

### Syntax:-

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

```
INHERIT a.name = "nameofnode" AND b.name = "nameofnode";
```

```
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 -n, create student and dept node and insert value of properties.

```
create (n:student {sid: "VITWUSOD", sname: "John",
deptname: "CSE"})
)
```

Output:- Added 1 table, created 1 node, set 3 properties, completed after 232 ms.

```
create (n:student {sid: "VITWUSOD", 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).
```

### Task II

#### 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 "key": value.

Syntax:- Following is the syntax to create a node with properties.

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

#### Returning the create node:-

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

```
MATCH (n) RETURN n.
```

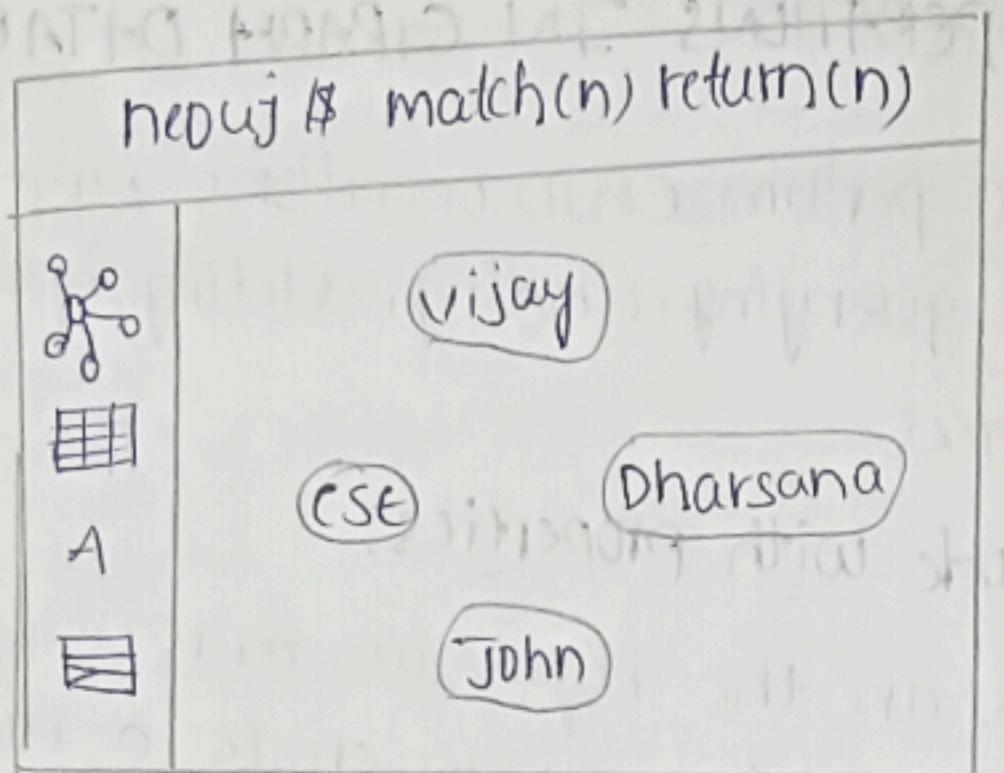
#### Creating relationships:-

We can create a relationship using the CREATE clause. We will specify relationship within square braces "[ ]" depending on relationship is placed between hyphen "-".

Syntax:- Following is the syntax to create a relationship using the CREATE(node-1)-[:Relationship]->(node2).

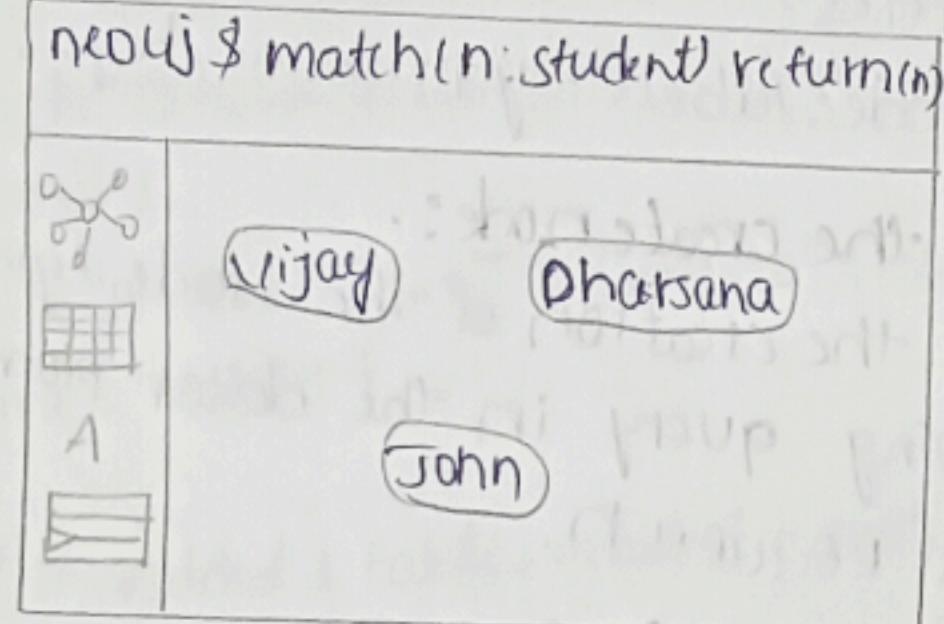
Creating a relationship between the existing nodes you can also create a relationship between the existing nodes using the MATCH clause.

Output:-



Match(n:student)return(n); -

Output:-

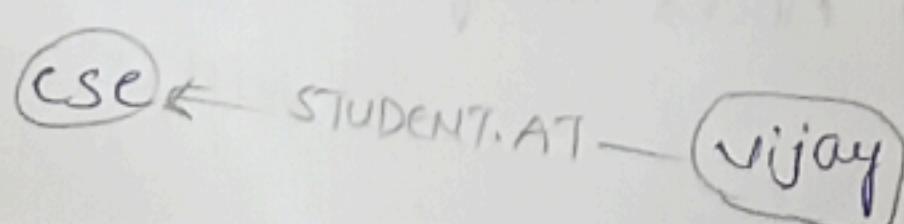


a) Create Relationship between student and CSE.

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

Output: -

