

DBMS LAB 10

NAME: SUBHAM DEY

SRN: PES1UG23CS604

```
■ free instance/neo4j // 4. Show student enrollments MATCH (a:Student {name: 'Alice'}), (cs:Course {code: 'CS101'})
Connect to Instance

✓ // 4. Show student enrollments
MATCH (a:Student {name: 'Alice'}), (cs:Course {code: 'CS101'})
CREATE (a)-[:ENROLLED_IN]→(cs);

Created 1 relationship Fetched 0 records

✓ MATCH (b:Student {name: 'Bob'}), (ec:Course {code: 'EC202'})
CREATE (b)-[:ENROLLED_IN]→(ec);

Created 1 relationship Fetched 0 records

✓ // 5. Show professor teaching assignments
MATCH (s:Professor {name: 'Dr. Smith'}), (cs:Course {code: 'CS101'})
CREATE (s)-[:TEACHES]→(cs);

Created 1 relationship Fetched 0 records

✓ MATCH (j:Professor {name: 'Dr. Jones'}), (ec:Course {code: 'EC202'})
CREATE (j)-[:TEACHES]→(ec);

Created 1 relationship Fetched 0 records

✓ // 6. Create friendship
MATCH (a:Student {name: 'Alice'}), (c:Student {name: 'Charlie'})
CREATE (a)-[:FRIENDS_WITH]→(c);

Created 1 relationship Fetched 0 records
```

```
