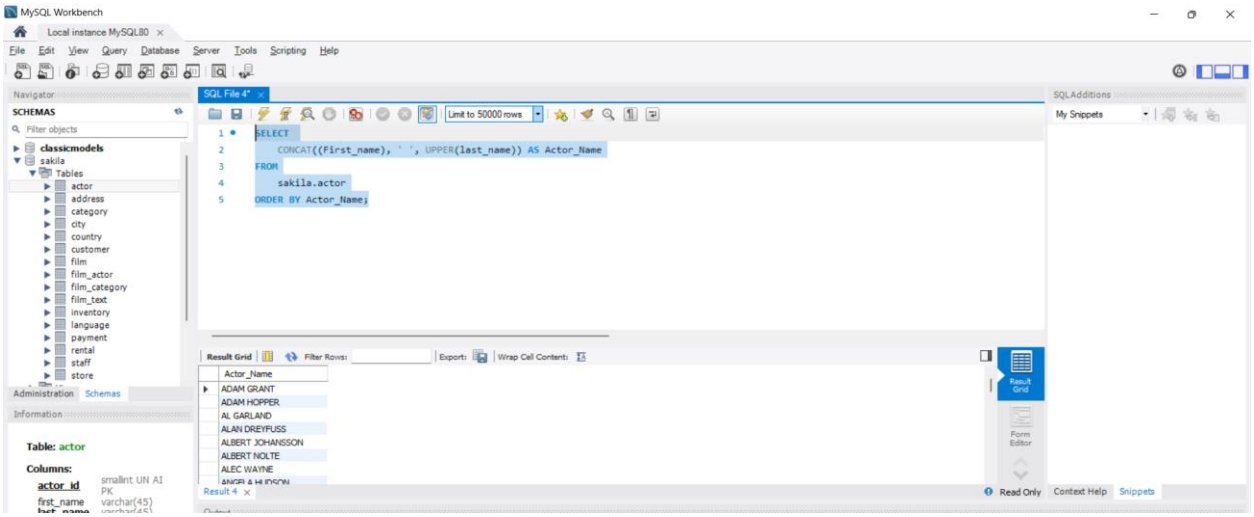
Exercises

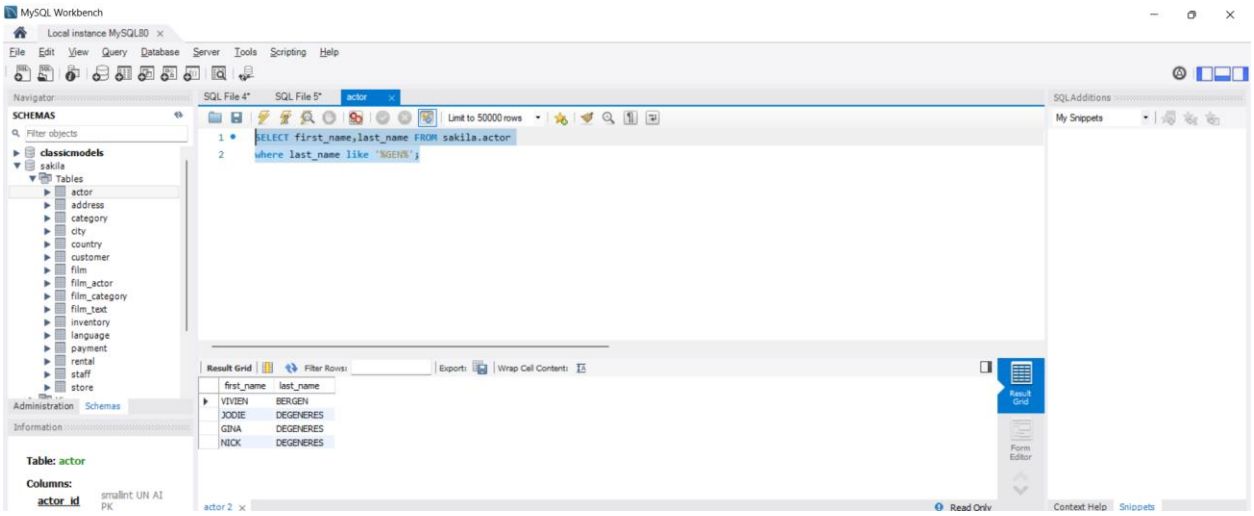
1. Display the first and last name of each actor in a single column in upper case letters in alphabetic order. Name the column Actor Name.

```
SELECT
    CONCAT((First_name), ' ', UPPER(last_name)) AS Actor_Name
FROM     sakila.actor ORDER BY Actor_Name;
```



