**作业1：知识图谱数据模型与查询语言**

**小组成员：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

3. 使用Neo4j导入IMDB电影知识图谱数据集（imdb-data）的actors.csv、movies.csv、roles.csv（从智慧树下载imdb-data.zip压缩包解压），执行课程视频“实战-Neo4j.mp4”中所演示的所有Cypher查询，并给出查询结果截图。

**给出执行neo4j-admin import批量导入数据集成功截图：**

(1) 查询1

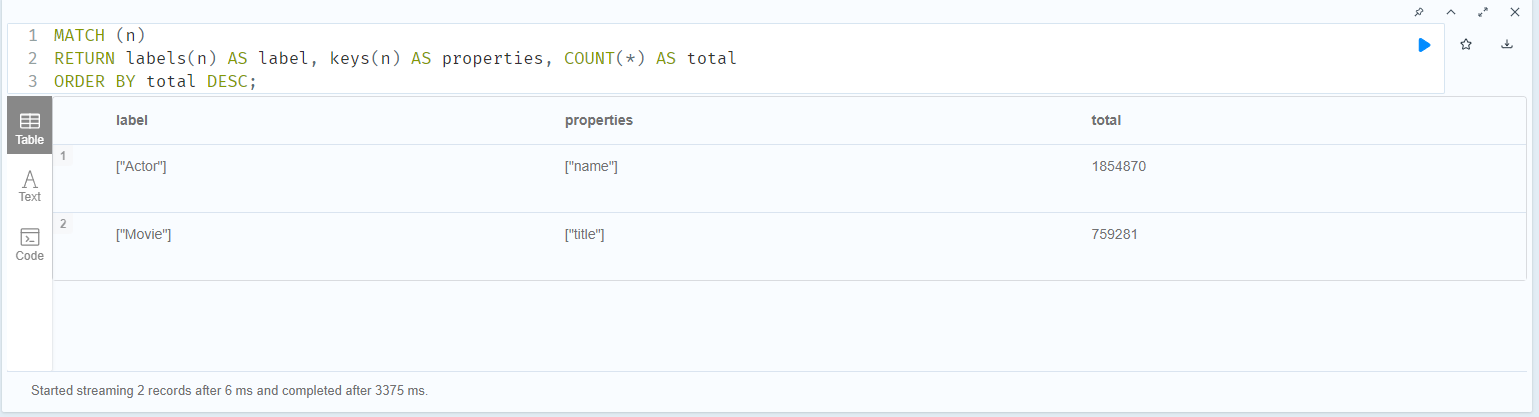
**Cypher语句：**

MATCH(n)

RETURN labels(n) AS labels, keys(n) AS properties, COUNT(\*) AS total

ORDER BY total DESC;

**查询结果截图：**

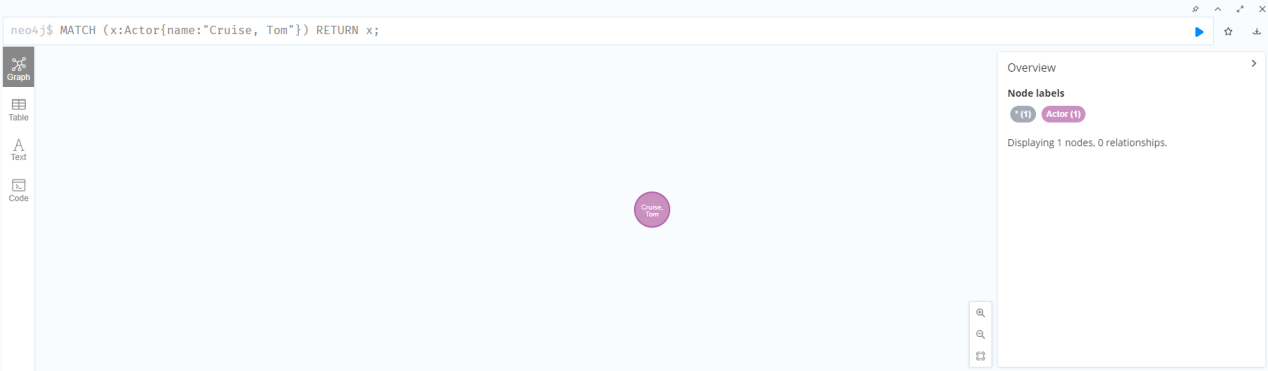


(2) 查询2

**Cypher语句：**

MATCH (x:Actor{name:"Cruise, Tom"}) RETURN x;

