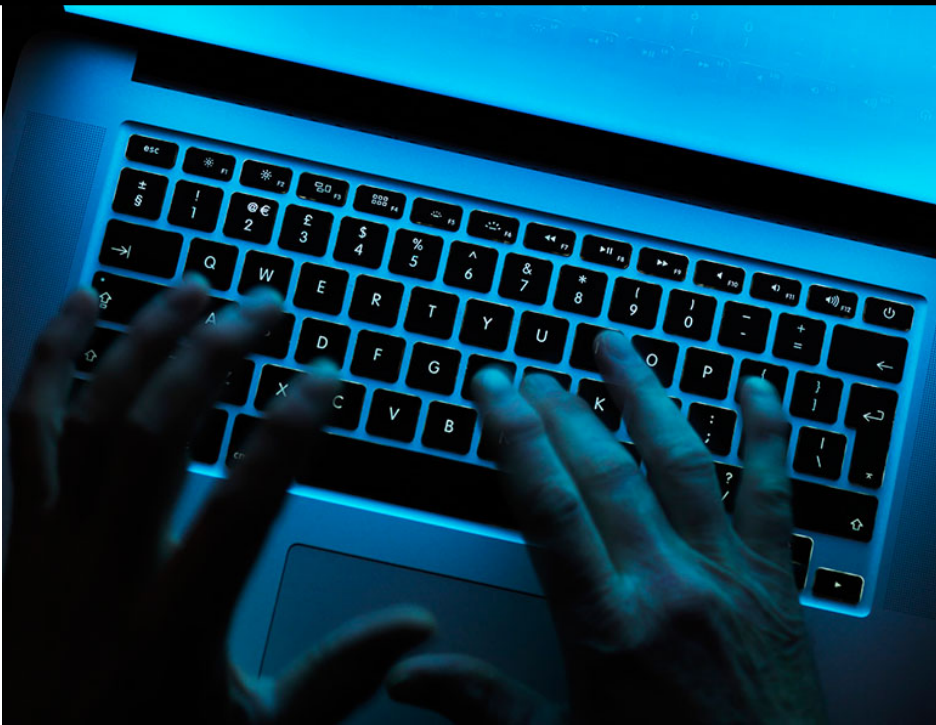


UNIDAD 4

REALIZACIÓN DE CONSULTAS



López Coronel David

DAM/ Base de Datos / Grupo (A)

UNIDAD 4

¿Qué te pedimos que hagas?

Para poder acceder a información de una Base de Datos, ésta debe estar creada y debe contener registros previamente. Por tanto, lo primero que debes realizar es descargar el script que contiene las tablas y datos que encontrarás en el apartado 2.- "*Información de interés*", además de seguir todos los consejos y recomendaciones para elaborar esta tarea que en dicho apartado se explican.

Lo que realmente se pide en la tarea es que ayudes a Ana redactando las sentencias SQL que ejecuten cada una de las siguientes consultas correctamente en MySQL.

IMPORTANTE: Las **sentencias deben estar escritas directamente en el documento** que se entregue y no incluida en ninguna captura de pantalla.

APARTADO A:

- 1.- Obtener un listado de todas las especialidades.....**Pág: 4.**
- 2.- Obtener el id de las incidencias junto a la descripción, los puntos que cuesta, el número de cliente que la ha puesto y la fecha de apertura de todas aquellas cuyo estado sea abierta..... **Pág: 4.**
- 3.- Obtener todos los campos de los usuarios que tienen un valor en puntos acumulados entre 100 y 500..... **Pág: 4.**
- 4.- Obtener el identificador y la descripción de las incidencias que tengan fecha de aprobación en el último trimestre del 2019 y además ya estén adjudicadas..... **Pág: 4.**

APARTADO B:

- 5.- Obtener por cada técnico, su correspondiente número junto al total de incidencias que cada uno tiene asignada..... **Pág: 5.**
- 6.- Obtener por cada mes (con el formato de nombre y no de número ej. Noviembre), la cantidad de incidencias que se han reparado en cada uno de ellos..... **Pág: 5.**
- 7.- Obtener un listado con el nombre completo del usuario en el mismo campo y anteponiendo los apellidos al nombre de la siguiente forma (ej. Serrá Martínez, Vicente) que tenga como especialidad "Albañilería" o "Fontanería". Ordenar el listado por apellido..... **Pág: 5.**

8.- Obtener el nombre, apellidos y la valoración total de aquellos usuarios que superen o igualen la media de todas las valoraciones. Ordena el listado de mayor a menor valoración..... **Pág: 6.**

9.- Obtener un listado con el nombre y apellidos de los usuarios que no han puesto ninguna incidencia..... **Pág: 6.**

10.- Obtener el identificador, la descripción, el estado, los puntos de coste de las incidencias y el nombre del usuario completo (en el mismo campo) que ha abierto la incidencia de aquellas que estén asignadas al técnico de nombre "Sandra". Ordena el listado de mayor a menor valor en los puntos de coste de las incidencias..... **Pág: 6.**