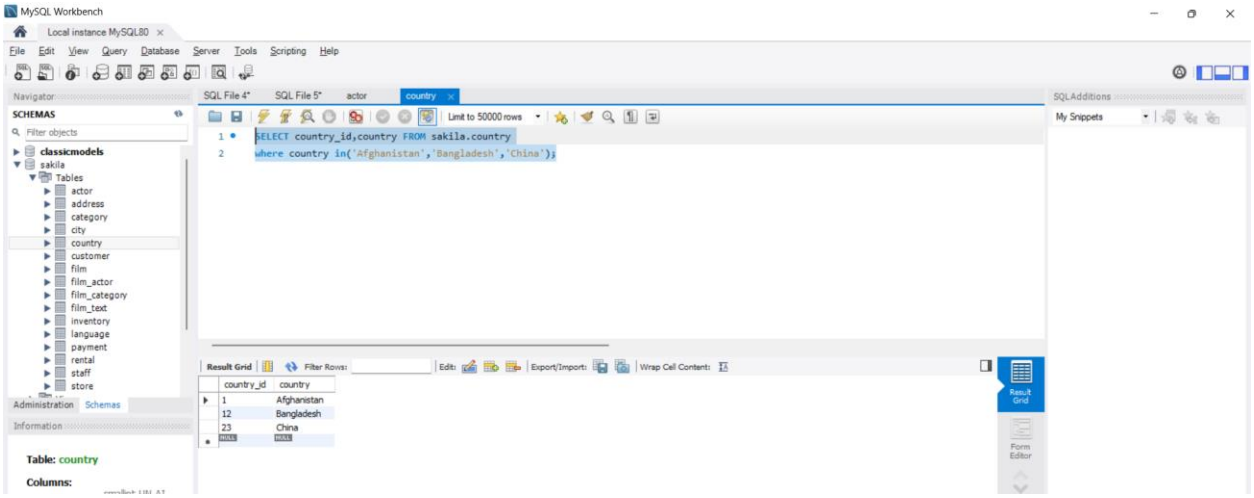
2. Find all actors whose last name contain the letters GEN:

```
SELECT first_name,last_name FROM sakila.actor
where last_name like '%GEN%';
```



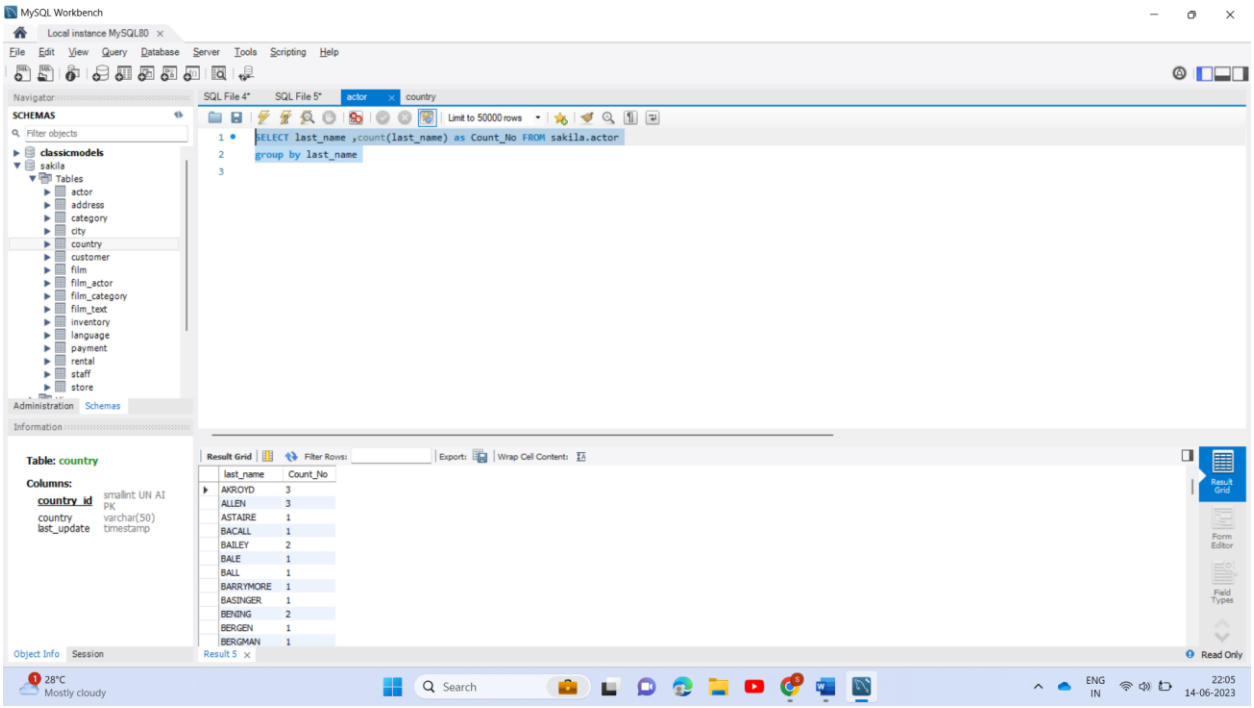
3. Using IN, display the country_id and country columns of the following countries: Afghanistan, Bangladesh, and China:

```
SELECT country_id,country FROM sakila.country
where country in('Afghanistan','Bangladesh','China');
```



