Task 11: CRUD OPERATIONS IN GRAPH DATABASES

Aim: To implement the crud operations in graph database.

Create node with properties

Properties are key value pairs using which a node stores data. You can create a node with properties using CREATE clause.

Syntax

Following:

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

Returning the created Node

To verify creation the node type and

Execute the MATCH (n) RETURN n

Create Relationships

"[ ]" starts depending on direction to hyphen '-' and arrow "→".

Syntax:

CREATE (node1)-[:RelationshipType]->(node2)

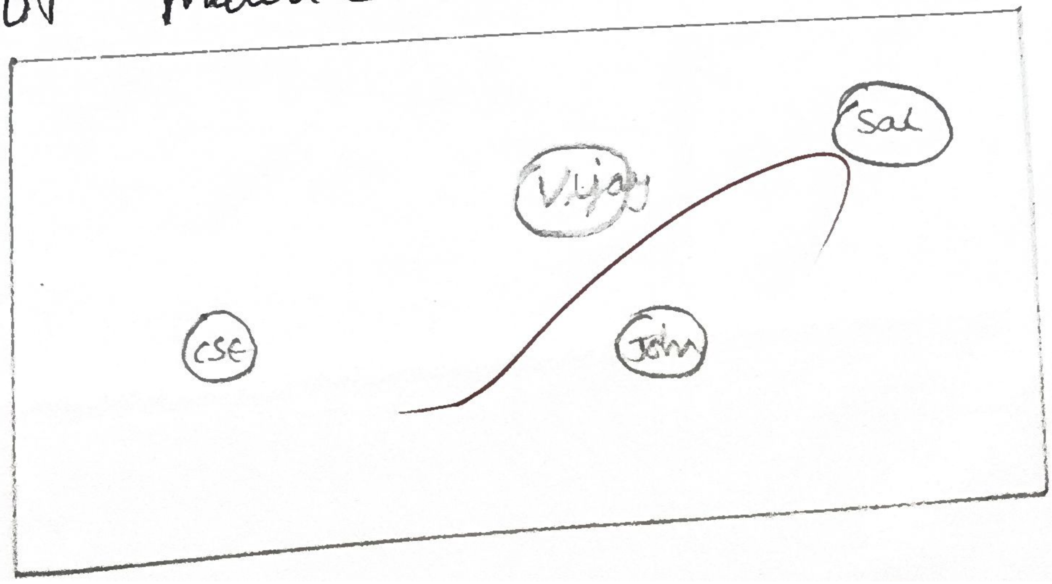
a Relationship between the existing

... (node1), (b: label {Node2})

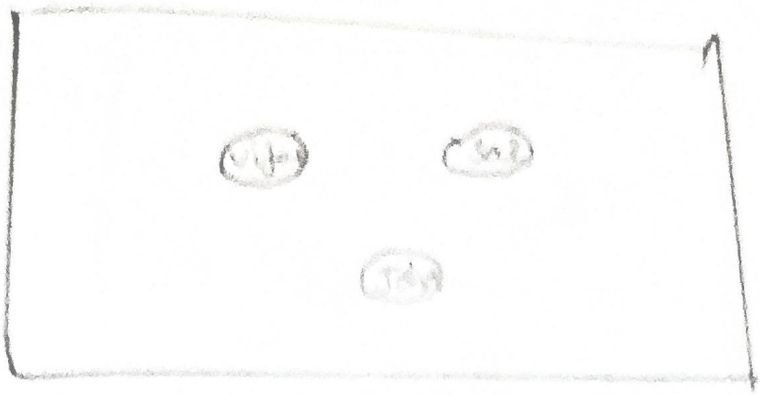
node ...

where (a: labelofNode 1), (b: labelof Node 2)
WHERE a. name = "name of node1" AND b.name
= "nameofnode 2"
return a, b.

