

## Practica 2 2ªEvaluación

### Ejercicios de SQL

Tenemos una base de datos “Empresas”.



Realiza las siguiente consultas. Muestra la select y el resultado.

1. Devuelve un listado con todos los pedidos que se han realizado. Los pedidos deben estar ordenados por la fecha de realización, mostrando en primer lugar los pedidos más recientes.

```
mysql> SELECT * FROM PEDIDO ORDER BY FECHA DESC;
```

ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
8	1983.43	2020-10-10	4	6
1	150.5	2020-10-05	5	2
3	65.26	2020-10-05	2	1
5	948.5	2020-09-10	5	2
12	3045.6	2020-04-25	2	1
14	145.82	2020-02-02	6	1
7	5760	2019-09-10	2	1
15	370.85	2019-03-11	1	5
16	2389.23	2019-03-11	1	5
13	545.75	2019-01-25	6	1
9	2480.4	2018-10-10	8	3
2	270.65	2018-09-10	1	5
4	110.5	2018-08-17	8	3
11	75.29	2018-08-17	3	7
6	2400.6	2018-07-27	7	1
10	250.45	2015-06-27	8	2

```
16 rows in set (0,00 sec)
```



2. Devuelve todos los datos de los dos pedidos de mayor valor.

```
mysql> SELECT * FROM PEDIDO ORDER BY TOTAL DESC LIMIT 2;
+-----+-----+-----+-----+-----+
| ID | TOTAL | FECHA      | ID_CLIENTE | ID_COMERCIAL |
+-----+-----+-----+-----+-----+
| 7 | 5760 | 2019-09-10 | 2 | 1 |
| 12 | 3045.6 | 2020-04-25 | 2 | 1 |
+-----+-----+-----+-----+-----+
2 rows in set (0,00 sec)
```

3. Devuelve un listado con los identificadores de los clientes que han realizado algún pedido. Tenga en cuenta que no debe mostrar identificadores que estén repetidos.

```
mysql> SELECT DISTINCT ID_CLIENTE FROM PEDIDO;
+-----+
| ID_CLIENTE |
+-----+
| 1 |
| 2 |
| 3 |
| 4 |
| 5 |
| 6 |
| 7 |
| 8 |
+-----+
8 rows in set (0,00 sec)
```

4. Devuelve un listado de todos los pedidos que se realizaron durante el año 2017, cuya cantidad sea superior a 500€.

```
mysql> SELECT * FROM PEDIDO WHERE FECHA LIKE '2017-*-*' AND TOTAL > 500;
Empty set (0,00 sec)
```

5. Devuelve un listado con el nombre y los apellidos de los comerciales que tienen una comisión entre 0.05 y 0.11.

```
mysql> SELECT NOMBRE,APELLIDO1,APELLIDO2,COMISIÓN FROM COMERCIAL WHERE COMISIÓN
BETWEEN 0.05 AND 0.11;
+-----+-----+-----+-----+
| NOMBRE | APELLIDO1 | APELLIDO2 | COMISIÓN |
+-----+-----+-----+-----+
| DIEGO | FLORES | SALAS | 0.11 |
| ANTONIO | VEGA | HERNÁNDEZ | 0.11 |
| ALFREDO | RUIZ | FLORES | 0.05 |
+-----+-----+-----+-----+
3 rows in set (0,00 sec)
```

6. Devuelve el valor de la comisión de mayor valor que existe en la tabla comercial.

```
mysql> SELECT MAX(COMISIÓN) FROM COMERCIAL;
+-----+
| MAX(COMISIÓN) |
+-----+
| 0.15 |
+-----+
1 row in set (0,02 sec)
```