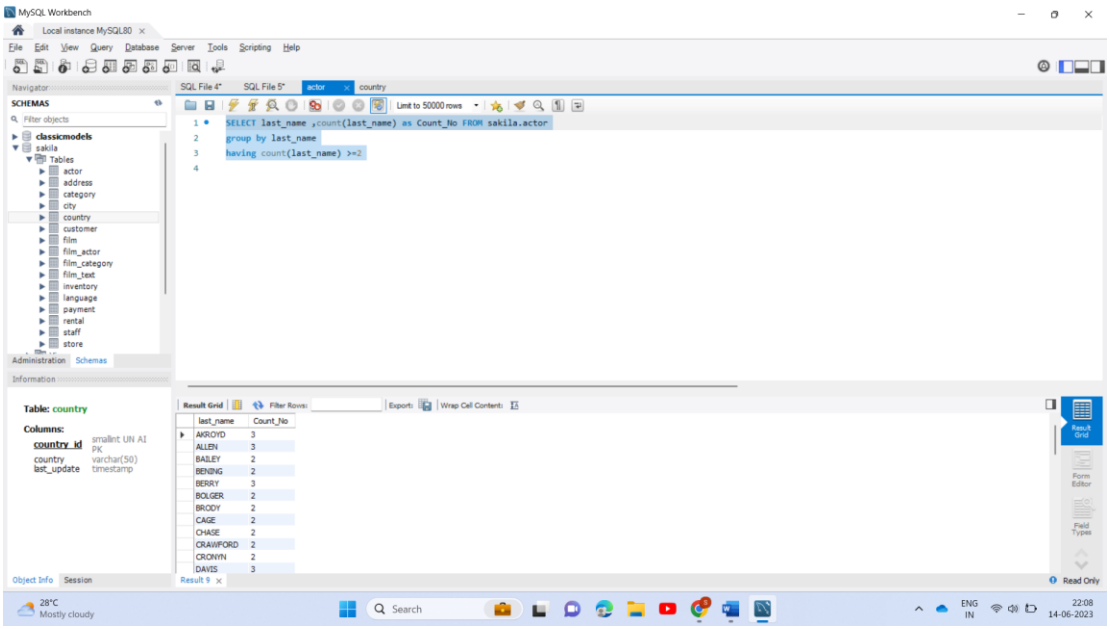
4. List the last names of actors, as well as how many actors have that last name.

```
SELECT last_name ,count(last_name) as Count_No FROM sakila.actor
group by last_name
```



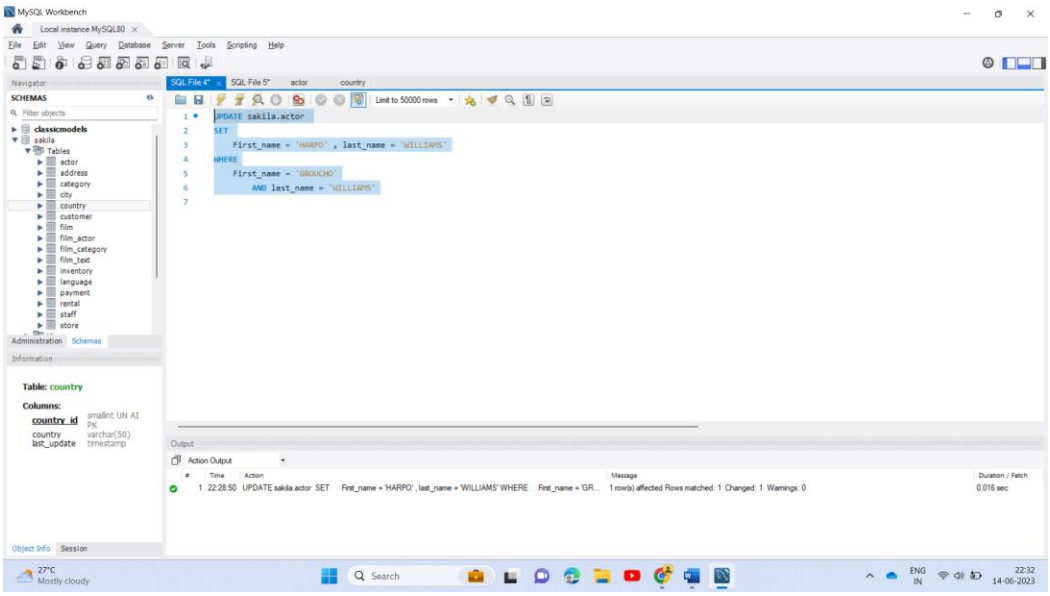
5. List last names of actors and the number of actors who have that last name, but only for names that are shared by at least two actors

```
SELECT last_name ,count(last_name) as Count_No FROM sakila.actor
group by last_name
having count(last_name) >=2
```



