

Practicar con la lista de selección

Estos son unos ejemplos para que te vayas familiarizando con las cláusulas de la SELECT. Aquí practicaremos los puntos vistos en la lista de selección.

Puedes probarlos utilizando la base de datos GestionA suministrada en la unidad 2.

Tienes el enunciado seguido de la respuesta, utiliza una hoja de papel para tapar la solución e intenta redactar la respuesta sin mirarla.

Un consejo: Para empezar plantea la consulta como varias preguntas en este orden:

- ¿Dónde están los datos que necesito? La respuesta me dará la FROM
- ¿Qué columnas quiero visualizar en el resultado? La respuesta me dará la lista de selección.
 - Si una de las columnas no se obtiene directamente de una columna de la tabla tendré que escribir una expresión.
 - Si quiero cambiar el encabezado de alguna columna tengo que añadir un alias de campo.

1.- Escribe 3 sentencias equivalentes para obtener todos los datos de la tabla Productos.

```
SELECT *  
FROM Productos;
```

```
SELECT Productos.*  
FROM Productos;
```

```
SELECT idfab, idproducto, descripcion, precio, existencias  
FROM Productos;
```

2.- Listar nombres, oficinas (código de la oficina), y fechas de contrato de todos los empleados

```
SELECT nombre, oficina, contrato  
FROM empleados;
```

3.- Cambia la consulta para que aparezca en el encabezado: Empleado | oficina | Fecha de contrato.

```
SELECT nombre AS Empleado, oficina, contrato AS [Fecha de contrato]  
FROM empleados;
```

4.- Listar una tarifa de productos, la tarifa es un listado donde aparece el código, nombre y precio de los productos, recuerda que el código del producto está formado por dos campos.

```
SELECT idfab, idproducto, descripcion, productos.precio  
FROM productos;
```

5.- Cambia la consulta anterior para que aparezca una sola columna para el código del producto y en esta columna aparecerá el idfab e idproducto juntos.

```
SELECT idfab + idproducto AS Codigo, descripcion, productos.precio  
FROM productos;
```

6.- Cambia la consulta anterior para que ahora en la columna Codigo aparezca el código del producto de la forma ACI – 41003 (el idfab e idproducto separados por un guión).

```
SELECT idfab + ' - ' + idproducto AS Codigo, descripcion, productos.precio  
FROM productos;
```

7.- Listar la ciudad, región y el superávit de cada oficina. El superávit es el volumen de ventas que se encuentran por encima o por debajo del objetivo de la oficina. Si la oficina ha vendido más que su objetivo el superavit será positivo, si ha vendido menos será negativo.

```
SELECT ciudad, region, (ventas-objetivo) AS superavit
FROM oficinas;
```

7.- De cada producto queremos saber el id de fabricante, id de producto, su descripción, sus existencias (unidades en el almacén), su precio y el valor de sus existencias (las existencias convertidas a euros).

```
SELECT idfab,idproducto,descripcion, existencias, precio,
(existencias*precio) AS valoracion
FROM productos;
```

8.- El precio que aparece en la tabla productos no lleva el IVA incluido, lista los productos añadiendo una columna que sea Precio Con IVA, todos nuestros productos tienen un IVA del 18%. Limitate a obtener el valor, luego en los ejercicios posteriores lo dejaremos "bonito".

```
SELECT idfab,idproducto,descripcion, existencias, precio,
(precio* 1.18) AS [Precio Con IVA]
FROM productos;
```

9.- En la consulta anterior el precio con IVA aparece con muchos decimales, modifica la consulta para que aparezca redondeado a 2 decimales. Lo mismo de antes, haz que salga el valor redondeado sin preocuparte del formato.

```
SELECT idfab,idproducto,descripcion, existencias, precio,
ROUND((precio* 1.18),2) AS [Precio Con IVA]
FROM productos;
```

10.- En la consulta anterior el precio con IVA ya aparece redondeado pero siguen los muchos decimales, lo tenemos que formatear. Para formatear número lo mejor es utilizar la función STR que además (si lo queremos) redondea.

```
SELECT idfab,idproducto,descripcion, existencias, precio,
ROUND(precio* 1.18,2) AS [Precio Con IVA],
STR(precio*1.18,6,2) AS [Con STR]
FROM productos;
```

Entonces, si la función STR me permite redondear y dejarlo más bonito, ¿para qué utilizar ROUND?

