Task 11: CRUD operation in Graph Database.

Aim:

To perform CRUD operation like Creating, inserting, querying, finding, deleting operations on graph space.

* Create Node with properties.

Properties are the key-value pairs using which a node store data. You can create a node with properties. Using the create clause, you need to specify three properties seperated by commas with the flower brackets "{}"

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 brackets "[]" Place btw hyphen "-" & arrow "->" as shown in the follow Syntax

Syntax

following is the syntax to create a relation ship btw the existing nodes using the match clause.

create (node1)-[:Relationship Type]->(node2)

* creating a Relationship Btw the Existing Nodes.

you can also create a relationship btw the existing nodes using the match clause.

Syntax

following in the syntax to create a relationship using the match.

match (node:label{properties.....})

DETACH DELETE node.

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

Create (n:student {sid: "VTU14500",

sname: "John"

deptname: "CSE"}

)

Output

Added 1 label, create 1 node, set 3 properties

create (n: student {sid: "VTU14501",

sname: "Dharasa"

deptname: "EEE"})

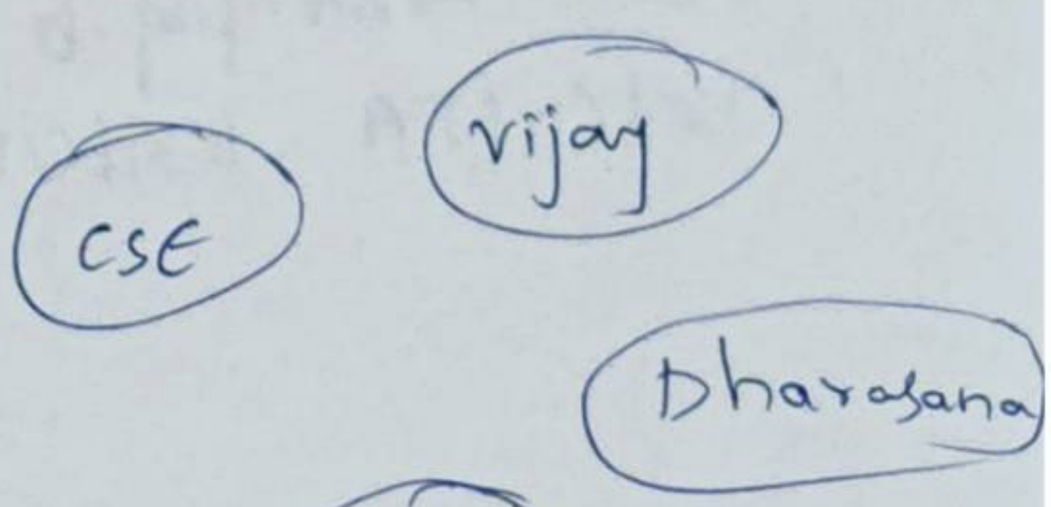
output

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

Select all the nodes in your database using match command.

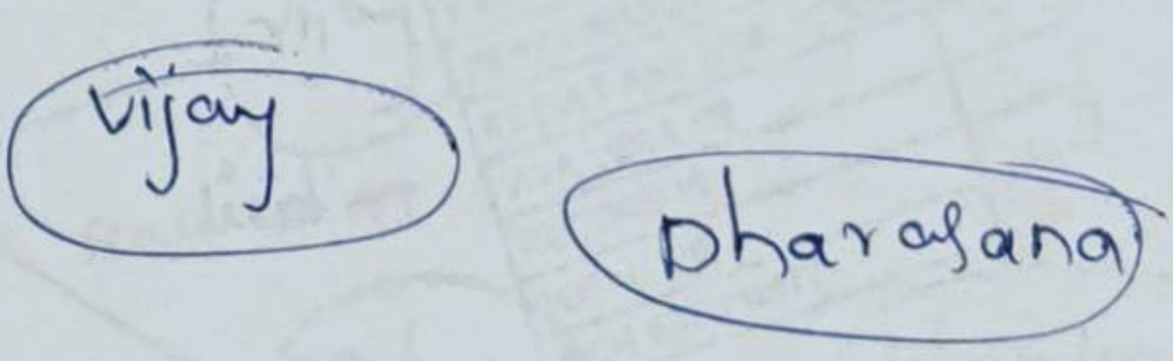
* match (n) return (n)

