

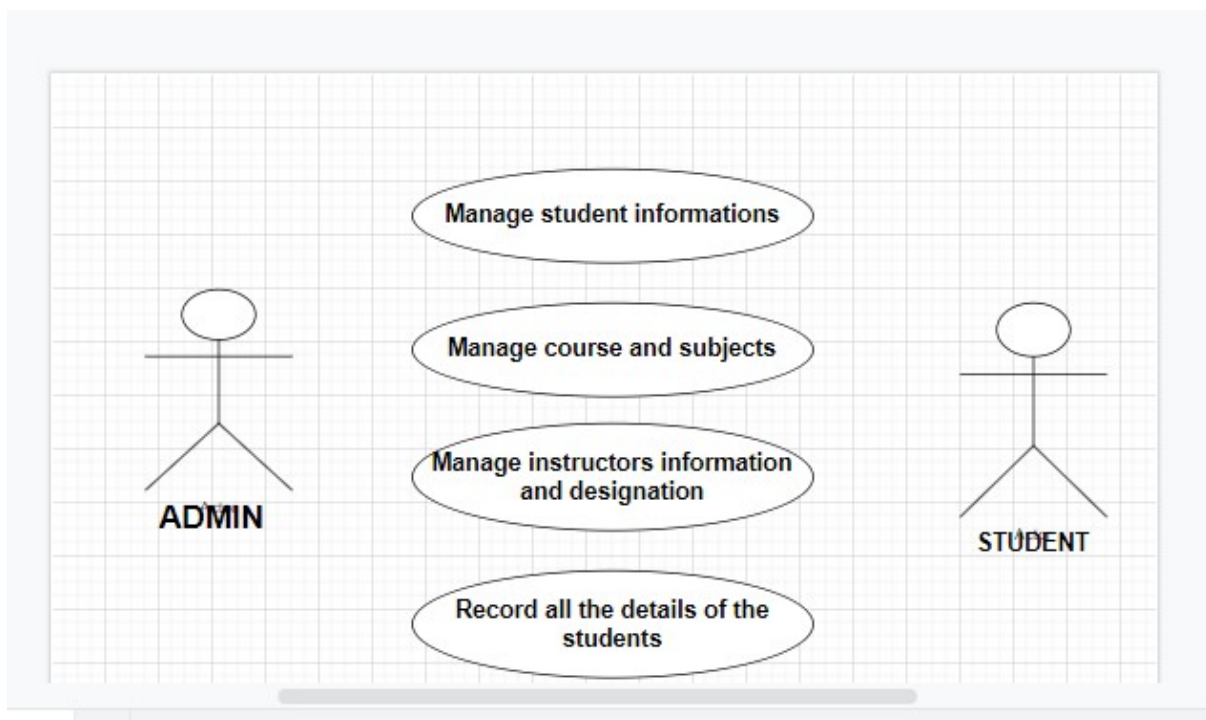
Neo4j DATABASE

UNIVERSITY DATABASE

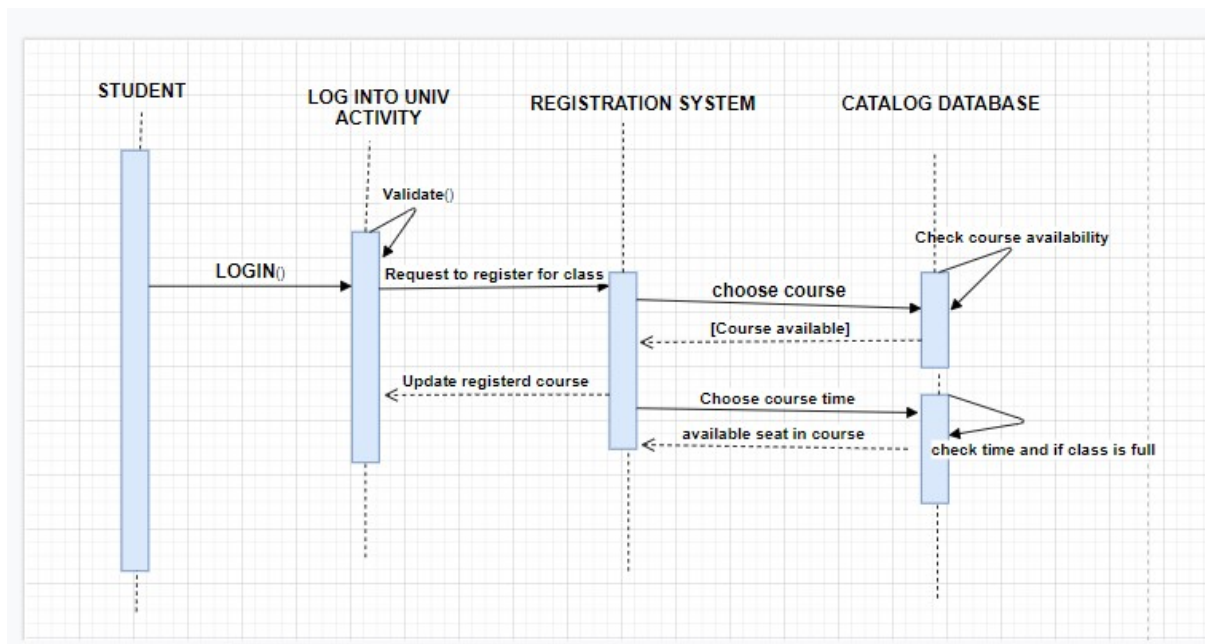
Koduri Gokul-19BCD7006

B.ARSHITH GOPAL-19BCE7402

USECASE DIAGRAM:



SEQUENCE DIAGRAM:



LIST OF QUERIES:

- 1.CREATE A NODE A?
- 2.CREATE A NODE B?
- 3.CREATE A NODE C?
- 4.CREATING A NODE D?
- 5.CREATING A RELATION?
- 6.RETRIEVING A PARTICULAR RELATION?
- 7.DELETE A RELATION?
- 8.CREATING AN INDEX?
- 9.TO DISPLAY ALL THE AVAILABLE INDEXES IN SCHEMAS?
- 10.INDEX HINTS?
- 11.CREATE CONSTRAINT?
- 12.TO RETRIEVE A PARTICULAR DATA
- 13.TO RETRIEVE A RELATIONSHIP?
- 14.DROPPING INDEX
- 15.DROPPING CONSTRAINT
- 16.TO DELETE A NODE?

17. TO DELETE A NODE IN RELATION?

18. TO DELETE A RELATION?

19. TO DISPLAY ALL THE STUDENT DETAILS:

20. TO DISPLAY ALL THE COURSES AND NAME OF STUDENTS:

21. TO DISPLAY ALL THE STUDENTS OF MTSE or MECH:

22. TO DISPLAY ALL THE STUDENTS OF MTSE AND ECE:

23. TO DISPLAY ALL THE NODES NAMES IN THE GRAPH IN ASCENDING ORDER:

24. TO FIND THE COURSE FROM STUDENT ID:

25. TO DISPLAY ALL THE DEPARTMENT NAMES:

26. TO DISPLAY THE STUDENT IDS WHO ARE STUDYING IN NOSQL COURSE:

QUERIES:

1. CREATING NODE A

```
CREATE (a:studentdetails { id:"7166", name : "prasanna", address : "Guntur",  
phone:"6300199052", course:"DBA", slot:"A", dept:"MTSE", age:"21" })  
return a
```



2. CREATING NODE B

```
CREATE (b:studentdetails { id:"7052", name : "anu", address : "vijaywada",  
phone:"6765764847", course:"SPM", slot:"C", dept:"ECE", age:"20" })  
return b
```



