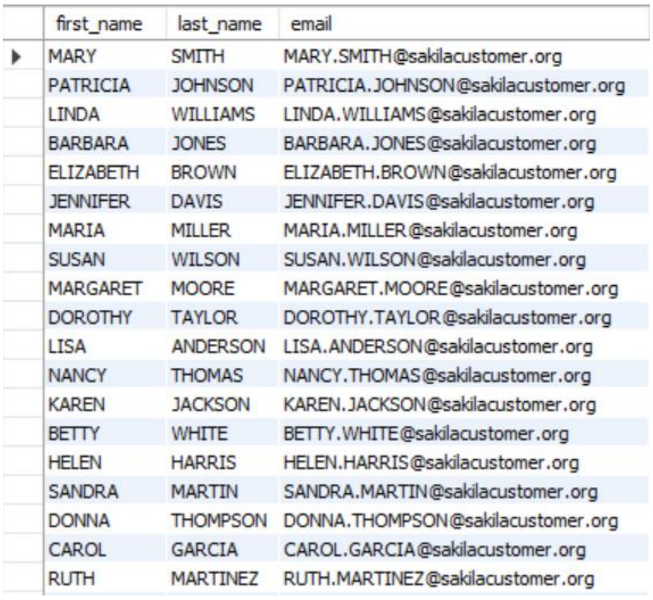
1.- SELECT & FROM

"I'm going to send an email letting our customers now there has been a management change.Could you pull a list of the **first name**,**last name** and **email** of each of our customers?"

Result:



first_name	last_name	email
MARY	SMITH	MARY.SMITH@sakilacustomer.org
PATRICIA	JOHNSON	PATRICIA.JOHNSON@sakilacustomer.org
LINDA	WILLIAMS	LINDA.WILLIAMS@sakilacustomer.org
BARBARA	JONES	BARBARA.JONES@sakilacustomer.org
ELIZABETH	BROWN	ELIZABETH.BROWN@sakilacustomer.org
JENNIFER	DAVIS	JENNIFER.DAVIS@sakilacustomer.org
MARIA	MILLER	MARIA.MILLER@sakilacustomer.org
SUSAN	WILSON	SUSAN.WILSON@sakilacustomer.org
MARGARET	MOORE	MARGARET.MOORE@sakilacustomer.org
DOROTHY	TAYLOR	DOROTHY.TAYLOR@sakilacustomer.org
LISA	ANDERSON	LISA.ANDERSON@sakilacustomer.org
NANCY	THOMAS	NANCY.THOMAS@sakilacustomer.org
KAREN	JACKSON	KAREN.JACKSON@sakilacustomer.org
BETTY	WHITE	BETTY.WHITE@sakilacustomer.org
HELEN	HARRIS	HELEN.HARRIS@sakilacustomer.org
SANDRA	MARTIN	SANDRA.MARTIN@sakilacustomer.org
DONNA	THOMPSON	DONNA.THOMPSON@sakilacustomer.org
CAROL	GARCIA	CAROL.GARCIA@sakilacustomer.org
RUTH	MARTINEZ	RUTH.MARTINEZ@sakilacustomer.org

Answer:

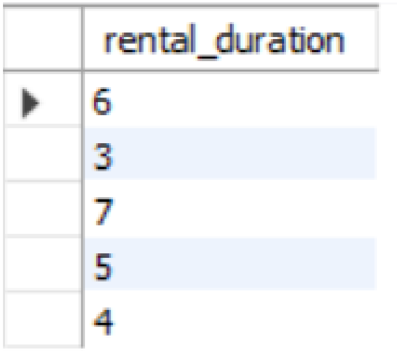
```
USE mavenmovies;

SELECT first_name,last_name,email FROM customer;
```

2.-SELECT DISTINCT

“My understanding is that we have titles that we rent for duration of 3,5 or 7 days. Could you pull the record of our films and see if there are any other rental durations?”

