

GBD04- Solución a la TAREA

REALIZACIÓN DE CONSULTAS.

Partiendo de la base de datos TalleresFaber, creada y con datos que puedes descargar del recurso:

Insertar TalleresFaber.txt

Realiza las siguientes consultas:

1. Pasadas las fiestas navideñas Talleres Faber quiere enviar publicidad a sus clientes. Para ello quiere seleccionar a aquéllos a los que no se les ha emitido ninguna factura desde el 6 de Enero. Obtener un listado con Nombre, Apellidos y Dirección de esos clientes.
2. El gerente quiere conocer los apellidos y el nombre del empleado que más horas ha trabajado. La consulta debe mostrar 2 columnas:
 - Los apellidos y el nombre, separados por una coma y un espacio con el encabezado 'Apellidos y nombre'
 - Las horas totales trabajadas, con el encabezado 'Horas totales trabajadas'
3. ¿Cuál es el importe total facturado este mes por Talleres Faber?
 - a. Tener en cuenta todas las facturas registradas en la base de datos.
 - b. Es necesario recoger:
 - Importe total de los recambios: PrecioReferencia*Unidades
 - Importe total de las actuaciones: Horas*50
 - c. La consulta devolverá una sola fila con los siguientes resultados:
 - Número total de facturas emitidas
 - Importe total facturado
 - d. Utilizar alias para cambiar los encabezados.
4. Obtener un listado con el número de reparaciones en las que ha intervenido cada uno de los empleados. El listado debe mostrar:
 - a. El número de reparaciones, los apellidos y el nombre del empleado.
 - b. El número de reparaciones debe aparecer en orden descendente.
5. Los socios del taller pretenden dar un servicio cada vez más rápido a sus clientes. Para ello han previsto que las reparaciones se realicen antes de 2 días. Interesa conocer las reparaciones que han superado esa previsión y los empleados que la han realizado. Las columnas que se deben mostrar son:
 - Nombre de los empleados, Fecha de entrada, Fecha de salida y Fecha de salida prevista (2 días a partir de la fecha de entrada).
 - Combinar las tablas utilizando la Combinación interna (INNER JOIN).
6. La administración quiere comprobar que los únicos empleados que no han intervenido en reparaciones son aquéllos que se encargan de labores de organización y administración del taller. Hacer una consulta que devuelva los nombres y categorías de los empleados que no han intervenido en ninguna reparación.
7. Interesa conocer cuál es el vehículo con más kilómetros que ha visitado el taller. Mostrar los kilómetros, la marca, el modelo, el color y la matrícula.

8. En Talleres Faber están pensando en hacer más rentable la gestión de los recambios. Se pretende optimizar el stock, para ello se solicita un listado con los siguientes datos: IdRecambio, Descripción, UnidadBase y Stock, de los artículos que superen el stock considerado como óptimo, según los siguientes criterios:
- Para los artículos cuya unidad base se mide como **una unidad** se considera el stock óptimo hasta 10 artículos.
 - Para los artículos cuya unidad base **no sea una unidad** el stock considerado como óptimo es hasta 4 artículos.
9. Con relación a las actuaciones, se considera importante saber cuántas reparaciones se realizan de importe superior a la media. La consulta nos devolverá un solo dato: el número de reparaciones realizadas cuyo importe supera el importe medio de las actuaciones.
10. Obtener un listado que nos informe del número de facturas que hemos emitido a cada cliente. El listado tendrá dos columnas: el nombre completo del cliente y el número de facturas que le corresponden.

Nota para el tutor:

La solución propuesta a continuación es orientativa. Algunas consultas pueden resolverse de distintas formas.

