</talentlabs>

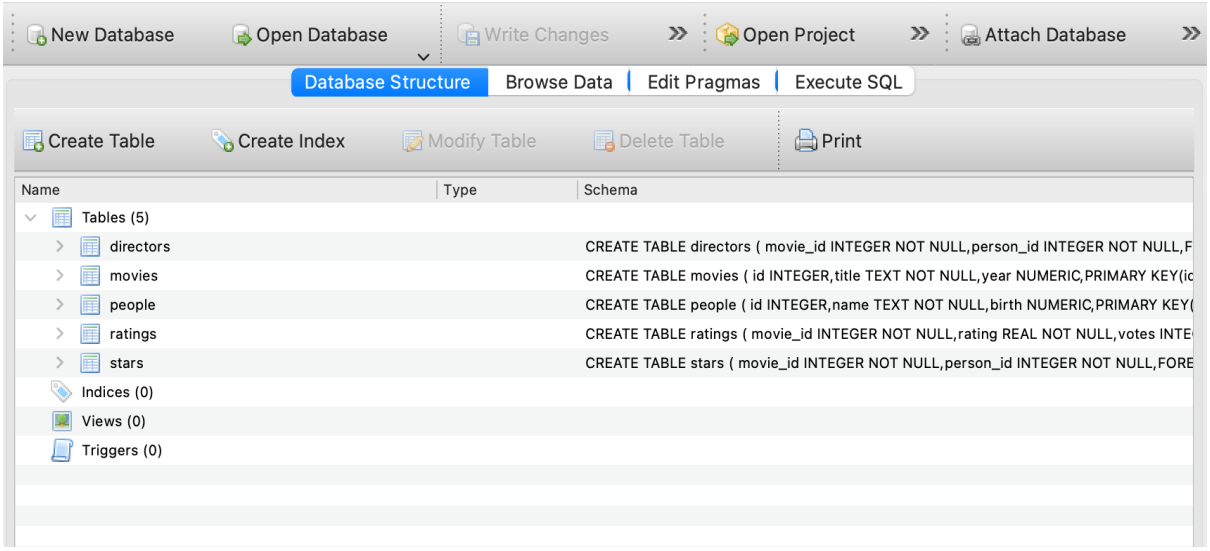
Lab Assignment 6 - Complex SELECT statements

Instructions

1. Answer the below question in the boxes.
2. Please submit the assignment through TalentLabs Learning System.

Open the Movies database

Follow the step illustrated in Chapter 3 to open the Movies database using DB Browser for SQLite. You should see 5 tables in the database.



</talentlabs>

Understanding the database

1. Study the table schema and the data in the “people” and “directors” table and describe the relation between the tables “people” and “directors”

   The “people” table will be binding with “directors” table which “people” id was located as the “directors” person_id.

2. Study the table schema and the data in the “movies” and “directors” table and describe the relation between the tables “movies” and “directors”

   The “movies” table will be binding with “directors” table which “movies” id was located the “directors” movie_id.

Query Exercises

1. Write a SQL query to obtain the movie_id who is directed by “Joris Ivens” without using WITH keyword
   Expected Output: a table with a single column for the movie_id of the director’s movie.

   ```
   SELECT movie_id FROM directors WHERE person_id IN (SELECT id FROM people WHERE name= 'Joris Ivens')
   ```

2. Write a SQL query to obtain the movie title who is directed by “Joris Ivens”
   Expected Output: a table with a single column for the movie title of the director’s movie.

```
WITH
Name_Joris AS
(SELECT id FROM people WHERE name = 'Joris Ivens'),
movieID AS
(SELECT movie_id FROM directors WHERE person_id IN Name_Joris)
SELECT title FROM movies WHERE id IN movieID
```

3. Organize and rewrite the SQL query of Q1 using WITH keyword
   Expected Output: The SQL query in WITH keyword

```
WITH mv_id AS
(
SELECT id FROM people WHERE name = 'Joris Ivens'
)
SELECT movie_id FROM directors where person_id IN mv_id
```

4. Write a SQL query to show each person’s name and whether the person is born before 1970, born in 1970, born after 1970
   Expected Output: The SQL query fulfilling the required data

```
SELECT name,
CASE
WHEN birth < 1970 THEN 'born before 1970'
WHEN birth > 1970 THEN 'born after 1970'
WHEN birth = 1970 THEN 'born in 1970'
ELSE NULL
END AS born_year FROM people
```

5. Write a SQL query to count the number of people in the “people” table by each birth year.
   Expected Output: The SQL query fulfilling the required data. Note that having the NULL birth year on the query result is normal.

```
SELECT birth, COUNT(*) FROM people GROUP BY birth
```

6. Write a SQL query to count the number of directors by each birth year. Only the years with more than 500 directors born are interested.
   Expected Output: a table with two columns for the birth year and count of directors.

```
WITH
directorsID AS
(SELECT DISTINCT person_id FROM directors)
SELECT birth,COUNT(*) AS count FROM people WHERE id IN directorsID
AND birth IS NOT NULL
GROUP BY birth HAVING count>500
```

- End of Assignment -