Result:



rental_duration
6
3
7
5
4

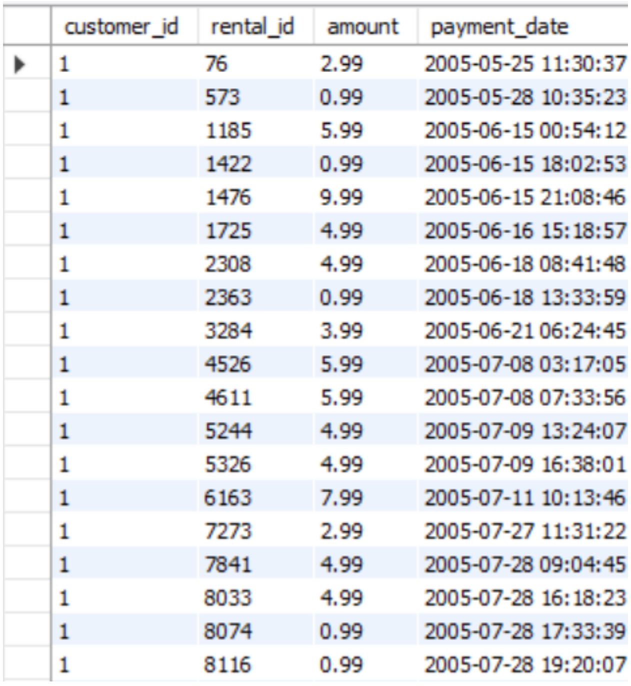
Answer:

```
SELECT DISTINCT(rental_duration) FROM film;
```

3.-WHERE

"I'd like to look at payment records for our long-term customers to learn about their purchase patterns. Could you pull all payments from our first 100 customers(based on customer ID)"

Result:



customer_id	rental_id	amount	payment_date
1	76	2.99	2005-05-25 11:30:37
1	573	0.99	2005-05-28 10:35:23
1	1185	5.99	2005-06-15 00:54:12
1	1422	0.99	2005-06-15 18:02:53
1	1476	9.99	2005-06-15 21:08:46
1	1725	4.99	2005-06-16 15:18:57
1	2308	4.99	2005-06-18 08:41:48
1	2363	0.99	2005-06-18 13:33:59
1	3284	3.99	2005-06-21 06:24:45
1	4526	5.99	2005-07-08 03:17:05
1	4611	5.99	2005-07-08 07:33:56
1	5244	4.99	2005-07-09 13:24:07
1	5326	4.99	2005-07-09 16:38:01
1	6163	7.99	2005-07-11 10:13:46
1	7273	2.99	2005-07-27 11:31:22
1	7841	4.99	2005-07-28 09:04:45
1	8033	4.99	2005-07-28 16:18:23
1	8074	0.99	2005-07-28 17:33:39
1	8116	0.99	2005-07-28 19:20:07

Answer:

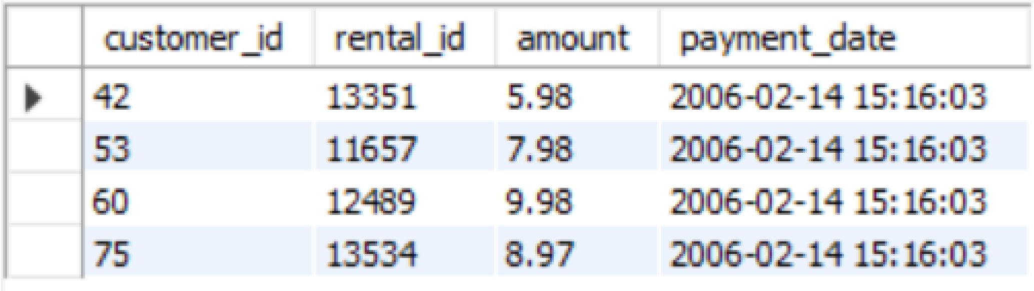
```
SELECT customer_id,rental_id, amount, payment_date FROM payment

WHERE customer_id < 100;
```

4.-WHERE & AND

"The payment data you gave me on our first 100 customers was great – thank you!. Now I'd love to see just payments over $ for those same customers,since January 1st, 2006."