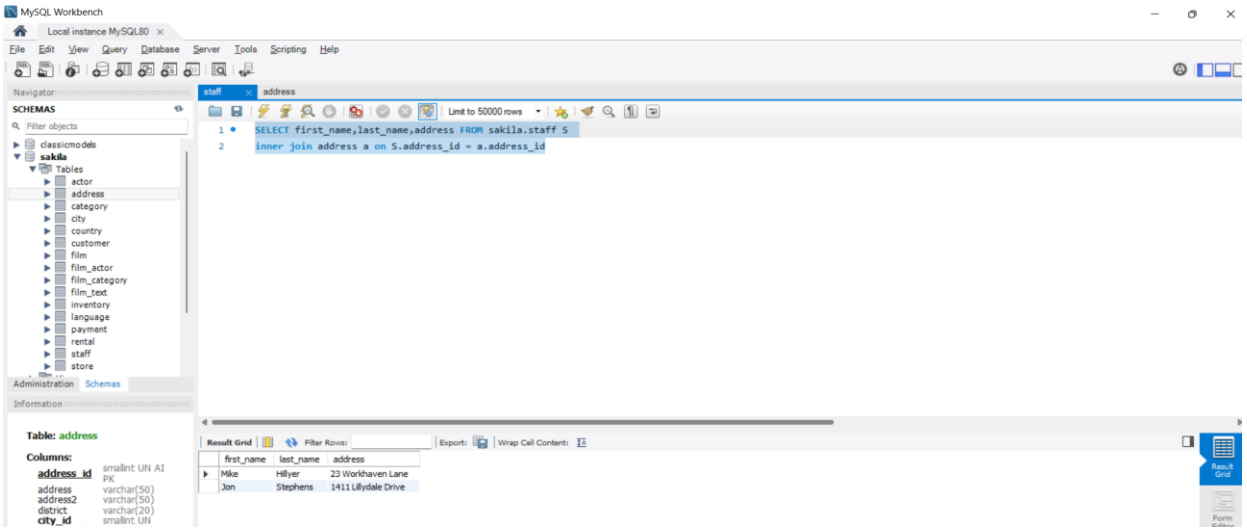
6. The actor HARPO WILLIAMS was accidentally entered in the actor table as GROUCHO WILLIAMS. Write a query to fix the record.

```
UPDATE sakila.actor
SET
    First_name = 'HARPO' , last_name = 'WILLIAMS' WHERE
    First_name = 'GROUCHO' AND last_name = 'WILLIAMS'
```



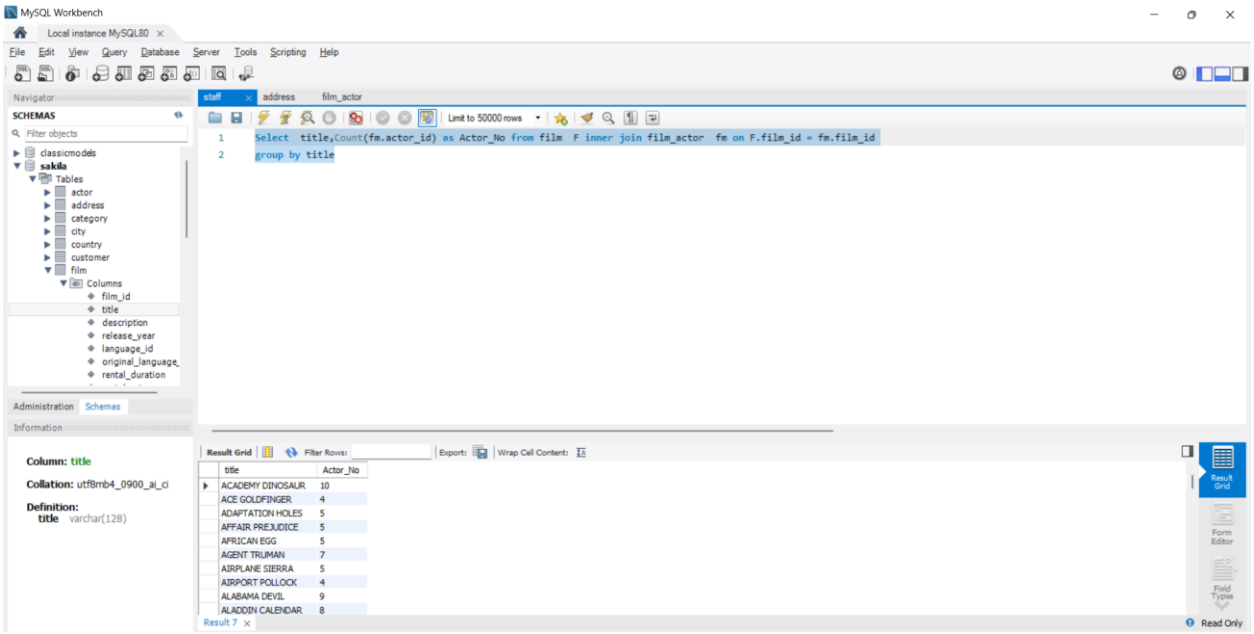
7. Use JOIN to display the first and last names, as well as the address, of each staff member. Use the tables staff and address:

```
SELECT first_name,last_name,address FROM sakila.staff S
inner join address a on S.address_id = a.address_id
```



