**Introduction**

This document is relative with Neo4j program about timetable of GMIT. It can describe all relationship like courses, room, tutors and others.

**What to store**

In Neo4j, data stored as node, edge. Both of them have a number of properties.

This graph contains nodes and relationships. Each of nodes have relationship with other.

Nodes represent events and relationships connect every nodes by using edge

Nodes can be divided by labels. Label groups nodes with same label, nodes with same label mean they are same group and can be query by using label as a key.

Every nodes have id attribute, each of nodes’ id is different with other. It can be queried with label.

List of nodes:

Course has name and available time attributes and relates with room and tutor

//Lab has name and available time attributes and relates with group

Room has number and seats attributes also relates with course and lab

Tutor has name and relate with course and lab

Group has group number to divide students

Date has weekday attribute

List of relationships:

Room—BEUSED->Course

Date—RELATE-> Course

Tutor—TEACH-> Course

Group—LEARN-> Course

**About Noe4j**

**Neo4j** is a [graph database](https://en.wikipedia.org/wiki/Graph_database) management system developed by Neo Technology, Inc. Described by its developers as an [ACID](https://en.wikipedia.org/wiki/ACID)-compliant transactional database with native graph storage and processing,[[3]](https://en.wikipedia.org/wiki/Neo4j#cite_note-product-details-3)Neo4j is the most popular graph database according to db-engines.com.[[4]](https://en.wikipedia.org/wiki/Neo4j#cite_note-DB-Engines-4)

Neo4j is available in a [GPL3](https://en.wikipedia.org/wiki/GNU_General_Public_License)-licensed [open-source](https://en.wikipedia.org/wiki/Open-source_software) "community edition", with [online backup](https://en.wikipedia.org/wiki/Remote_backup_service) and [high availability](https://en.wikipedia.org/wiki/High_availability) extensions licensed under the terms of the [Affero General Public License](https://en.wikipedia.org/wiki/Affero_General_Public_License). Neo also licenses Neo4j with these extensions under closed-source commercial terms.[[5]](https://en.wikipedia.org/wiki/Neo4j#cite_note-LICENSE-5)

Neo4j is implemented in [Java](https://en.wikipedia.org/wiki/Java_(programming_language)) and accessible from software written in other languages using the [Cypher Query Language](https://en.wikipedia.org/wiki/Cypher_Query_Language) through a transactional HTTP endpoint, or through the binary 'bolt' protocol.

**How to store**

To store data into neo4j, it should use nodes and relationships to connect them

Node: user create a node and give it several attributes to specific and connect to other nodes

By using: create (xxx:Xxx{attribute1:’ ’ , attribute2:’ ’ })

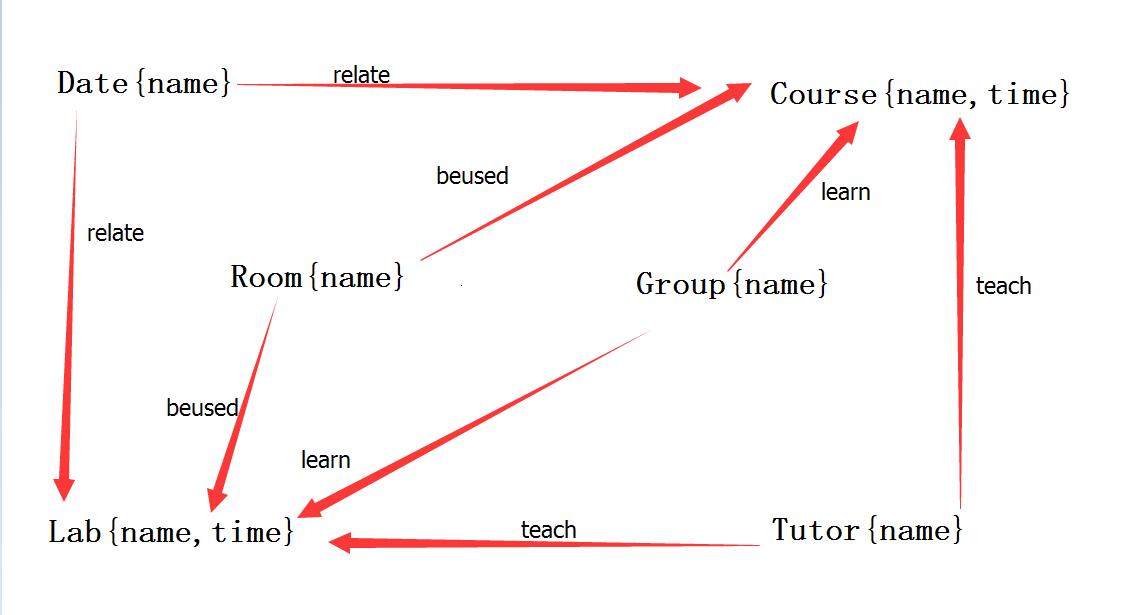
Relationship: this is a link between two nodes to display their relationship

·Group has relationship--LEARN with course

all groups connect with course of lecture, but they connect different labs and will not affect each other

·Course and Lab have relationships—BEUSED with room, RELATE with date, TEACH with tutor and LEARN with group

course link all of groups



Search all graph

MATCH (n) RETURN n

Search all group , course and lab with learn relationship

MATCH p=()-[r:LEARN]->() RETURN p

Search all nodes with beused relationship

MATCH p=()-[r:BEUSED]->() RETURN p

Search all nodes with teach relationship

MATCH p=()-[r:TEACH]->() RETURN p

Search all course

MATCH(N:Course) RETURN N

Search specific Group, Course and Date

MATCH(B:Group {name:'Group B'})-[:LEARN]->(GT:Course {name:'Graph Theory'})<-[:RELATE]-(Friday:Date{name:'Friday'}) RETURN B,GT,Friday

Search specific Group, Course time and Date

MATCH(B:Group {name:'Group B'})-[:LEARN]->(GT:Course {name:'Graph Theory',time:'12:00-13:00'})<-[:RELATE]-(Friday:Date{name:'Friday'}) RETURN B,GT,Friday