Result:



customer_id	rental_id	amount	payment_date
42	13351	5.98	2006-02-14 15:16:03
53	11657	7.98	2006-02-14 15:16:03
60	12489	9.98	2006-02-14 15:16:03
75	13534	8.97	2006-02-14 15:16:03

Answer:

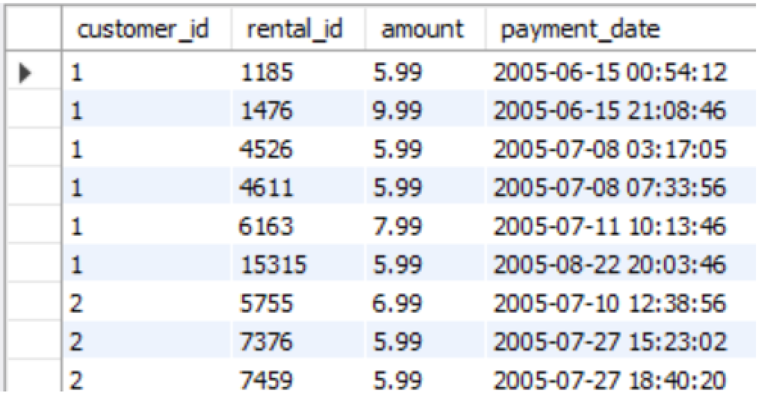
```
SELECT customer_id,rental_id, amount, payment_date FROM payment

WHERE customer_id < 100 AND amount > 5 AND payment_date > 20060101;
```

5.-WHERE & OR

“The data you shared previously on customers 42,53,60 and 75 was good to see. Now could you please write a query to pull all payments from those specific customers along with payments over $5, from any customer?”

Result:



	customer_id	rental_id	amount	payment_date
▶	1	1185	5.99	2005-06-15 00:54:12
	1	1476	9.99	2005-06-15 21:08:46
	1	4526	5.99	2005-07-08 03:17:05
	1	4611	5.99	2005-07-08 07:33:56
	1	6163	7.99	2005-07-11 10:13:46
	1	15315	5.99	2005-08-22 20:03:46
	2	5755	6.99	2005-07-10 12:38:56
	2	7376	5.99	2005-07-27 15:23:02
	2	7459	5.99	2005-07-27 18:40:20

Answer:

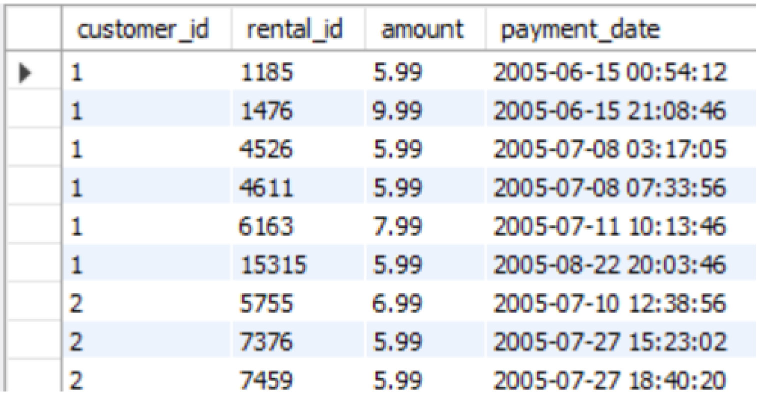
SELECT customer_id,rental_id, amount, payment_date FROM payment

WHERE customer_id = 42 AND customer_id =53 AND customer_id =60 AND customer_id =75 OR amount > 5;

6.-IN

“The data you shared previously on customers 42,53,60 and 75 was good to see. Now could you please write a query to pull all payments from those specific customers along with payments over $5, from any customer?”

