

Conocimientos SQL.

1.1 Describe el funcionamiento general de la sentencia JOIN.

Se usa para obtener la información de diferentes tablas que tengan alguna relación entre sus datos. Normalmente esta relación es la clave primaria de una tabla en el dato de clave foránea de otra.

1.2 ¿Cuáles son los tipos de JOIN y cuál es el funcionamiento de los mismos?

INNER JOIN devuelve los datos que si tienen relación.

LEFT JOIN devuelve los datos de la tabla a la izquierda del join y los datos que están relacionado entre las tablas.

RIGHT JOIN devuelve los datos de la tabla a la derecha del join y los datos que están relacionado entre las tablas.

FULL JOIN devuelve los datos de la tabla a la izquierda y derecha del join así como sus datos relacionados.

1.3 ¿Cuál es el funcionamiento general de los TRIGGER y qué propósito tiene?

Se definen un conjunto de acciones a ejecutarse de manera automática antes o después de que se realice una acción de insertar, actualizar, o eliminar un registro en alguna tabla.

El propósito principal es automatizar procesos para la validación de acciones, registros de movimientos o consistencia en datos.

1.4 ¿Qué es y para qué sirve un STORED PROCEDURE?

Es un conjunto de acciones predefinidas listas para ser ejecutadas cuando se manden llamar. Algo como las funciones en los lenguajes de programación. Sirven para también automatizar tareas, reducir código duplicado, y mejoran el rendimiento pues se ejecutan más rápido que las sentencias normales.

1.5 Traer todos los productos que tengan una venta.

```
SELECT P.NOMBRE, V.CANTIDAD  
FROM PRODUCTOS P INNER JOIN VENTAS V  
ON P.IDPRODUCTO = V.IDPRODUCTO  
WHERE V.CANTIDAD = 1;
```

1.6 Traer todos los productos que tengan ventas y la cantidad total de productos vendidos.

```
SELECT P.NOMBRE, SUM(V.CANTIDAD) AS CANTIDAD  
FROM PRODUCTOS P INNER JOIN VENTAS V  
ON P.IDPRODUCTO = V.IDPRODUCTO  
GROUP BY P.NOMBRE  
ORDER BY CANTIDAD DESC;
```

1.7 Traer todos los productos (independientemente de si tienen ventas o no) y la suma total (\$) vendida por producto.

```
SELECT P.NOMBRE, IFNULL(P.PRECIO * SUM(V.CANTIDAD), 0) AS VENTA_TOTAL  
FROM PRODUCTOS P LEFT JOIN VENTAS V  
ON P.IDPRODUCTO = V.IDPRODUCTO  
GROUP BY P.NOMBRE, P.PRECIO  
ORDER BY VENTA_TOTAL DESC;
```