output



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

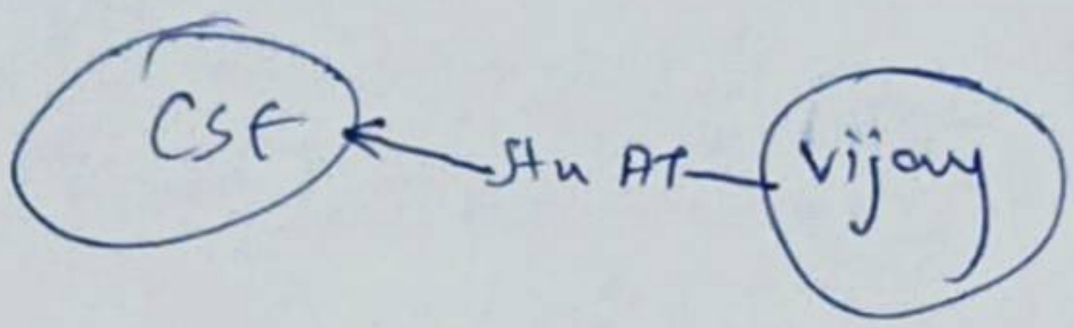
Output:



(a) Create relationship between Student & CSE

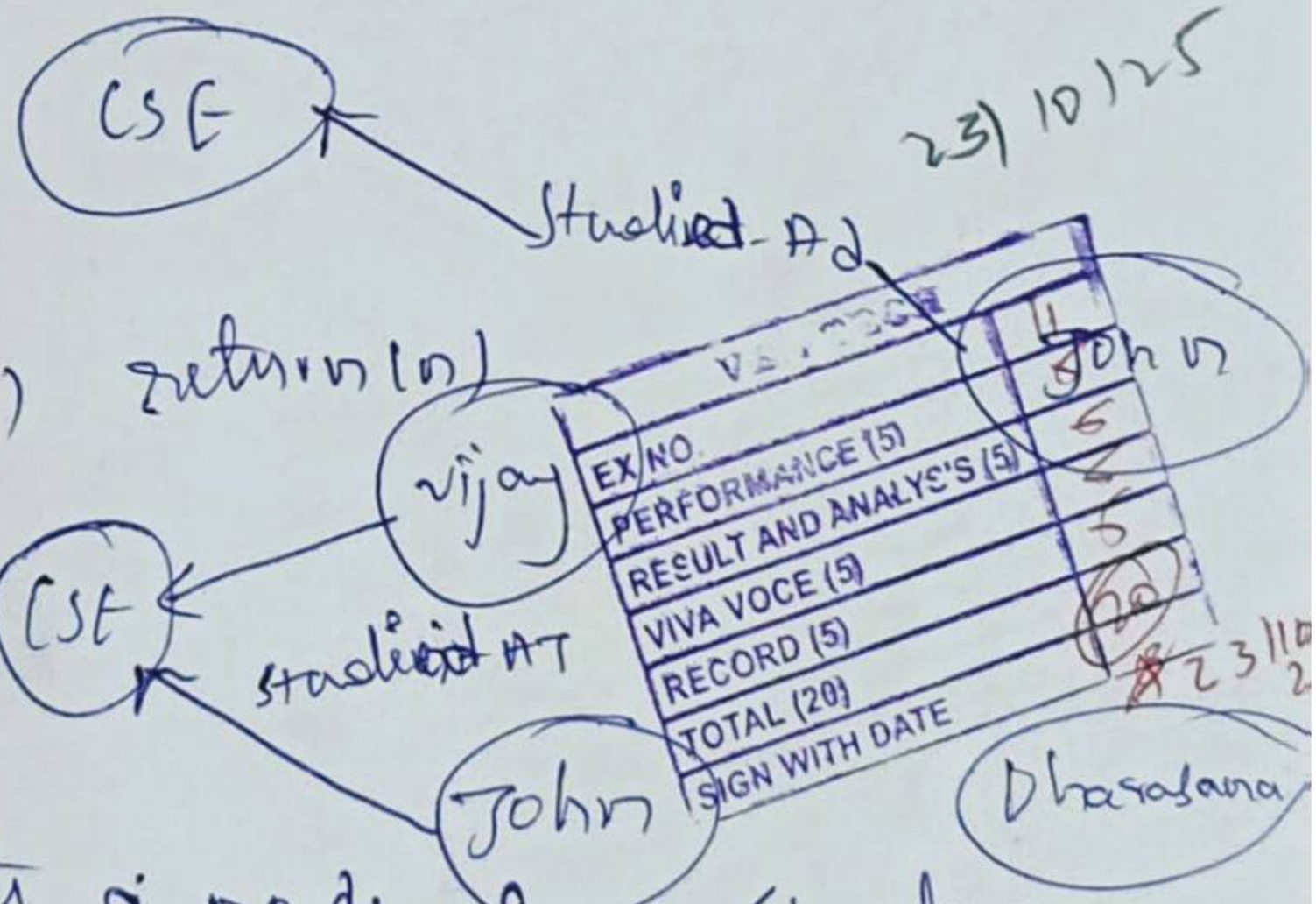
MATCH (s:student), (d:dept) WHERE s.sname = 'vijay' AND d.dptname = 'cse' create (s) -[st: STUDIED_AT]-> (d) return s, d.