Result:



	customer_id	rental_id	amount	payment_date
▶	1	1185	5.99	2005-06-15 00:54:12
	1	1476	9.99	2005-06-15 21:08:46
	1	4526	5.99	2005-07-08 03:17:05
	1	4611	5.99	2005-07-08 07:33:56
	1	6163	7.99	2005-07-11 10:13:46
	1	15315	5.99	2005-08-22 20:03:46
	2	5755	6.99	2005-07-10 12:38:56
	2	7376	5.99	2005-07-27 15:23:02
	2	7459	5.99	2005-07-27 18:40:20

Answer:

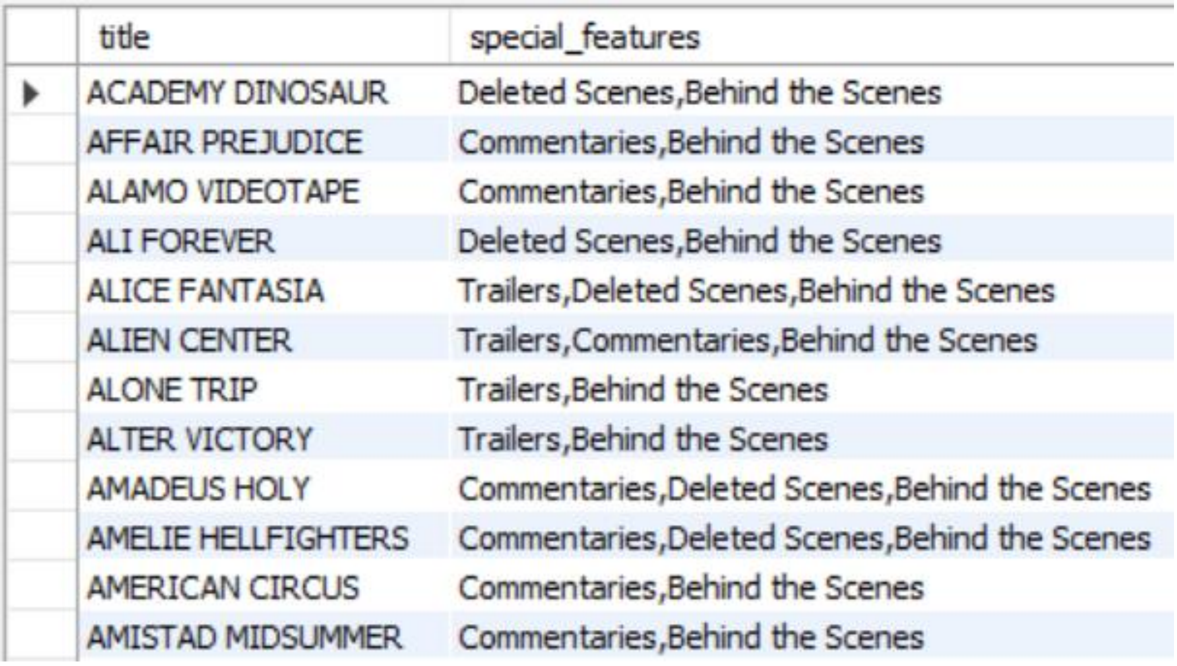
SELECT customer_id,rental_id, amount, payment_date FROM payment

WHERE customer_id in(42,53,60,75) OR amount > 5;

7.- LIKE

“We need to understand the special features in our films. Could you pull a list of films wich include a **behind the scenes** special features ”

