

# Applied Databases

## Neo4j I Exercise Sheet

### Setup

- On the VM Neo4j is located here:

`C:\Users\appDB\Documents\neo4j-community-5.3.0-windows\neo4j-community-5.3.0`

- `neo4j.conf` is located here:

`C:\Users\appDB\Documents\neo4j-community-5.3.0-windows\neo4j-community-5.3.0\conf\neo4j.conf`

- On the VM the Username and Password are:

Username = `neo4j`

Password = `neo4jneo4j`

### Part 1

- Use a new database called */6p1* (by updating `neo4j.conf`).
- Run Neo4j as follows:
  - Open a Windows Command prompt/PowerShell and change to the `bin` folder of the Neo4j installation.
  - Run `neo4j console`

```
C:\>cd \Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\bin
C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\bin>neo4j console
Directories in use:
home:          C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3
config:        C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\conf
logs:          C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\logs
plugins:       C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\plugins
import:        C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\import
data:          C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\data
certificates:  C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\certificates
licenses:      C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\licenses
run:           C:\Users\GHarrison\Documents\neo4j-community-4.3.3-windows\neo4j-community-4.3.3\run
Starting Neo4j.
2021-09-29 19:21:36.589+0000 INFO Starting...
2021-09-29 19:21:38.479+0000 INFO ===== Neo4j 4.3.3 =====
2021-09-29 19:21:39.424+0000 INFO Performing postInitialization step for component 'security-users' with version 3 and
status CURRENT
2021-09-29 19:21:39.425+0000 INFO Updating the initial password in component 'security-users'
2021-09-29 19:21:40.085+0000 INFO Bolt enabled on 127.0.0.1:7687.
2021-09-29 19:21:40.621+0000 INFO Remote interface available at http://localhost:7474/
2021-09-29 19:21:40.623+0000 INFO Started.
```

- Then open a browser to <http://localhost:7474> and select the database just created (*/6p1*).

- Create the following nodes with a label :Student with the following properties:

- name: "Tom"
- sid: "G001"
- age: 23
- sex: "M"
- address: "Galway"
- hair: "brown"
- email: "tom@gmail.com"
  
- name: "Sean"
- sid: "G002"
- age: 19
- sex: "M"
- address: "Galway"
- email: [sean@gmail.com](mailto:sean@gmail.com)
  
- name: "Bob"
- sid: "G003"
- age: 22
- sex: "M"
- address: "Mayo"
- email: [bob123@hotmail.com](mailto:bob123@hotmail.com)
- twitter: "@bob123"
  
- name: "Mary"
- sid: "G004"
- age: 20
- sex: "F"
- address: "Mayo"
- hair: "blonde"
- email: [mary19@gmail.com](mailto:mary19@gmail.com)
- twitter: "@mary19"
- snapchat: "mary19"
  
- name: "Alice"
- sid: "G005"
- age: 28
- sex: "F"
- address: "Roscommon"
- email: [alice@hotmail.com](mailto:alice@hotmail.com)
- snapchat: "alice123"

- **name:** "Pat"
- **sid:** "G006"
- **age:** 24
- **sex:** "M"
- **address:** "Roscommon"
- **email:** "[pat@hotmail.com](mailto:pat@hotmail.com)"
- **twitter:** "patABC"

- Create the following nodes with a label : **Lecturer** with the following properties:

- **name:** "Alan"
- **sid:** "L001"
- **age:** 57
- **sex:** "M"
- **address:** "Galway"
- **email:** "alan@gmit.ie",
- **twitter:** "@alan"

- **name:** "Mary"
- **sid:** "L002"
- **age:** 47
- **sex:** "F"
- **address:** "Mayo"
- **email:** "mary@gmit.ie"
- **hair:** "brown"

- Find the average age of Students, rounded to the nearest whole number.
- Show the name of each student and his/her age.
- Find the age of the youngest Student.
- Show the names of students who have a *twitter* attribute.
- Show the number of students who have a *twitter* attribute.

