INFORMATION TECHNOLOGY ENGINEERING

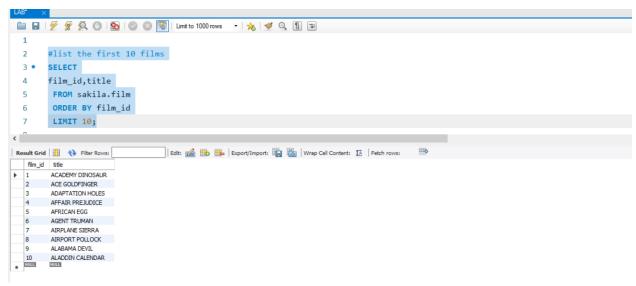
CLASS: ITE M1

NAME: Thouerng sreymealea

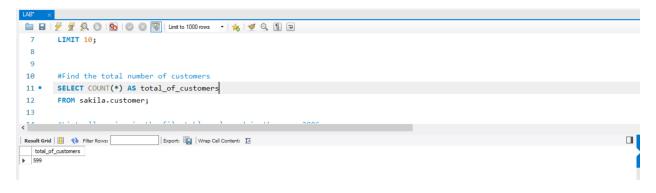
Part 1: Basic Queries

1. Retrieve Basic Data

• Write a query to list the first 10 films, displaying the film_id, title, and release_year

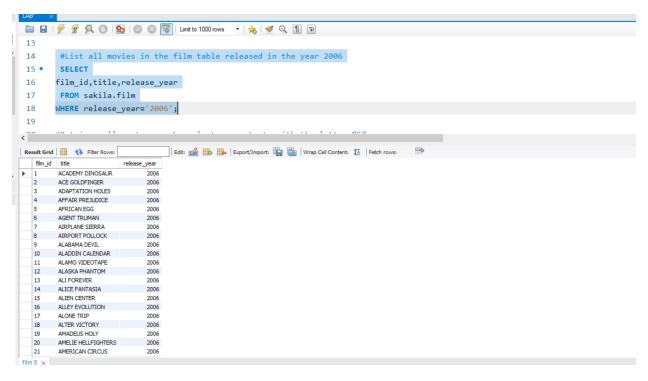


• Find the total number of customers in the database

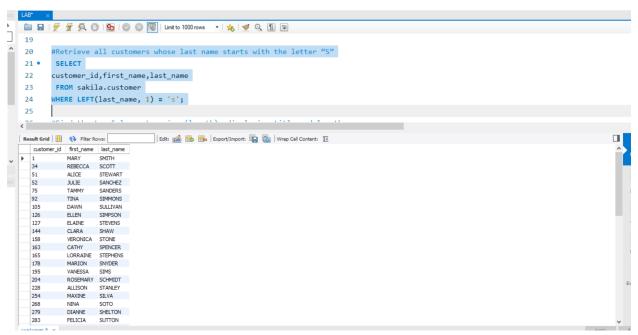


2. Filter Data

• List all movies in the film table released in the year 2006

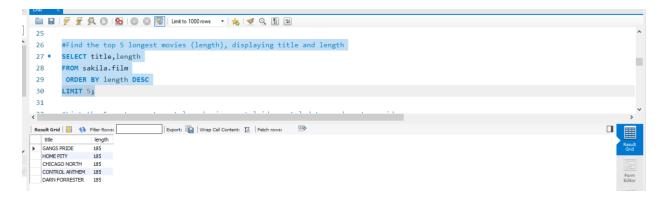


• Retrieve all customers whose last name starts with the letter "S"

