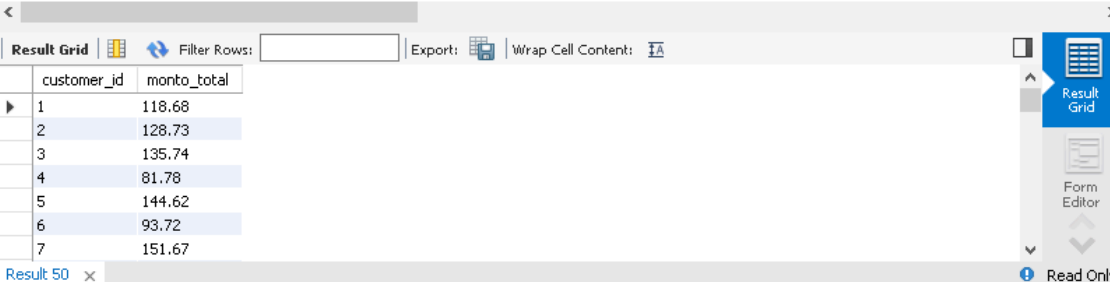


# ACTIVIDAD 13

**Rodrigo Martin Alvarez**

*Calcule el monto de las rentas de cada empleado con durante el mes de mayo*

```
3 -- Calcule el monto de las rentas de cada empleado con durante el mes de mayo
4 • SELECT customer_id, SUM(amount) AS monto_total FROM payment GROUP BY customer_id;
```



customer_id	monto_total
1	118.68
2	128.73
3	135.74
4	81.78
5	144.62
6	93.72
7	151.67

Result 50 x

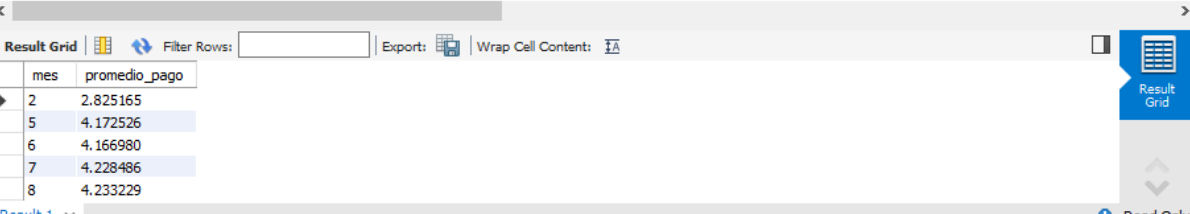
Output

Action Output

#	Time	Action	Message
✓ 1	09:18:00	USE sakila	0 row(s) affected
✓ 2	10:06:03	SELECT customer_id, SUM(amount) AS monto_total FROM payment GROUP BY customer_id ...	599 row(s) returned

*Calcule el promedio de los pagos por cada mes*

```
3 • SELECT MONTH(payment_date) AS mes, AVG(amount) AS promedio_pago FROM payment GROUP BY MONTH(payment_date);
4
```



mes	promedio_pago
2	2.825165
5	4.172526
6	4.166980
7	4.228486
8	4.233229

Result 1 x

Output

Action Output

#	Time	Action	Message
✓ 1	20:53:12	SELECT MONTH(payment_date) AS mes, AVG(amount) AS promedio_pago FROM payment G...	5 row(s) returned

*Calcule el promedio de los pagos de cada cliente*

Query 1 x

Limit to 1000 rows

```

1  USE sakila;
2
3  • SELECT MONTH(payment_date) AS mes, AVG(amount) AS promedio_pago FROM payment GROUP BY MONTH(payment_date);
4
5  • SELECT customer_id, AVG(amount) AS promedio_pago FROM payment GROUP BY customer_id;
6

```

Result Grid

customer_id	promedio_pago
1	3.708750
2	4.767778
3	5.220769
4	3.717273
5	3.805789

Result 2 x

Output

Action Output

#	Action	Message
1	SELECT customer_id, AVG(amount) AS promedio_pago FROM payment GROUP BY customer_id	599 row(s) returned