**查询结果截图：**

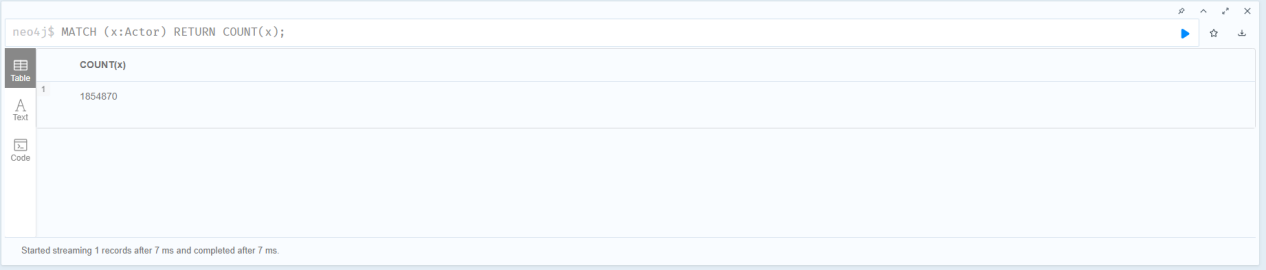


(3) 查询3

**Cypher语句：**

MATCH (x:Actor) RETURN COUNT(x);

**查询结果截图：**



(4) 查询4

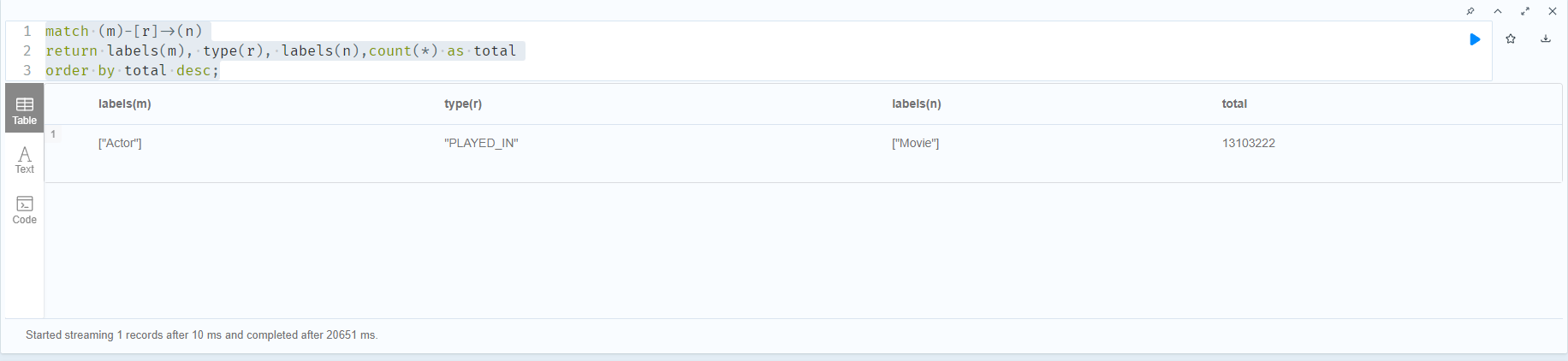
**Cypher语句：**

match (m)-[r]->(n)

return labels(m), type(r), labels(n),count(\*) as total

order by total desc;