- Show the average of age of people in their 20s, 30s and 40s rounded to one decimal place.
- Show all the properties for the Student *Tom*.
- Increase everyone's age by 1.
- Return the name and age of all males living in Galway.
- Create the following nodes with both :Student and :Lecturer labels
  - `name: "Yvonne"`
  - `age: 37`
  - `sex: "F"`
  - `address: "Galway"`
  - `email: yvonne@gmit.ie`
  - `twitter: "@yv12"`
  
  - `name: "Walter"`
  - `age: 44`
  - `address: "Galway"`
  - `email: walter@gmit.ie`
  - `hair: "black"`
- Show the name, age and hair colour of everyone who is both a Student and a Lecturer.
- Update the *twitter* attribute of all lectures to have GMIT after their existing twitter name. E.g. "@alan" should become "@alanGMIT".

- Find the average age of Males and find the youngest Male(s).  
Then return the name (as *Name*) and age (as *Age*) of the youngest Male(s) as well as the average age of Males (as *AverageAge*) and the difference in age between the youngest Male(s) and the average age (as *Difference*).

E.g., If the average age of Males was 30, and the youngest Male was called “Tony” aged 20, the following should be returned:

<b>Name</b>	<b>Age</b>	<b>AverageAge</b>	<b>Difference</b>
Tony	20	30	10

## Part 2

- Use a new database, *l6p2*, (by updating neo4j.conf).
- In the Neo4j Browser, select the new database and type the following command:

```
tdb$ :play movies
```

- This will return the following:

**Create**

To the right is a giant code block containing a single Cypher query statement composed of multiple CREATE clauses. This will create the movie graph.

Click on the code block  
Notice it gets copied to the editor above ↑  
Click the editor's play button to execute  
Wait for the query to finish  
WARNING: This adds data to the current database, each time it is run!

```
Real World'})  
CREATE (Keanu:Person {name:'Keanu Reeves', born:1964})  
CREATE (Carrie:Person {name:'Carrie-Anne Moss', born:1967})  
CREATE (Laurence:Person {name:'Laurence Fishburne', born:1961})  
CREATE (Hugo:Person {name:'Hugo Weaving', born:1960})  
CREATE (LillyW:Person {name:'Lilly Wachowski', born:1967})  
CREATE (LanaW:Person {name:'Lana Wachowski', born:1965})  
CREATE (Joels:Person {name:'Joel Silver', born:1952})  
CREATE  
(Keanu)-[:ACTED_IN {roles:['Neo']}]>(TheMatrix),
```

2/8

- Go to page 2 and follow steps circled above.
- A series of 171 nodes (representing Movies and People) and 253 relationships (such as ACTED\_IN, DIRECTED etc.) between the nodes should now be created.
- Type MATCH(n) RETURN n to see all nodes and relationships:

Database Information

Use database: tdb

Node Labels: (171) Movie (133) Person

Relationship Types: (253) ACTED\_IN (172) DIRECTED (64) PRODUCED (15) WRITING (10) FOLLOWING (3) REVIEWED (50)

Property Keys: born, name, rating, released, roles, summary, tagline, title

Connected as: Username: neo4j, Roles: []

tdb\$ MATCH(n) RETURN n

Displaying 171 nodes, 253 relationships.

- Show each movie node for movies that were released between 2000 and 2010

- Set an attribute called `olderThan70` to true for all Persons born in the 1930s.
- Show the movie title and the year it was released for the first 10 movies in alphabetical order.
- Show the unique years in which movies were released in chronological order.
- Show the title and tagline for movies released in 1999.
- Show the names of the people (as *People*) and the year they were born (as *YOB*) for everyone older than "Robin Williams".
- Show the number of movies released in 2006 (as *Releases\_in\_2006*).
- Show the name (as *Name*) and born (as *YOB*) the youngest Person(s).