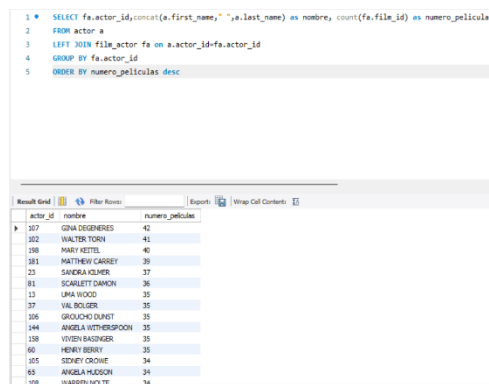


Actividad 18

- Liste los nombres de los actores y en cuantas películas del catalogo participa

```
SELECT fa.actor_id,concat(a.first_name," ",a.last_name) as nombre,  
count(fa.film_id) as numero_peliculas  
FROM actor a  
LEFT JOIN film_actor fa on a.actor_id=fa.actor_id  
GROUP BY fa.actor_id  
ORDER BY numero_peliculas desc
```



The screenshot shows a SQL query in a text editor and its results in a table grid. The query is:

```
1 SELECT fa.actor_id,concat(a.first_name," ",a.last_name) as nombre, count(fa.film_id) as numero_peliculas  
2 FROM actor a  
3 LEFT JOIN film_actor fa on a.actor_id=fa.actor_id  
4 GROUP BY fa.actor_id  
5 ORDER BY numero_peliculas desc
```

The results table has three columns: actor_id, nombre, and numero_peliculas. The data is sorted by numero_peliculas in descending order.

actor_id	nombre	numero_peliculas
107	GINA DEGENERES	42
102	WALTER TORN	41
109	MARY KETTEL	40
181	MATTHEW CARREY	39
23	SANDRA KILMER	37
81	SCARLETT DANKIN	36
13	UMA WOOD	35
37	VAL BOGGER	35
106	GROUCHO DANKIN	35
144	ANGELA WITHERSPOON	35
159	VIVIAN BASSINGER	35
60	HENRY BERRY	35
105	STONEY CROWE	34
65	ANGELA HUDSON	34
108	WILLIAM MARY	34

La razón del “ORDER BY” es para ordenar a los actores que mas tienen mas participaciones en películas (de forma descendente)

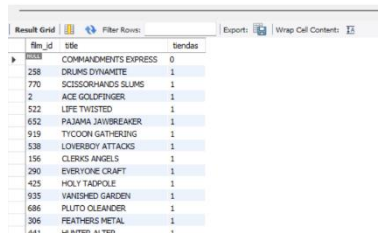
- Liste los nombres de las peliculas y en cuantas tiendas estan registradas

```
SELECT i.film_id,f.title,count(DISTINCT i.store_id) as tiendas  
FROM film f  
LEFT JOIN inventory i on f.film_id=i.film_id  
GROUP BY i.film_id  
ORDER BY tiendas
```



The screenshot shows a SQL query in a text editor and its results in a table grid. The query is:

```
1 SELECT i.film_id,f.title,count(DISTINCT i.store_id) as tiendas  
2 FROM film f  
3 LEFT JOIN inventory i on f.film_id=i.film_id  
4 GROUP BY i.film_id  
5 ORDER BY tiendas
```



The screenshot shows a SQL query in a text editor and its results in a table grid. The query is:

```
1 SELECT i.film_id,f.title,count(DISTINCT i.store_id) as tiendas  
2 FROM film f  
3 LEFT JOIN inventory i on f.film_id=i.film_id  
4 GROUP BY i.film_id  
5 ORDER BY tiendas
```

The results table has three columns: film_id, title, and tiendas. The data is sorted by tiendas in ascending order.

film_id	title	tiendas
108	COMMANDEMENT'S EXPRESS	0
258	DRUMS DYNAMITE	1
770	SCISSORHANDS SLIMS	1
2	ACE GOLDFINGER	1
522	LIPE TWISTED	1
652	PAJANA JAWBREAKER	1
919	TYCOON GATHERING	1
538	LOVERBOY ATTACKS	1
156	CLOSING ANGELS	1
290	EVERYONE CRAFT	1
425	HOLY TADPOLE	1
935	VANISHED GARDEN	1
686	PLUTO OLEANDER	1
306	FEATHERS METAL	1
441	LA RIVIERA	1

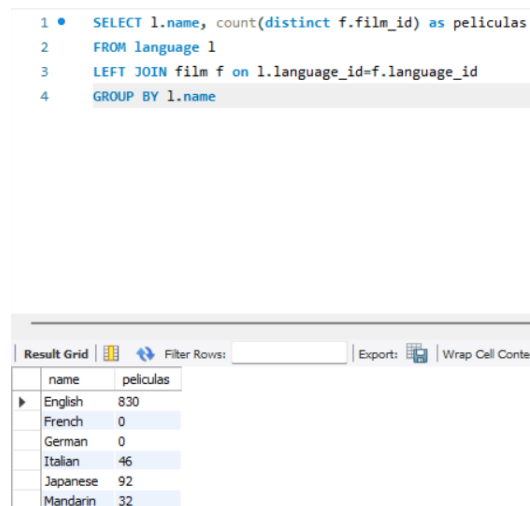
La razón del “ORDER BY” es que liste a las