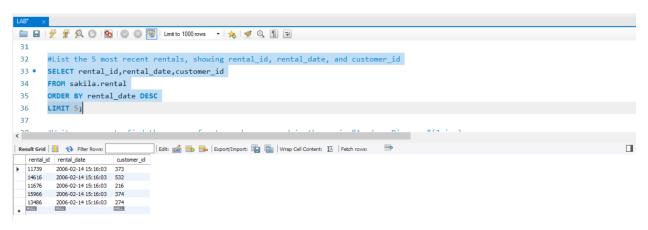


3. Sort and Limit Results

• Find the top 5 longest movies (length), displaying title and length



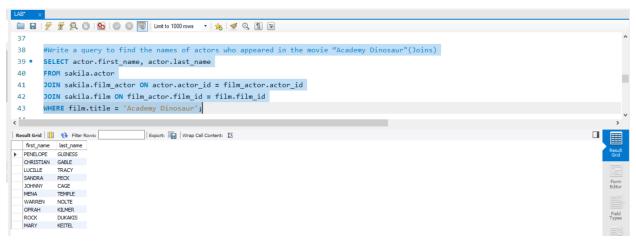
• List the 5 most recent rentals, showing rental_id, rental_date, and customer_id



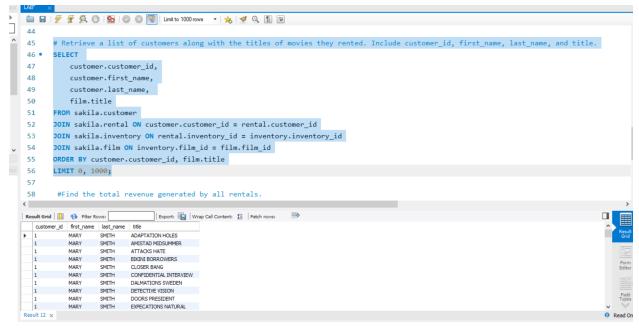
Part 2: Advanced Queries

4. Joins

· Write a query to find the names of actors who appeared in the movie "Academy Dinosaur"

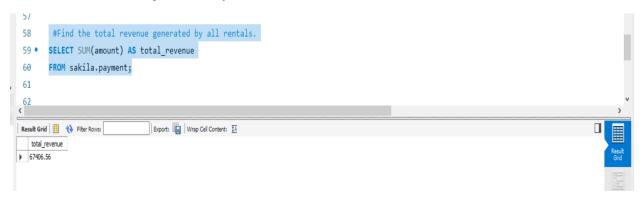


• Retrieve a list of customers along with the titles of movies they rented. Include customer_id, first_name, last_name, and title.

