

Resolución de la etapa 5

Para llevar a cabo esta etapa, debes tener abierto **MySQL Workbench** y la base de datos **TECNO_DB** en uso. Será necesario que hayas completado el proyecto integrador anterior.

Te proporcionamos, en las siguientes *slides*, una solución modelo que puedes usar como referencia.



Ejercicio 1

1. Utilizar la tabla **Clientes** y generar una consulta que muestre las columnas **ClienteID** y **Dirección**. En el resultado de la consulta, se debe mostrar una nueva columna llamada **Nombre completo** que concatene las columnas **Apellido**, **Nombre**, separando los valores de estos campos por un **guion**.

```
SELECT
    ClienteID, Direccion,
    CONCAT(Apellido, '-', Nombre) AS NombreCompleto
FROM
    CLIENTES;
```

2. Repite el ejercicio anterior, utilizando la función ***CONCAT_WS***.

```
SELECT
    ClienteID, Direccion,
    CONCAT_WS('-', Apellido, Nombre) AS NombreCompleto
FROM
    CLIENTES;
```

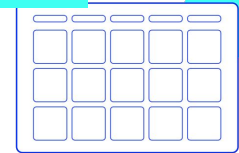


3. Modificar el ejercicio anterior para mostrar **en mayúsculas** los valores cargados en el campo **Nombre**.

```
SELECT
    ClienteID, Direccion,
    CONCAT_WS('-', Apellido, UPPER(Nombre)) AS NombreCompleto
FROM
    CLIENTES;
```

4. Generar una consulta que muestre la columna **Nombre**. En el resultado de la consulta, se debe mostrar una **nueva columna** con el nombre **Inicial** que muestre la **inicial** de los datos cargados en el **campo Nombre**.

```
SELECT
    Nombre,
    LEFT(Nombre, 1) AS Inicial
FROM
    CLIENTES;
```



5. Utilizar la tabla **Facturas** y obtener una lista de **todas las facturas** cargadas en la tabla. Generar una **nueva columna** con el nombre **IVA** que calcule el **21% del total de cada factura**, obteniendo un valor numérico que, **como máximo, contenga 2 decimales**.

```
SELECT
    FacturaID,
    Letra,
    Numero,
    ClienteID,
    Fecha,
    Monto,
    ROUND(Monto * 0.21, 2) AS IVA
FROM
    FACTURAS;
```

6. Modificar la consulta anterior, agregando una nueva columna con el nombre **NETO**, que calcule el **total a pagar por cada cliente por las compras realizadas** (es decir, sumándole el IVA al cargo original, manteniendo **como máximo 2 decimales**).

```
SELECT
    FacturaID,
    Letra,
    Numero,
    ClienteID,
    Fecha,
    Monto,
    ROUND(Monto * 0.21, 2) AS IVA,
    ROUND(Monto * 1.21, 2) AS NETO
FROM
    FACTURAS;
```



7. Modificar la consulta anterior, agregando una **nueva columna** con el nombre ***REDONDEO A FAVOR CLIENTE*** que devuelva el **valor entero inferior del neto** calculado anteriormente.

```
SELECT
    FacturaID,
    Letra,
    Numero,
    ClienteID,
    Fecha,
    Monto,
    ROUND(Monto * 0.21, 2) AS IVA,
    ROUND(Monto * 1.21, 2) AS NETO,
    FLOOR(ROUND(Monto * 1.21, 2)) AS 'REDONDEO A FAVOR CLIENTE'
FROM
    FACTURAS;
```


Ejercicio 2

1. Utilizando la tabla **Facturas**, obtener una lista de todas aquellas facturas efectuadas a lo largo del **año 2021**.

```
SELECT * FROM Facturas  
WHERE YEAR(Fecha) = 2021;
```



2. Obtener una lista de todas aquellas **Facturas** efectuadas durante los **meses de marzo y septiembre del año 2021**.

```
SELECT * FROM Facturas WHERE  
YEAR(Fecha) = 2021 AND MONTH(Fecha) BETWEEN  
3 AND 9;
```

3. Obtener una lista de todas aquellas **Facturas** efectuadas el **primer día de cada mes de cualquier año**.

```
SELECT * FROM Facturas  
WHERE DAY(Fecha) = 1;
```

4. Obtener una lista de **todas las Facturas** contenidas en la tabla. En una **nueva columna** llamada ***DIAS TRANSCURRIDOS***, mostrar la cantidad de días transcurridos desde la fecha en que fue realizada cada factura, al día de hoy.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS'  
FROM Facturas;
```

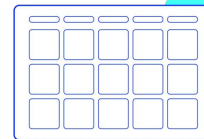
5. Modifica la consulta anterior y **agregar otra columna** con el nombre **DIA**, que refleje el **nombre del día** en el que se efectuó cada una de las facturas.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS',  
DAYNAME(Fecha) AS DIA  
FROM Facturas;
```



6. Modifica la consulta anterior y agrega otra **columna** llamada **DIA DEL AÑO**, que refleje el **número del día del año** en el que se efectuó cada una de las facturas.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS',  
DAYNAME(Fecha) AS DIA,  
DAYOFYEAR(Fecha) AS 'DIA DEL AÑO'  
FROM Facturas;
```



7. Modifica la consulta anterior y **agrega otra columna** con el nombre **MES**, que refleje el **nombre del mes del año** en el que se efectuó cada uno de los pedidos.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS',  
DAYNAME(Fecha) AS DIA,  
DAYOFYEAR(Fecha) AS 'DIA DEL AÑO', MONTHNAME(Fecha) AS 'MES'  
FROM Facturas;
```

8. Modifica la consulta anterior y **agrega otra columna** llamada ***PRIMER VENCIMIENTO*** que calcule el **primer vencimiento de cada factura**, suponiendo que dicho vencimiento es a los **30 días** de haber sido emitida.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS',  
DAYNAME(Fecha) AS DIA,  
DAYOFYEAR(Fecha) AS 'DIA DEL AÑO', MONTHNAME(Fecha) AS 'MES',  
ADDDATE(Fecha, INTERVAL 30 DAY) AS 'PRIMER VENCIMIENTO'  
FROM Facturas;
```

9. Modifica la consulta anterior y **agrega otra columna** llamada ***SEGUNDO VENCIMIENTO*** que calcule el **segundo vencimiento de cada factura**, suponiendo que el mismo es a los **2 meses** de haber sido emitida.

```
SELECT *, DATEDIFF(CURDATE(), Fecha) AS 'DIAS TRANSCURRIDOS',  
DAYNAME(Fecha) AS DIA,  
DAYOFYEAR(Fecha) AS 'DIA DEL AÑO', MONTHNAME(Fecha) AS 'MES',  
ADDDATE(Fecha, INTERVAL 30 DAY) AS 'PRIMER VENCIMIENTO',  
ADDDATE(Fecha, INTERVAL 2 MONTH) AS 'SEGUNDO VENCIMIENTO'  
FROM Facturas;
```


**¡Sigamos
trabajando!**