3. CREATING NODE C

```
CREATE (c:studentdetails { id:"7197" ,name : "BINCY", address : "HYDERABAD",
phone:"9177693023", course:"ML", slot:"B", dept:"MECH",age:"21" })
return c
```



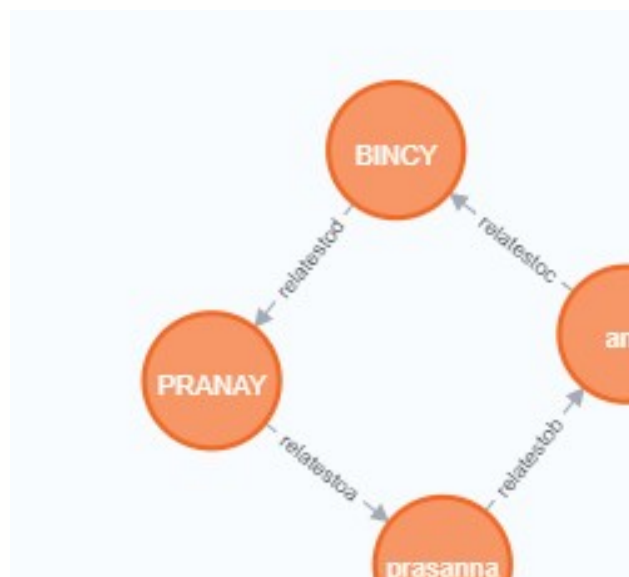
4.CREATING NODE D

```
CREATE (d:studentdetails { id:"7207" ,name : "PRANAY", address : "JAGTIAL",
phone:"9618040926", course:"NOSQL", slot:"D", dept:"CSE",age:"21" })
return d
```



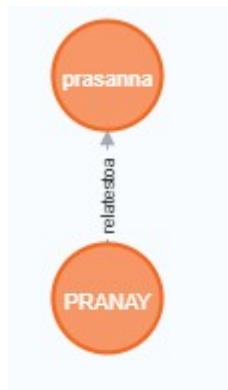
5.CREATING A RELATION

```
MATCH (a:studentdetails),(b:studentdetails),(c:studentdetails),(d:studentdetails)
WHERE a.id = "7166" AND b.id = "7052" AND c.id="7197" AND d.id="7207"
CREATE (a)-[relo:relatestob]->(b),(b)-[relt:relatestoc]->(c),(c)-[relh:relatestod]->(d),(d)-
[relf:relatestoa]->(a)
RETURN a,b,c,d
```

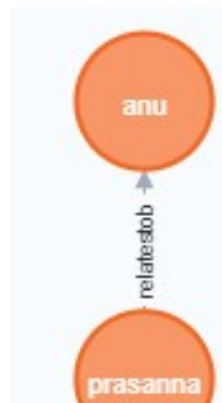


6.RETREIVING A PARTICULAR RELATION:

```
MATCH p=()-[r:relatestoa]->() RETURN p LIMIT 25
```



`MATCH p()-[r:relatestob]->>() RETURN p LIMIT 25`



`MATCH p()-[r:relatestoc]->>() RETURN p LIMIT 25`



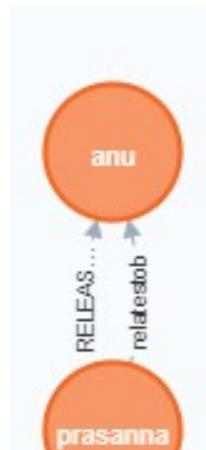
`MATCH p()-[r:relatestod]->>() RETURN p LIMIT 25`



7.DELETE A RELATION

```

MATCH (a:studentdetails),(b:studentdetails)
WHERE a.id = "7166" AND b.id = "7052"
CREATE (a)-[r:RELEASED]->(b)
RETURN a,b
  
```



```

MATCH (:studentdetails)-[r:RELEASED]-(:studentdetails)
DELETE r
  
```

8.CREATING AN INDEX:

