



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

#### MYSQL CON PHP

**Ejercicio 01:** Muestre todas las bases de datos usando el siguiente query SQL:

```
show databases;
```

```
1 • use sakila;
2 -- ejer1 olarte zuasnabar
3 • show databases;
```

Result Grid | Filter Rows: | Export: | Wrap Cell

Database
information_schema
mysql
performance_schema
sakila
sys

**Ejercicio 02:** Ubique la ruta donde se encuentran las bases de datos en MySQL:

```
select @@datadir;
```

```
5 -- ejer2 olarte zuasnabar
6 • select @@datadir;
```

Result Grid | Filter Rows: | Export

@@datadir
C:\ProgramData\MySQL\MySQL Server 8.0\Data\

Result 3 x

**Ejercicio 03:** Usando la base de datos **sakila**; muestre las columnas: “first\_name” y “last\_name”.

```
select first_name, last_name from actor;
```

```
8 -- ejer3 olarte zuasnabar
9 • select first_name, last_name from actor;
```

Result Grid | Filter Rows: | Export

first_name	last_name
JENNIFER	DAVIS
JOHNNY	LOLLOBRIGIDA
BETTE	NICHOLSON
GRACE	MOSTEL
MATTHEW	JOHANSSON
JOE	SWANK
CHRISTIAN	GABLE



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

**Ejercicio 04:** Del ejercicio anterior, renombre los nombres del campo o field “first\_name” y “last\_name”, por apellidos y nombres.

```
select first_name as 'Nombres', last_name as 'Apellidos' from actor;
```

**Practica:** ahora ordenarlos los apellidos en forma ascendente y luego descendente.

```
11 -- ejer4 olarte zuasnabar
12 • select first_name as 'Nombres', last_name as 'Apellidos' from actor;
13
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Nombres	Apellidos
▶	PENELOPE	GUINNESS
	NICK	WAHLBERG
	ED	CHASE
	JENNIFER	DAVIS
	JOHNNY	LOLLOBRIGIDA
	BETTE	NICHOLSON
	GRACE	MOSTEL
	MATTHEW	JOHANSSON

```
14 -- ejer4 ascendente
15 • SELECT first_name AS 'Nombres', last_name AS 'Apellidos'
16 FROM actor
17 ORDER BY last_name ASC;
18
19
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Nombres	Apellidos
▶	KIRSTEN	AKROYD
	DEBBIE	AKROYD
	CHRISTIAN	AKROYD
	KIM	ALLEN
	MERYL	ALLEN
	CUBA	ALLEN
	ANGELINA	ASTAIRE
	RUSSELL	BACALL



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

```
19 -- ejer4 dexcendente
20 • SELECT first_name AS 'Nombres', last_name AS 'Apellidos'
21 FROM actor
22 ORDER BY last_name DESC;
23
24
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Nombres	Apellidos		
MINNIE	ZELLWEGER		
JULIA	ZELLWEGER		
CAMERON	ZELLWEGER		
CAMERON	WRAY		
UMA	WOOD		
FAY	WOOD		
ANGEL A	WITHERSPOON		

**Ejercicio 05:** Del ejercicio anterior muestre sólo los apellidos que inicien con la letra B.

```
select first_name as 'Nombres', last_name as 'Apellidos' from actor where last_name like 'b%'
order by last_name ASC;
```

**Practica:** ahora ordenarlos los apellidos en forma ascendente y luego descendente.

**Practica:** ahora muestre solo los apellidos en forma ascendente pero estas no debe repetirse.

```
14 -- ejer5 olarte zuasnabar
15 • select first_name as 'Nombres', last_name as 'Apellidos' from actor where last_name like 'b%'
16 order by last_name ASC;
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Nombres	Apellidos		
RUSSELL	BACALL		
JESSICA	BAILEY		
AUDREY	BAILEY		
HARRISON	BALE		
RENEE	BALL		
JULIA	BARRYMORE		
VIVIEN	BASINGER		
CHARLOTTE	BENING		



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

```
18 -- ejer5(2) olarte zuasnabar
19 • SELECT first_name AS 'Nombres', last_name AS 'Apellidos'
20 FROM actor
21 WHERE last_name LIKE 'b%'
22 ORDER BY last_name DESC;
23
24 -- ejer6 olarte zuasnabar
```

	Nombres	Apellidos
▶	LAURENCE	BULLOCK
	LAURA	BRODY
	GOLDIE	BRODY
	CHRIS	BRIDGES
	MICHAEL	BOLGER
	VAL	BOLGER
	KEVIN	BLOOM

```
24 -- ejer5(3) olarte zuasnabar
25 • SELECT DISTINCT last_name AS 'Apellidos'
26 FROM actor
27 WHERE last_name LIKE 'b%'
28 ORDER BY last_name ASC;
29
30
```

	Apellidos
▶	BACALL
	BAILEY
	BALE
	BALL
	BARRYMORE
	BASINGER
	BENING

**Ejercicio 06:** Del ejercicio anterior cuantos cumplen con dicha condición.

```
select count(first_name) as 'Total' from actor where last_name like 'b%' order by last_name ASC;
```

```
18 -- ejer6 olarte zuasnabar
19 • select count(first_name) as 'Total' from actor where last_name like 'b%' order by last_name ASC;
```

	Total
▶	22

**Ejercicio 07:** Usando la tabla película o “film”, liste las películas que describan el tipo documental o “Documentary”.