■ free instance/neo4j // 1. Create Student Nodes CREATE (a:Student {name: 'Alice', age: 21, major: 'CSE'}), (
Connect to Instance

✓ // 1. Create Student Nodes
CREATE (a:Student {name: 'Alice', age: 21, major: 'CSE'}),
(b:Student {name: 'Bob', age: 22, major: 'ECE'}),
(c:Student {name: 'Charlie', age: 20, major: 'CSE'});

Created 3 nodes, set 9 properties, added 3 labels Fetched 0 records

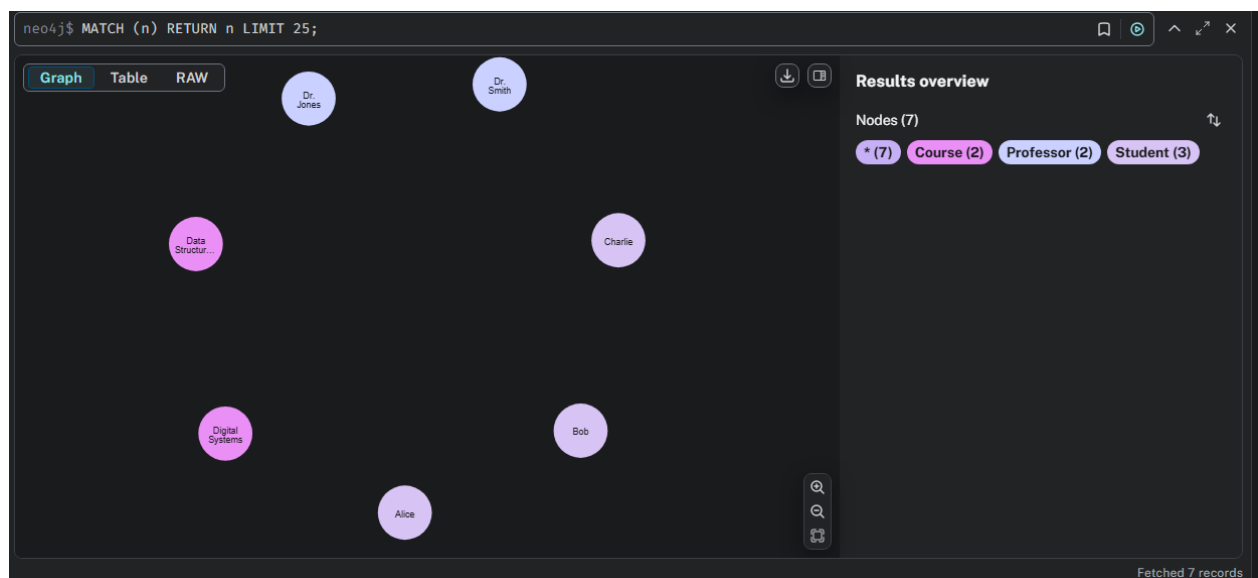
✓ // 2. Create Professor Nodes
CREATE (s:Professor {name: 'Dr. Smith', department: 'CSE'}),