*Calcule el promedio de los pagos de cada día de la semana*

```

7  • SELECT DAYOFWEEK(payment_date) AS dia, AVG(amount) AS promedio_pago
8  FROM payment
9  GROUP BY DAYOFWEEK(payment_date);
10

```

Result Grid

dia	promedio_pago
1	4.200776
2	4.159114
3	4.148343
4	4.153156
5	4.243636
6	4.210070
7	4.297776

Result 4 x

Output

Action Output

#	Action	Message
1	SELECT DAYOFWEEK(payment_date) AS dia, AVG(amount) AS promedio_pago FROM payment GROUP BY DAYOFWEEK(payment_date)	7 row(s) returned

*Calcule cuantas rentas tiene cada cliente*

```

11 • SELECT customer_id, COUNT(*) AS cuantos
12 FROM rental
13 GROUP BY customer_id;
14

```

Result Grid

customer_id	cuantos
1	32
2	27
3	26
4	22
5	38
6	28
7	33

Result 6 x

Output

Action Output

#	Action	Message
1	SELECT customer_id, COUNT(*) AS cuantos FROM rental GROUP BY customer_id LIMIT 0, ...	599 row(s) returned

### Calcule cuantas rentas tiene cada empleado

```
15 • SELECT staff_id, COUNT(*) AS cuantos
16 FROM rental
17 GROUP BY staff_id;
18
19
```

Result Grid

staff_id	cuantos
1	8040
2	8004

Result 7 x

Output

Action Output

#	Action	Message
✓ 1	SELECT staff_id, COUNT(*) AS cuantos FROM rental GROUP BY staff_id LIMIT 0, 1000	2 row(s) returned

### Calcule cuantos clientes tiene cada tienda

```
19 • SELECT store_id, COUNT(DISTINCT customer_id) AS numero_clientes
20 FROM customer
21 GROUP BY store_id;
22
23
```

Result Grid

store_id	numero_clientes
1	326
2	273

Result 8 x

Output

Action Output

#	Action	Message
✓ 1	SELECT store_id, COUNT(DISTINCT customer_id) AS numero_clientes FROM customer GRO...	2 row(s) returned

### Calcule cuantas películas tiene cada categoría de rating

```

23 • SELECT rating, COUNT(*) AS numero_peliculas
24 FROM film
25 GROUP BY rating;
26

```

	rating	numero_peliculas
▶	G	178
	PG	194
	PG-13	223
	R	195
	NC-17	210

Result 10 x

Output		
Action Output		
#	Action	Message
✓ 1	SELECT rating, COUNT(*) AS numero_categorias FROM film GROUP BY rating LIMIT 0, 1000	5 row(s) returned
✓ 2	SELECT rating, COUNT(*) AS numero_peliculas FROM film GROUP BY rating LIMIT 0, 1000	5 row(s) returned

*Calcule cuantas ciudades están registradas por país*

```

27 • SELECT country_id, COUNT(DISTINCT city_id) AS ciudades_registradas
28 FROM city
29 GROUP BY country_id;
30

```

	country_id	ciudades_registradas
▶	1	1
	2	3
	3	1
	4	2
	5	1
	6	13
	7	1

Result 12 x

Output		
Action Output		
#	Action	Message
✓ 1	SELECT country_id, COUNT(DISTINCT city_id) AS ciudades_registradas FROM city GROUP ...	109 row(s) returned

*Calcule cuantas películas hay en cada tienda*

```

31 • SELECT store_id, COUNT(*) AS numero_peliculas_en_tienda
32 FROM inventory
33 GROUP BY store_id;

```

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

	store_id	numero_peliculas_en_tienda
▶	1	2270
	2	2311

Result 14 ×

Output

Action Output

#	Action	Message
✓ 1	SELECT store_id, COUNT(*) AS numero_peliculas_en_tienda FROM inventory GROUP BY sto...	2 row(s) returned