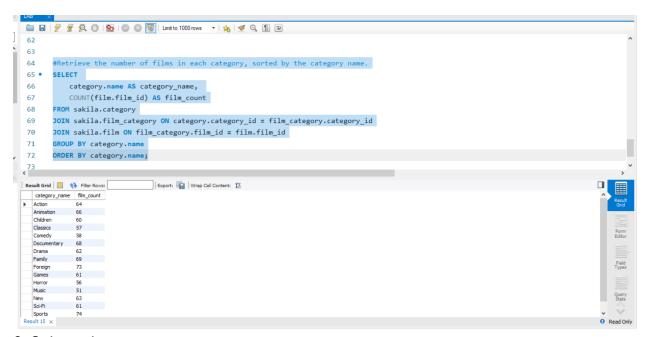


5. Aggregations

• Find the total revenue generated by all rentals

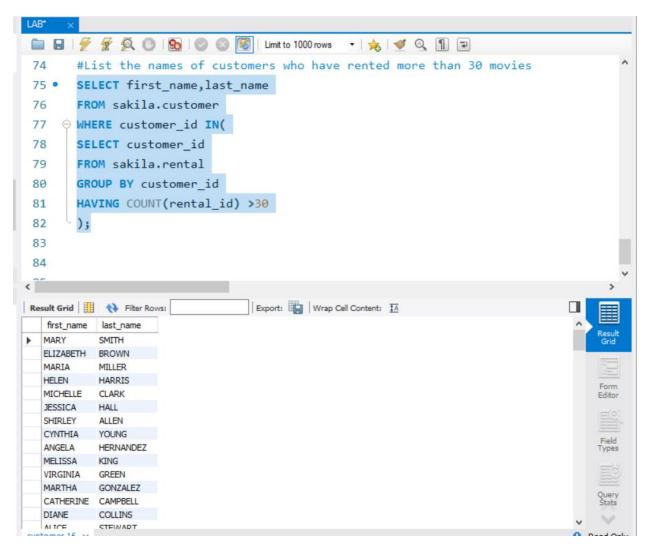


• Retrieve the number of films in each category, sorted by the category name

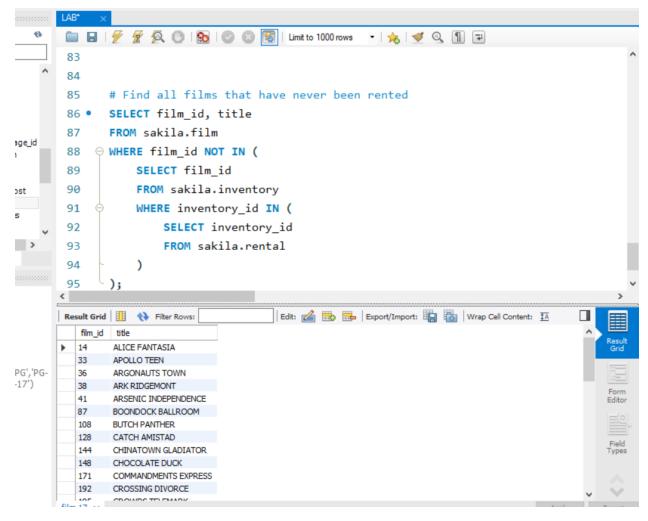


6. Subqueries

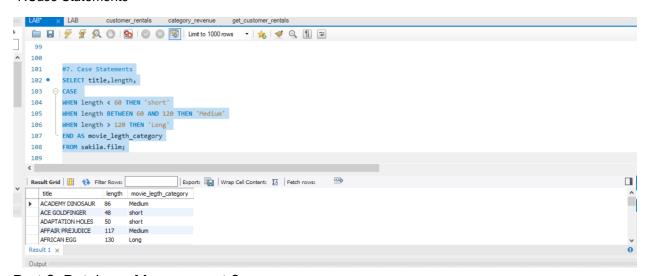
• List the names of customers who have rented more than 30 movies



• Find all films that have never been rented



7.Case Statements



Part 3: Database Management 8.

8. Insert Data

.Insert customer

```
106
 107
         #insert data to table
 108 •
         INSERT INTO sakila.customer (first_name, last_name, email, address_id, activ
         VALUES ('John', 'Doe', 'john.doe@example.com', 1, 1, NOW(), 1);
 109
 110 •
         SELECT *
 111
         FROM sakila.customer
 112
         WHERE first_name = 'John' AND last_name = 'Doe';
 110
 Edit: 🕍 📆 Export/Import: 📳 📸 | Wrap Cell Content: 🔣
     customer_id store_id
                     first_name
                             last name
                                     email
                                                      address id
                                                                    create date
                                                                                   last upda
                                     john.doe@example.com
                                                                    2024-12-13 13:23:23
                                                                                   2024-12-1
```

.Insert film

```
110
        INSERT INTO sakila.film (title, description, release_year, language_id, rent
114 •
        VALUES ('Epic Adventure', 'A thrilling tale of courage and survival.', 2024,
115
        SELECT *
116 •
        FROM sakila.film
117
        WHERE title = 'Epic Adventure';
118
110
Edit: 🚄 🖶 Export/Import: 📳 🦝 Wrap Cell Content: 🟗
                     description
   film_id
                                               release_year language_id
                                                                  original_language_id
                                                                  NULL
  1001
         Epic Adventure
                    A thrilling tale of courage and survival.
· NULL
                                                                  NULL
                                                                                 NULL
                                                     HULL HULL
```