(j:Professor {name: 'Dr. Jones', department: 'ECE'});

Created 2 nodes, set 4 properties, added 2 labels Fetched 0 records

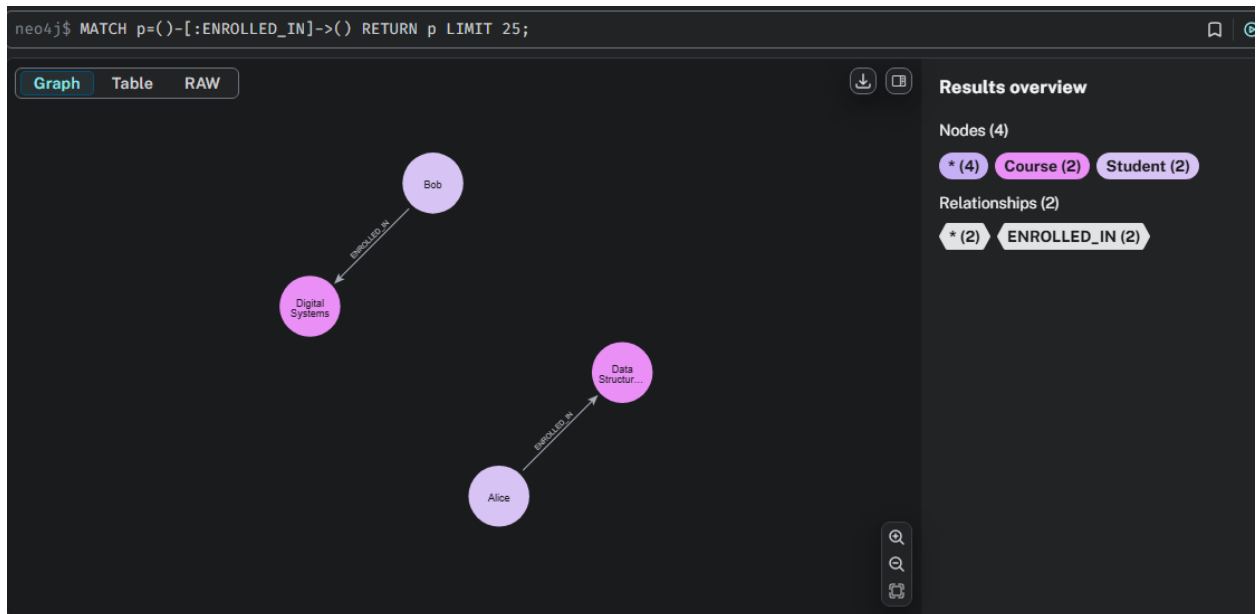
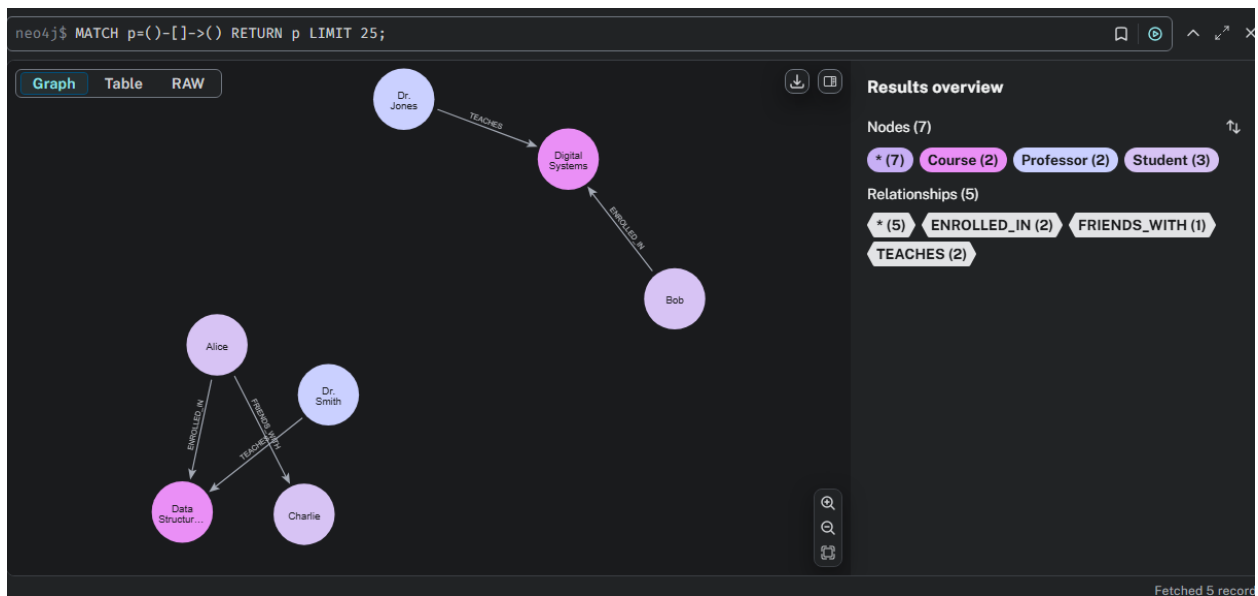
✓ // 3. Create Course Nodes
CREATE (cs:Course {code: 'CS101', name: 'Data Structures'}),
(ec:Course {code: 'EC202', name: 'Digital Systems'});

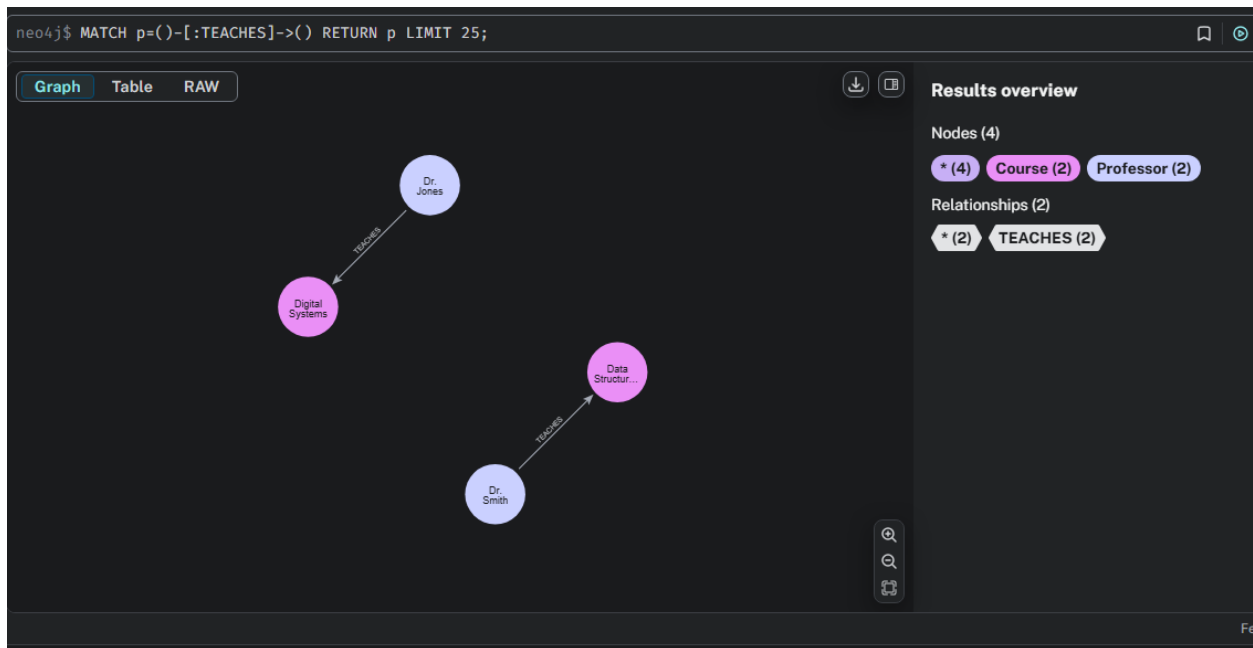
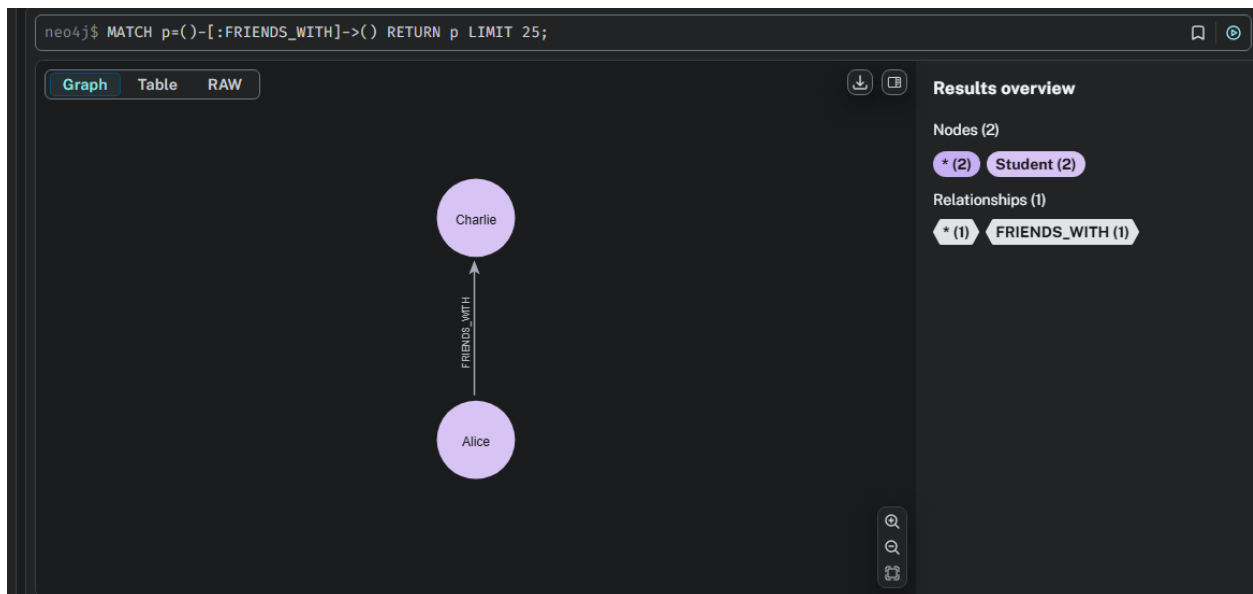
Created 2 nodes, set 4 properties, added 2 labels Fetched 0 records

$ :welcome
```