*Calcule el monto total de pagos de cada cliente*

```

35 • SELECT customer_id, SUM(amount) AS total_pago
36 FROM payment
37 GROUP BY customer_id;

```

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

	customer_id	total_pago
▶	1	118.68
	2	128.73
	3	135.74
	4	81.78
	5	144.62
	6	93.72
	7	151.67

Result 16 ×

Output

Action Output

#	Action	Message
✓ 1	SELECT customer_id, SUM(amount) AS total_pago FROM payment GROUP BY customer_id ...	599 row(s) returned

*Liste los empleados que han generado pagos con un monto total mayor a \$100*

```

3 • SELECT staff_id, SUM(amount)
4 FROM payment
5 GROUP BY staff_id
6 HAVING SUM(amount) > 100;

```

staff_id	SUM(amount)
1	33482.50
2	33924.06

Result 1 x Read Only

Output

Action Output

#	Time	Action	Message
✓ 1	08:09:22	USE sakila	0 row(s) affected
✓ 2	08:09:22	SELECT staff_id, SUM(amount) FROM payment GROUP BY staff_id HAVING SUM(amount) > 100;	2 row(s) returned

*-Calcule el número de pagos realizados por cada cliente en el mes de junio*

```

8 • SELECT customer_id, COUNT(*)
9 FROM payment
10 WHERE MONTH(payment_date) = 6
11 GROUP BY customer_id;

```

customer_id	COUNT(*)
1	7
2	1
3	4
4	6
5	5
6	4
7	5

Result 2 x

Output

Action Output

#	Time	Action	Message
✓ 1	08:10:19	SELECT customer_id, COUNT(*) FROM payment WHERE MONTH(payment_date) = 6 GROUP BY customer_id;	590 row(s) returned

*Calcule la cantidad de películas por cada duración de renta (rental\_duration)*

```

13 • SELECT rental_duration, COUNT(*)
14 FROM film
15 GROUP BY rental_duration;
16

```

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

Result 3 x

Output

Action Output

#	Time	Action	Message
✓ 1	08:11:11	SELECT rental_duration, COUNT(*) FROM film GROUP BY rental_duration LIMIT 0, 1000	5 row(s) returned

*Calcule la cantidad total de pagos recibidos cada año*

```

17 • SELECT YEAR(payment_date), SUM(amount) As total_pagos
18 FROM payment
19 GROUP BY YEAR(payment_date);
20
21
22

```

YEAR(payment_date)	total_pagos
2005	66892.38
2006	514.18

Result 5 x

Output

Action Output

#	Action	Message
✓ 1	SELECT YEAR(payment_date), SUM(amount) FROM payment GROUP BY YEAR(payment_d...	2 row(s) returned
✓ 2	SELECT YEAR(payment_date), SUM(amount) As total_pagos FROM payment GROUP BY YE...	2 row(s) returned

*Calcule el monto total de pagos por cada día de la semana*

```

21 • SELECT DAYOFWEEK(payment_date) AS dia_semana, SUM(amount) AS total_pagos
22 FROM payment
23 GROUP BY dia_semana;

```

Result Grid

	dia_semana	total_pagos
▶	1	9745.80
	2	9345.53
	3	10217.37
	4	9265.69
	5	9336.00
	6	9565.28
	7	9930.89

Result 7

Output

Action Output

#	Action	Message
✓ 1	SELECT DAYOFWEEK(payment_date) AS dia_semana, SUM(amount) FROM payment GRO...	7 row(s) returned
✓ 2	SELECT DAYOFWEEK(payment_date) AS dia_semana, SUM(amount) AS total_pagos FROM ...	7 row(s) returned

*Liste los días donde el total de pagos fue mayor a \$500*

```

25 • SELECT DATE(payment_date) AS fecha, SUM(amount) AS pago
26 FROM payment
27 GROUP BY DATE(payment_date)
28 HAVING SUM(amount) > 500;
29