7. Devuelve el identificador, nombre y primer apellido de aquellos clientes cuyo segundo apellido no es NULL. El listado deberá estar ordenado alfabéticamente por apellidos y nombre.

```
mysql> SELECT ID,NOMBRE,APELLIDO1 FROM CLIENTE WHERE APELLIDO2 IS NOT NULL ORDER BY APELLIDO1,NOMBRE;
+-----+
| ID | NOMBRE | APELLIDO1 |
+-----+
| 9 | GUILLERMO | LÓPEZ |
| 5 | MARCOS | LOYOLA |
| 1 | AARÓN | RIVERO |
| 3 | ADOLFO | RUBIO |
| 8 | PEPE | RUIZ |
| 2 | ADELA | SALAS |
| 10 | DANIEL | SANTANA |
| 6 | MARÍA | SANTANA |
+-----+
8 rows in set (0,00 sec)
```

8. Devuelve un listado de los nombres de los clientes que empiezan por A y terminan por n y también los nombres que empiezan por P. El listado deberá estar ordenado alfabéticamente.

```
mysql> SELECT NOMBRE FROM CLIENTE WHERE NOMBRE LIKE "A%n" OR NOMBRE LIKE "P%" ORDER BY NOMBRE;
+-----+
| NOMBRE |
+-----+
| AARÓN |
| ADRIÁN |
| PEPE |
| PILAR |
+-----+
4 rows in set (0,00 sec)
```

9. Devuelve un listado de los nombres de los clientes que no empiezan por A. El listado deberá estar ordenado alfabéticamente.

```
mysql> SELECT NOMBRE FROM CLIENTE WHERE NOT NOMBRE LIKE "A%" ORDER BY NOMBRE;
+-----+
| NOMBRE |
+-----+
| DANIEL |
| GUILLERMO |
| MARCOS |
| MARÍA |
| PEPE |
| PILAR |
+-----+
6 rows in set (0,00 sec)
```

10. Devuelve un listado con los nombres de los comerciales que terminan por "o". Tenga en cuenta que se deberán eliminar los nombres repetidos.

```
mysql> SELECT DISTINCT NOMBRE FROM COMERCIAL WHERE NOMBRE LIKE "%o";
+-----+
| NOMBRE |
+-----+
| DIEGO |
| ANTONIO |
| ALFREDO |
+-----+
3 rows in set (0,00 sec)
```

11. Devuelve un listado con el identificador, nombre y los apellidos de todos los clientes que han realizado algún pedido. El listado debe estar ordenado alfabéticamente y se deben eliminar los elementos repetidos.

```
mysql> SELECT DISTINCT PEDIDO.ID_CLIENTE,NOMBRE,APELLIDO1,APELLIDO2 FROM CLIENTE,PEDIDO WHERE PEDIDO.ID_CLIENTE=CLIENTE.ID ORDER BY NOMBRE;
```

ID_CLIENTE	NOMBRE	APELLIDO1	APELLIDO2
1	AARÓN	RIVERO	GÓMEZ
2	ADELA	SALAS	DÍAZ
3	ADOLFO	RUBIO	FLORES
4	ADRIÁN	SUÁREZ	NULL
5	MARCOS	LOYOLA	MÉNDEZ
6	MARÍA	SANTANA	MORENO
8	PEPE	RUIZ	SANTANA
7	PILAR	RUIZ	NULL

8 rows in set (0,00 sec)

12. Devuelve un listado que muestre todos los pedidos que ha realizado cada cliente. El resultado debe mostrar todos los datos de los pedidos y del cliente. El listado debe mostrar los datos de los clientes ordenados alfabéticamente.

```
mysql> SELECT * FROM CLIENTE, PEDIDO WHERE PEDIDO.ID_CLIENTE = CLIENTE.ID ORDER BY CLIENTE.ID;
```

