

# Task - 11 CRUD Operations In Graph Databases

## Aim

To perform CRUD operations like creating, inserting, querying, finding, deleting operations on graph spreads.

### \* 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 specify these properties separated by commas within the flower braces "{}".

### 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 relations -hips 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

To create a relationship using the create clause.

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

\* Creating a relationship between the existing nodes (using match clause)

### Syntax

`MATCH (a:LabelOfNode1), (b:LabelOfNode2)`

`WHERE a.name = "nameOfNode1" AND`

`b.name = "nameOfNode2"`

`Create (a) - [:RelationshipType] → (b)`

Return a, b

\* Deleting a particular node

You need to specify the details of the node in the place of "n" in the above query.

### Syntax

`MATCH (node:label { properties ... })`

`DETACH DELETE node`

Create a graph database for student course registration, create student and dept node and insert value of properties.

Create (n: Student { id: "VTD14500",  
Sname : "John",  
deptname : "CSE"})

}

Output

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

Create (n: Student { id: "VTD14501",  
Sname : "Dharsana",  
deptname : "EEE"})

Output

Added 1 label, created 1 node, set 3 properties completed after 16 ms.

Create (n: student { id: "VTD14502",  
Sname : "Vijay",  
deptname : "CSE")

})

Select all the nodes in your database using match command

\* match (n) return (n)

Vijay

CSE

Dhark  
-ana

John

\* match (n: student) returns (n)

Output :

Vijay

Dhark  
-ana

John

a) Create relationship between student and CSE

Match (s: student), [d: dept] WHERE s.Sname = 'vijay' AND d.deptname = 'CSE'

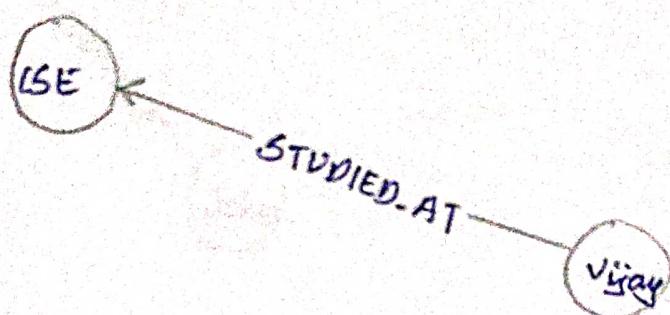
(Create (s)- [STUDIED\_AT] → (d))

return s,d

Output

match (s: Student), (d: dept) where s.sname = 'vijay' AND d.deptname = 'cse'

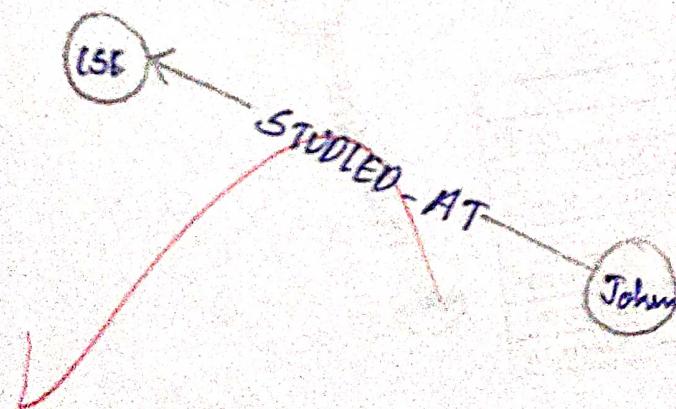
create (s) - [st: STUDIED\_AT] -> (d)  
return s,d



