

Example of Data Analysis Using SQL

Rifah Nur Hasanah

1. Display first_name and last_name in the actor table

```
SELECT first_name, last_name
```

```
FROM actor a
```

	ABC first_name ▼	ABC last_name ▼
1	PENELOPE	GUINNESS
2	NICK	WAHLBERG
3	ED	CHASE
4	JENNIFER	DAVIS
5	JOHNNY	LOLLOBRIGIDA

2. Display the combined first_name and last_name in the actor table in lower case. Hint : Use || to combine 2 columns, example: SELECT a || b

```
SELECT LOWER(first_name || " " || last_name) AS Actor_Name
```

```
FROM actor a
```

	ABC Actor_Name ▼
1	penelope guinness
2	nick wahlberg
3	ed chase
4	jennifer davis
5	johnny lollobrigida

3. Look for the named actor "JOE" then displays actor_id, first_name, and last_name

```
SELECT actor_id, first_name, last_name
```

```
FROM actor a
```

```
WHERE first_name = 'JOE'
```

	123 actor_id ▼	ABC first_name ▼	ABC last_name ▼
1	9	JOE	SWANK

4. Still like the previous problem, now look for people whose last name contains the word "LI", but try to sort them by last name. If there are people with the same last name, then prioritize sorting by first name.

```
SELECT actor_id, first_name, last_name
```

```
FROM actor a
```

```
WHERE last_name LIKE '%LI%'
```

```
ORDER BY last_name, first_name
```

	123 actor_id ▼	ABC first_name ▼	ABC last_name ▼
1	86	GREG	CHAPLIN
2	82	WOODY	JOLIE
3	34	AUDREY	OLIVIER
4	15	CUBA	OLIVIER
5	172	GROUCHO	WILLIAMS
6	137	MORGAN	WILLIAMS
7	72	SEAN	WILLIAMS
8	83	BEN	WILLIS
9	96	GENE	WILLIS
10	164	HUMPHREY	WILLIS

5. Dengan fungsi IN di SQL, Tampilkan ID Negara Afghanistan, Bangladesh, dan China

```
SELECT country_id, country
```

```
FROM country c
```

```
WHERE country IN ('Afghanistan','Bangladesh','China')
```

	123 country_id ▼	ABC country ▼
1	1	Afghanistan
2	12	Bangladesh
3	23	China

6. Tampilkan semua nama belakang aktor dan juga hitung berapa banyak aktor dengan nama belakang tersebut

```
SELECT last_name, COUNT(last_name) AS Jumlah
```

```
FROM actor a
```

```
GROUP BY last_name
```

	ABC last_name ▼	123 Jumlah ▼
1	AKROYD	3
2	ALLEN	3
3	ASTAIRE	1
4	BACALL	1
5	BAILEY	2
6	BALE	1
7	BALL	1
8	BARRYMORE	1

7. Sama seperti dengan nomot 6, akan tetapi sekarang buang nama belakang aktor yang hanya dimiliki 1 orang

SELECT last_name, **COUNT**(last_name) **AS** Jumlah

FROM actor a

GROUP BY last_name

HAVING Jumlah >1

	ABC last_name ▼	123 Jumlah ▼
1	AKROYD	3
2	ALLEN	3
3	BAILEY	2
4	BENING	2
5	BERRY	3
6	BOLGER	2
7	BRODY	2
8	CAGE	2

8. Find the first name, last name, and address of each staff member, using the staff and address table

SELECT s.first_name, s.last_name, a.address

FROM staff s **LEFT JOIN** address a **ON** s.address_id = a.address_id

	ABC first_name ▼	ABC last_name ▼	ABC address ▼
1	Mike	Hillyer	23 Workhaven Lane
2	Jon	Stephens	1411 Lillydale Drive

9. Find how many sales/amount each staff member made in August 2005, use the staff and payment table

```
SELECT s.staff_id, SUM(p.amount)
FROM staff s LEFT JOIN payment p
ON s.staff_id = p.staff_id
WHERE p.payment_date >= '2005-08-01' AND p.payment_date < '2005-09-01'
GROUP BY s.staff_id
```