output:



match (s: student), (d: dept) WHERE s.sname = 'John' and d.deptname = 'CSE'
Create (s)-[ST: STUDied - AT] ->(d)
return s,d.

Output



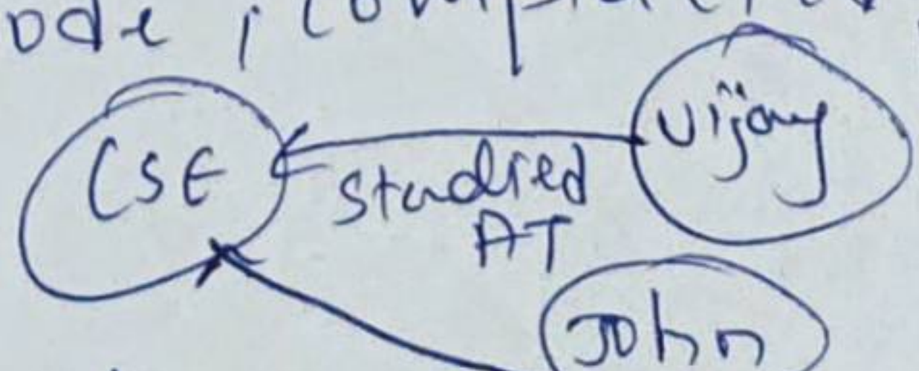
match(n) return(n)

b) Delete a node from student

match (n: student {s name: 'Dharsana'})
DELETE (n)

Output:

Delete 1 node, completed after 1083 ms



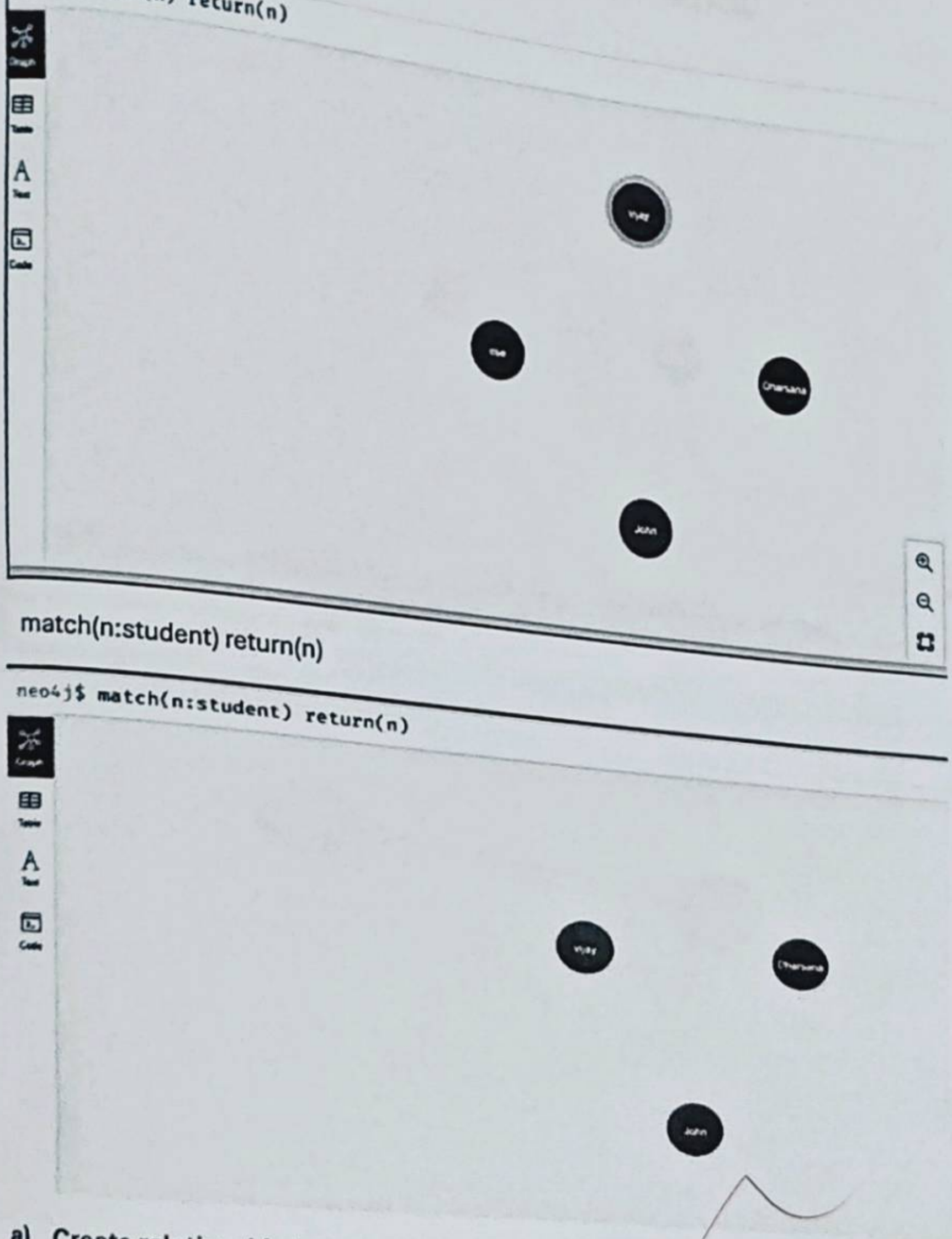
Result: The implementation of CRUD operation like creating, inserting, finding & removing operation is successfully executed

Ex: 11 CRUD operation in Graph database.

Select all the nodes in your database using match command.

match(n) return(n)