ID	NOMBRE	APELLIDO1	APELLIDO2	CIUDAD	CATEGORÍA	ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
1	AARÓN	RIVERO	GÓMEZ	ALMERÍA	100	2	270.65	2018-09-10	1	5
1	AARÓN	RIVERO	GÓMEZ	ALMERÍA	100	15	370.85	2019-03-11	1	5
1	AARÓN	RIVERO	GÓMEZ	ALMERÍA	100	16	2389.23	2019-03-11	1	5
2	ADELA	SALAS	DÍAZ	GRANADA	200	3	65.26	2020-10-05	2	1
2	ADELA	SALAS	DÍAZ	GRANADA	200	7	5760	2019-09-10	2	1
2	ADELA	SALAS	DÍAZ	GRANADA	200	12	3045.6	2020-04-25	2	1
3	ADOLFO	RUBIO	FLORES	SEVILLA	NULL	11	75.29	2018-08-17	3	7
4	ADRIÁN	SUÁREZ	NULL	JAÉN	300	8	1983.43	2020-10-10	4	6
5	MARCOS	LOYOLA	MÉNDEZ	ALMERÍA	200	1	150.5	2020-10-05	5	2
5	MARCOS	LOYOLA	MÉNDEZ	ALMERÍA	200	5	948.5	2020-09-10	5	2
6	MARÍA	SANTANA	MORENO	CÁDIZ	100	13	545.75	2019-01-25	6	1
6	MARÍA	SANTANA	MORENO	CÁDIZ	100	14	145.82	2020-02-02	6	1
7	PILAR	RUIZ	NULL	SEVILLA	300	6	2400.6	2018-07-27	7	1
8	PEPE	RUIZ	SANTANA	HUELVA	200	4	110.5	2018-08-17	8	3
8	PEPE	RUIZ	SANTANA	HUELVA	200	9	2480.4	2018-10-10	8	3
8	PEPE	RUIZ	SANTANA	HUELVA	200	10	250.45	2015-06-27	8	2

16 rows in set (0,00 sec)

13. Devuelve un listado que muestre todos los pedidos en los que ha participado un comercial. El resultado debe mostrar todos los datos de los pedidos y de los comerciales. El listado debe mostrar los datos de los comerciales ordenados alfabéticamente.

```
mysql> SELECT * FROM COMERCIAL, PEDIDO WHERE PEDIDO.ID_COMERCIAL = COMERCIAL.ID ORDER BY COMERCIAL.NOMBRE;
```

ID	NOMBRE	APELLIDO1	APELLIDO2	COMISIÓN	ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
5	ANTONIO	CARRETERO	ORTEGA	0.12	2	270.65	2018-09-10	1	5
5	ANTONIO	CARRETERO	ORTEGA	0.12	15	370.85	2019-03-11	1	5
5	ANTONIO	CARRETERO	ORTEGA	0.12	16	2389.23	2019-03-11	1	5
7	ANTONIO	VEGA	HERNÁNDEZ	0.11	11	75.29	2018-08-17	3	7
1	DANIEL	SÁEZ	VEGA	0.15	3	65.26	2020-10-05	2	1
1	DANIEL	SÁEZ	VEGA	0.15	6	2400.6	2018-07-27	7	1
1	DANIEL	SÁEZ	VEGA	0.15	7	5760	2019-09-10	2	1
1	DANIEL	SÁEZ	VEGA	0.15	12	3045.6	2020-04-25	2	1
1	DANIEL	SÁEZ	VEGA	0.15	13	545.75	2019-01-25	6	1
1	DANIEL	SÁEZ	VEGA	0.15	14	145.82	2020-02-02	6	1
3	DIEGO	FLORES	SALAS	0.11	4	110.5	2018-08-17	8	3
3	DIEGO	FLORES	SALAS	0.11	9	2480.4	2018-10-10	8	3
2	JUAN	GÓMEZ	LÓPEZ	0.13	1	150.5	2020-10-05	5	2
2	JUAN	GÓMEZ	LÓPEZ	0.13	5	948.5	2020-09-10	5	2
2	JUAN	GÓMEZ	LÓPEZ	0.13	10	250.45	2015-06-27	8	2
6	MANUEL	DOMÍNGUEZ	HERNÁNDEZ	0.13	8	1983.43	2020-10-10	4	6

16 rows in set (0,00 sec)

14. Calcula la cantidad total de todos los pedidos que aparecen en la tabla pedido.

```
mysql> SELECT SUM(TOTAL) AS "SUMA(TOTAL)" FROM PEDIDO;
+-----+
| SUMA(TOTAL) |
+-----+
| 20992.829999999998 |
+-----+
1 row in set (0,00 sec)
```