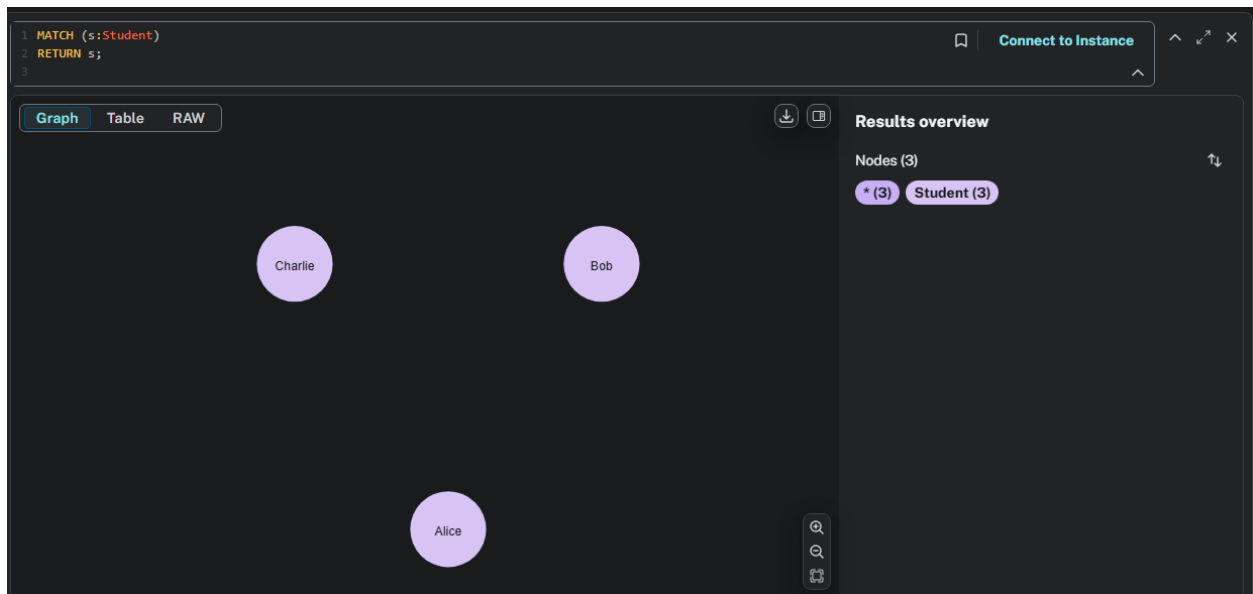




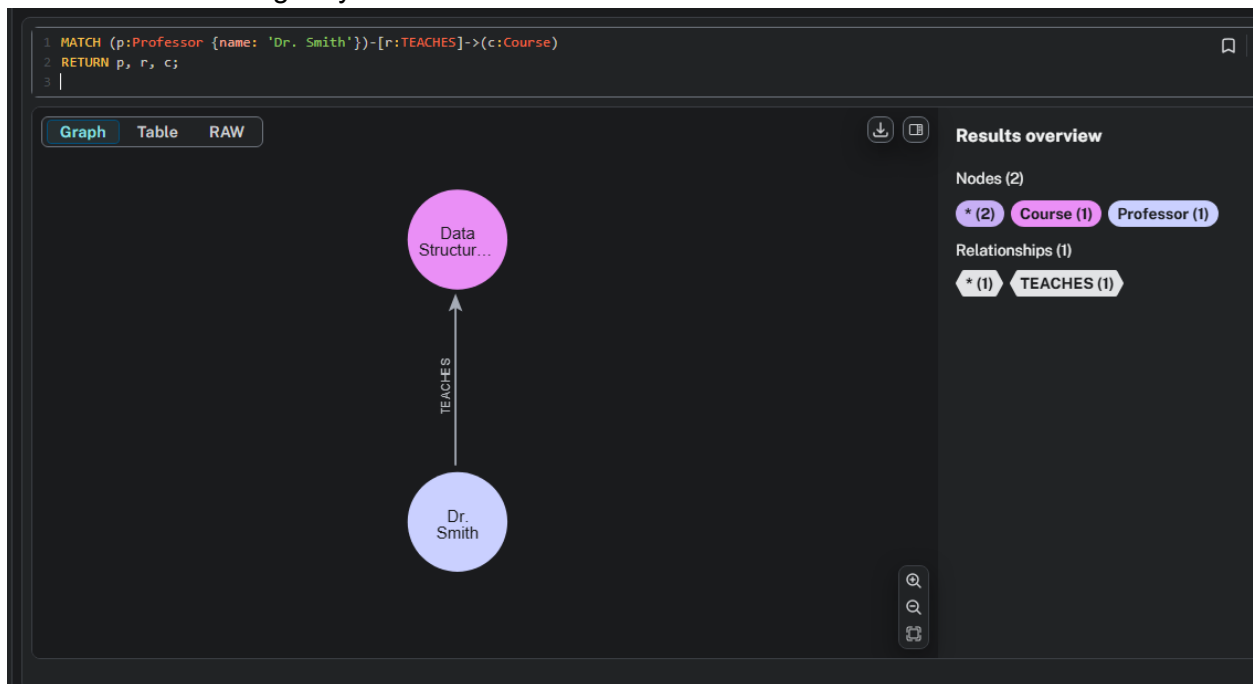




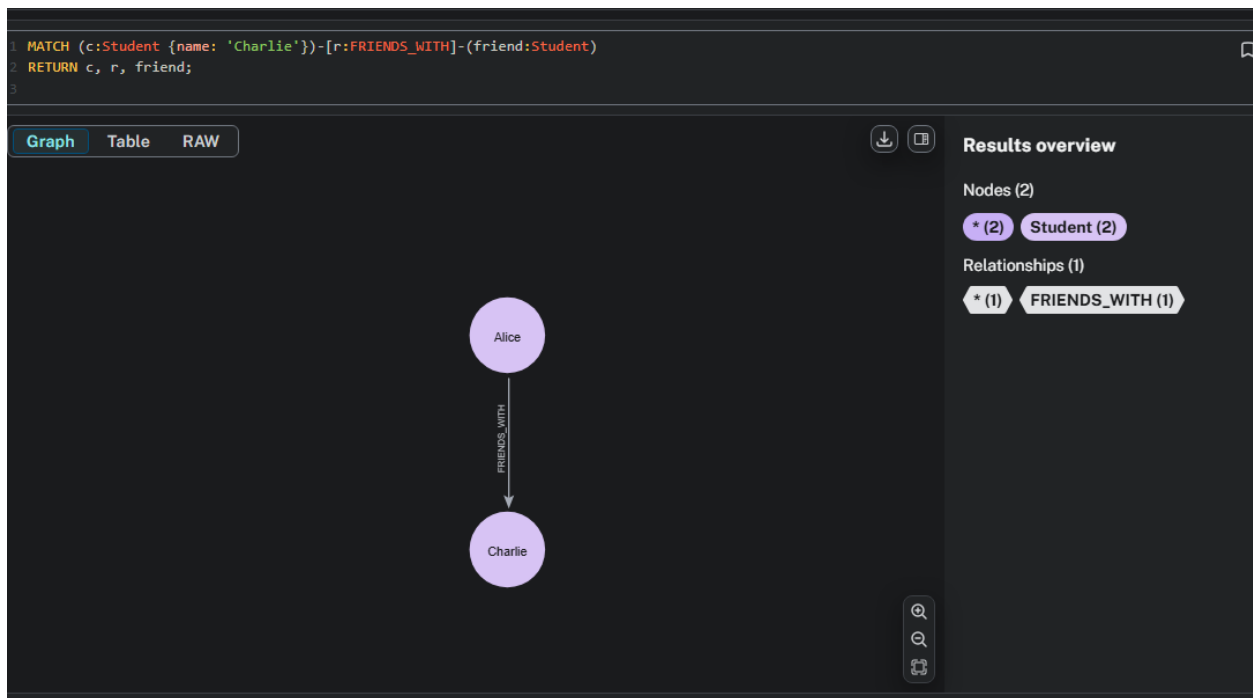
7. List All Students



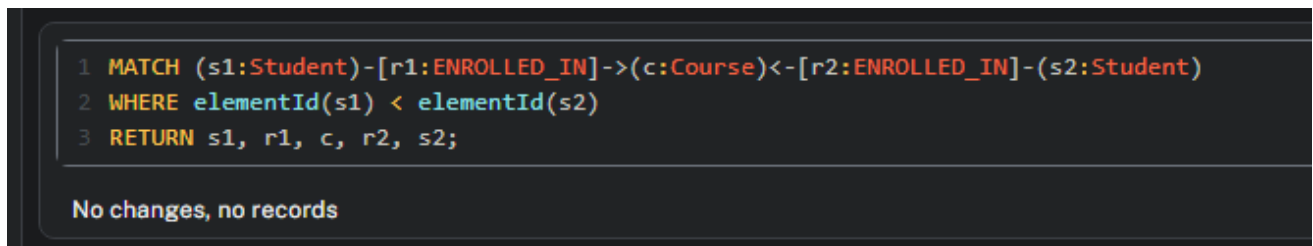
8. Find Courses Taught by Dr. Smith



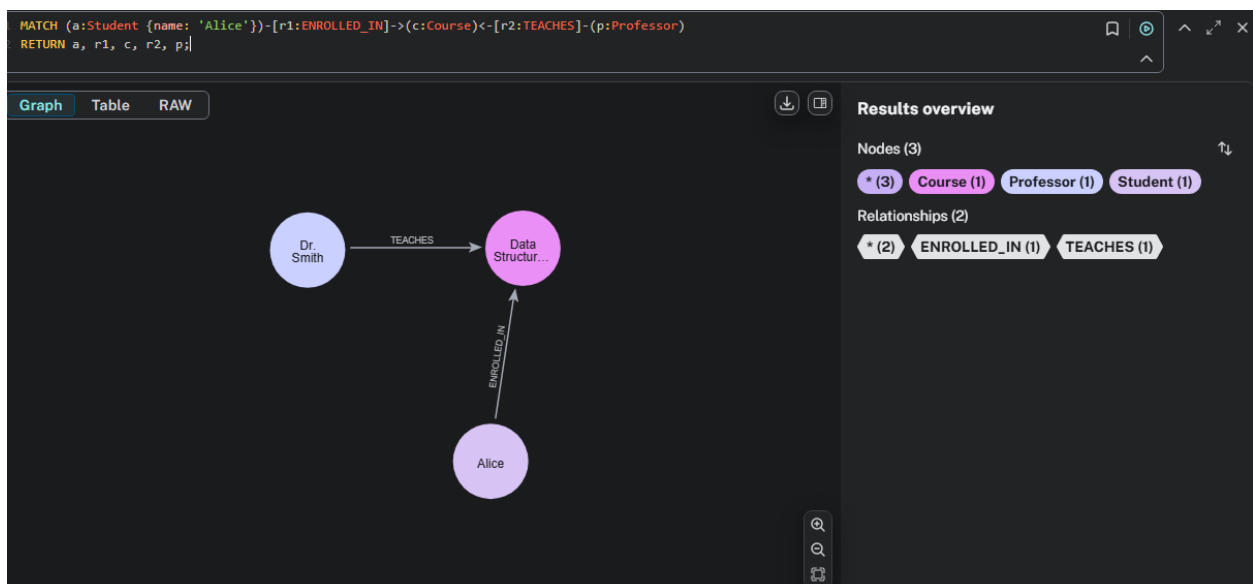
9. Find Friends of Charlie



10. List All Students in the Same Course



11. Find Professors Who Teach Alice's Courses



12. Find Students Who Are Friends and Enrolled in the Same Course

```
1 MATCH (s1:Student)-[r1:FRIENDS_WITH]-(s2:Student),
2       (s1)-[r2:ENROLLED_IN]->(c:Course)<-[r3:ENROLLED_IN]-(s2:Student)
3 WHERE elementId(s1) < elementId(s2)
4 RETURN s1, r1, s2, r2, c, r3;
```

No changes, no records

Find

13. Courses with More Than One Student Enrolled

```
1 MATCH (s:Student)-[:ENROLLED_IN]->(c:Course)
2 WITH c, count(s) AS studentCount
3 WHERE studentCount > 1
4 MATCH (s_final:Student)-[r_final:ENROLLED_IN]->(c)
5 RETURN s_final, r_final, c;
```

No changes, no records

14. Count how many students each professor teaches.

```
1 MATCH (p:Professor)-[:TEACHES]->(c:Course)<-[:ENROLLED_IN]-(s:Student)
2 RETURN p.name AS Professor, count(DISTINCT s) AS NumberOfStudents;
```

Table RAW

	Professor	NumberOfStudents
1	"Dr. Smith"	1
2	"Dr. Jones"	1