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

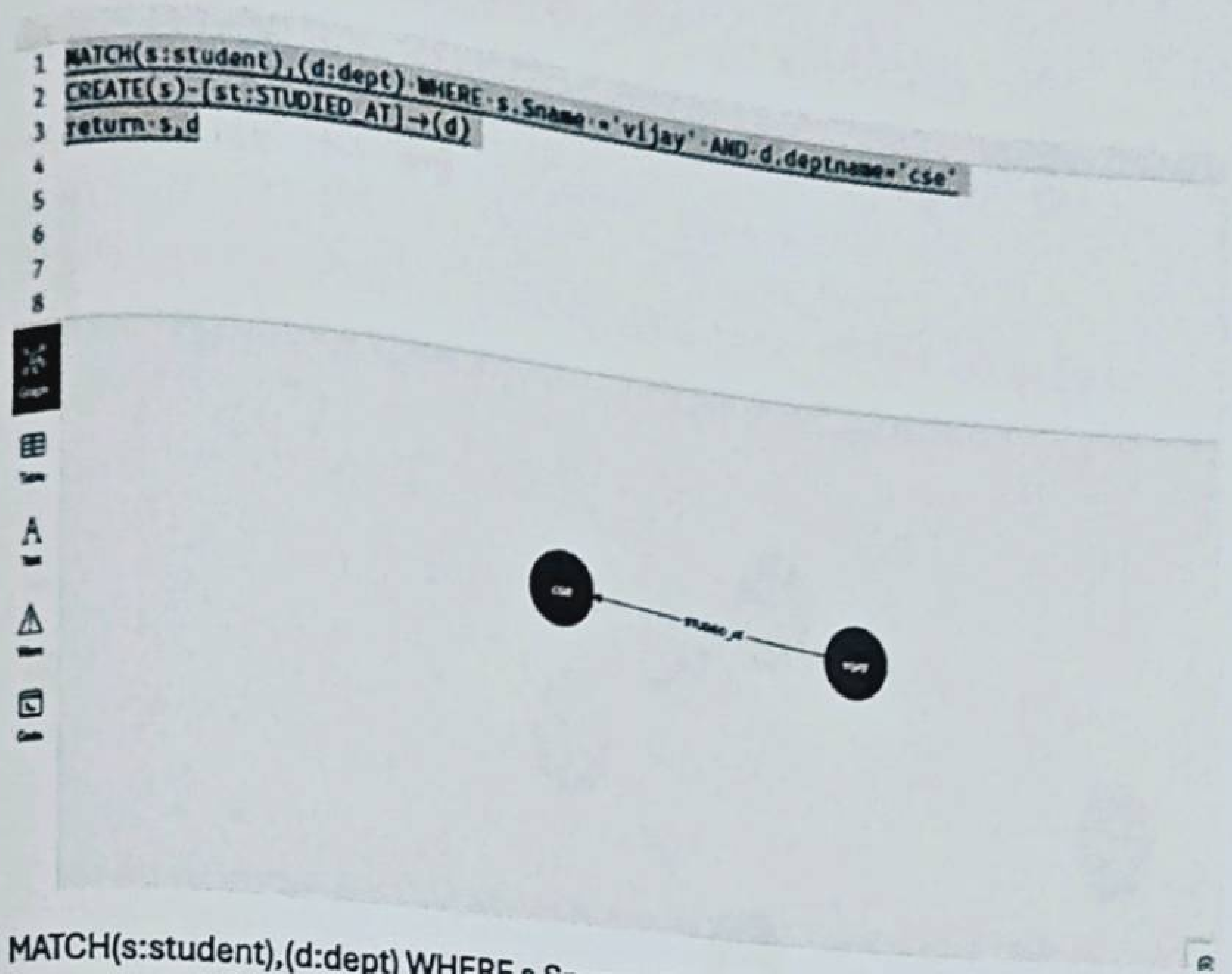


match(n:student) return(n)

```
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='cse'
CREATE(s)-[st:STUDIED_AT]->(d)
return s,d
```



MATCH(s:student),(d:dept) WHERE s.Sname ='John' AND d.deptname='cse'