**查询结果截图：**



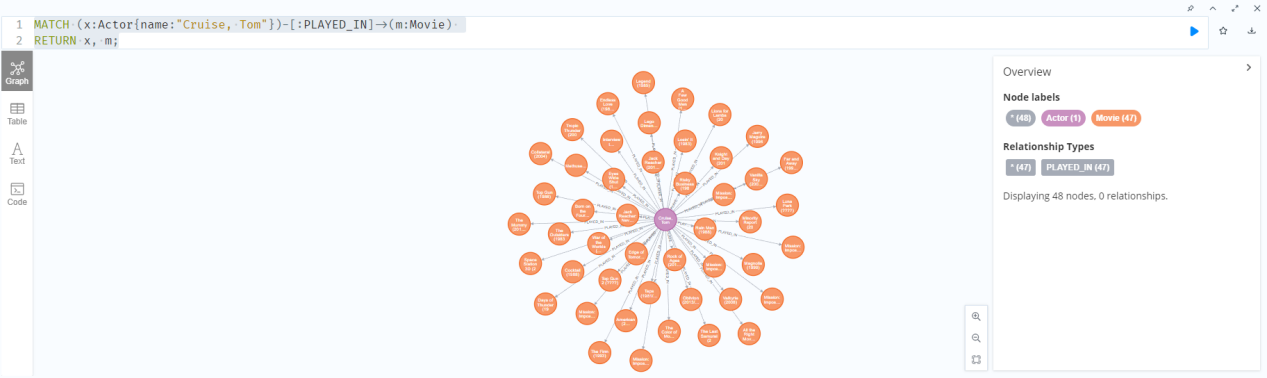
(5) 查询5

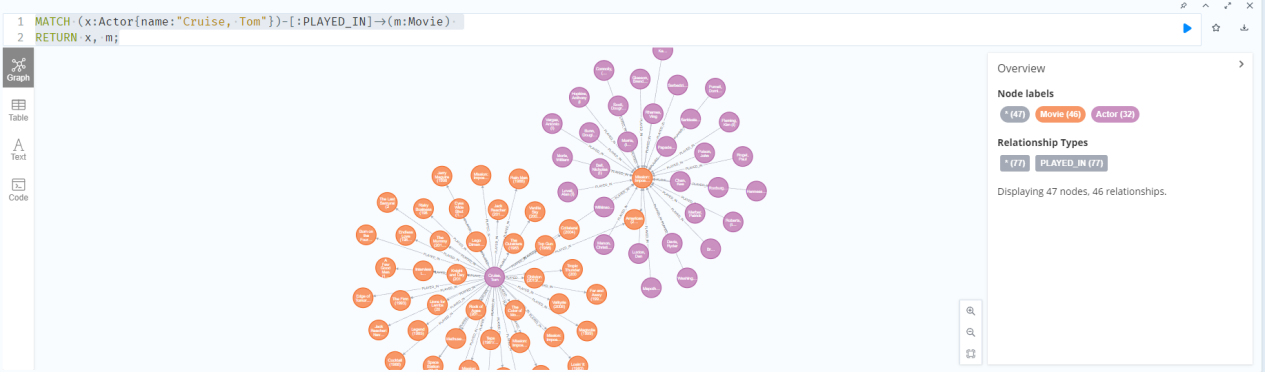
**Cypher语句：**

MATCH (x:Actor{name:"Cruise, Tom"})-[:PLAYED\_IN]->(m:Movie)

RETURN x, m;

**查询结果截图：**





(6) 查询6

**Cypher语句：**

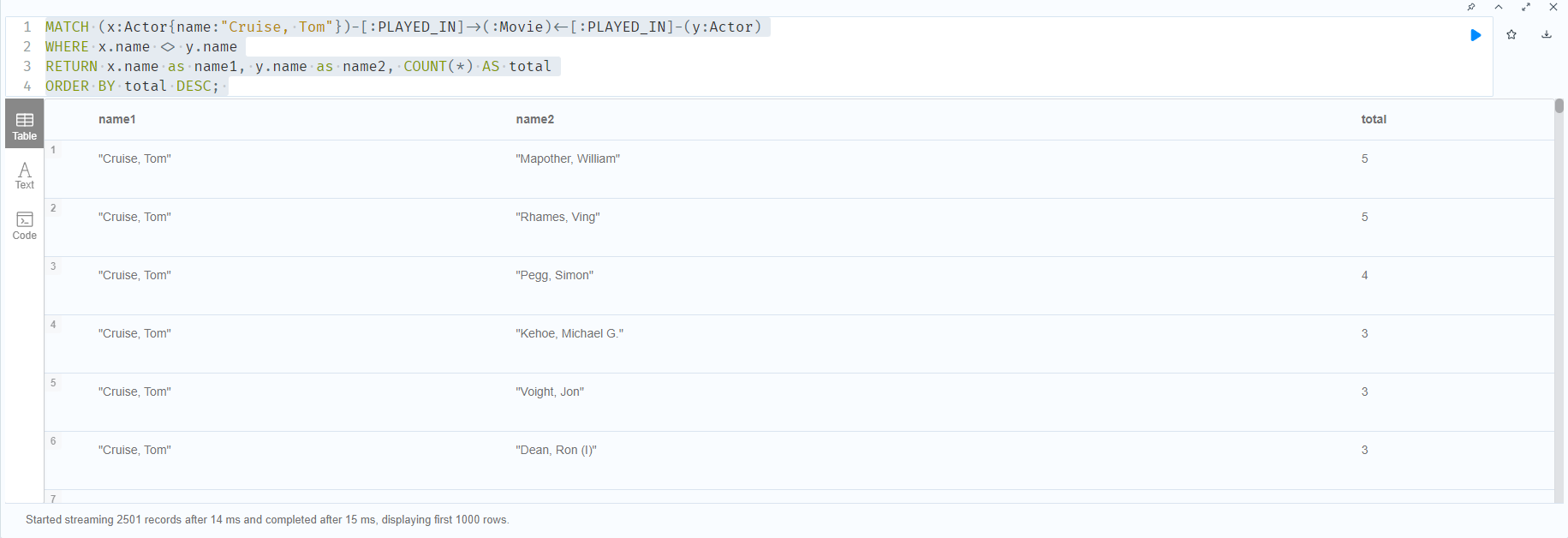
MATCH (x:Actor{name:"Cruise, Tom"})-[:PLAYED\_IN]->(:Movie)<-[:PLAYED\_IN]-(y:Actor)

WHERE x.name <> y.name

RETURN x.name as name1, y.name as name2, COUNT(\*) AS total

ORDER BY total DESC;

**查询结果截图：**



(7) 查询7

**Cypher语句：**

MATCH (x:Actor{name:"Cruise, Tom"})-[:PLAYED\_IN\*2]-(y:Actor)

WHERE x.name <> y.name

RETURN x.name as name1, y.name as name2, COUNT(\*) AS total

ORDER BY total DESC;

**查询结果截图：**