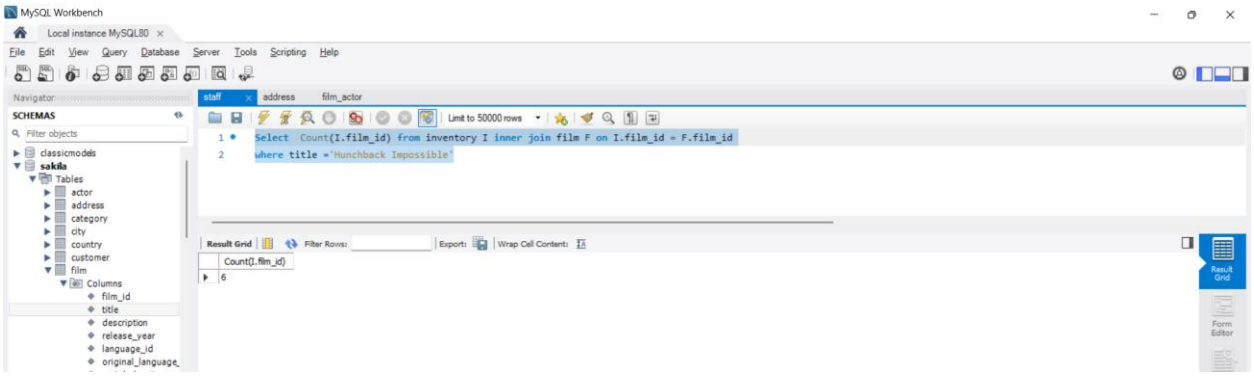
8. List each film and the number of actors who are listed for that film. Use tables film_actor and film. Use inner join.

```
Select  title,Count(fm.actor_id) as Actor_No from film  F inner
join film_actor  fm on F.film_id = fm.film_id
group by title
```



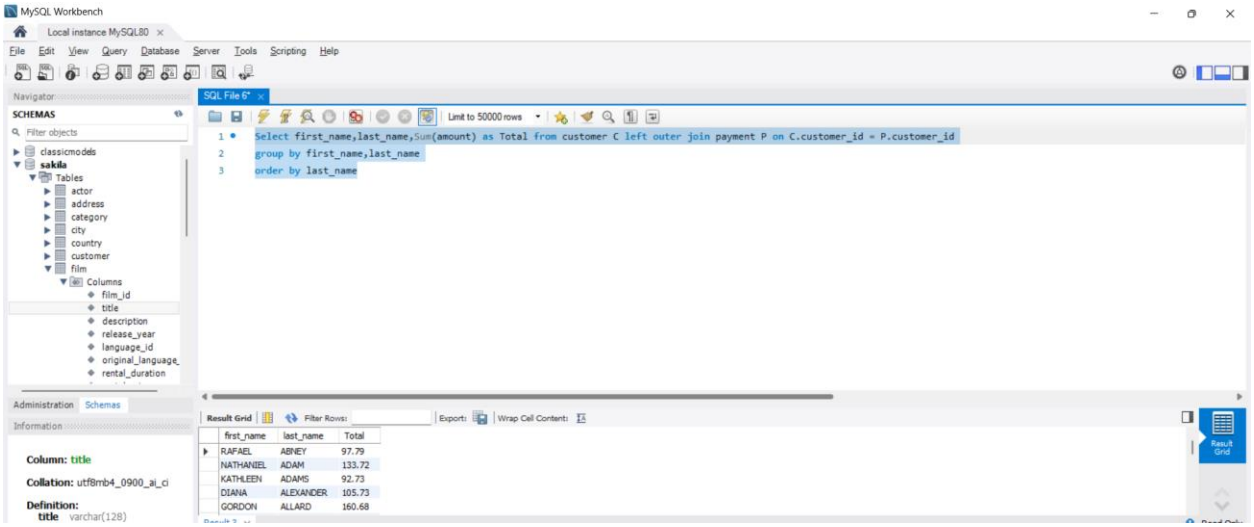
9. How many copies of the film Hunchback Impossible exist in the inventory system?

```
Select  Count(I.film_id) from inventory I inner join film F on
I.film_id = F.film_id
where title ='Hunchback Impossible'
```



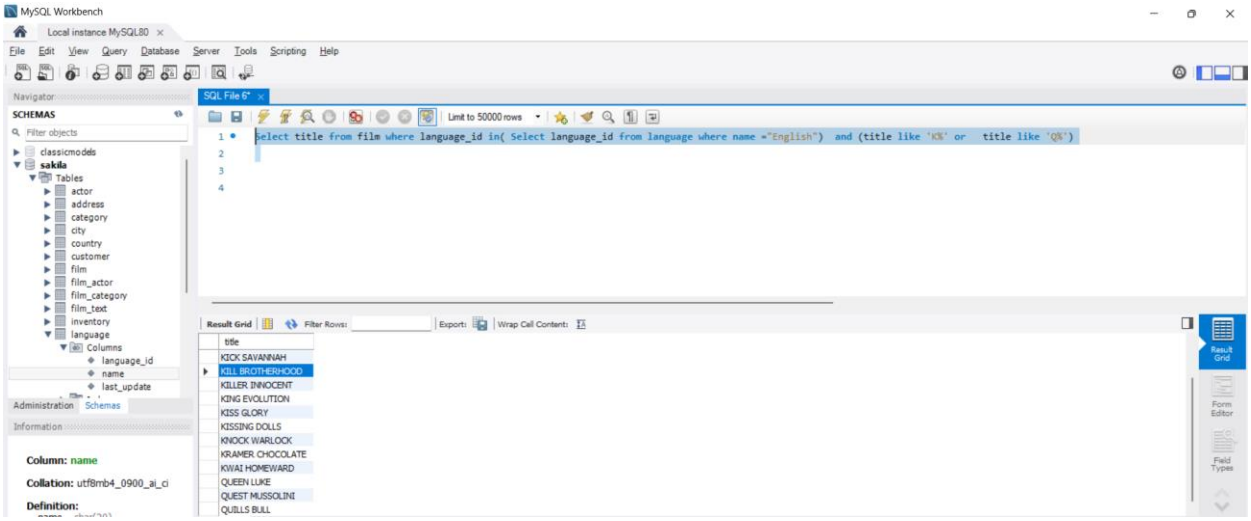
10. Using the tables payment and customer and the JOIN command, list the total paid by each customer. List the customers alphabetically by last name

```
Select first_name,last_name,Sum(amount) as Total from customer C
left outer join payment P on C.customer_id = P.customer_id
group by first_name,last_name
order by last_name
```