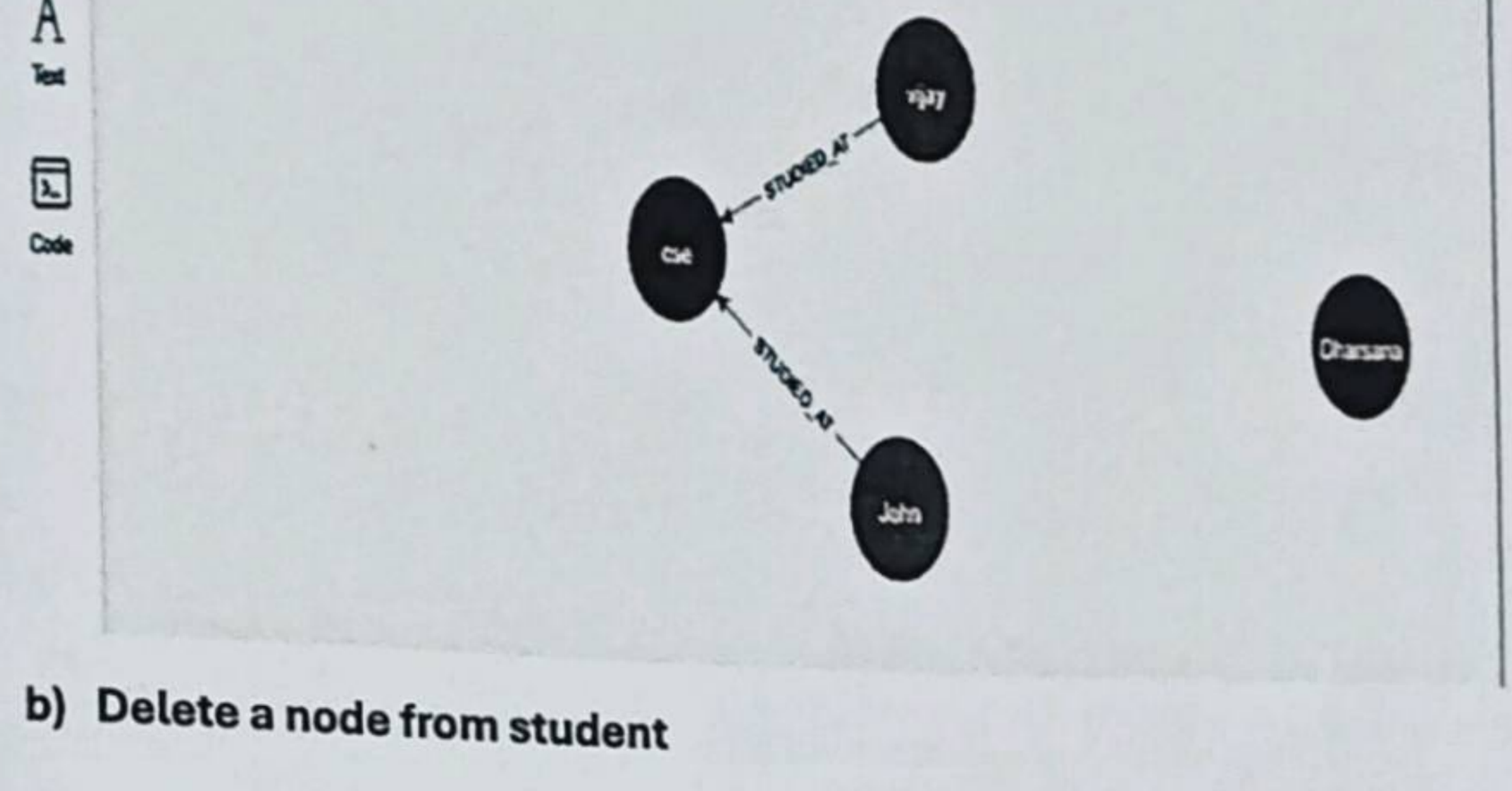
CREATE(s)-[st:STUDIED_AT]->(d)

return s,d

shown in the follow Synta

match(n) return(n)