```

Result Grid

	fecha	pago
▶	2005-05-25	573.63
	2005-05-26	754.26
	2005-05-27	684.34
	2005-05-28	804.04
	2005-05-29	648.46
	2005-05-30	628.42
	2005-05-31	700.37

Result 8

Output

Action Output

#	Action	Message
✓ 1	SELECT DATE(payment_date) AS fecha, SUM(amount) AS pago FROM payment GROUP B...	36 row(s) returned

*Calcule cuántas películas hay en cada duración (length) específica*



```

30 • SELECT length AS duracion, COUNT(*) AS cantidad
31 FROM film
32 GROUP BY length;

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	duracion	cantidad			
▶	46	5			
	47	7			
	48	11			
	49	5			
	50	9			
	51	7			
	52	7			

Result 10 x

Output

Action Output

#	Action	Message
✓ 1	SELECT length AS duracion, COUNT(*) AS cantidad FROM film GROUP BY length LIMIT 0, ...	140 row(s) returned

*Calcule la cantidad de pagos realizados en cada tienda*

```

34 • SELECT staff_id, COUNT(payment_id) AS pagos_realizados
35 FROM payment
36 GROUP BY staff_id;
37

```

Result Grid			Filter Rows:	Export:	Wrap Cell Content:
	staff_id	pagos_realizados			
▶	1	8054			
	2	7990			

Result 11 x

Output

Action Output

#	Action	Message
✓ 1	SELECT staff_id, COUNT(payment_id) AS pagos_realizados FROM payment GROUP BY staff...	2 row(s) returned

*Calcule el monto total de pagos por cada cliente, pero solo para aquellos que han realizado más de 5 pagos*

```

37
38 SELECT customer_id, SUM(amount) as monto_total_de_los_pagos
39 FROM payment
40 GROUP BY customer_id
41 HAVING COUNT(*) > 5;
42

```

customer_id	monto_total_de_los_pagos
1	118.68
2	128.73
3	135.74
4	81.78
5	144.62
6	93.72
7	151.67

Result 10

Output

#	Time	Action	Message
7	08:16:48	SELECT MONTH(payment_date) as mes, COUNT(*) as pagos, SUM(amount) as total_pagos...	5 row(s) returned
8	08:17:31	SELECT rental_duration, COUNT(*) as peliculas FROM film WHERE rental_duration > 5 GR...	2 row(s) returned
9	08:18:20	SELECT customer_id, AVG(amount) as pagos FROM payment GROUP BY customer_id HAV...	23 row(s) returned
10	08:18:33	SELECT customer_id, AVG(amount) as pagos_promedio FROM payment GROUP BY custo...	23 row(s) returned
11	08:18:52	SELECT rating, COUNT(*) as peliculas FROM film GROUP BY rating LIMIT 0, 1000	5 row(s) returned
12	08:19:34	SELECT customer_id, SUM(amount) as monto_total_de_los_pagos FROM payment GROUP...	599 row(s) returned

Calcule el número de películas según su clasificación (rating)