Tiendas de mayor a menor tomando en cuenta

En la cantidad de tiendas que están (de forma ascendente)

- Liste los idiomas y cuantas películas están habladas en ese idioma

```
SELECT l.name, count(distinct f.film_id) as peliculas
FROM language l
LEFT JOIN film f on l.language_id=f.language_id
GROUP BY l.name
```



The screenshot shows a SQL query editor with the following query:

```
1 • SELECT l.name, count(distinct f.film_id) as peliculas
2 FROM language l
3 LEFT JOIN film f on l.language_id=f.language_id
4 GROUP BY l.name
```

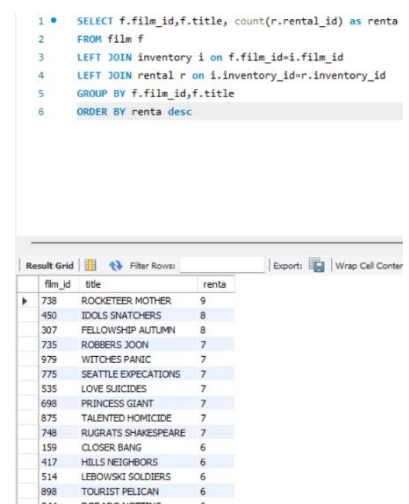
Below the query, the results are displayed in a table with the following data:

name	peliculas
English	830
French	0
German	0
Italian	46
Japanese	92
Mandarin	32

- Liste cuantas veces ha sido rentada cada película en el catalogo

```
SELECT f.film_id,f.title, count(r.rental_id) as renta
FROM film f
LEFT JOIN inventory i on f.film_id=i.film_id
LEFT JOIN rental r on i.inventory_id=r.inventory_id
GROUP BY f.film_id,f.title
ORDER BY renta desc
```

La razón del “ORDER BY” es que ordena el numero de veces que han sido rentadas (de forma decendente)



The screenshot shows a SQL query editor with the following query:

```
1 • SELECT f.film_id,f.title, count(r.rental_id) as renta
2 FROM film f
3 LEFT JOIN inventory i on f.film_id=i.film_id
4 LEFT JOIN rental r on i.inventory_id=r.inventory_id
5 GROUP BY f.film_id,f.title
6 ORDER BY renta desc
```

Below the query, the results are displayed in a table with the following data:

film_id	title	renta
738	ROCKETEER MOTHER	9
450	IDOLS SHATCHERS	8
307	FELLOWSHIP ALUMIN	8
735	ROBBERS JOON	7
979	WITCHES PANIC	7
775	SEATTLE EXPECTATIONS	7
535	LOVE SUICIDES	7
698	PRINCESS GIANT	7
875	TALENTED HOMICIDE	7
748	RUGRATS SHAKESPEARE	7
159	CLOSER BANG	6
417	HILLS NEIGHBORS	6
514	LEBOWSKI SOLDIERS	6
898	TOURIST PELICAN	6
744	DORADO MOTTING	6

- Liste cuantos pagos se realizaron en cada renta

```
SELECT p.payment_id,count(r.rental_id) as renta
FROM payment p
LEFT JOIN rental r on p.rental_id=r.rental_id
GROUP BY p.payment_id
```

payment_id	renta
1	1
2	1
3	1
4	1
5	1
6	1
33	1
60	1
61	1
62	1
63	1
64	1
65	1
67	1
88	1

La razón del “ORDER BY” es que ordene la lista tomando en cuenta el identificador del pago (de forma ascendente)

- Liste cuantas películas ha rentado cada cliente

```
SELECT c.customer_id,concat(c.first_name," ",c.last_name) as cliente,
count(r.rental_id) as cantidad
FROM customer c
LEFT JOIN rental r on c.customer_id=r.customer_id
GROUP BY cliente
ORDER BY c.customer_id
```

customer_id	cliente	cantidad
2	PATRICK JOHNSON	1
3	LINDA WILSON	5
4	BANBARA JONES	4
5	ELIZABETH BROWN	5
6	BRUNO EAGLE	6
7	MARIA MILLER	6
8	SHARON WILSON	2
9	MARGARET TUCKER	3
10	JOANETTE PATTON	3
11	USA ANDERSON	3
12	MARC THOMPAS	3
13	KAREN JACKSON	1
14	BETTY WHITE	6
15	HELEN HARRIS	0
16	KAROLIA MARTIN	4

La razón del “ORDER BY” es que ordene la lista tomando en cuenta el identificador del cliente (de forma ascendente)

- **Liste cuantas películas ha rentado cada miembro del staff**

```
SELECT concat(a.first_name," ",a.last_name) as nombre, count(r.rental_id) as  
películas_rentadas  
FROM staff a  
LEFT JOIN rental r on a.staff_id=r.staff_id  
GROUP BY nombre
```



The screenshot shows a SQL query editor with the following query:

```
1 SELECT concat(a.first_name," ",a.last_name) as nombre, count(r.rental_id) as películas_rentadas
2 FROM staff a
3 LEFT JOIN rental r on a.staff_id=r.staff_id
4 GROUP BY nombre
```

Below the query editor, there is a 'Result Grid' section with a table showing the results of the query. The table has two columns: 'nombre' and 'películas_rentadas'. The results are as follows:

nombre	películas_rentadas
Jon Stephens	1016
Mike Hillyer	983