	123 staff_id ▼	123 SUM(p.amount) ▼
1	1	11,853.6499999995
2	2	12,218.4799999995

10. Display movie titles and how many actors are in each movie

```
SELECT f.title, COUNT(fa.actor_id )
FROM film f INNER JOIN film_actor fa
ON f.film_id = fa.film_id
GROUP BY f.title
```

	ABC title ▼	123 COUNT(fa.actor_id) ▼
1	ACADEMY DINOSAUR	10
2	ACE GOLDFINGER	4
3	ADAPTATION HOLES	5
4	AFFAIR PREJUDICE	5
5	AFRICAN EGG	5
6	AGENT TRUMAN	7
7	AIRPLANE SIERRA	5

11. Look for customer information and how much each customer incurs. Sort search results alphabetically by customer's last name.

```
SELECT C.customer_id, c.last_name, SUM(p.amount)
FROM customer c LEFT JOIN payment p
ON c.customer_id = p.customer_id
GROUP BY C.customer_id
```

ORDER BY c.last_name

	customer_id	last_name	SUM(p.amount)
1	505	ABNEY	97.79
2	504	ADAM	133.72
3	36	ADAMS	92.73
4	96	ALEXANDER	105.73
5	470	ALLARD	160.68
6	27	ALLEN	126.69
7	220	ALVAREZ	114.73
8	11	ANDERSON	106.76
9	326	ANDREW	96.75

12. The company wants to create a marketing campaign via email in Canada. Search for all customer emails originating from Canada, use CTE if possible

```
WITH Location_info AS (
    SELECT address_id, a.city_id, c.country_id, city, country
    FROM address a INNER JOIN city c ON a.address_id = c.city_id
    INNER JOIN country c2 ON c.city_id = c2.country_id
)
SELECT email
FROM customer c3
WHERE address_id IN (SELECT address_id FROM Location_info WHERE
country='Canada')
```

	email
1	SANDRA.MARTIN@sakilacustomer.org

13. Display the names of the most frequently rented films and sort them from the most popular--Gabungkan 3 tabel

```
SELECT f.title, COUNT(r.rental_id)
FROM rental r INNER JOIN inventory i
ON r.inventory_id = i.inventory_id
```

INNER JOIN film f

ON i.film_id = f.film_id

GROUP BY f.film_id

ORDER BY COUNT(r.rental_id) DESC

	ABC title	123 COUNT(r.rental_id)
1	BUCKET BROTHERHOOD	34
2	ROCKETEER MOTHER	33
3	SCALAWAG DUCK	32
4	RIDGEMONT SUBMARINE	32
5	JUGGLER HARDLY	32
6	GRIT CLOCKWORK	32
7	FORWARD TEMPLE	32
8	ZORRO ARK	31
9	WIFE TURN	31

14. Create a query that shows how much revenue/amount each store earned.

SELECT s.store_id, SUM(p.amount)

FROM payment p INNER JOIN rental r ON P.rental_id = R.rental_id

INNER JOIN inventory i ON R.inventory_id = I.inventory_id

INNER JOIN store s ON i.store_id = s.store_id

GROUP BY s.store_id

	123 store_id	123 SUM(p.amount)
1	1	33,679.790000005
2	2	33,726.770000005

15. Look for the 5 film genres with the highest gross revenue, then sort them from the best-selling ones first. Hint ; Genres are found in the film_category table.

SELECT c.name AS genre, SUM(amount) AS gross_revenue

FROM payment p LEFT JOIN rental r on P.rental_id = R.rental_id

LEFT JOIN inventory i ON i.inventory_id = r.inventory_id

LEFT JOIN film f ON f.film_id = i.inventory_id

LEFT JOIN film_category fc ON fc.film_id = i.film_id

LEFT JOIN category c ON c.category_id = fc.category_id

GROUP BY genre

ORDER BY gross_revenue DESC

LIMIT 5

	ABC genre ▼	123 gross_revenue ▼
1	Sports	5,314.20999999998
2	Sci-Fi	4,756.97999999999
3	Animation	4,656.29999999999
4	Drama	4,587.38999999999
5	Comedy	4,383.57999999999