

task 11 — CRUD operations in Graph Database 16/10/25

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

* 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 those properties separated by commas within the 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 with in the square braces " {} "

Syntax:-

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

~~`CREATE (node1)-[:Relationship:Type]->(node2)`~~

* Creating a Relationship Between the Existing Nodes:- You can also create a relationship between the existing nodes using the MATCH clause.

"bEPGd9)P3OPPCG8P21bFGJ)" ls log : "b"
"DABHBT" u "mst."
28 - "utp"

Output

(Vijay)

(CSF)

(Dhan)

(John)

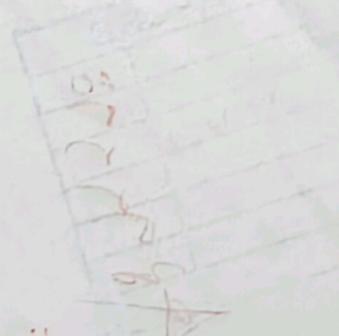
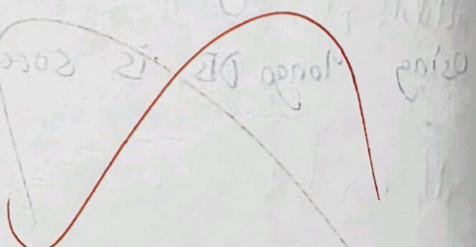
Output

(Vijay)

(Dhanshara)

(John)

creat logo (05/09/2017) - 125
visit logo. C John bao jenben, antsepi, pntok). olik
business 11.02.2018 21.210 agoal pntok



01/01

Syntax :-

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

```
MATCH (a:label of Node1), (b: label of Node2)
```

```
WHERE a.name = "name of node 1" AND b.name =  
"name of node 2".
```

```
CREATE (a)-[C: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 of properties) --- (n)
```

```
DETACH DELETE node;
```

* Create a graph database for student course registration
Create student and dept node and insert values of properties.

```
Create (n:student {id : "VTU14500"})
```

```
{name: "John"
```

```
deptname: "CSE"}
```

```
}
```

6th Period - 10:00 AM - 10:30 AM

CSE - Computer Organization - Dr. K. S. Jayaraman
CSCE - Computer Organization - Dr. K. S. Jayaraman
CSCE - Computer Organization - Dr. K. S. Jayaraman

CREATE - Computer Organization - Dr. K. S. Jayaraman
CREATE - Computer Organization - Dr. K. S. Jayaraman
CREATE - Computer Organization - Dr. K. S. Jayaraman

CREATE - Computer Organization - Dr. K. S. Jayaraman

CREATE - Computer Organization - Dr. K. S. Jayaraman

Output:-

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

Create.(n:student{sid:"VTU11501",
Sname:"Dhag Sana",
deptname:"EEE"})

Output:-

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

Create.(n:student{sid:"VTU11502",

Sname:"Vijay",

dept name:"CSE".

g,

Output:-

Added 1 label 1, created 1 node, set 3 properties,

Completed after 12 ms

Create.(n:dept{deptname:"SSE", deptid:"d001"})

Output:-

Added 1 label 1, created 1 node, set 2 properties

Completed After 72ms

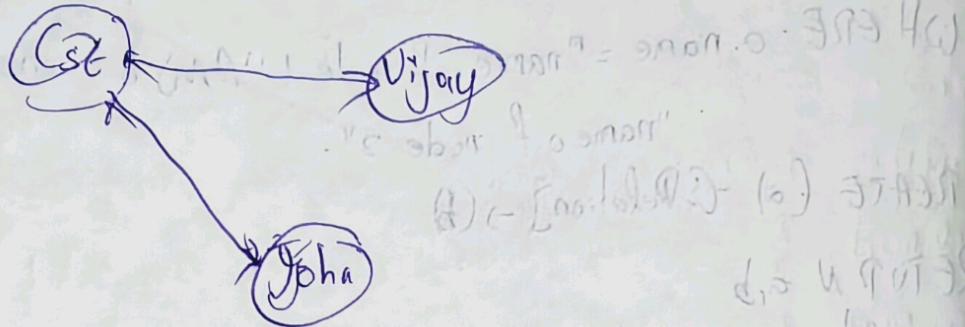
Select all the nodes in your database using match command

* match(n) .return(n).

* .match(n:student) .return(n)..

Output

(books) to school (l) (books). Good: P). HSTAM



(b) - [books] (l) (o) HSTAM

RC (L) (P)

RC (L) (P) (G)

To define a begin node a begin node will be defined
and in "n" no node will update its position with
space. Now

when visiting a node at time t it is visited, then

when it is visited at time t

(C - - visit[books] (l) (books)). HSTAM

books. HSTAM. EGI

not visit[books]. If visit[books] is false then

visit[books]. Then books have been visited

(books). books. books. books. books. books. books

"and": true

"or": false

a) Create relationship between student node cse:

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

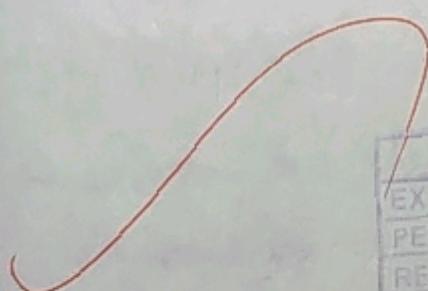
* match (n) return (n)

b) Delete a node from student

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

Output :- Delete 1 node, completed after 10834 ms

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



VEL TECH-CSE	
EX NO.	4
PERFORMANCE (5)	F
RESULT AND ANALYSIS (5)	F
VIVA VOCE (5)	F
CGPA (5)	7.0
Total (20)	14
DATE	14/10