STAT 37820
HW 1
Siying Cao
UCID 10445633

Q0. How many tables are there in the Sakila database?

```
USE sakila;
SHOW TABLES;
# Method 1 (this includes views)
SELECT FOUND_ROWS();
# Method 2 (this only counts tables, excluding views)
SELECT COUNT(*) FROM information_schema.tables
WHERE table_schema='sakila'
AND table_type='base table';
```

Output: 23 tables; but if we exclude views, only 16.

Q1. Provide the code to list the DVDs that were returned later than their due date

```
SELECT DISTINCT inventory.inventory_id FROM inventory, rental, film
WHERE inventory.inventory_id=rental.inventory_id
AND film.film_id=inventory.film_id
AND return_date IS NOT NULL
AND (rental_date + INTERVAL film.rental_duration DAY) < return_date;
```

Q2. How many DVSs were returned on time? Provide the code and the output

```
SELECT COUNT(t1.inventory_id)
FROM
(SELECT inventory.inventory_id, COUNT(rental.rental_id) AS total_rent
FROM rental, inventory
WHERE inventory.inventory_id=rental.inventory_id
GROUP BY inventory.inventory_id) AS t1
LEFT JOIN
(SELECT inventory.inventory_id, COUNT(rental.rental_id) AS on_time
FROM rental, inventory, film
```

```
WHERE inventory.inventory_id=rental.inventory_id
AND film.film_id=inventory.film_id
AND return_date IS NOT NULL
AND (rental_date + INTERVAL film.rental_duration DAY) > return_date
GROUP BY inventory.inventory_id) AS t2
ON t1.inventory_id=t2.inventory_id
WHERE total_rent=on_time;
```

Output: 603 out of 4581 DVDs were returned on time. I approach the question by first counting the number of times a DVD has been rent, and then counting the times that inventory was returned on time among all its rentals. A DVD has been returned on time if and only if the two numbers coincide.

Q3. Provide the code to find the category name, category id, and the number of movies in each category

```
SELECT category.name, category.category_id, COUNT(film_category.film_id) AS total
FROM category, film_category
WHERE film_category.category_id=category.category_id
GROUP BY category.category_id
ORDER BY category.name;
```

Q4. Provide the code to find the names of actors and the drama movies that they acted in. How many records are there in your query result?

```
SELECT actor.actor_id, CONCAT(actor.first_name, ',', actor.last_name) AS Actor_Name,
film.title
FROM actor, film, film_actor, category, film_category
WHERE film_actor.actor_id=actor.actor_id
AND film_actor.film_id=film.film_id
AND category.name='Drama'
AND film_category.category_id=category.category_id
AND film_category.film_id=film.film_id
ORDER BY Actor_Name, title;
```

Output: there are 350 rows of records returned under my query. Below is the first ten rows returned.

actor_id	Actor_Name	title
132	ADAM,HOPPER	TORQUE BOUND
165	AL,GARLAND	CHITTY LOCK
165	AL,GARLAND	DALMATIONS SWEDEN
165	AL,GARLAND	JACKET FRISCO
173	ALAN,DREYFUSS	GREEDY ROOTS
146	ALBERT,JOHANSSON	APOLLO TEEN
146	ALBERT,JOHANSSON	DECEIVER BETRAYED
146	ALBERT,JOHANSSON	LEBOWSKI SOLDIERS
146	ALBERT,JOHANSSON	WEST LION

Q5. Provide the code to find the number of customers living in each city

```
SELECT city.city_id, city.city, COUNT(customer.customer_id) AS total
FROM customer, address, city
WHERE customer.address_id=address.address_id
AND address.city_id=city.city_id
GROUP BY city.city
#HAVING total>1
ORDER BY city.city;
```

Q6. Provide the code to find the names of the top 20 customers who have rented the most movies. Print the first and last rows of your query result.

```
SELECT customer.customer_id, CONCAT(customer.first_name, ',', customer.last_name) AS name,COUNT(rental_id)
FROM customer, rental
WHERE customer.customer_id=rental.customer_id
GROUP BY customer.customer_id
ORDER BY -COUNT(rental_id)
LIMIT 20;
```

Output:

customer_id	name	COUNT(rental_id)
148	ELEANOR,HUNT	46
	.....	
348	ROGER,QUINTANILLA	36

Q7. Find the names of the top 5 most rented movies. How many times were they rented? Provide the code and output.

SELECT film.film_id, film.title, COUNT(rental.rental_id)

FROM rental, inventory, film

WHERE rental.inventory_id=inventory.inventory_id

AND inventory.film_id=film.film_id

GROUP BY film.film_id

ORDER BY -COUNT(rental.rental_id), title

LIMIT 5;

Output:

film_id	title	COUNT(rental.rental_id)
103	BUCKET BROTHERHOOD	34
738	ROCKETEER MOTHER	33
331	FORWARD TEMPLE	32
382	GRIT CLOCKWORK	32
489	JUGGLER HARDLY	32

Q8. Provide the code to find the movies that each actor has acted in

SELECT actor.actor_id, CONCAT(first_name, ',', last_name) AS actor_name, film.film_id, film.title

FROM actor, film, film_actor

WHERE film_actor.actor_id=actor.actor_id

AND film_actor.film_id=film.film_id;

Q9. Find the top 5 actors whose movies were rented the most. Provide the code and output.

SELECT actor.actor_id, CONCAT(first_name, ',', last_name) AS actor_name, COUNT(rental.rental_id)

```
FROM actor, film, film_actor, rental, inventory
WHERE film_actor.actor_id=actor.actor_id
AND film_actor.film_id=film.film_id
AND rental.inventory_id=inventory.inventory_id
AND inventory.film_id=film.film_id
GROUP BY actor.actor_id
ORDER BY -COUNT(rental.rental_id)
LIMIT 5;
```

Output:

actor_id	actor_name	COUNT(rental.rental_id)
107	GINA,DEGENERES	753
181	MATTHEW,CARREY	678
198	MARY,KEITEL	674
144	ANGELA,WITHERSPOON	654
102	WALTER,TORN	640