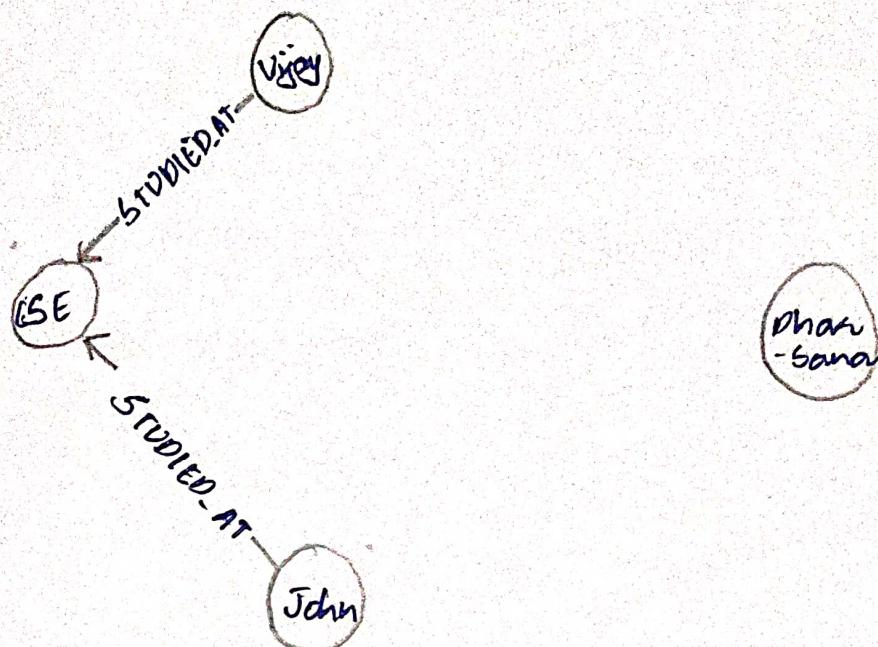
MATCH (s: Student), (d: dept) WHERE s.sname = 'JOHN' AND d.deptname = 'CSE'

CREATE (s) - [st: STUDIED\_AT] -> (d)  
RETURN s,d

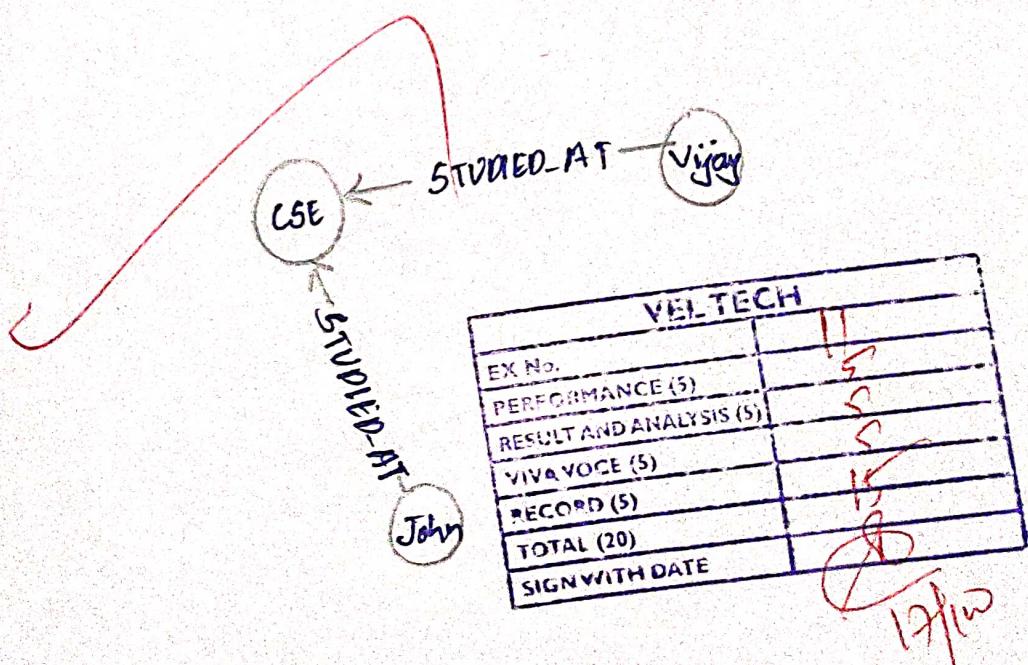
Output



match (n) return (n)



- b) Delete a node from student  
match (n: Student { Sname : 'Dhananjai' })  
Delete (n)



Result:

The implementation of LRD operations like creating, inserting, finding and removing operations using hash DB is successfully executed.