Task-11: CRUD Operations in Graph Databases.

Date: 21/10/25

Aim: To perform CRUD operations like creating, inserting, querying, finding, deleting operations on graph spaces.

* Create Node with Properties.

Properties are the key-value pairs using which a node stores data. You can create a node with properties using the CREATE clause. You need to satisfy these properties separated by commas within the flower braces "{}".

Syntax

Following is the syntax to create a node with properties.

CREATE (node: label {key1: Value, key2: Value, ----- }).

* Returning the Created Node

To verify the creation of the node, type and execute the following query in the dollar prompt.

MATCH (n) RETURN n.

* Creating Relationships.

We can create a relationship using the CREATE clause. We will specify relationship within the square braces "[]" depending on the direction of the relationship. It is placed between hyphen "-" and arrow "→" as shown in the following syntax

Syntax

Following is the syntax to create a relationship using the CREATE clause.

CREATE (node1)-[:RelationshipType]→(node2)

* Creating a Relationship Between the Existing Nodes.

You can also create a Relationship between the existing nodes using the MATCH clause.

Syntax.

Following is the syntax to create a relationship using the MATCH clause

MATCH (a: labeofNode1), (b: labeofNode2)
WHERE a.name = "nameofnode1" AND b.name = "nameof node2"

CREATE (a)-[:Relation]→(b)
RETURN a,b.

* Deleting a Particular Node.

To delete a particular node, you need to specify the details of the node in the place of "n" in the above query.

Syntax

Following is the syntax to delete a particular node from Neo4j using the DELETE clause.

MATCH (node: Label {properties------})
DETACH DELETE node.

create a graph database for student course registration, create student and dept node and insert values of properties

create (n: student {sid: "VTU14500",
Sname: "John",
deptname: "CSE"}.
)

Output

Added 1 label, created 1 node, set 3 properties completed after 232ms.

Create (n: student {sid: "VTU14502",
Sname: "vijay", "Harsoma",
deptname: "EEE"
}).

Output

Added 1 label, created 1 node, set 3 properties, completed after 15ms.

~~Create (n: dept {deptname: "CSE", deptid: "d001"})~~

Create (n: student {sid: "VTU14502",
Sname: "vijay",
deptname: "CSE"
})

Output

Added 1 label, created 1 node, set 3 properties, completed after 12 ms.

create (n: dept { deptname : "CSE", deptid : "d001"}).

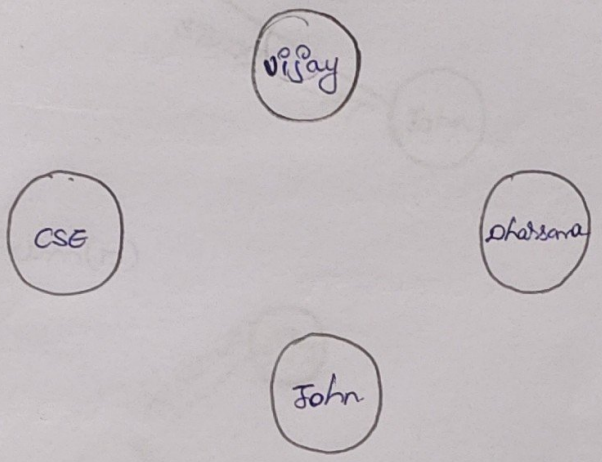
Output

Added 1 label, created 1 node, set 2 properties, completed after ~~72~~ ms

select all the nodes in your database using match command

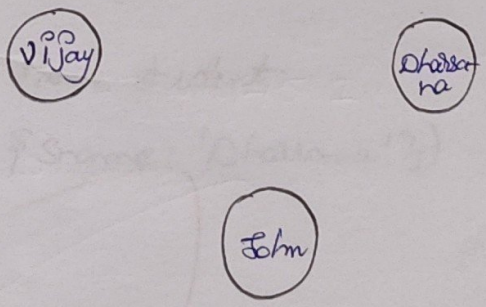
* match(n) return(n)

Output



* match (n: student) return (n)

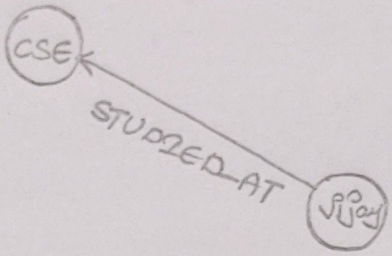
Output



a) create relationship between student and cse.

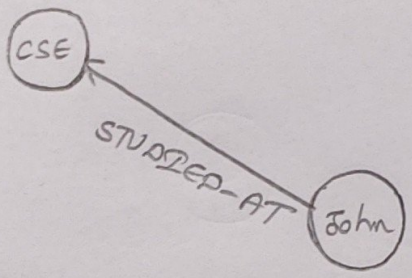
MATCH (s: student), (d: dept) WHERE s.Sname = 'vijay'
AND d.deptname = 'cse'
CREATE (s)-[st: STUDIED_AT]->(d)
return s, d.

Output:

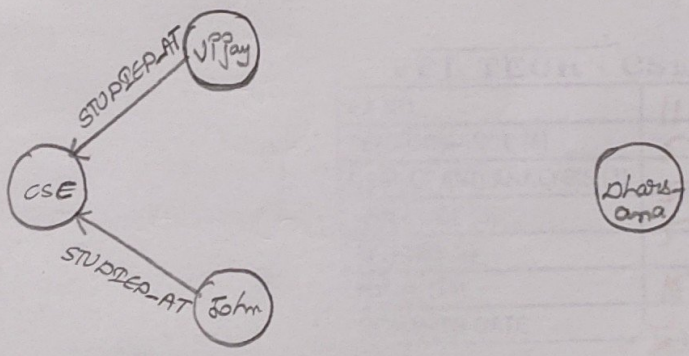


MATCH (s: student), (d: dept) WHERE s.Sname = 'John' AND d.deptname = 'CSE'
CREATE (s) - [st: STUDIED_AT] → (d)
return s, d.

Output:



match(n) return(n)

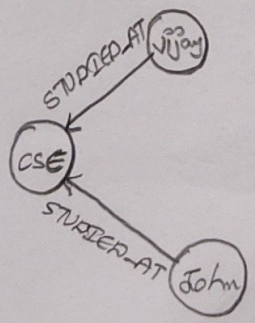


b) Delete a node from student

match (n: student {Sname: 'Dharsana'})
DELETE(n)

Output:

Deleted 1 node, completed after 108 ms.



VEL TECH - CSE	
EX NO.	11
PERFORMANCE (5)	5
RESULT AND ANALYSIS (3)	5
VIVA VOCE (3)	5
RECORD (4)	
TOTAL (15)	15
SIGN WITH DATE	

15/10

Result: The Implementation of CRUD operations like creating, inserting, finding and removing operations using Graph DB is successfully executed.