La función STR convierte el resultado a cadena (valor alfanumérico), mientras que ROUND sigue dando un número, lo verás con el siguiente ejemplo.

11.- Añade a la consulta anterior la valoración de las existencias pero ahora cogiendo como precio el precio con IVA.

```
SELECT idfab,idproducto,descripcion, existencias, precio,
ROUND((precio* 1.18),2) AS [Precio Con IVA],
STR(precio*1.18,6,2) AS [Con STR],
existencias * ROUND((precio* 1.18),2) AS Valoracion,
existencias * STR(precio*1.18,6,2) -- esto da error porque no se puede
-- multiplicar un número por un caracter
FROM productos;
```

12.- Listar el nombre, mes y año de contrato de cada empleado (quiero el mes y el año en columnas separadas con los encabezados Mes de contrato y Año de contrato).

```
SELECT nombre, MONTH(contrato) AS [Mes de contrato], YEAR(contrato) AS [Año de contrato]
FROM empleados;
```

13.- De cada empleado quiero su código, nombre, fecha de contrato y la fecha de hoy.

```
SELECT numemp, nombre, contrato, GETDATE() AS Hoy
FROM empleados;
```

14.- Quiero que ahora las fechas aparezcan sin la hora, la fecha de contrato con el año de 4 dígitos, y la fecha de hoy con 2 dígitos para el año.

```
SELECT numemp, nombre, contrato,
CONVERT(CHAR(10),contrato,103) AS Contrato,
CONVERT(CHAR(8),GETDATE(),3) AS Hoy
FROM empleados;
```

15.- Añade a la consulta anterior el número de días trabajados.

```
SELECT numemp, nombre, contrato,
CONVERT(CHAR(10),contrato,103) AS Contrato,
CONVERT(CHAR(8),GETDATE(),3) AS Hoy,
DATEDIFF( day , contrato , GETDATE() ) AS [Dias Trabajados]
FROM empleados;
```

Puedes practicar con las funciones de fechas.

15.- Listar las ventas de cada oficina con el formato: 22 tiene ventas de 186,042.00 €
Tienen que aparecer 3 columnas.

```
SELECT oficina, 'tiene ventas de ' AS [ ], ventas
FROM oficinas;
```

Cuando no queremos que aparezca nada en el encabezado rellenamos los corchetes con blancos.

Aparece siempre en el resultado tantas columnas como términos (separados por comas) aparezcan en la lista de selección.

16.- Rectifica el ejercicio anterior para que el literal aparezca en la primera columna. Sólo tendremos dos columnas.

Pista: Ojo la columna oficina es de tipo numérico, luego nos dará problemas en la concatenación.

```
SELECT STR(oficina) + ' tiene ventas de ' AS [ ], ventas
FROM oficinas;
```

Si intentamos concatenar oficina + ' tiene ventas de ' nos sale el siguiente error:

“Error de conversión al convertir el valor varchar ' tiene ventas de ' al tipo de datos int”.

Según nos dice ha intentado convertir ' tiene ventas de ' al tipo int ¿Por qué?

Se ha encontrado un campo entero (oficina) seguido de + , luego entiende que tiene que sumar oficina a lo que viene detrás 'tiene ventas de' como no es entero lo intenta convertir para hacer la suma y claro, da error.

Esto nos indica que tenemos que hacer que oficina sea una cadena para que pueda concatenar, para ello utilizamos la función STR() que convierte números a alfanuméricos.

Recuerda que tienes que aprender a interpretar los mensajes de error.