SOLUCIÓN EJERCICIO 1:

```
SELECT Apellidos, Nombre, Direccion FROM CLIENTES
WHERE CodCliente NOT IN
(SELECT CodCliente FROM FACTURAS WHERE FechaFactura BETWEEN '2011-01-06' AND '2011-01-31');
```

SOLUCIÓN EJERCICIO 2:

```
SELECT CONCAT(Apellidos,', ', Nombre) AS 'Apellidos y Nombre', SUM(Horas) AS 'Horas Totales'
FROM EMPLEADOS E, Intervienen I
WHERE E.CodEmpleado=I.CodEmpleado
GROUP BY DNI
ORDER BY SUM(Horas) DESC
LIMIT 1;
```

SOLUCIÓN EJERCICIO 3:

```
SELECT count(IdFactura) as 'Numero de facturas',
(
SELECT sum(Unidades*PrecioReferencia)
FROM Incluyen Ic, RECAMBIOS REC
WHERE Ic.IdRecambio = REC.IdRecambio
)
+
(
SELECT sum(Horas*50)
FROM Realizan Rz, ACTUACIONES AC
WHERE Rz.Referencia = AC.Referencia
)
AS 'Importe total sin IVA'
FROM FACTURAS;
```

SOLUCIÓN EJERCICIO 4:

```
SELECT count(R.IdReparacion) AS 'Número de reparaciones', Apellidos, Nombre
FROM REPARACIONES R, Intervienen I, EMPLEADOS E
WHERE R.IdReparacion=I.IdReparacion
AND I.CodEmpleado=E.CodEmpleado
GROUP BY Apellidos, Nombre
ORDER BY 'Número de reparaciones' DESC;
```

SOLUCIÓN EJERCICIO 5:

```
SELECT EMPLEADOS.Nombre, FechaEntrada, FechaSalida, DATE_ADD(FechaEntrada, INTERVAL 2 DAY) AS  
'Salida prevista'  
FROM (REPARACIONES INNER JOIN Intervienen ON REPARACIONES.IdReparacion=Intervienen.IdReparacion)  
INNER JOIN EMPLEADOS ON Intervienen.CodEmpleado=EMPLEADOS.CodEmpleado  
WHERE FechaSalida > date_add(FechaEntrada, INTERVAL 2 DAY);
```

SOLUCIÓN EJERCICIO 6:

```
SELECT EMPLEADOS.Nombre, Categoria  
FROM EMPLEADOS LEFT JOIN Intervienen ON EMPLEADOS.CodEmpleado=Intervienen.CodEmpleado  
WHERE Intervienen.CodEmpleado IS NULL;
```

SOLUCIÓN EJERCICIO 7:

```
SELECT VEHICULOS.Matricula, Km, Marca, Modelo, Color  
FROM REPARACIONES, CLIENTES, VEHICULOS  
WHERE REPARACIONES.Matricula=VEHICULOS.Matricula  
AND VEHICULOS.CodCliente=CLIENTES.CodCliente AND Km =  
(SELECT MAX(Km) FROM REPARACIONES);
```

SOLUCIÓN EJERCICIO 8:

```
SELECT IdRecambio, Descripcion, UnidadBase FROM RECAMBIOS  
WHERE Stock>4 AND UnidadBase NOT LIKE 'Unidad'  
OR Stock>10 AND UnidadBase LIKE 'Unidad';
```

SOLUCIÓN EJERCICIO 9:

```
SELECT COUNT(R.IdReparacion) AS 'Número de reparaciones con actuaciones que superan el importe medio'  
FROM REPARACIONES R, Realizan Re  
WHERE R.IdReparacion=Re.IdReparacion  
AND Referencia IN  
(SELECT Referencia FROM ACTUACIONES WHERE Importe > (SELECT AVG(Importe) FROM actuaciones));
```

SOLUCIÓN EJERCICIO 10:

```
SELECT CONCAT(Apellidos,', ', Nombre) AS 'Cliente', Count(*) AS 'Número de facturas'  
FROM CLIENTES C, FACTURAS F  
WHERE C.CodCliente=F.CodCliente  
GROUP BY C.CodCliente;
```