

TASK 11:- CRUD OPERATIONS IN GRAPH DATABASES

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: {}.

Syntax:-

Following is the syntax to create a node with properties.

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

• Returning the CreateNode:-

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 square braces "[]". depending of relationship it is placed between hyphen "-".

Syntax:- Following is the syntax to create a relationship using the CREATE Clause.

CREATE (node1)-[:Relationship]-(node2).

• Creating a Relationship Between the Existing Nodes-

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

• 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 create a relationship using the MATCH Clause.

WHERE a.name = "nameofnode1" AND b.name = "nameofnode2"
CREATE (a)-[:Relation]-(b)
RETURN a,b;

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 table, Created 1 node, Set 3 Properties,
Completed after 232 ms.

Create (n:Student {sid: "VTU14501", sname: "Dharsana",
deptname: "EEE"})

Output:- Added 1 label, Created 1 node, Set 2 Properties,
Completed after 12 ms.

Create (n:dept {deptname: "Cse", deptid: "d01"}).

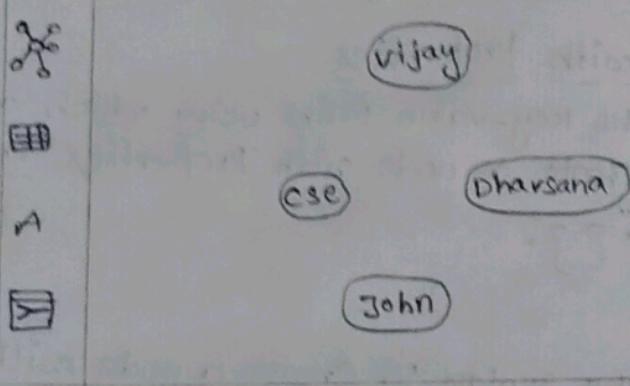
Output:- Added 1 label, Created 1 node, Set 2 Properties,
Completed after 72 ms.

Select all the nodes in your database using match Command.

match(n) return(n).

Output:-

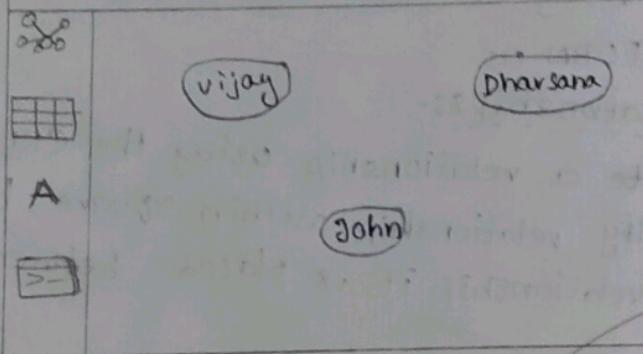
```
| neo4j# match(n) return(n)
```



Match (n:Student) return (n).

Output:-

```
neo4j$ match(n: Student) return(n)
```



a) Create Relationship between student and cse.

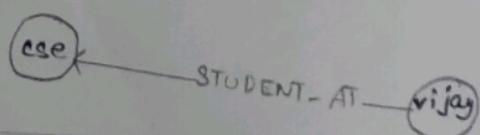
~~MATCH (s:student), (d:dept) WHERE s.sname = 'vijay' AND
d.deptname = 'CREATE (s)-[st: STUDENT_AT]-(d)
return s,d;~~

Output:-

```

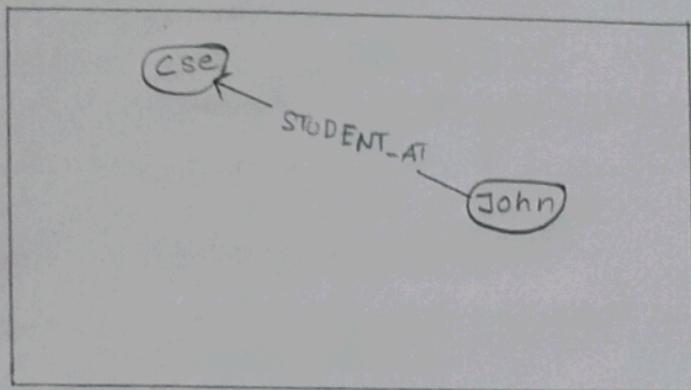
1 MATCH (s:student),(dept:d) WHERE s.sname = 'Vijay'
2 AND d.department = '(sel'
3 CREATE (s)-[STUDENT_AT]-(d)
4 return s,d.