Result:



	title	special_features
▶	ACADEMY DINOSAUR	Deleted Scenes,Behind the Scenes
	AFFAIR PREJUDICE	Commentaries,Behind the Scenes
	ALAMO VIDEOTAPE	Commentaries,Behind the Scenes
	ALI FOREVER	Deleted Scenes,Behind the Scenes
	ALICE FANTASIA	Trailers,Deleted Scenes,Behind the Scenes
	ALIEN CENTER	Trailers,Commentaries,Behind the Scenes
	ALONE TRIP	Trailers,Behind the Scenes
	ALTER VICTORY	Trailers,Behind the Scenes
	AMADEUS HOLY	Commentaries,Deleted Scenes,Behind the Scenes
	AMELIE HELLFIGHTERS	Commentaries,Deleted Scenes,Behind the Scenes
	AMERICAN CIRCUS	Commentaries,Behind the Scenes
	AMISTAD MIDSUMMER	Commentaries,Behind the Scenes

Answer:

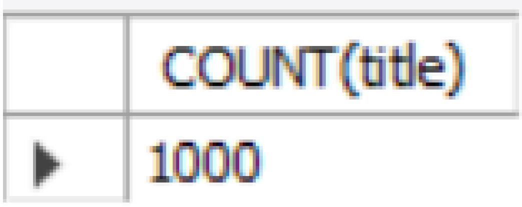
SELECT title, special_features FROM film

WHERE special_features LIKE '%Behind the Scenes%';

8.-COUNT

“Please count the total of films”

Result:



	COUNT(title)
▶	1000

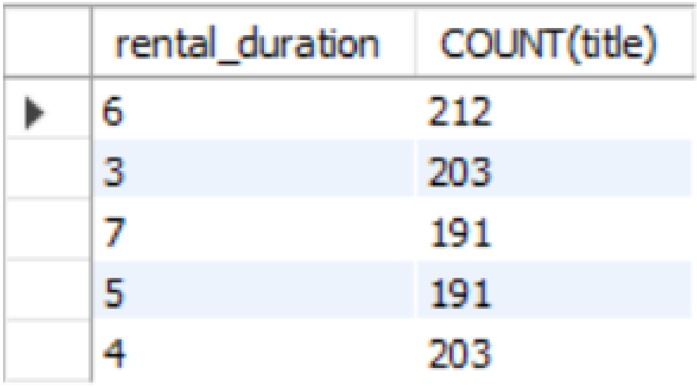
Answer:

SELECT COUNT(title) FROM film;

7.-GROUP BY

“I need to get a quick overview of how long our movies tend to be rented out for. Could you please pull a**count of tittle** sliced by** rental duration**”

Result:



rental_duration	COUNT(title)
6	212
3	203
7	191
5	191
4	203

Answer:

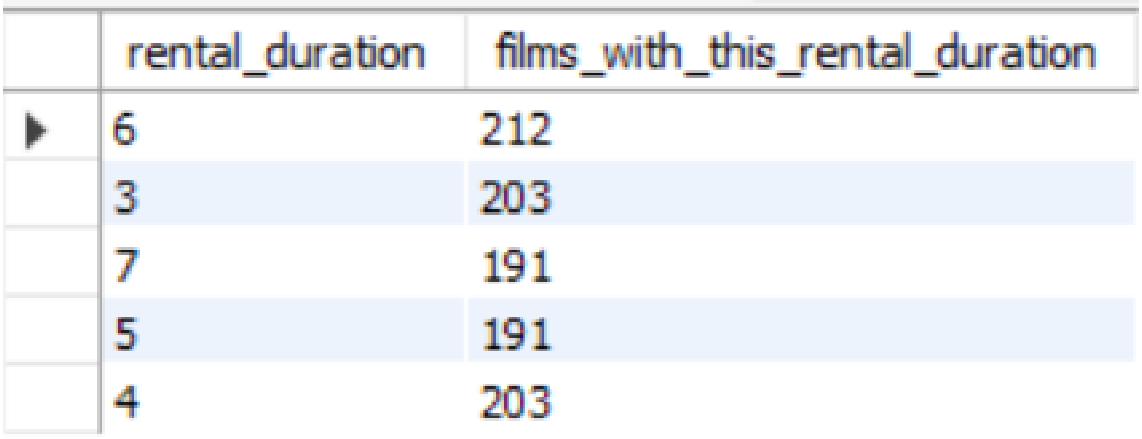
SELECT rental_duration, COUNT(title) FROM film

