* 1. Describe el funcionamiento general de la sentencia **JOIN**.

Unión de tablas a través de una columna de relación

* 1. ¿Cuáles son los tipos de **JOIN** y cuál es el funcionamiento de los mismos?

INNER JOIN

Devuelve solo las filas que tienen una coincidencia en ambas tablas.

LEFT JOIN (LEFT OUTER JOIN)

Devuelve todas las filas de la tabla izquierda y las filas coincidentes de la tabla derecha

RIGHT JOIN (RIGHT OUTER JOIN)

Devuelve todas las filas de la tabla derecha y las filas coincidentes de la tabla izquierda

FULL JOIN (FULL OUTER JOIN)

Devuelve todas las filas de ambas tablas, incluyendo tanto las coincidentes como las que no

CROSS JOIN

Devuelve todas las combinaciones posibles de filas entre las dos tablas, sin tener en cuenta ninguna condición de unión

* 1. **¿**Cuál el funcionamiento general de los **TRIGGER** y qué propósito tienen?

Son disparadores que se ejecutan automáticamente al realizar alguna inserción, actualización y/o eliminación de alguna tabla y uno de sus propósitos es la automatización de procesos

* 1. ¿Qué es y para qué sirve un **STORED PROCEDURE**?

Es un procedimiento para ejecutar ciertas tareas/acciones en la base de datos, la cual nos sirve para automatizar procesos con nuestra misma base de datos.

* 1. Traer todos los productos que tengan una venta.

select

p.nombre

from [dbo].[productos] p

inner join [dbo].[ventas] v on v.idProducto = p.idProducto

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

select

p.nombre

,v.cantidad

from [dbo].[productos] p

inner join [dbo].[ventas] v on v.idProducto = p.idProducto

order by

p.nombre

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

select

p.nombre

,[cantidad] = isnull(sum(v.cantidad),0)

from [dbo].[productos] p

left join [dbo].[ventas] v on v.idProducto = p.idProducto

group by

p.nombre

order by

p.nombre