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

```
select title as 'Título', description as 'Detalle' from film where description like '%Documentary%';
```

**Practica:** ahora mencione cuantas películas cumplen esa condición.

```
21 -- ejer7 olarte zuasnabar
22 • select title as 'Título', description as 'Detalle' from film where description like '%Documentary%';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Título	Detalle
▶	AFFAIR PREJUDICE	A Fanciful Documentary of a Frisbee And a Lum...
	AFRICAN EGG	A Fast-Paced Documentary of a Pastry Chef An...
	ARSENIC INDEPENDENCE	A Fanciful Documentary of a Mad Cow And a W...
	BALLROOM MOCKINGBIRD	A Thrilling Documentary of a Composer And a M...
	BEVERLY OUTLAW	A Fanciful Documentary of a Womanizer And a ...
	BIRD INDEPENDENCE	A Thrilling Documentary of a Car And a Student ...
	BLANKET BEVERLY	A Emotional Documentary of a Student And a Gi...
	BLUES INSTINCT	A Insightful Documentarv of a Boat And a Compo...

```
46 -- ejer7 olarte zuasnabar
47 • select title as 'Título', description as 'Detalle' from film where descript
48
49 -- ejer7 olarte
50 • SELECT COUNT(*) AS 'Cantidad de Películas'
51 FROM film
52 WHERE description LIKE '%Documentary%';
53
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: |

	Cantidad de Películas
▶	101

**Ejercicio 08:** Liste las películas que describan el tipo documental o “Documentary” y que su costo (replacement\_cost) sea superior a 20.0, además agregue una columna correlativo y mostrar ordenado por costo.

```
select ROW_NUMBER() OVER () AS '#', title as 'Título', description as 'Detalle', replacement_cost as 'Costo'
from film where description like '%Documentary%' AND replacement_cost > 20.0 order by Costo desc;
```



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

```
24 -- ejer8 olarte zuasnabar
25 • select ROW_NUMBER() OVER () AS '#', title as 'Título', description as 'Deta
```

Result Grid				
Filter Rows: <input type="text"/> Export:  Wrap Cell Content: <input type="checkbox"/>				
	#	Título	Detalle	Costo
▶	1	BALLROOM MOCKINGBIRD	A Thrilling Documentary of a Composer And a M...	29.99
	2	SASSY PACKER	A Fast-Paced Documentary of a Dog And a Tea...	29.99
	3	CLOCKWORK PARADISE	A Insightful Documentary of a Technical Writer ...	29.99
	4	FANTASIA PARK	A Thoughtful Documentary of a Mad Scientist A...	29.99
	5	GILMORE BOILED	A Unbelievable Documentary of a Boat And a ...	29.99
	6	SONG HEDWIG	A Amazing Documentary of a Man And a Husba...	29.99
	7	VIRGIN DAISY	A Awe-Inspiring Documentary of a Robot And a...	29.99
	8	MAGNOLIA FORRESTER	A Thoughtful Documentarv of a Composer And ...	28.99

**Ejercicio 09:** Muestre las películas que empiecen con la letra A ordenados, además muestre sus autores ordenados, acompañe con la columna correlativo.

```
select row_number() over() as '#', A.title as 'Título', C.first_name as 'Actores' from film A inner join
film_actor B on A.film_id = B.film_id inner join actor C on B.actor_id = C.actor_id where A.title like 'a%'
order by A.title, C.first_name;
```

```
27 -- ejer9 olarte zuasnabar
28 • select row_number() over() as '#', A.title as 'Título', C.first_name as 'Actores' from film A inner join f
```

Result Grid			
Filter Rows: <input type="text"/> Export:  Wrap Cell Content: <input type="checkbox"/>			
	#	Título	Actores
▶	2	ACADEMY DINOSAUR	CHRISTIAN
	5	ACADEMY DINOSAUR	JOHNNY
	3	ACADEMY DINOSAUR	LUCILLE
	10	ACADEMY DINOSAUR	MARY
	6	ACADEMY DINOSAUR	MENA
	8	ACADEMY DINOSAUR	OPRAH
	1	ACADEMY DINOSAUR	PENELOPE
	9	ACADEMY DINOSAUR	ROCK

**Ejercicio 10:** Muestre el valor de la película mas económica.

```
select min(replacement_cost) from film;
```

**Practica:** ahora mencione el valor de la película mas cara.

```
30 -- ejer10 olarte zuasnabar
31 • select min(replacement_cost) from film;
```

Result Grid	
Filter Rows: <input type="text"/> Export:  Wrap C	
	min(replacement_cost)
▶	9.99



## PRACTICA DIRIGIDA

### TALLER DE PROGRAMACIÓN WEB – SESIÓN 13

```
61 -- ejer10 olarte zuasnabar
62 • select min(replacement_cost) from film;
63
64 -- ejer10 olarte
65 • SELECT MAX(replacement_cost) AS 'Película más cara'
66 FROM film;
67
```

Result Grid		Filter Rows:	Export:	Wrap Cell Center
	Película más cara			
▶	29.99			