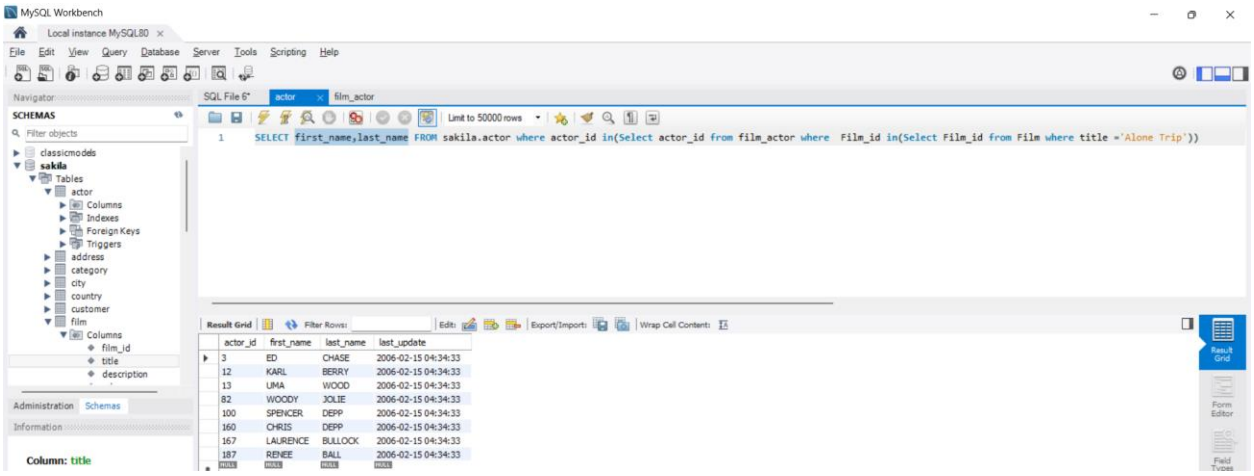
11. The music of Queen and Kris Kristofferson have seen an unlikely resurgence. As an unintended consequence, films starting with the letters `K` and `Q` have also soared in popularity. Use subqueries to display the titles of movies starting with the letters `K` and `Q` whose language is English.

```
Select title from film where language_id in( Select language_id
from language where name ="English")  and (title like 'K%' or
title like 'Q%')
```



12. Use subqueries to display all actors who appear in the film `Alone Trip`.

```
SELECT first_name,last_name FROM sakila.actor where actor_id
in(Select actor_id from film_actor where  Film_id in(Select
Film_id from Film where title ='Alone Trip'))
```