```

42
43 SELECT rating, COUNT(*) as peliculas
44 FROM film
45 GROUP BY rating;
46
47 SELECT customer_id, AVG(amount) as pagos_promedio
48 FROM payment
49 GROUP BY customer_id;

```

rating	peliculas
G	178
PG	194
PG-13	223
R	195
NC-17	210

Result 9

Output

#	Time	Action	Message
6	08:15:23	SELECT MONTH(payment_date), COUNT(*), SUM(amount) as total_pagos FROM payment ...	5 row(s) returned
7	08:16:48	SELECT MONTH(payment_date) as mes, COUNT(*) as pagos, SUM(amount) as total_pagos...	5 row(s) returned
8	08:17:31	SELECT rental_duration, COUNT(*) as peliculas FROM film WHERE rental_duration > 5 GR...	2 row(s) returned
9	08:18:20	SELECT customer_id, AVG(amount) as pagos FROM payment GROUP BY customer_id HAV...	23 row(s) returned
10	08:18:33	SELECT customer_id, AVG(amount) as pagos_promedio FROM payment GROUP BY custo...	23 row(s) returned
11	08:18:52	SELECT rating, COUNT(*) as peliculas FROM film GROUP BY rating LIMIT 0, 1000	5 row(s) returned

Liste los clientes que han realizado pagos con un monto promedio mayor a \$5

46  
47 `SELECT customer_id, AVG(amount) as pagos_promedio`  
48 `FROM payment`  
49 `GROUP BY customer_id`  
50 `HAVING AVG(amount) > 5;`  
51

Result Grid

customer_id	pagos_promedio
3	5.220769
19	5.240000
22	5.171818
82	5.028462
181	5.137059
187	5.704286
200	5.064074

Result 8 x

Output

Action Output

#	Time	Action	Message
5	08:14:22	SELECT rating.COUNT(*) as no_de_peliculas FROM film GROUP BY rating HAVING COUN...	2 row(s) returned
6	08:15:23	SELECT MONTH(payment_date), COUNT(*), SUM(amount) as total_pagos FROM payment ...	5 row(s) returned
7	08:16:48	SELECT MONTH(payment_date) as mes, COUNT(*) as pagos, SUM(amount) as total_pagos...	5 row(s) returned
8	08:17:31	SELECT rental_duration.COUNT(*) as peliculas FROM film WHERE rental_duration > 5 GR...	2 row(s) returned

*Calcule la cantidad de películas en cada duración de renta (rental\_duration) mayor a 5 días*

54 `WHERE rental_duration > 5`  
55 `GROUP BY rental_duration;`  
56  
57 `SELECT MONTH(payment_date) as mes, COUNT(*) as pagos, SUM(amount) as total_pagos`  
58 `FROM payment`

Result Grid

rental_duration	peliculas
6	212
7	191

Result 6 x

Output

*Calcule el total de pagos recibidos por cada mes, pero solo los meses con más de 100 pagos*

Query 1: Query File 1



Limit to 1000 rows

```
57 SELECT MONTH(payment_date) as mes, COUNT(*) as pagos, SUM(amount) as total_pagos
58 FROM payment
59 GROUP BY MONTH(payment_date)
60 HAVING COUNT(*) > 100;
61
62 SELECT rating, COUNT(*) as no_de_peliculas
63 FROM film
64 GROUP BY rating;
```

Result Grid

	mes	pagos	total_pagos
▶	2	182	514.18
	5	1156	4823.44
	6	2311	9629.89
	7	6709	28368.91
	8	5686	24070.14

Filter Rows:

Export:  Wrap Cell Content: 

*Liste las clasificaciones de películas (rating) que tienen más de 200 películas registradas*

```

61
62 SELECT rating, COUNT(*) as no_de_peliculas
63 FROM film
64 GROUP BY rating
65 HAVING COUNT(*) > 200;
66


```

Result Grid   Filter Rows:  Export:  Wrap Cell Content: 

	rating	no_de_peliculas
▶	PG-13	223
	NC-17	210

Result 3 x

Output

 Action Output ▼

	#	Time	Action	Message
✓	1	08:08:57	use sakila	0 row(s) affected
✗	2	08:08:57	SELECT employee_id, SUM(amount) FROM payment GROUP BY employee_id WHERE SUM(...	Error Code: 1064. Yo
✓	3	08:13:37	SELECT rating, COUNT(*) FROM film GROUP BY rating HAVING COUNT(*) > 200 LIMIT 0, 1...	2 row(s) returned
✓	4	08:14:04	SELECT rating, COUNT(*) as clasificacion FROM film GROUP BY rating HAVING COUNT(*) > ...	2 row(s) returned
✓	5	08:14:22	SELECT rating, COUNT(*) as no_de_peliculas FROM film GROUP BY rating HAVING COUNT(...	2 row(s) returned