```

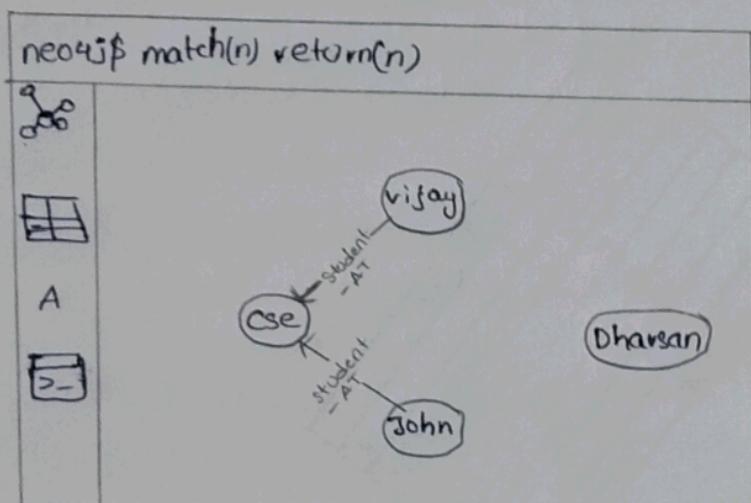


MATCH (s:STUDENT), (d:dept) WHERE s.sname='John' AND d.dept_name = 'CSE' CREATE(s)-[s+STODIED_AT]->(d).
 return s,d.

Output:-



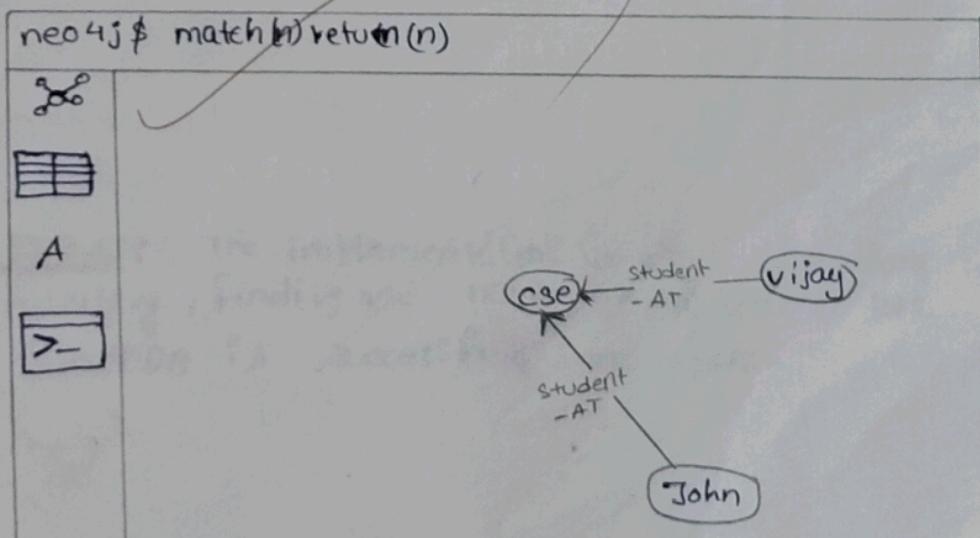
match(n) return(n)



(b) Delete a node from student.

match (n: Student {sname:'Dharsana'}) DELETE(n)

Output:- Delete 1 node, completed after 10834 ms.



VEL TECH	
EX No	PERFORMANCE
1	9
2	9
3	9
4	9
5	9
6	9
7	9
8	9
9	9
10	9
11	9
12	9
13	9
14	9
15	9
16	9
17	9
18	9
19	9
20	9
21	9
22	9
23	9
24	9
25	9

RESULT:- The implementation of CRUD operations like creating, inserting, finding and removing operations like using GraphDB is successfully executed.

130023