

## Extra Practice Writing Queries

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 1) Find the ids and names of everyone in the database who acted in *Avatar*.

```
SELECT P.id, P.name
FROM Person P, Actor A, Movie M
WHERE P.id = A.actor_id
      AND M.id = A.movie_id
      AND M.name = 'Avatar';
```

## Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 2) How many people in the database did not act in *Avatar*?

Will this work?

```
SELECT COUNT(*)
FROM Person P, Actor A, Movie M
WHERE P.id = A.actor_id AND M.id = A.movie_id
      AND M.name != 'Avatar';
```

If not, what will?

```
SELECT COUNT(*)
FROM Person
WHERE id NOT IN (SELECT actor_id
                  FROM Actor A, Movie M
                  WHERE A.movie_id = M.id
                      AND M.name = 'Avatar');
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 3) How many people in the database who were born in California have won an Oscar? (assume dob format is city, state, country)

```
SELECT COUNT(DISTINCT P.id)
FROM Person P, Oscar O
WHERE P.id = O.person_id
AND P.pob LIKE '%,%California, USA';
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 4) Find the ids and names of everyone in the database who has acted in a movie directed by James Cameron.  
(Hint: One table is needed twice!)

```
SELECT DISTINCT ActP.id, ActP.name
FROM Person ActP, Actor A,
     Director D, Person DirP
WHERE ActP.id = A.actor_id
AND A.movie_id = D.movie_id
AND D.director_id = DirP.id
AND DirP.name = 'James Cameron';
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 4) Find the ids and names of everyone in the database who has acted in a movie directed by James Cameron.

*an alternate answer using set membership and a subquery:*

```
SELECT id, name
FROM Person
WHERE id IN (SELECT A.actor_id
             FROM Actor A, Director D, Person P
             WHERE A.movie_id = D.movie_id
                 AND D.director_id = P.id
                 AND P.name = 'James Cameron');
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 4) Find the ids and names of everyone in the database who has acted in a movie directed by James Cameron.

*a third answer that also uses a subquery:*

```
SELECT DISTINCT P.id, P.name
FROM Person P, Actor A, Director D
WHERE P.id = A.actor_id
  AND A.movie_id = D.movie_id
  AND D.director_id = (SELECT id
                      FROM Person
                      WHERE name = 'James Cameron');
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 5) Which movie ratings have an average runtime that is greater than 120 minutes, and what are their average runtimes?

```
SELECT rating, AVG(runtime)
FROM Movie
GROUP BY rating
HAVING AVG(runtime) > 120;
```

### Extra Practice Writing Queries (cont.)

Person(id, name, dob, pob)  
Movie(id, name, year, rating, runtime, genre, earnings\_rank)  
Actor(actor\_id, movie\_id) Director(director\_id, movie\_id)  
Oscar(movie\_id, person\_id, type, year)

- 6) For each person in the database born in Boston, find the number of movies in the database (possibly 0) in which the person has acted.

```
SELECT name, COUNT(movie_id)
FROM Person LEFT OUTER JOIN Actor
      ON id = actor_id
WHERE pob LIKE 'Boston, Mass%'
GROUP BY name;
```