LABSHEET: 14

NAME: Swetha.K

ROLL NO: 235229143

Question 1:

```
movie$ :use movie
```

```
movie$ CREATE (m1:MOVIE{id:"thalaiva",title:"Thalaiva",year:2013}),(m2:MOVIE{id:"mersal",title:"Mersal",year:2017}),
    (m3:MOVIE{id:"rajavinparvaiyile",title:"Rajavin Parvayile",year:1995}),(m4:MOVIE{id:"master",title:"Master
    ",year:2020}),(a1:ACTOR{id:"vijay",title:"Vijay Jospeh",year:1973}),(a2:ACTOR{id:"ajith",title:"Ajith
    Kumar",year:1971}),(a3:ACTOR{id:"vadivelu",title:" Vadivelu",year:1963}),(a4:ACTOR{id:"sathiyaraj",title:"Sathiya
    Raj",year:1955}),                    (m1)-[c1:PLAY{role:"Vishwa Bhai"}]→(a1),(m1)-[c2:PLAY{role:"Ramadorai Anna"}]→
    (a4),(m2)-[c3:PLAY{role:"doctor"}]→(a1),(m2)-[c4:PLAY{role:"compounder"}]→(a3),(m2)-[c5:PLAY{role:"police officer"}]-
    >(a4),(m3)-[c6:PLAY{role:"Raja"}]→(a1),(m3)-[c7:PLAY{role:"Chandru",award:"Only film of Vijay and Ajith together"}]→
    (a2),(a1)-[f1:KNOW]→(a2),(a1)-[f2:KNOW]→(a3),(a2)-[f3:KNOW]→(a3),(a4)-[f4:KNOW]→(a2);
```
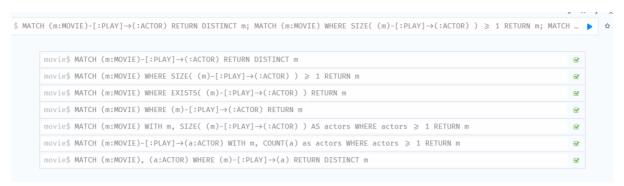
Added 8 labels, created 8 nodes, set 32 properties, created 11 relationships, completed after 216 ms.

Question 2:

```
$ MATCH (m:MOVIE {id: "rajavinparvaiyile"}) RETURN m, m.title; MATCH (m:MOVIE) WHERE m.id = "rajavinparvaiyile" RETURN m, m.titl…
```

```
movie$ MATCH (m:MOVIE {id: "rajavinparvaiyile"}) RETURN m, m.title
movie$ MATCH (m:MOVIE) WHERE m.id = "rajavinparvaiyile" RETURN m, m.title
```

Question 3:

```
movie$ MATCH (a:ACTOR) WHERE a.year ≥ 1971 RETURN a.name, a.year ORDER BY a.year DESC, a.name ASC;
```

a.name	a.year
null	1973
null	1971

Question 4:

```
movie$ MATCH (:ACTOR {name: "Ajith Kumar"})←[:PLAY]-(n:MOVIE) RETURN n.title;
```

(no changes, no records)

Question 5:

```
$ MATCH (m:MOVIE)-[:PLAY]→(:ACTOR) RETURN DISTINCT m; MATCH (m:MOVIE) WHERE SIZE( (m)-[:PLAY]→(:ACTOR) ) ≥ 1 RETURN m; MATCH …  ▶  ☆
```

```
movie$ MATCH (m:MOVIE)-[:PLAY]→(:ACTOR) RETURN DISTINCT m                                                    ☑
movie$ MATCH (m:MOVIE) WHERE SIZE( (m)-[:PLAY]→(:ACTOR) ) ≥ 1 RETURN m                                       ☑
movie$ MATCH (m:MOVIE) WHERE EXISTS( (m)-[:PLAY]→(:ACTOR) ) RETURN m                                         ☑
movie$ MATCH (m:MOVIE) WHERE (m)-[:PLAY]→(:ACTOR) RETURN m                                                   ☑
movie$ MATCH (m:MOVIE) WITH m, SIZE( (m)-[:PLAY]→(:ACTOR) ) AS actors WHERE actors ≥ 1 RETURN m             ☑
movie$ MATCH (m:MOVIE)-[:PLAY]→(a:ACTOR) WITH m, COUNT(a) as actors WHERE actors ≥ 1 RETURN m              ☑
movie$ MATCH (m:MOVIE), (a:ACTOR) WHERE (m)-[:PLAY]→(a) RETURN DISTINCT m                                    ☑
```

Question 6:

```
movie$ MATCH (s:ACTOR {name: "Vijay Joseph"}) ←[:PLAY]-(m:MOVIE)-[:PLAY]→ (a:ACTOR) RETURN DISTINCT a;      ▶  ☆
```

Table	(no changes, no records)
Code	

Question 7:

```
movie$ MATCH (m:MOVIE) OPTIONAL MATCH (m)-[:PLAY]→(a:ACTOR) RETURN m.title, a.name;                          ▶  ☆  ⬇
```

m.title	a.name
"Thalaiva"	null
"Thalaiva"	null
"Mersal"	null
"Mersal"	null
"Mersal"	null
"Rajavin Parvayile"	null
"Rajavin Parvayile"	null
"Master "	null

Question 8:

```
movie$ MATCH (m:MOVIE) WITH m, SIZE( (m)-[:PLAY]→(:ACTOR) ) AS actors WITH AVG(actors) AS average MATCH (m:MOVIE) WHERE SIZ…  ▶  ☆  ⬇
```

a.name
null

Question 9:

```
movie$ MATCH (m:MOVIE) OPTIONAL MATCH (m)-[:PLAY]→(a:ACTOR) WITH m, COUNT(a) AS actors WITH AVG(actors) AS average MATCH (m…
```

a.name
null