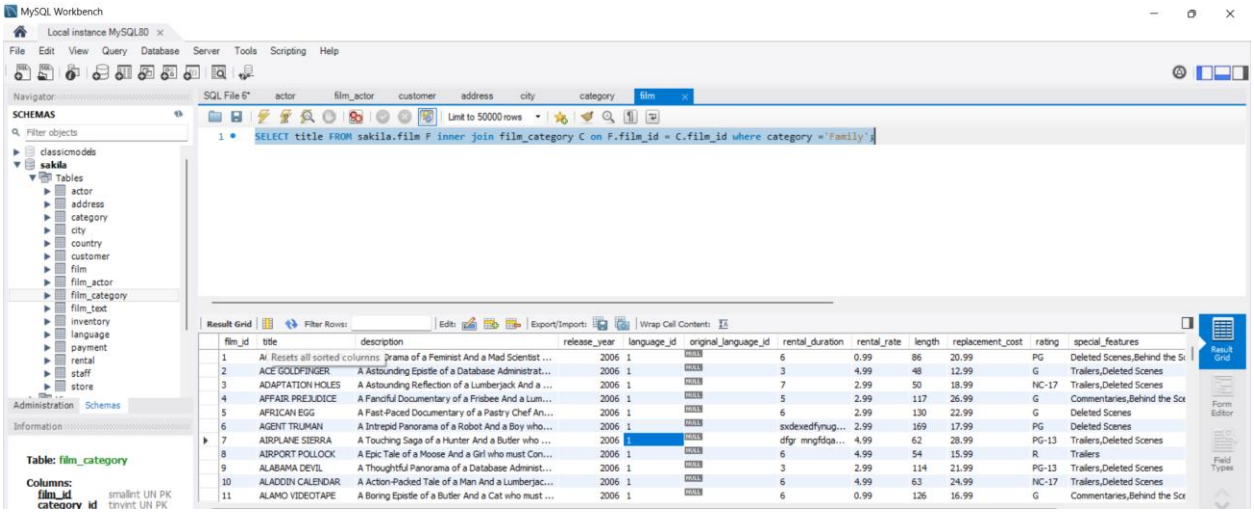


13. You want to run an email marketing campaign in Canada, for which you will need the names and email addresses of all Canadian customers. Use joins to retrieve this information.

```
SELECT First_name,last_name,email FROM sakila.Customer C inner
join sakila.address A on C.Address_id = A.address_id
inner join sakila.City Ci on Ci.city_id = A.city_id
inner join sakila.country Cn on Cn.country_id = Ci.country_id
where Cn.country ='Canada'
```

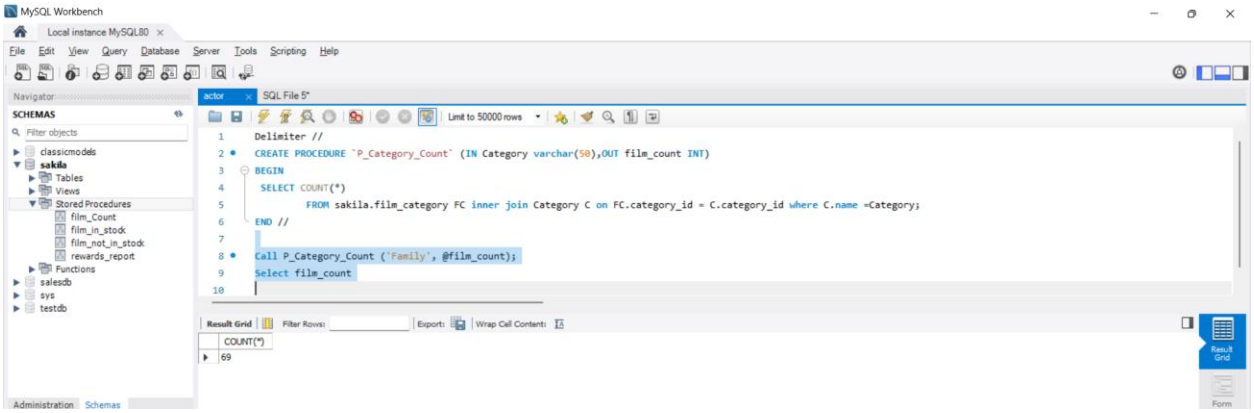
14. Sales have been lagging among young families, and you wish to target all family movies for a promotion. Identify all movies categorized as famiy films.

```
SELECT title FROM sakila.film F inner join film_category C on
F.film_id = C.film_id where category ='Family';
```



15. Create a Stored procedure to get the count of films in the input category (IN category_name, OUT count)
16.

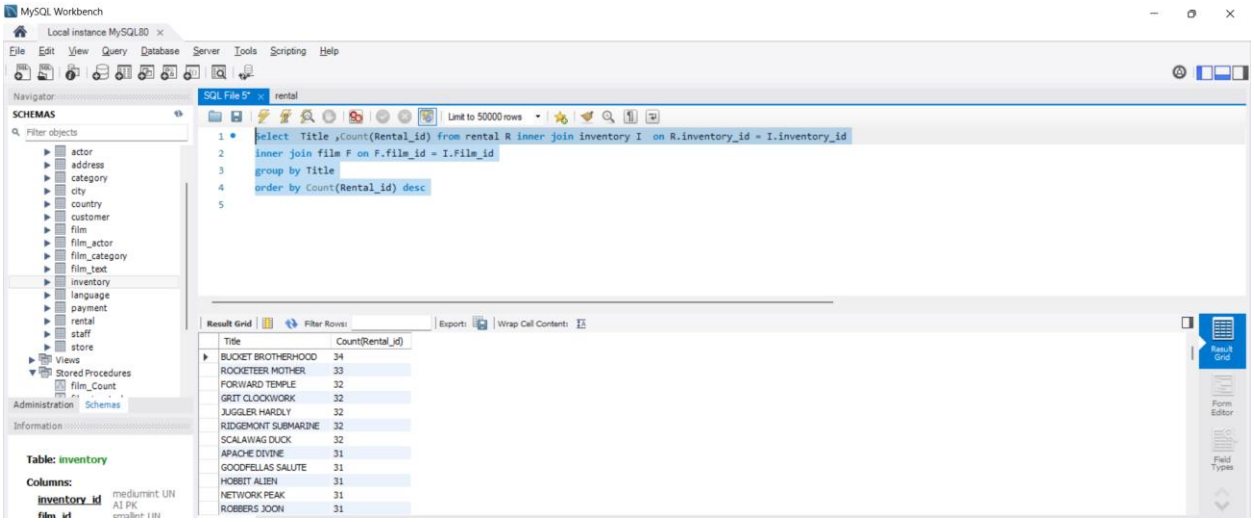
```
Delimiter //
CREATE PROCEDURE `P_Category_Count` (IN Category varchar(50),OUT
film_count INT)
BEGIN
 SELECT COUNT(*)
            FROM sakila.film_category FC inner join Category C on
FC.category_id = C.category_id where C.name =Category;
END //
```