GROUP BY rental_duration;

8.-AS

“I need to get a quick overview of how long our movies tend to be rented out for. Could you please pull a**count of tittle** sliced by** rental duration**”

Result:



rental_duration	films_with_this_rental_duration
6	212
3	203
7	191
5	191
4	203

Answer:

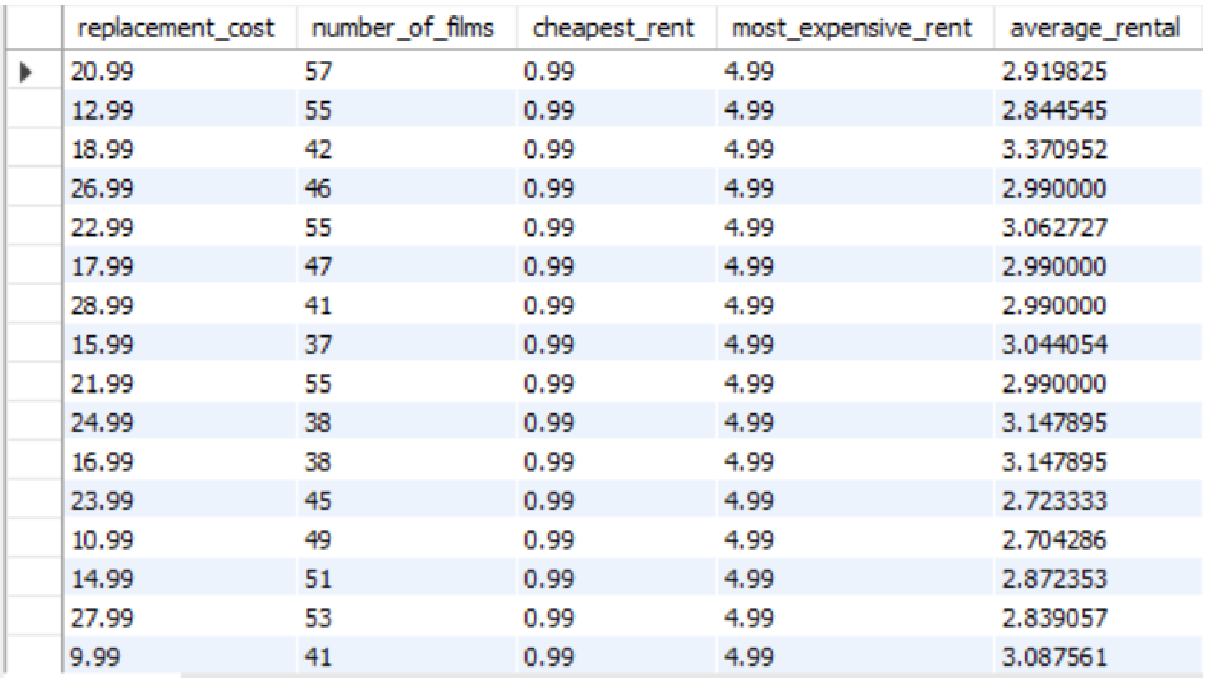
SELECT rental_duration, COUNT(title) AS films_with_this_rental_duration FROM film

GROUP BY rental_duration;

9.- Aggregate functions

“I’m wondering if we charge more for a rental when the replacement cost is higher. Can you help me pull a *count of films*, along with the *average*, *min* and *max rental rate* grouped by replacement cost?”

