (10)
4. Enable the user to perform the following actions on the database:
    - Count the number of actor nodes in the network. (4)
    - For each actor, count the number of movies they have acted in. Show only the top 100 results. (6)
    - Show the names of actors who have acted in the same movie. (6)

Note: The results for the above actions must be output to an appropriate GUI component.