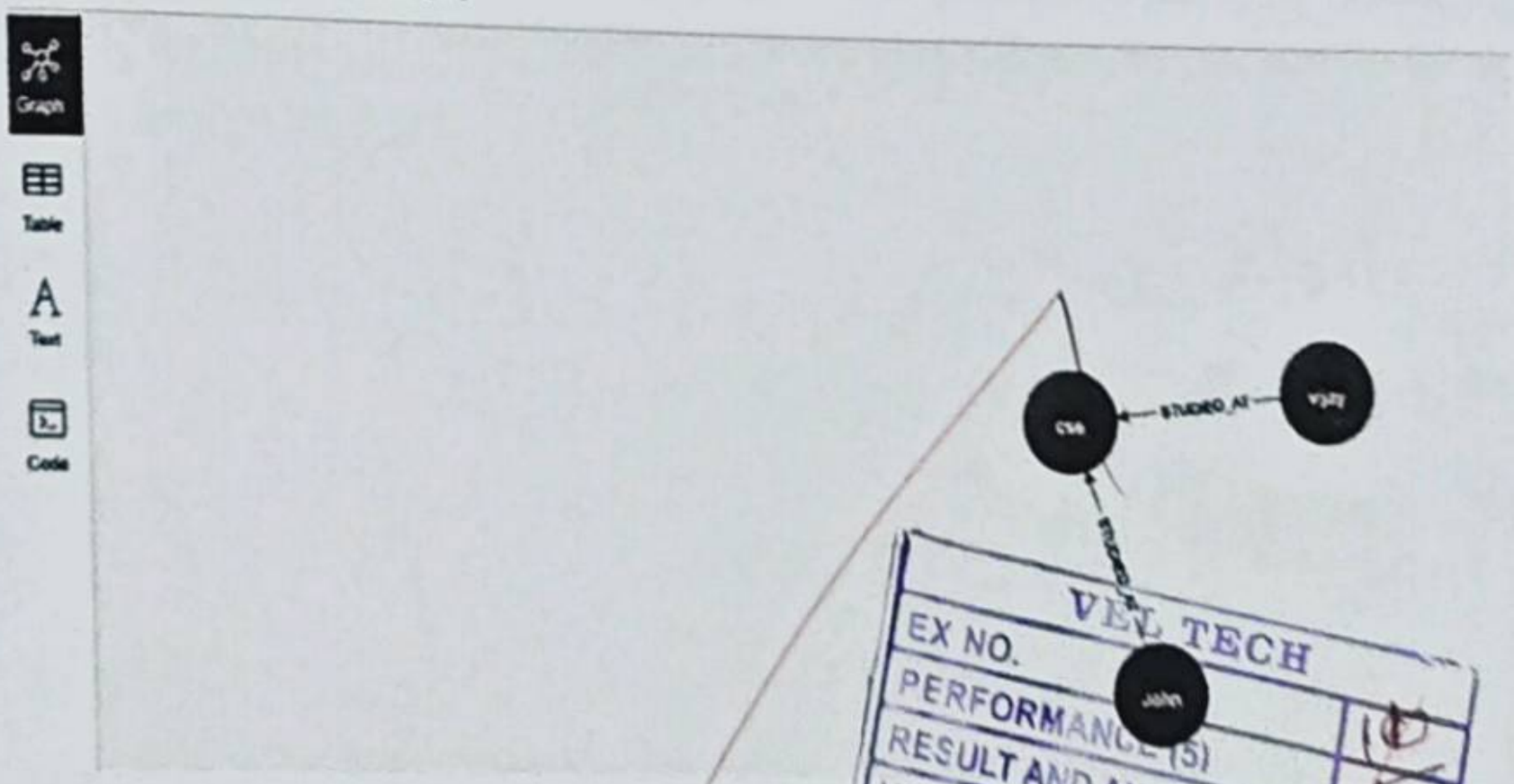
```
neo4j$ match(n) return(n)
```



b) Delete a node from student

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

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



Result: The the program was executed.

VEL TECH

EX NO.	
PERFORMANCE (5)	
RESULT AND ANALYSIS (5)	
VIVA VOCE (5)	
RECORD (5)	
TOTAL (20)	
SIGN WITH DATE	