OUTPUT
Added 1 label, created 1 node, set 3 properties
completed after 232 ms.
create (n: student {sId: "VTU 29364";
sname: "yaswanth",
deptname: "EEE"})
output:
Added 1 label, created 1 node, set 3 properties,
completed after 12 ms.
create (n: dept {deptname: "cse", deptId: "d001"})
output:
Added 1 label, created 1 node, set 2 properties,
completed after 72 ms.
select all the nodes in your database
OUTPUT match (n) return (n)



match (n:students) return (n)



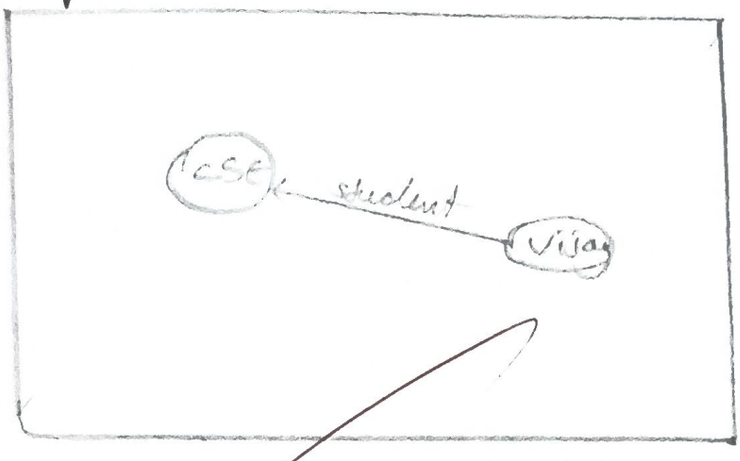
a) create relationship

MATCH (s:student), (d:dept) where s.sname = 'vijay' AND d.deptname = 'CSE'.

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

return s, d

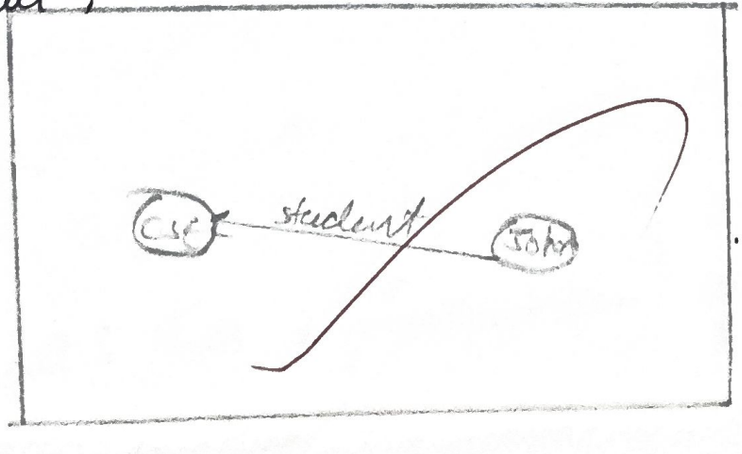
output:



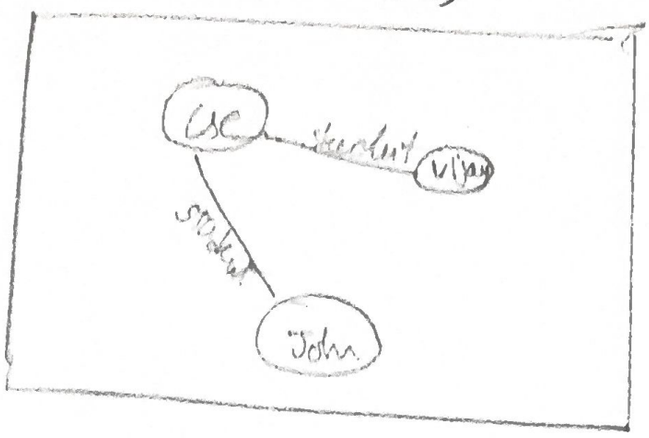
Match (s:student), (d:dept) WHERE s.sname = 'John' AND d.deptname = 'CSE'.

return s, d.

output:



match (n) return (n)



deleted 1 node, completed after 10834 ms.

VEL TECH	
EX NO.	5
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	15
TOTAL (30)	

14/3/25

Result: Thus implementation of read operations like creating, inserting, removing using graph DB in Neo4j executed successfully.