9. Update data

```
124
           #9. UPDATE DATA
  125
  126
           #Update the email of a specific customer in the customer table
  127 •
           UPDATE sakila.customer
           SET email = 'newjohn.doe@example.com'
  128
           WHERE first_name = 'John' AND last_name = 'Doe';
  129
  130
           SELECT customer_id, first_name, last_name, email
           FROM sakila.customer
  131
           WHERE first_name = 'John' AND last_name = 'Doe';
  132
  133
  134
           #9. Change the rental rate for all movies in the film table to 4.99 where the length is greater th
           SET SQL_SAFE_UPDATES = 0;
  135 •
  136 •
           UPDATE sakila.film
  137
           SET rental rate = 4.99
 <
                                             Edit: 🚄 🖶 Export/Import: 📳 🐻 | Wrap Cell Content: 🔣
 customer_id
                first_name
                           last_name
                                     email
                                     newjohn.doe@example.com
 133

    Change the rental rate for all movies in the film table to 4.99 where the length is greater th

 134
135 •
          SET SQL_SAFE_UPDATES = 0;
          UPDATE sakila.film
136 •
          SET rental_rate = 4.99
137
          WHERE length > 120;
138
          SET SQL_SAFE_UPDATES = 1;
139 •
140 •
          SET SQL SAFE UPDATES = 0;
141
142
          #10. Delete Data
<
Output :
Action Output
         Time
                 Action
                                                           Message
                                                                                                    Duration / Fetch
      6 10:43:22 SELECT customer_id, first_name, last_name, email F...
                                                          2 row(s) returned
                                                                                                    0.000 sec / 0.000 sec
      7 10:44:05 SET SQL_SAFE_UPDATES = 0
                                                          0 row(s) affected
                                                                                                    0.000 sec
      8 10:44:05 UPDATE sakila film SET rental_rate = 4.99 WHERE ...
                                                          0 row(s) affected Rows matched: 457 Changed: 0 ...
                                                                                                    0.032 sec
      9 10:44:05 SET SQL SAFE UPDATES = 1
                                                          0 row(s) affected
                                                                                                    0.000 sec
10 10-44-05 SET SQL SAFE UPDATES = 0
                                                          0 row(s) affected
                                                                                                    0.000 sec
```

10. Delete Data

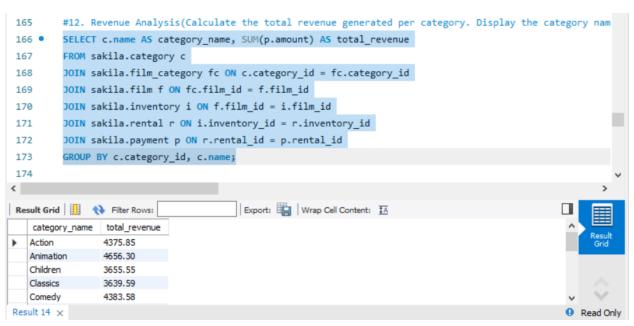
```
#10. Delete Data
 142
          # Delete a customer from the customer table who hasn't rented any movies.
 143
          DELETE FROM sakila.customer

→ WHERE customer_id NOT IN (
 145
              SELECT DISTINCT customer_id
 146
              FROM sakila.rental
 147
 148
 149
          # Remove all movies in the film table that were released before 2000.
 150
         DELETE FROM sakila.film
 151 •
         WHERE release_year < 2000;
 152
          SELECT film id, title, release year
 153 •
          FROM sakila.film;
154
155
<
Output ::
Action Output
        Time
               Action
                                                                                                    Duration / Fetch
                                                          Message
      1 10:49:29 DELETE FROM sakila.customer WHERE customer_id ... 0 row(s) affected
                                                                                                    0.000 sec
 149
          # Remove all movies in the film table that were released before 2000.
 150
 151 •
          DELETE FROM sakila.film
 152
          WHERE release year < 2000;
          SELECT film_id, title, release_year
 153 •
          FROM sakila.film;
 154
 155
 156
          # Create a Report
                                           Edit: 🕍 🐯 🖶 Export/Import: 📳 🐻 | Wrap Cell Content: 🏗 | Fetch row
 film_id title
                             release_year
           ACADEMY DINOSAUR
                                    2006
          ACE GOLDFINGER
                                    2006
           ADAPTATION HOLES
          AFFAIR PREJUDICE
                                    2006
    5
           AFRICAN EGG
                                    2006
                                    2006
   6
          AGENT TRUMAN
film 10 ×
           ATDDI ANE CTEDDA
```