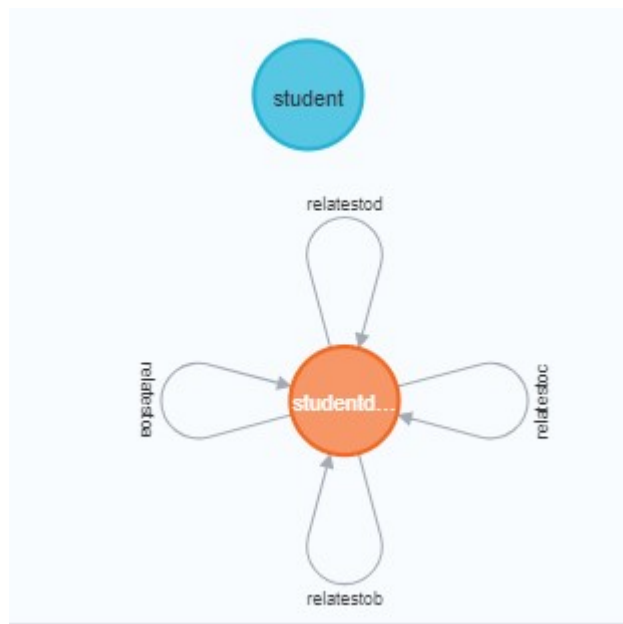
```

CREATE INDEX ON :student(id)
  
```

9.TO DISPLAY ALL THE SCHEMAS IN AVAILABLE INDEX :schema

Index Name	Type	Uniqueness	EntityType	LabelsOrTypes	Prop
index_343aff4e	LOOKUP	NONUNIQUE	NODE	[]	[]
index_aa360668	BTREE	NONUNIQUE	NODE	["student"]	["id"]
index_f7700477	LOOKUP	NONUNIQUE	RELATIONSHIP	[]	[]
Constraints					

CALL db.schema.visualization



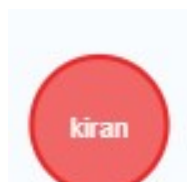
10. INDEX HINTS:

```
MATCH (a:studentdetails {id: "7166"})  
USING INDEX a:studentdetails(id)  
RETURN a
```

<div>Graph</div> <div>Table</div> <div>Text</div> <div>Code</div>	<div>a</div> <pre>{ "identity": 0, "labels": ["studentdetails"], "properties": { "address": "Guntur", "phone": "6300199052", "name": "prasanna", "course": "DBA", "id": "7166", "dept": "MTSE", "slot": "A",</pre>
---	--

11.CREATE A CONSTRAINT;

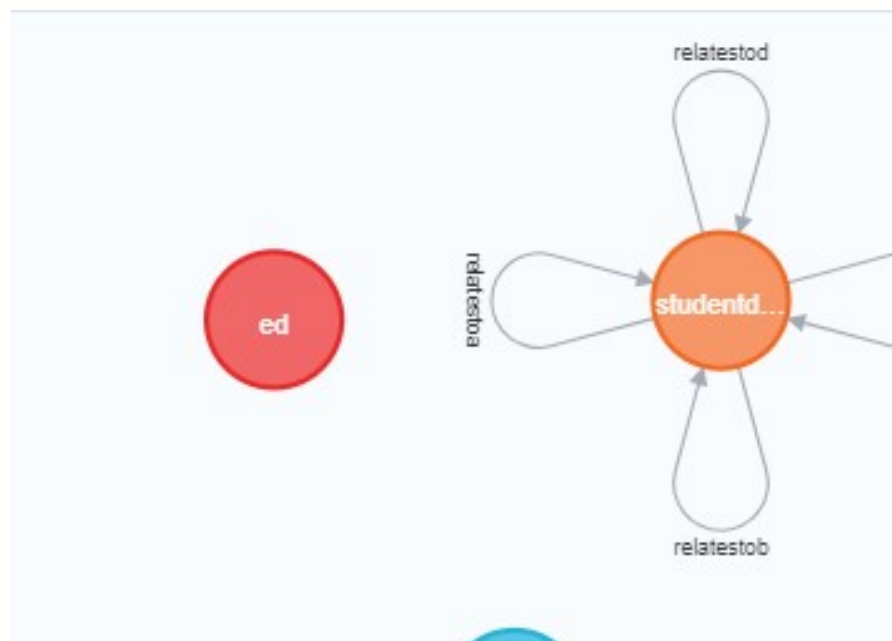
```
CREATE (e:ed { id:"7237",name : "kiran", address : "srikakulam",
phone:"6300199053", course:"MAD", slot:"G", dept:"BBA", age:"21"}) return e
```



```
CREATE CONSTRAINT ON (e:ed) ASSERT e.id IS UNIQUE
:schema
```