15. Calcula la cantidad media de todos los pedidos que aparecen en la tabla pedido.

```
mysql> SELECT AVG(TOTAL) AS "MEDIA(TOTAL)" FROM PEDIDO;
+-----+
| MEDIA(TOTAL) |
+-----+
| 1312.0518749999999 |
+-----+
1 row in set (0,00 sec)
```

16. Calcula el número total de clientes que aparecen en la tabla cliente.

```
mysql> SELECT COUNT(ID) AS "TOTAL CLIENTES" FROM CLIENTE ;
+-----+
| TOTAL CLIENTES |
+-----+
| 10 |
+-----+
1 row in set (0,00 sec)
```

RONDA DE ATOCHA, 27 | MADRID | 28012 | WWW.SALESIANOSATOCCHA.ES

17. Calcula cuál es la mayor cantidad que aparece en la tabla pedido.

```
mysql> SELECT MAX(TOTAL) AS "MÁXIMO(TOTAL)" FROM PEDIDO;
+-----+
| MÁXIMO(TOTAL) |
+-----+
| 5760 |
+-----+
1 row in set (0,01 sec)
```

18. Calcula cuál es el valor máximo de categoría para cada una de las ciudades que aparece en la tabla cliente.

```
mysql> SELECT MAX(CATEGORÍA), CIUDAD FROM CLIENTE GROUP BY CIUDAD;
+-----+-----+
| MAX(CATEGORÍA) | CIUDAD |
+-----+-----+
| 200 | ALMERÍA |
| 225 | GRANADA |
| 300 | SEVILLA |
| 300 | JAÉN |
| 100 | CÁDIZ |
| 200 | HUELVA |
+-----+-----+
```

19. Calcula cuál es el máximo valor de los pedidos realizados durante el mismo día para cada uno de los clientes. Es decir, el mismo cliente puede haber realizado varios pedidos de diferentes cantidades el mismo día. Se pide que se calcule cuál es el pedido de máximo valor para cada uno de los días en los que un cliente ha realizado un pedido. Muestra el identificador del cliente, nombre, apellidos, la fecha y el valor de la cantidad.

```
mysql> SELECT CLIENTE.ID, NOMBRE, APELLIDO1, APELLIDO2, FECHA, MAX(TOTAL) FROM PEDIDO, CLIENTE WHERE CLIENTE.ID=PEDIDO.ID_CLIENTE GROUP BY FECHA;
```

ID	NOMBRE	APELLIDO1	APELLIDO2	FECHA	MAX(TOTAL)
8	PEPE	RUIZ	SANTANA	2015-06-27	250.45
7	PILAR	RUIZ	NULL	2018-07-27	2400.6
3	ADOLFO	RUBIO	FLORES	2018-08-17	110.5
1	AARÓN	RIVERO	GÓMEZ	2018-09-10	270.65
8	PEPE	RUIZ	SANTANA	2018-10-10	2480.4
6	MARÍA	SANTANA	MORENO	2019-01-25	545.75
1	AARÓN	RIVERO	GÓMEZ	2019-03-11	2389.23
2	ADELA	SALAS	DÍAZ	2019-09-10	5760
6	MARÍA	SANTANA	MORENO	2020-02-02	145.82
2	ADELA	SALAS	DÍAZ	2020-04-25	3045.6
5	MARCOS	LOYOLA	MÉNDEZ	2020-09-10	948.5
2	ADELA	SALAS	DÍAZ	2020-10-05	150.5
4	ADRIÁN	SUÁREZ	NULL	2020-10-10	1983.43

13 rows in set (0.00 sec)

20. Calcula cuál es el máximo valor de los pedidos realizados durante el mismo día para cada uno de los clientes, teniendo en cuenta que sólo queremos mostrar aquellos pedidos que superen la cantidad de 2000 €.

```
mysql> SELECT MAX(TOTAL),FECHA FROM CLIENTE,PEDIDO WHERE PEDIDO.TOTAL>2000 GROUP BY FECHA;
```

