

EXAMEN SQL DE BASE DE DATOS

```
mysql> show tables;
+-----+
| Tables_in_examen_sql |
+-----+
| detalles_pedido       |
| pedidos               |
| productos              |
+-----+
3 rows in set (0.00 sec)
```

```
mysql> describe pedidos;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| PEDIDO_ID  | int(11)   | YES  |     | NULL    |       |
| CLIENTE_ID | int(11)   | YES  |     | NULL    |       |
| PEDIDO_TOTAL | int(11)   | YES  |     | NULL    |       |
| PEDIDO_TIMESTAMP | text      | YES  |     | NULL    |       |
| USER_ID    | int(11)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

```
mysql> describe productos;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| PRODUCTO_ID | int(11)   | YES  |     | NULL    |       |
| PRODUCTO_NAME | text      | YES  |     | NULL    |       |
| PRODUCTO_DESCRIPTION | text      | YES  |     | NULL    |       |
| CATEGORIA    | text      | YES  |     | NULL    |       |
| PRODUCTO_AVAIL | text      | YES  |     | NULL    |       |
| PRECIO       | int(11)   | YES  |     | NULL    |       |
| PRODUCTO_IMAGE | text      | YES  |     | NULL    |       |
| MIMETYPE     | text      | YES  |     | NULL    |       |
| FILENAME     | text      | YES  |     | NULL    |       |
| IMAGE_LAST_UPDATE | text      | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

```
mysql> describe detalles_pedido;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| PEDIDO_ITEM_ID | int(11)   | YES  |     | NULL    |       |
| PEDIDO_ID      | int(11)   | YES  |     | NULL    |       |
| PRODUCTO_ID    | int(11)   | YES  |     | NULL    |       |
| UNIT_PRICE     | int(11)   | YES  |     | NULL    |       |
| CANTIDAD       | int(11)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

1-Productos que nunca se han vendido:

```
mysql> SELECT producto_name FROM productos INNER JOIN detalles_pedido ON productos.producto_id=detalles_pedido.producto_id WHERE detalles_pedido.cantidad IS NULL;
Empty set (0.00 sec)
```

2-Se desea añadir un código nuevo a los productos que contenga para cada producto: las tres primeras letras de la categoría, un guión, las tres últimas del nombre del producto junto con el número de caracteres de la descripción. Realiza un listado donde aparezca el nombre del producto, la categoría y el código nuevo. Por ejemplo para <<Business Shirt>> Mens el código nuevo sería: Men-irt34.

```
mysql> SELECT producto_id, producto_name, CONCAT(LEFT(categoria,3), "-", RIGHT(producto_name,3), LENGTH(producto_description)) AS new_code FROM productos;
+-----+-----+-----+
| producto_id | producto_name | new_code |
+-----+-----+-----+
| 1 | Business Shirt | Men-irt34 |
| 2 | Trousers | Men-ers46 |
| 3 | Jacket | Men-ket79 |
| 4 | Blouse | Wom-use40 |
| 5 | Skirt | Wom-irt18 |
| 6 | Ladies Shoes | Wom-oes81 |
| 7 | Belt | Acc-elt12 |
| 8 | Bag | Acc-Bag76 |
| 9 | Mens Shoes | Men-oes37 |
| 10 | Wallet | Acc-let99 |
+-----+-----+-----+
10 rows in set (0.00 sec)
```

3-Cuánto se factura por meses. Ordena cronológicamente.

```
mysql> SELECT DISTINCT MONTH(pedido_timestamp), SUM(pedido_total) FROM pedidos GROUP BY MONTH(pedido_timestamp) ORDER BY MONTH(pedido_timestamp) ASC;
+-----+-----+
| MONTH(pedido_timestamp) | SUM(pedido_total) |
+-----+-----+
| NULL | 13030 |
+-----+-----+
1 row in set, 11 warnings (0.00 sec)
```

Debido al formato de las tablas, se ha acordado que se resolvería de esta forma a pesar de que el resultado no saliese correctamente.

4-Para cada pedido, comprueba si el campo PEDIDO TOTAL coincide exactamente con lo consumido para dicho pedido.

5-Pedidos que se realizaron el 15 de noviembre y el 31 de diciembre de cualquier año.

```
mysql> SELECT pedido_id FROM pedidos WHERE pedido_timestamp LIKE '15/11/%' OR pedido_timestamp LIKE '31/12/%';
Empty set (0.00 sec)
```

Aquí pasa igual que en el 3, debido al formato de fecha se ha acordado colocarlo de esta forma, a pesar de que igualmente no saldría ningún resultado debido a las tablas.

6-Clientes (CLIENTE_ID) que han consumido más de 1000\$ (considera los campos cantidad y unit_price de DETALLES_PEDIDO).

```
mysql> SELECT DISTINCT cliente_id, SUM<pedido_total> FROM pedidos GROUP BY cliente_id HAVING SUM<pedido_total>>1000;
```

cliente_id	SUM<pedido_total>
7	2760
1	2380
2	2370
5	1090
3	2420
4	1060

6 rows in set (0.00 sec)

7-Indicar los tres productos que más se han vendido (tener en cuenta el campo Cantidad).

```
mysql> SELECT producto_id, COUNT<producto_id> FROM detalles_pedido GROUP BY producto_id ORDER BY COUNT<producto_id>DESC LIMIT 3;
```

producto_id	COUNT<producto_id>
2	6
3	6
8	6

3 rows in set (0.00 sec)

8-El producto más caro.

```
mysql> SELECT producto_name, precio FROM productos WHERE precio=(SELECT MAX<precio> FROM productos);
```

producto_name	precio
Jacket	150

1 row in set (0.00 sec)