Index Name	Type	Uniqueness	EntityType	LabelsOrTypes	Pro
constraint_bbba1aa	BTREE	UNIQUE	NODE	["ed"]	["i
index_343aff4e	LOOKUP	NONUNIQUE	NODE	[]	[]
index_aa360668	BTREE	NONUNIQUE	NODE	["student"]	["i
index_f7700477	LOOKUP	NONUNIQUE	RELATIONSHIP	[]	[]

CALL db.schema.visualization



```
CREATE (e:ed {id: "7207"})
RETURN e
```

ERROR Neo.ClientError.Schema.ConstraintValidationFailed

Node(5) already exists with label `ed` and property `id` = '7207'

12.TO RETRIEVE A PARTICULAR DATA:

```
MATCH (a:studentdetails)
WHERE a.id = "7166"
RETURN a
```



```
MATCH (b:studentdetails)
WHERE b.id = "7052"
RETURN b
```



13.TO RETREIVE A RELATIONSHIP

```
MATCH (a:studentdetails)-[:relatestob]->(b:studentdetails)
WHERE b.name = "anu"
RETURN a
```

```
{
  "identity": 0,
  "labels": [
    "studentdetails"
  ],
  "properties": {
    "address": "Guntur",
    "phone": "6300199052",
    "name": "prasanna",
    "course": "DBA",
    "id": "7166",
    "dept": "MTSE",
    "slot": "A",
    "age": "21"
  }
}
```

MATCH (n) RETURN n

```
{
  "identity": 0,
  "labels": [
    "studentdetails"
  ],
  "properties": {
    "address": "Guntur",
    "phone": "6300199052",
    "name": "prasanna",
    "course": "DBA",
    "id": "7166",
    "dept": "MTSE",
    "slot": "A",
    "age": "21"
  }
}
```

```
{
  "identity": 1,
  "labels": [
    "studentdetails"
  ],
  "properties": {
    "address": "vijaywada",
    "phone": "6765764847",
    "name": "anu",
    "course": "SPM",
    "id": "7052",
    "dept": "ECE",
    "slot": "C",
    "age": "20"
  }
}
```