11. Create a Report

```
156
         # Create a Report
157
         #11. showing the top 10 customers who rented the most movies
         SELECT c.customer_id, c.first_name, c.last_name, COUNT(r.rental_id) AS movies_rented
158 •
         FROM sakila.customer c
159
160
         JOIN sakila.rental r ON c.customer id = r.customer id
         GROUP BY c.customer_id, c.first_name, c.last_name
161
         ORDER BY movies_rented DESC
162
163
         LIMIT 10;
         #12. Revenue Analysis(Calculate the total revenue generated per category. Display the category nam 🔻
165
Export: Wrap Cell Content: A Fetch rows:
   customer_id
              first_name
                        last_name
                                 movies_rented
   148
              ELEANOR
                       HUNT
                                 46
   526
              KARL
                       SEAL
                                 45
   144
              CLARA
                       SHAW
                                 42
              MARCIA
                       DEAN
                                 42
   236
   75
              TAMMY
                        SANDERS
Result 13 ×
                                                                                                    Read Only
Output
```

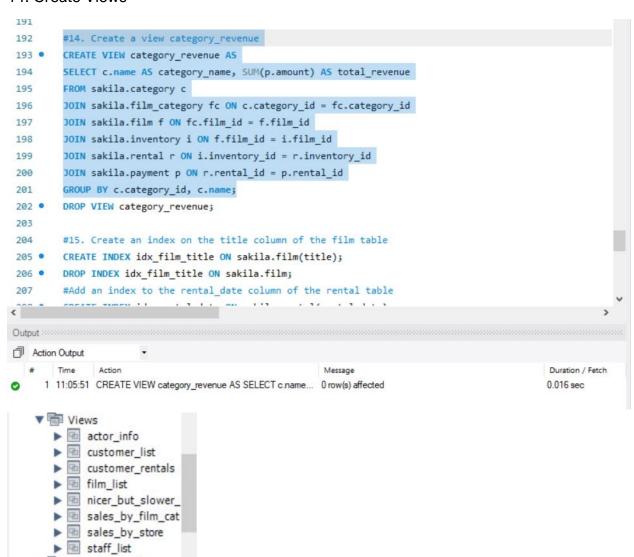
12. Revenue Analysis



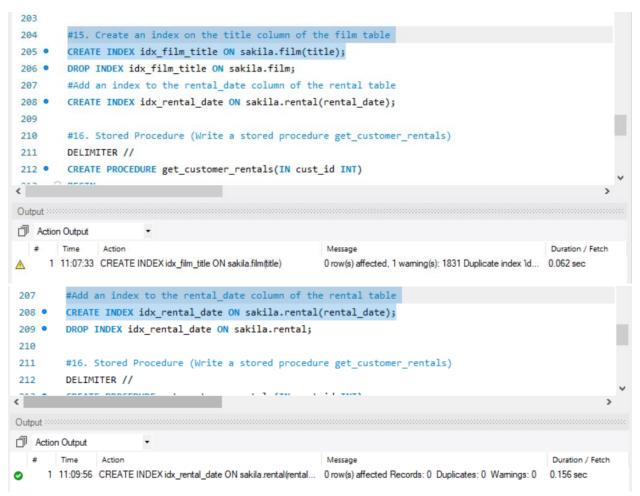
13. Rental Trends

```
174
         #13. Rental Trends( Identify the busiest rental month. Display the month and the total number of
175
         SELECT MONTH(r.rental_date) AS rental_month, COUNT(r.rental_id) AS total_rentals
176 •
177
         FROM sakila.rental r
         GROUP BY rental_month
178
179
         ORDER BY total_rentals DESC
         LIMIT 1;
180
181
182
        #Create Views
<
Export: Wrap Cell Content: 🔣 | Fetch rows:
   rental_month total_rentals
> 7
              6709
```

14. Create Views



15. Create Indexes



16. Stored Procedure