MAX(TOTAL)	FECHA
2400.6	2018-07-27
2480.4	2018-10-10
2389.23	2019-03-11
5760	2019-09-10
3045.6	2020-04-25

5 rows in set (0.00 sec)

21. Devuelve el pedido más caro que existe en la tabla pedido usando algún tipo de subconsulta.

```
mysql> SELECT * FROM PEDIDO WHERE TOTAL LIKE (SELECT MAX(TOTAL) FROM PEDIDO);
```

ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
7	5760	2019-09-10	2	1

1 row in set (0.00 sec)

22. Calcula el máximo valor de los pedidos realizados para cada uno de los comerciales durante el año 2019. Muestra el identificador del comercial, nombre, apellidos y total.

```
mysql> SELECT PEDIDO.ID_COMERCIAL,NOMBRE,APELLIDO1,APELLIDO2,MAX(TOTAL) FROM PEDIDO,COMERCIAL WHERE PEDIDO.ID_COMERCIAL=COMERCIAL.ID AND FECHA BETWEEN '2019-01-01' AND '2019-12-31' GROUP BY NOMBRE;
```

ID_COMERCIAL	NOMBRE	APELLIDO1	APELLIDO2	MAX(TOTAL)
5	ANTONIO	CARRETERO	ORTEGA	2389.23
1	DANIEL	SÁEZ	VEGA	5760

2 rows in set (0.03 sec)

23. Devuelve un listado con el identificador de cliente, nombre y apellidos y el número total de pedidos que ha realizado cada uno de los clientes.

```
mysql> SELECT CLIENTE.ID,NOMBRE, APELLIDO1, APELLIDO2, COUNT(PEDIDO.ID) FROM CLIENTE,PEDIDO WHERE PEDIDO.ID_CLIENTE=CLIENTE.ID GROUP BY CLIENTE.ID ;
```

ID	NOMBRE	APELLIDO1	APELLIDO2	COUNT(PEDIDO.ID)
1	AARÓN	RIVERO	GÓMEZ	3
2	ADELA	SALAS	DÍAZ	3
3	ADOLFO	RUBIO	FLORES	1
4	ADRIÁN	SUÁREZ	NULL	1
5	MARCOS	LOYOLA	MÉNDEZ	2
6	MARÍA	SANTANA	MORENO	2
7	PILAR	RUIZ	NULL	1
8	PEPE	RUIZ	SANTANA	3

8 rows in set (0.00 sec)

24. Devuelve un listado con el identificador de cliente, nombre y apellidos y el número total de pedidos que ha realizado cada uno de los clientes durante el año 2020.

```
mysql> SELECT CLIENTE.ID,NOMBRE, APELLIDO1, APELLIDO2, COUNT(PEDIDO.ID) FROM CLIENTE,PEDIDO WHERE PEDIDO.ID_CLIENTE=CLIENTE.ID GROUP BY CLIENTE.ID ;
```

ID	NOMBRE	APELLIDO1	APELLIDO2	COUNT(PEDIDO.ID)
1	AARÓN	RIVERO	GÓMEZ	3
2	ADELA	SALAS	DÍAZ	3
3	ADOLFO	RUBIO	FLORES	1
4	ADRIÁN	SUÁREZ	NULL	1
5	MARCOS	LOYOLA	MÉNDEZ	2
6	MARÍA	SANTANA	MORENO	2
7	PILAR	RUIZ	NULL	1
8	PEPE	RUIZ	SANTANA	3

8 rows in set (0.00 sec)

25. Devuelve un listado que muestre el identificador de cliente, nombre, primer apellido y el valor de la máxima cantidad del pedido realizado por cada uno de los clientes. El resultado debe mostrar aquellos clientes que no han realizado ningún pedido indicando que la máxima cantidad de sus pedidos realizados es 0. Puede hacer uso de la función IF NULL.

```
mysql> SELECT CLIENTE.ID, NOMBRE, APELLIDO1,IFNULL(MAX(TOTAL),0)FROM CLIENTE LEFT JOIN PEDIDO ON CLIENTE.ID = PEDIDO.ID_CLIENTE GROUP BY CLIENTE.ID;
```