11.- Obtener por cada técnico, su nombre, la cantidad de incidencias que no siguen abiertas (cualquier otro estado) y la media de días que han tardado en aprobar las incidencias desde que se abrieron redondeando con dos decimales ese valor. Ordenar el listado de menor a mayores días..... **Pág: 7.**

12.- Obtener un listado con dos registros. En uno debe aparecer el identificador, la descripción y el coste de puntos cuyo valor del campo puntos_coste sea el mayor de todas las incidencias y en el otro el identificador, la descripción y el coste de puntos cuyo valor sea el menor en todas ellas..... **Pág: 7.**

13.- Obtener por cada especialidad, el nombre de cada una y el número total de incidencias que tienen asignadas los usuarios en cada una de las especialidades teniendo en cuenta que el total de incidencias sea superior o igual a dos. Ordena el listado de mayor a menor número de incidencias..... **Pág: 8.**

APARTADO C:

14.- Obtener un listado con el nombre y apellidos de los usuarios) que hayan reparado alguna incidencia y por tanto estén finalizadas obteniendo además el total de incidencias que haya reparado junto a la suma total de puntos que ha conseguido por las incidencias reparadas siempre y cuando esa suma total sea mayor que la media de los puntos de coste del total de incidencias..... **Pág: 8.**

15.- Obtener un listado con el nombre completo (nombre y apellidos juntos) de los usuarios clientes (que han abierto la incidencia), el identificador y la descripción de incidencia que abren y también el nombre completo en el mismo campo del usuario que repara la incidencia, así como los días que transcurren desde que se aprobó hasta que finalizó siempre y cuando las incidencias tengan que ver con las especialidades de Fontanería, Pintura o Albañilería..... **Pág: 9.**

APARTADO A:**1.- Obtener un listado de todas las especialidades.**

```
SELECT nombre FROM especialidad;
```

2.- Obtener el id de las incidencias junto a la descripción, los puntos que cuesta, el número de cliente que la ha puesto y la fecha de apertura de todas aquellas cuyo estado sea abierta.

```
SELECT id, descripcion, puntos_coste, num_cliente, f_abierta FROM  
incidencia
```

```
WHERE estado='abierta';
```

3.- Obtener todos los campos de los usuarios que tienen un valor en puntos acumulados entre 100 y 500.

```
SELECT * FROM usuario
```

```
WHERE puntos_acumulados BETWEEN 100 AND 500;
```

4.- Obtener el identificador y la descripción de las incidencias que tengan fecha de aprobación en el último trimestre del 2019 y además ya estén adjudicadas.

```
SELECT id, descripcion FROM incidencia
```

```
WHERE f_aprobada >= '2019-10-01' AND f_aprobada <= '2019-12-31'
```

```
AND estado = 'adjudicada'
```

```
;
```

APARTADO B:

5.- Obtener por cada técnico, su correspondiente número junto al total de incidencias que cada uno tiene asignada.

```
SELECT n_tecnico, COUNT(n_tecnico) AS num_incidencias  
FROM incidencia  
GROUP BY n_tecnico ;
```

6.- Obtener por cada mes (con el formato de nombre y no de número ej. Noviembre), la cantidad de incidencias que se han reparado en cada uno de ellos.

```
SELECT MONTHNAME(f_abierta) Mes, YEAR(f_abierta) Año,  
COUNT(f_abierta) Total_incidencias  
FROM incidencia  
GROUP BY Mes  
ORDER BY (f_abierta);
```

7.- Obtener un listado con el nombre completo del usuario en el mismo campo y anteponiendo los apellidos al nombre de la siguiente forma (ej; Serrá Martínez, Vicente) que tenga como especialidad "Albañilería" o "Fontanería". Ordenar el listado por apellido.

```
SELECT CONCAT(apellidos,', ', u.nombre) Nombre_Usuario  
FROM usuario u, especialidad e  
WHERE (especialista = codigo) AND (codigo= 11 OR codigo= 22)  
ORDER BY apellidos  
;
```

8.- Obtener el nombre, apellidos y la valoración total de aquellos usuarios que superen o igualen la media de todas las valoraciones. Ordena el listado de mayor a menor valoración.

```
SELECT nombre, apellidos, valoracion_total FROM usuario  
  
WHERE valoracion_total >=  
  
(SELECT AVG(valoracion_total) FROM usuario)  
  
ORDER BY valoracion_total DESC;
```

9.- Obtener un listado con el nombre y apellidos de los usuarios que no han puesto ninguna incidencia.

```
SELECT usuario.nombre, usuario.apellidos  
  
FROM usuario  
  
LEFT OUTER JOIN incidencia ON  
(usuario.numero=incidencia.num_cliente)  
  
WHERE num_cliente IS NULL  
  
;
```