```
1 MATCH (s:student),(dept:d) WHERE s.name =  
2 'vijay' AND d.department = 'cse'  
3 CREATE (s)-[student_AT]->(d)  
4 return s.id.
```

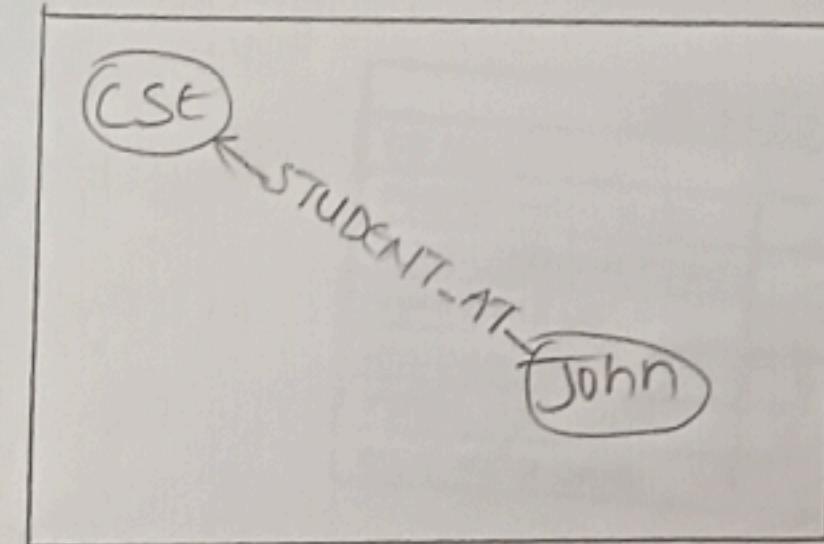


## Deleting a participant from a task.

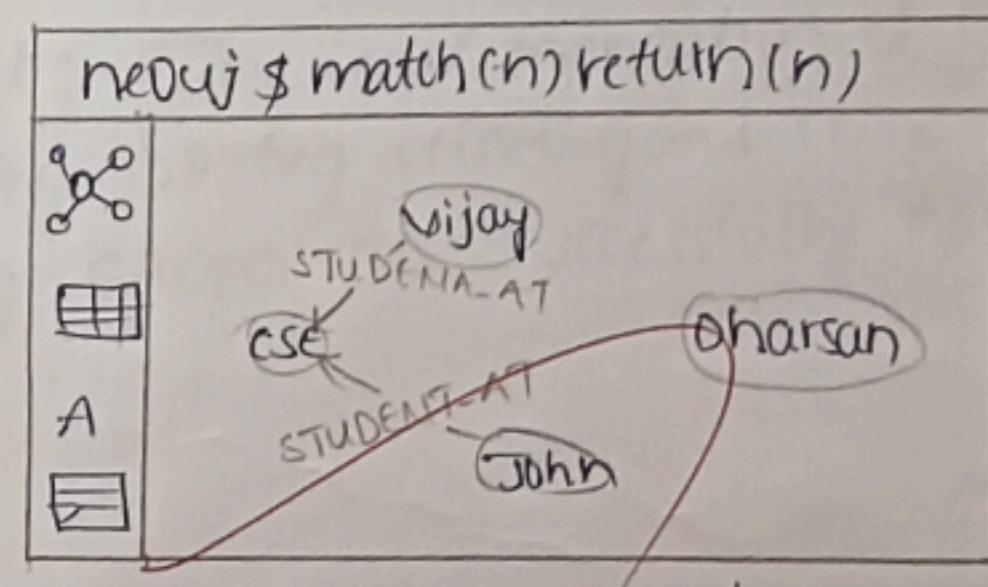
To delete a particular node, you need to specify the details of the node in the place of "n" in the above query.