ID	NOMBRE	APELLIDO1	IFNULL(MAX(TOTAL),0)
1	AARÓN	RIVERO	2389.23
2	ADELA	SALAS	5760
3	ADOLFO	RUBIO	75.29
4	ADRIÁN	SUÁREZ	1983.43
5	MARCOS	LOYOLA	948.5
6	MARÍA	SANTANA	545.75
7	PILAR	RUIZ	2400.6
8	PEPE	RUIZ	2480.4
9	GUILLERMO	LÓPEZ	0
10	DANIEL	SANTANA	0

10 rows in set (0.00 sec)

26. Devuelve cuál ha sido el pedido de máximo valor que se ha realizado cada año.

```
mysql> SELECT * FROM PEDIDO WHERE TOTAL = ANY (SELECT MAX(TOTAL) FROM PEDIDO GROUP BY YEAR(FECHA));
```

ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
7	5760	2019-09-10	2	1
9	2480.4	2018-10-10	8	3
10	250.45	2015-06-27	8	2
12	3045.6	2020-04-25	2	1

4 rows in set (0.03 sec)

27. Devuelve un listado con todos los pedidos que ha realizado Adela Salas Díaz.

```
mysql> SELECT * FROM PEDIDO WHERE ID_CLIENTE LIKE (SELECT ID FROM CLIENTE WHERE NOMBRE LIKE "ADELA" AND APELLIDO1 LIKE "SALAS" AND APELLIDO2 LIKE "DÍAZ");
```

ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
3	65.26	2020-10-05	2	1
7	5760	2019-09-10	2	1
12	3045.6	2020-04-25	2	1

3 rows in set (0.00 sec)

28. Devuelve el número de pedidos en los que ha participado el comercial Daniel Sáez Vega.

```
mysql> SELECT COUNT(*) FROM PEDIDO WHERE ID_COMERCIAL LIKE (SELECT ID FROM COMERCIAL WHERE NOMBRE LIKE "DANIEL" AND APELLIDO1 LIKE "SÁEZ" AND APELLIDO2 LIKE "VEGA");
```

COUNT(*)
6

1 row in set (0.01 sec)

29. Devuelve los datos del cliente que realizó el pedido más caro en el año 2019.

```
mysql> SELECT * FROM CLIENTE WHERE ID LIKE (SELECT ID_CLIENTE FROM PEDIDO WHERE TOTAL LIKE (SELECT MAX(TOTAL) FROM PEDIDO WHERE YEAR(FECHA) LIKE 2019));
```

ID	NOMBRE	APELLIDO1	APELLIDO2	CIUDAD	CATEGORÍA
2	ADELA	SALAS	DÍAZ	GRANADA	200

1 row in set (0.00 sec)



30. Devuelve un listado con los datos de los clientes y los pedidos, de todos los clientes que han realizado un pedido durante el año 2020 con un valor mayor o igual al valor medio de los pedidos realizados durante ese mismo año.

```
mysql> SELECT * FROM CLIENTE, PEDIDO WHERE CLIENTE.ID = PEDIDO.ID_CLIENTE AND YEAR(FECHA) LIKE 2020 AND TOTAL >= (SELECT AVG(TOTAL) FROM PEDIDO WHERE YEAR(FECHA) LIKE 2020);
```

ID	NOMBRE	APELLIDO1	APELLIDO2	CIUDAD	CATEGORÍA	ID	TOTAL	FECHA	ID_CLIENTE	ID_COMERCIAL
4	ADRIÁN	SUÁREZ	NULL	JAÉN	300	8	1983.43	2020-10-10	4	6
2	ADELA	SALAS	DÍAZ	GRANADA	200	12	3045.6	2020-04-25	2	1

```
2 rows in set (0.00 sec)
```