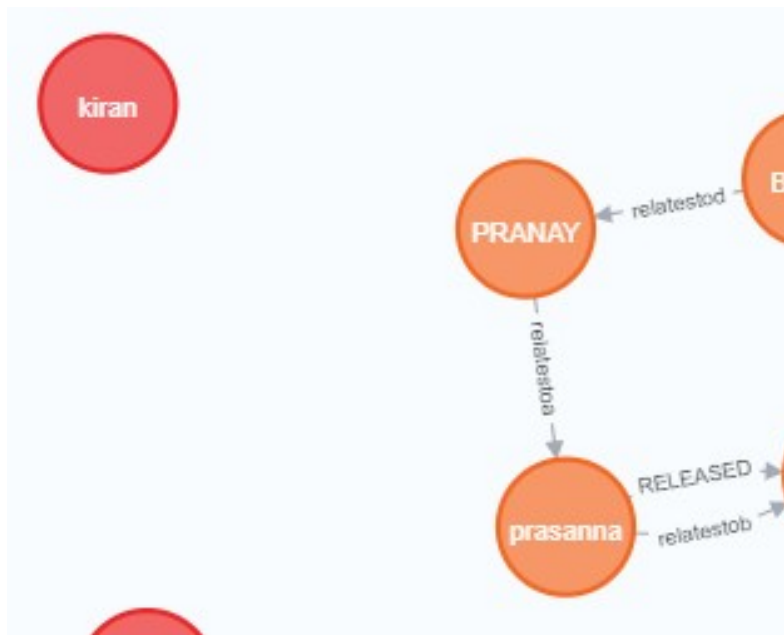
```
{
  "identity": 2,
  "labels": [
    "studentdetails"
  ],
  "properties": {
    "address": "HYDERABAD",
    "phone": "9177693023",
    "name": "BINCY",
    "course": "ML",
    "id": "7197",
    "dept": "MECH",
  }
}
```

```
"slot": "B",  
"age": "21"  
}  
}
```

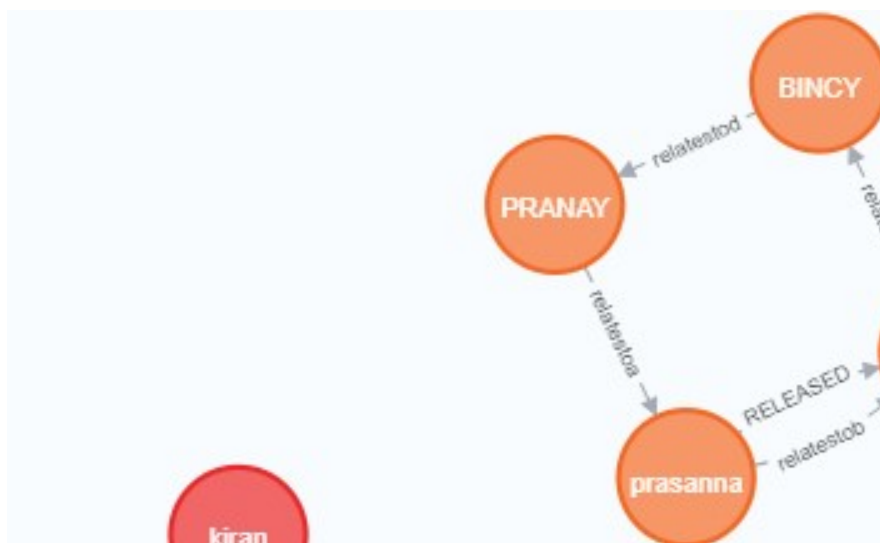
```
{  
  "identity": 3,  
  "labels": [  
    "studentdetails"  
  ],  
  "properties": {  
    "address": "JAGTIAL",  
    "phone": "9618040926",  
    "name": "PRANAY",  
    "course": "NOSQL",  
    "id": "7207",  
    "dept": "CSE",  
    "slot": "D",  
    "age": "21"  
  }  
}
```

```
{  
  "identity": 4,  
  "labels": [  
    "ed"  
  ],  
  "properties": {  
    "address": "srikakulam",  
    "phone": "6300199053",  
    "name": "kiran",  
    "course": "MAD",  
    "id": "7237",  
    "dept": "BBA",  
    "slot": "G",  
    "age": "21"  
  }  
}
```

```
{  
  "identity": 5,  
  "labels": [  
    "ed"  
  ],  
  "properties": {  
    "id": "7207"  
  }  
}
```



MATCH (n) RETURN n
LIMIT 5



14.DROPPING INDEX

DROP INDEX ON :student(id)

15.DROPPING CONSTRAINT;

DROP CONSTRAINT ON (e:ed) ASSERT e.id IS UNIQUE

16.TO DELETE A NODE IN RELATION:

MATCH (c:studentdetails {id: "7197"}) **DETACH DELETE** c

17.DELETE A NODE:

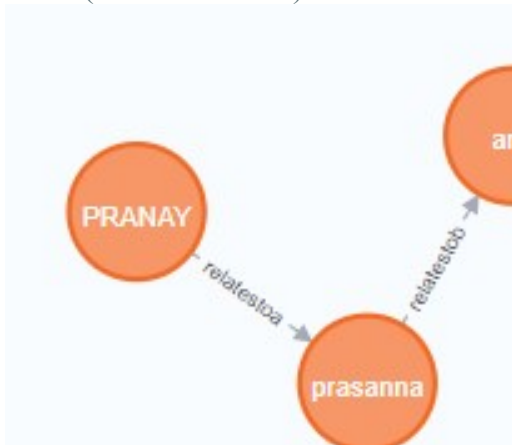
MATCH (c:studentdetails {id: "7197"}) **DELETE** c

18.TO DELETE A RELATION:

MATCH (:studentdetails)-[r:RELEASED]-(:studentdetails)
DELETE r

19.TO DISPLAY ALL THE STUDENT DETAILS:

match(a:studentdetails) **return** a



20.TO DISPLAY ALL THE COURSES AND NAME OF STUDENTS:

MATCH(n:studentdetails) **return** n.course,n.name

	n.course	n.name
1	"DBA"	"prasanna"
2	"SPM"	"anu"
3	"NOSQL"	"PRANAY"

21.TO DISPLAY ALL THE STUDENTS OF MTSE or MECH:

MATCH(n:studentdetails) **where**(n.dept="MTSE" **AND** n.name="prasanna") **or** (n.dept="MECH" **AND** n.name="BINCY") **return** n.name,n.dept

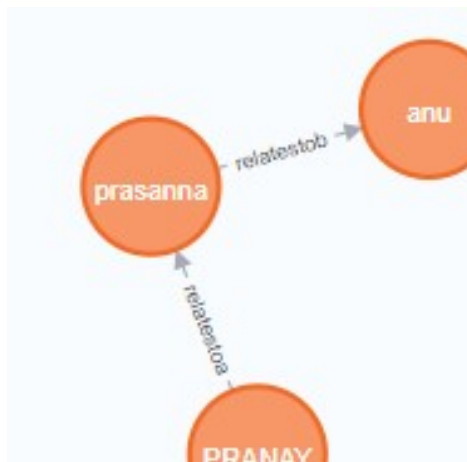
n.name	
1	"prasanna"

22.TO DISPLAY ALL THE STUDENTS OF MTSE AND ECE:

MATCH(n:studentdetails) **where**(n.dept="MTSE" **AND** n.name="prasanna") **AND** (n.dept="MECH" **AND** n.name="BINCY") **return** n.name,n.dept

23.TO DISPLAY ALL THE NODES NAMES IN THE GRAPH IN ASCENDING ORDER:

MATCH(n:studentdetails) **return** n **ORDER BY** n.id



"n"
{ "address": "vijaywada", "phone": "6765764847", "name": "anu", "id": "7052", "dept": "ECE", "slot": "C", "age": "20" }
{ "address": "Guntur", "phone": "6300199052", "name": "prasanna", "id": "7166", "dept": "MTSE", "slot": "A", "age": "21" }

24.TO FIND THE COURSE FROM STUDENT ID:

MATCH (n:studentdetails) **where** n.id="7207" **return** n.course

n.course
"NOSOI "

25.TO DISPLAY ALL THE DEPARTMENT NAMES:

MATCH (n:studentdetails) **return** n.dept

n.dept	
1	"MTSE"
2	"ECE"

26.TO DISPLAY THE STUDENT IDS WHO ARE STUDYING IN NOSQL COURSE:

MATCH (n:studentdetails {course:"**NOSQL**"}) **return** n.id

"n.id"
"7207"