17. Display the most frequently rented movies in descending order.

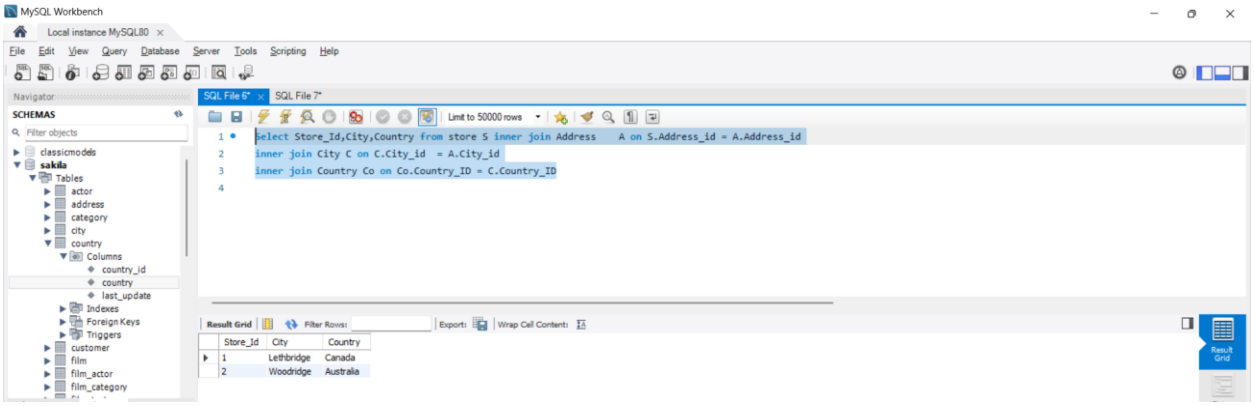
```
Select  Title ,Count(Rental_id) from rental R inner join
inventory I  on R.inventory_id = I.inventory_id
inner join film F on F.film_id = I.Film_id
```

```
group by Title
order by Count(Rental_id) desc
```



18. Write a query to display for each store its store ID, city, and country.

```
Select Store_Id,City,Country from store S inner join
Address    A on S.Address_id = A.Address_id
inner join City C on C.City_id  = A.City_id
inner join Country Co on Co.Country_ID = C.Country_ID
```



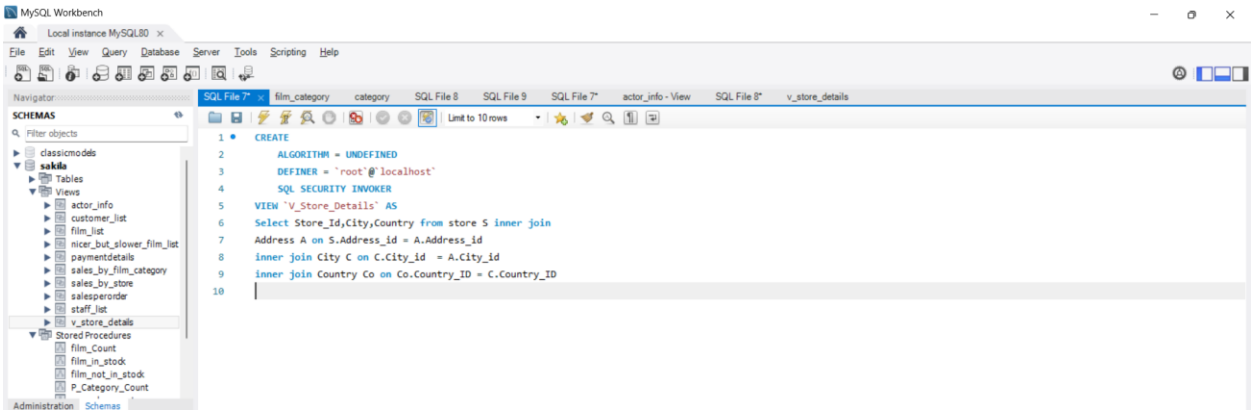
19. List the genres and its gross revenue.

```
SELECT C.name, Sum(p.Amount) FROM category C inner join
film_category FC on C.category_id = FC.category_id
inner join  inventory I inner join rental R on I.inventory_id =
R.inventory_id
inner join payment P on P.rental_id on R.rental_id
group by   C.name
order by Sum(p.Amount) desc limit 5
```

20. Create a View for the above query(18)

```
CREATE
  ALGORITHM = UNDEFINED
    DEFINER = `root`@`localhost`
```

```
    SQL SECURITY INVOKER
VIEW `V_Store_Details` AS
Select Store_Id,City,Country from store S inner join
Address    A on S.Address_id = A.Address_id
inner join City C on C.City_id  = A.City_id
inner join Country Co on Co.Country_ID = C.Country_ID
```



21. Select top 5 genres in gross revenue view.