```
MATCH (s:STUDENT),(d:dept) WHERE s.sname='John'
AND d.dept-name = 'CSE' CREATE (s)-[st:STUDIED-AT]->(d).
```

Output:-



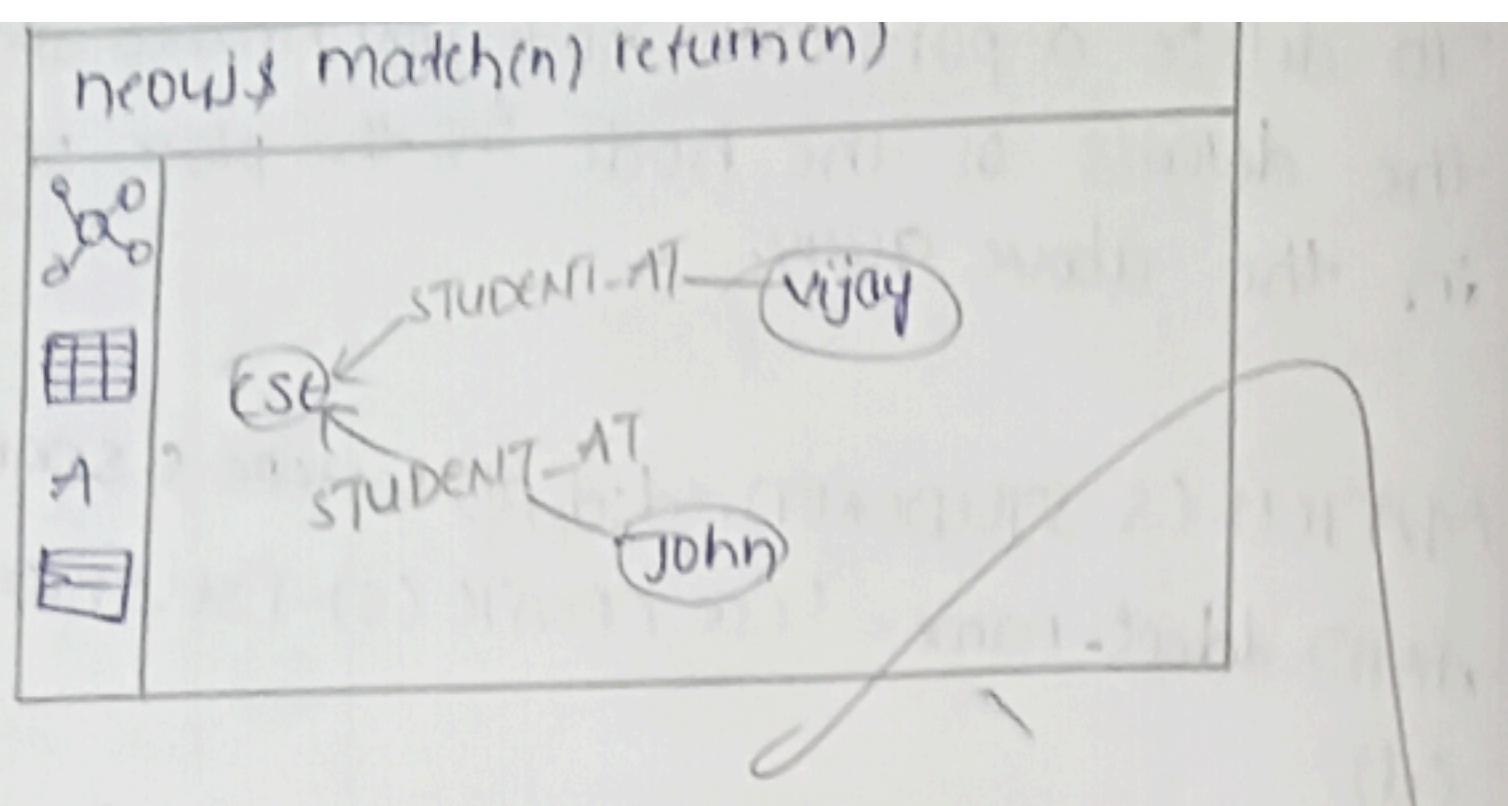
match(n) return(n)



b) Delete a node from Student.

```
match(n:student{S.name:'Dharsana'}) DELETE(n)
```

Output:- Delete 1 node completed after 10834 ms.



VELTECH	
EX No.	11
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	3
VIVA VOCE (3)	2
RECORD (4)	4
TOTAL (15)	15
SIGN WITH DATE	✓

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