IST769 Homework Problem Set I

Graph Model: Neo4J

In this lab, we will explore the graph model with the Neo4j database. We will learn how to query data with the Cypher query language and access data from Apache Spark with Cypher. Learning Outcomes

At the end of this lab, you should be able to:

* Query a graph model using Neo4j’s Cypher query language
* Import and export data from Neo4j into Spark DataFrames
* Build a graph from a table of data

# Prerequisites

Before you begin:

* Open a terminal window in the lab environment.
* Set the current working directory to **advanced-databases**.
* Start the following services required by the lab Spark and Redis:

**jupyter neo4j**.

# Tools Used in This Lab

The following tools will be used in this lab:

1. To access Jupyter Lab from your Windows host:

[http://localhost:8888](http://localhost:8888/) The password is **SU2orange!**

1. To access the Neo4j admin UI:

[http://localhost:7474](http://localhost:7474/)

Lab Problem Set

**QUESTIONS:**

**For each question, include a copy of the code required to complete the question along with a screenshot of the code and a screenshot of the output.**

1. Using the **:play northwind-graph** command, build the Northwind Product Catalog in Neo4j.

Diagram, table

Description automatically generated

Make sure to load the Product, Categories, and Suppliers nodes along with the PART\_OF and SUPPLIES relationships by running the sample code provided. This should just be a matter of following the commands in the first three steps of the Northwind graph.

As proof you've completed this correctly, write a Cypher query to display all Products, Suppliers, and Categories using both relationships. Your screenshot should include your Cypher code plus the graph output. If you did it correctly, there should be eight categories, 77 products, 29 supplier notes (and lines connecting nodes).

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Select a supplier contact name of your choosing. Write a Cypher query to display the supplier’s name, the supplier’s company name, the names of the products it supplies, and the unit price of those products for only products that are not discontinued. Display this information in a table.

A screenshot of a computer

Description automatically generated

1. Select two products of your choosing. Write a Cypher query to display a graph of the supplier and category for those products. Based on the graph displayed as output, are the two products you selected in the same category?

A diagram of a company

Description automatically generated

1. You just sold 30 units of `laughing lumberjack lager`; update the node to reflect the proper stock and display the output.

A screenshot of a computer

Description automatically generated

1. Load a Spark DataFrame of USA country suppliers and their products for products that are not discontinued. Include supplier company and contact names, country, and phone. From products, include product name, discontinued, unit price, and units in stock.

A screenshot of a computer

Description automatically generated

1. Load the **/datasets/fudgemart/fudgemart-employees.json** into a Spark DataFrame. Make sure to create a column **employee\_name**, which combines the first and last names together.

A screenshot of a computer

Description automatically generated

1. In Spark, load the employees into Neo4j under the label node Employee; include employee\_name, employee\_department, employee\_id, and employee\_jobtitle as node attributes. Make sure employee\_name is the first attribute as this will be the node’s visible label.

Provide evidence the nodes were created in Neo4i UI with a Cypher query.

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

1. In Spark add a SUPERVISES relationship to the nodes. Basically, you must match two nodes—one where the ID is the employee\_id and the other where the ID is the employee\_supervisor\_id—and then merge a relationship.

Provide evidence the relationships were created in Neo4i UI with a Cypher query to show the employees of a supervisor of your choosing.

A screenshot of a computer program

Description automatically generated

1. In Neo4J build an organizational chart by starting at the supervisor who is the “CEO” and the employees recursively four levels deep. To learn how to query recursively, check out:

<https://stackoverflow.com/questions/31079881/simple-recursive-cypher-query> Display the graph.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

**IMPORTANT:** When you are finished with the lab, execute:

**PS:> docker-compose stop**

To turn off all running services, then shut down your Azure Lab instance.