10.- Obtener el identificador, la descripción, el estado, los puntos de coste de las incidencias y el nombre del usuario completo (en el mismo campo) que ha abierto la incidencia de aquellas que estén asignadas al técnico de nombre "Sandra". Ordena el listado de mayor a menor valor en los puntos de coste de las incidencias.

```
SELECT ID, descripcion, estado, puntos_coste, CONCAT(usuario.nombre,  
' ', apellidos) Nombre_usuario  
  
FROM incidencia  
  
LEFT OUTER JOIN usuario ON (incidencia.num_cliente=usuario.numero)  
  
LEFT OUTER JOIN tecnico ON (incidencia.n_tecnico=tecnico.numero)  
  
WHERE tecnico.numero= 20  
  
ORDER BY puntos_coste DESC;
```

11.- Obtener por cada técnico, su nombre, la cantidad de incidencias que no siguen abiertas (cualquier otro estado) y la media de días que han tardado en aprobar las incidencias desde que se abrieron redondeando con dos decimales ese valor. Ordenar el listado de menor a mayores días.

```
SELECT nombre, COUNT(id) AS cantid_incidencias,
ROUND(AVG(DATEDIFF(f_aprobada, f_abierta)),2) as media_dias
FROM tecnico, incidencia
WHERE estado!= 'Abierta' AND numero= n_tecnico
GROUP BY nombre;
```

12.- Obtener un listado con dos registros. En uno debe aparecer el identificador, la descripción y el coste de puntos cuyo valor del campo puntos_coste sea el mayor de todas las incidencias y en el otro el identificador, la descripción y el coste de puntos cuyo valor sea el menor en todas ellas.

```
SELECT id, descripcion, puntos_coste
FROM incidencia
WHERE puntos_coste = (select max(puntos_coste) FROM incidencia)
UNION ALL
SELECT id, descripcion, puntos_coste
FROM incidencia
WHERE puntos_coste = (select min(puntos_coste) FROM incidencia)
GROUP BY incidencia.puntos_coste;
```

***Como resultado me da 3 registros;**

1 puntos MAX

2 puntos MIN

GROUP BY incidencia.puntos_coste; (pues si añado esto ya obtengo solo los dos registros que se pide en el ejercicio). Creo que ambos resultados estarían bien, por eso los menciono.

13.- Obtener por cada especialidad, el nombre de cada una y el número total de incidencias que tienen asignadas los usuarios en cada una de las especialidades teniendo en cuenta que el total de incidencias sea superior o igual a dos. Ordena el listado de mayor a menor número de incidencias.

```
SELECT especialidad.nombre, COUNT(num_cliente) AS  
num_total_incidencias  
  
FROM especialidad  
  
LEFT OUTER JOIN usuario ON (especialidad.codigo=usuario.especialista)  
  
LEFT OUTER JOIN incidencia ON  
(incidencia.num_cliente=usuario.numero)  
  
GROUP BY especialidad.nombre  
  
HAVING num_total_incidencias >= 2  
  
ORDER BY num_total_incidencias DESC;
```

APARTADO C:

14.- Obtener un listado con el nombre y apellidos de los usuarios) que hayan reparado alguna incidencia y por tanto estén finalizadas obteniendo además el total de incidencias que haya reparado junto a la suma total de puntos que ha conseguido por las incidencias reparadas siempre y cuando esa suma total sea mayor que la media de los puntos de coste del total de incidencias.

```
SELECT nombre, apellidos, COUNT(num_repara) AS  
total_incidencias_reparadas, SUM(puntos_coste) AS suma_total_puntos  
FROM incidencia, usuario  
  
WHERE estado= 'Finalizada' AND incidencia.num_repara=usuario.numero  
  
GROUP BY nombre  
  
HAVING suma_total_puntos >  
(SELECT AVG(incidencia.puntos_coste) FROM incidencia);
```

15.- Obtener un listado con el nombre completo (nombre y apellidos juntos) de los usuarios clientes (que han abierto la incidencia), el identificador y la descripción de incidencia que abren y también el nombre completo en el mismo campo del usuario que repara la incidencia, así como los días que transcurren desde que se aprobó hasta que finalizó siempre y cuando las incidencias tengan que ver con las especialidades de Fontanería, Pintura o Albañilería.

```
SELECT CONCAT(usu1.nombre, ' ', usu1.apellidos) AS Usuario_cliente,  
incidencia.ID, incidencia.descripcion,  
  
CONCAT(usu2.nombre, ' ', usu2.apellidos) AS Usuario_repara,  
DATEDIFF(f_repara, f_aprobada) AS dias_trancurridos  
  
FROM incidencia, usuario AS usu1, usuario AS usu2, especialidad  
  
WHERE incidencia.estado = 'Finalizada' AND incidencia.num_cliente =  
usu1.numero AND incidencia.num_repara = usu2.numero AND  
usu2.especialista = especialidad.codigo AND NOT especialidad.codigo =  
33 AND NOT especialidad.codigo = 44  
  
GROUP BY incidencia.ID;
```

David López Coronel