Result:



replacement_cost	number_of_films	cheapest_rent	most_expensive_rent	average_rental
20.99	57	0.99	4.99	2.919825
12.99	55	0.99	4.99	2.844545
18.99	42	0.99	4.99	3.370952
26.99	46	0.99	4.99	2.990000
22.99	55	0.99	4.99	3.062727
17.99	47	0.99	4.99	2.990000
28.99	41	0.99	4.99	2.990000
15.99	37	0.99	4.99	3.044054
21.99	55	0.99	4.99	2.990000
24.99	38	0.99	4.99	3.147895
16.99	38	0.99	4.99	3.147895
23.99	45	0.99	4.99	2.723333
10.99	49	0.99	4.99	2.704286
14.99	51	0.99	4.99	2.872353
27.99	53	0.99	4.99	2.839057
9.99	41	0.99	4.99	3.087561

Answer:

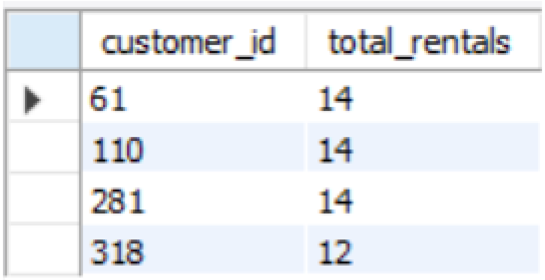
SELECT replacement_cost, COUNT(film_id) AS number_of_films, MIN(rental_rate) AS cheapest_rent, MAX(rental_rate) AS most_expensive_rent, AVG(rental_rate) AS average_rental FROM film

GROUP BY replacement_cost;

10.-HAVING

“I’d like to talk to customers that have not rented much form us to understand if there is something we could be doing better. Could you pull of customers_ids with less than 15 rentals all time”

Result:



customer_id	total_rentals
61	14
110	14
281	14
318	12

Answer:

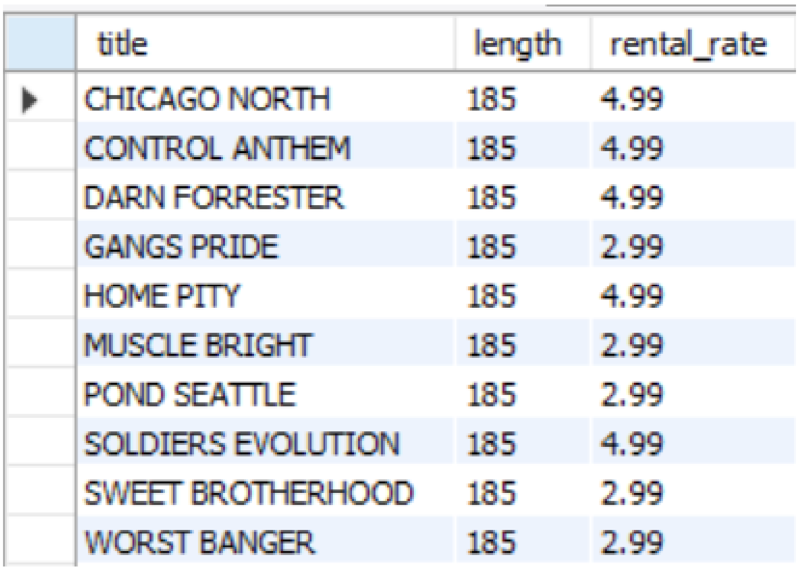
SELECT customer_id,Count(rental_id) AS total_rentals FROM rental

GROUP BY customer_id HAVING total_rentals <15;

11.-ORDER BY

“I’d like to see if out longest films also tend to be our most expensive rentals. Could you pull me a list of all film tittles along with their lengths and rental rates, and sort them from longest to shortest”

Result:



title	length	rental_rate
CHICAGO NORTH	185	4.99
CONTROL ANTHEM	185	4.99
DARN FORRESTER	185	4.99
GANGS PRIDE	185	2.99
HOME PITY	185	4.99
MUSCLE BRIGHT	185	2.99
POND SEATTLE	185	2.99
SOLDIERS EVOLUTION	185	4.99
SWEET BROTHERHOOD	185	2.99
WORST BANGER	185	2.99

