
OpenCourseWare

Database

Lourdes Moreno López

Paloma Martínez Fernández

José Luis Martínez Fernández

Rodrigo Alarcón García

Project 7 (neo4j (3.4))



OpenCourseWare



Data Base

Bachelor in Data Science and Engineering

SUBJECT: Project 7 (neo4j)

uc3m

Universidad
Carlos III
de Madrid

Exercise 1: ClimbingDB (social network climbing database)	2
Drills	2
Display Schema / Graph	2
Query / Update / Remove	2
Exercise 2: Create a Graph database	3
Drills	3
Create	3
Display Schema / Graph	3
Query	3
Requested information	3
Instructions, date	3

Exercise 1: ClimbingDB (social network climbing database)

This network provides a model that combines a database for routes and climbing sites as well as a social point where climbers could meet and interact with one another.

Data about routes, climbing sites, and countries containing them is highly connected.

Description of nodes:

- ROUTE is the way up the rock :), with attributes: `_name_`, `_grade_` (see the grading system above)
- CRAG, sometimes referred as climbing site or just the wall, with attributes: `_name_`, `_orientation_` (east, west, south, north)
- COUNTRY represents the country (English name), with attribute: `_name_`
- CLUB represents a climbing club, with attribute: `_name_`
- CLIMBER is the individual climber (the social aspect of this model), with attributes: `_first_name_`, `_last_name_`, `_date_of_birth_`

Description of relationships:

- Crag (climbing sites) are a summary of routes. Crag contains routes and routes are a part of crags. One route can be a part of only one crag but different routes have the same names (pretty common actually, but no two routes have the same name in one crag).
- Usually crags are managed by clubs and clubs are responsible for crags (sometimes this is not true but in majority of cases it is). Clubs originate from different countries and have members, climbers.
- As do clubs, so do climbers come from different countries and are part of (or not) clubs. Routes are climbed by climbers (obviously). This provides a sort of diary for individual climbers as well as some statistics for routes. Many climbers have a favorite climbing site which they like to visit the most.

Download climbingDB script (Source: <https://neo4j.com/graphgists/climbdb-social-networkclimbing-database-2/>), and insert climbing data (nodes, relationships, properties) into the graph.

Drills

Display Schema / Graph

Write the query and display the screenshot of the output (text, table or graph)

- 2.1. Write a query to display the schema of your database.
- 2.2. Recover all nodes and relationships (display graph).
- 2.3. Recover all American climbers ('USA').
- 2.4. Recover all the friends of the climber Alice.
- 2.5. Which club does the climber Annie belong to?
- 2.6. What country is the club 'Ljubljana CC' from?
- 2.7. List of all climbers with the number of routes climbed
- 2.8. Where do climbers climb (most visited climbing site)
- 2.9. Update the routes with a property called "height" and initially equal to 0
- 2.10. Delete the climber node 'Bob'

Exercise 2: Create a Graph database

Create a graph database that represents a network of all your classmates.

- It is necessary to include nodes of 'person' in which it is necessary to include the name of the person, NIA and other property that you have to invent.
- It is necessary to include more nodes that you invent, such as a university, etc.
- You need to create relationships between nodes, relationships are " is a project teammate of", "is a friend of", "studies in" and other relationships that you invent. Relationships must have some properties.

You can start working from the excel of classmates and the information of the project groups, both files uploaded in the global classroom platform

Drills

Create

- 2.1. Write the script, insert data (nodes, relationships, properties) into the graph, display the screenshot of the right menu (Database Information) and output (Table)

Display Schema / Graph

Write the query and display the screenshot of the output (text, table or graph)

- 2.2. Write a query to display the schema of your database.
- 2.3. Recover all nodes and relationships (display graph).

Query

Write the query and display the screenshot of the output (text, table or graph)

- 2.4. Recover all your friends.
- 2.5. List of the names of all classmates with the number of friends.
- 2.6. Invent three queries based on your scheme and data included in the graph.