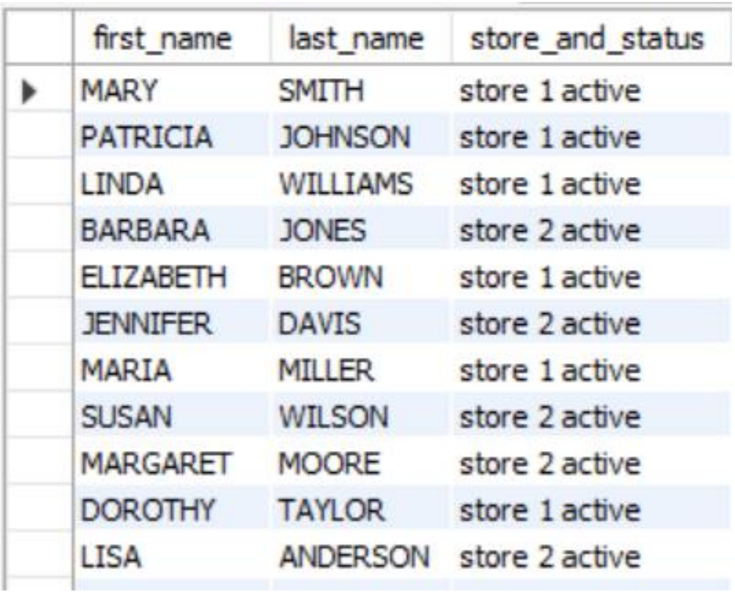
Answer:

```
SELECT title,length, rental_rate FROM film
 ORDER BY length DESC;
```

12.CASE

“I’d like to know which store each customer goes to, and whether or not they are active. Could you pull a list of first and last name of all customers, and label them as either ‘Store 1 active’, ‘Store 1 inactive’, ‘store 2 active ’ or ‘store 2 inactive’”?

Result:



first_name	last_name	store_and_status
MARY	SMITH	store 1 active
PATRICIA	JOHNSON	store 1 active
LINDA	WILLIAMS	store 1 active
BARBARA	JONES	store 2 active
ELIZABETH	BROWN	store 1 active
JENNIFER	DAVIS	store 2 active
MARIA	MILLER	store 1 active
SUSAN	WILSON	store 2 active
MARGARET	MOORE	store 2 active
DOROTHY	TAYLOR	store 1 active
LISA	ANDERSON	store 2 active

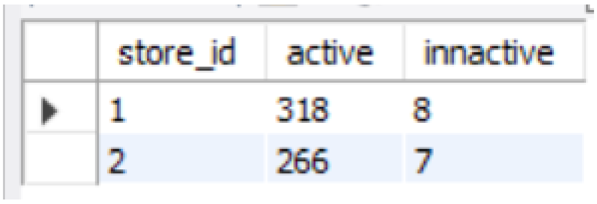
Answer:

```
SELECT first_name,last_name,
CASE
            WHEN store_id =1 AND active = 1 THEN 'store 1 active'
             WHEN store_id =1 AND active = 0 THEN 'store 1 active'
            WHEN store_id =2 AND active = 1 THEN 'store 2 active'
              WHEN store_id =2 AND active = 0 THEN 'store 2 active'
ELSE 'Review the logic'
END AS store_and_status
FROM customer;
```

13.COUNT & CASE

“I’m curious how many inactive customers we have at each store. Could you please create a table to count the number of customers broken down by store_id(in rows) and active status(in columns)”?

Result:



	store_id	active	innactive
▶	1	318	8
	2	266	7

Answer:

```
SELECT store_id,
COUNT(CASE WHEN active = 1 THEN customer_id ELSE NULL END) AS active,
COUNT(CASE WHEN active = 0 THEN customer_id ELSE NULL END) AS innactive
FROM customer
GROUP BY store_id;
```