

Procesual Hito 3

Manejo de conceptos

Nombre: Aaron Alvaro Huanca Salazar

Base de Datos I - 2021

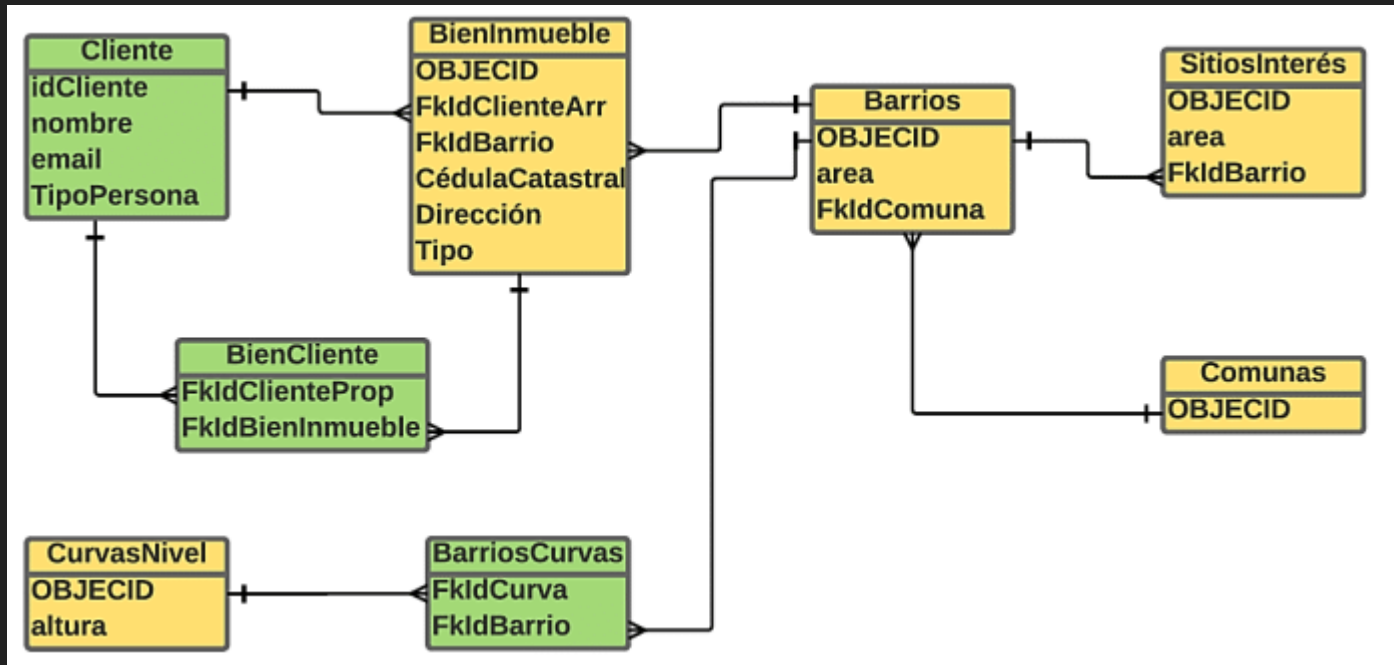
Que es el modelo entidad relación.

Un modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las "entidades", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema.



Que es el modelo lógico en bases de datos relacionales.

Se trata de una estructura formada por filas y columnas que almacena los datos referentes a una determinada entidad o relación del mundo real. Representa una propiedad que posee esa tabla.



Describe y menciona que formas(shapes) se utiliza para graficar un modelo entidad relación.

- **Relación uno a uno:** Un “individuo” de la entidad A solamente puede estar relacionado con un “individuo” de la entidad B, y ese “individuo” de la entidad B no puede estar relacionado con otros “individuos” de la entidad A. Por ejemplo, cada miembro de la entidad País se relaciona únicamente con un miembro de la entidad “Ciudad capital de un país”. Cada país puede tener una única capital y cada ciudad capital puede serlo únicamente de un país.
- **Relación uno a varios:** Un “individuo” de la entidad A puede estar relacionado con uno o varios “individuos” de la entidad B, y esos “individuos” de la entidad B no pueden estar relacionados con otros “individuos” de la entidad A. Por ejemplo, cada miembro de la entidad “Padre” puede estar relacionado con uno o varios miembros de la entidad “Hijo”, y cada miembro de la entidad “Hijo” solamente puede tener vínculo con un miembro de la entidad “Padre”.
- **Relación varios a varios:** Cada “individuo” de la entidad A puede estar relacionado con uno o varios “individuos” de la entidad B, y cada “individuo” de la entidad B puede estar relacionado con varios “individuos” de la entidad A. Por ejemplo, cada miembro de la entidad “Cliente” puede estar relacionado con uno o varios miembros de la entidad “Producto”, y cada miembro de la entidad “Producto” puede tener vínculo con varios miembros de la entidad “Cliente”.

Qué es una función de agregación.

- Las funciones de agregación realizan análisis estadísticos sobre números (y sobre fechas u horas para algunas funciones) en: varios campos de un registro, campos relacionados tanto si se muestran en un portal como si no, campos repetidos.



Muestra ejemplo del uso de 2 funciones de agregación.

Para determinar el número de proyectos activos en la organización, se aplica la función de agregación **Recuento(count)** al campo **Nombre del proyecto**.

Resultado El reporte muestra la cantidad de proyectos en su organización.

Para determinar el impacto del costo total de un proyecto se aplica la función de agregación **Suma(Sum)** al campo **Impacto del costo**.

Resultado El reporte muestra el impacto del costo total asociado con el proyecto

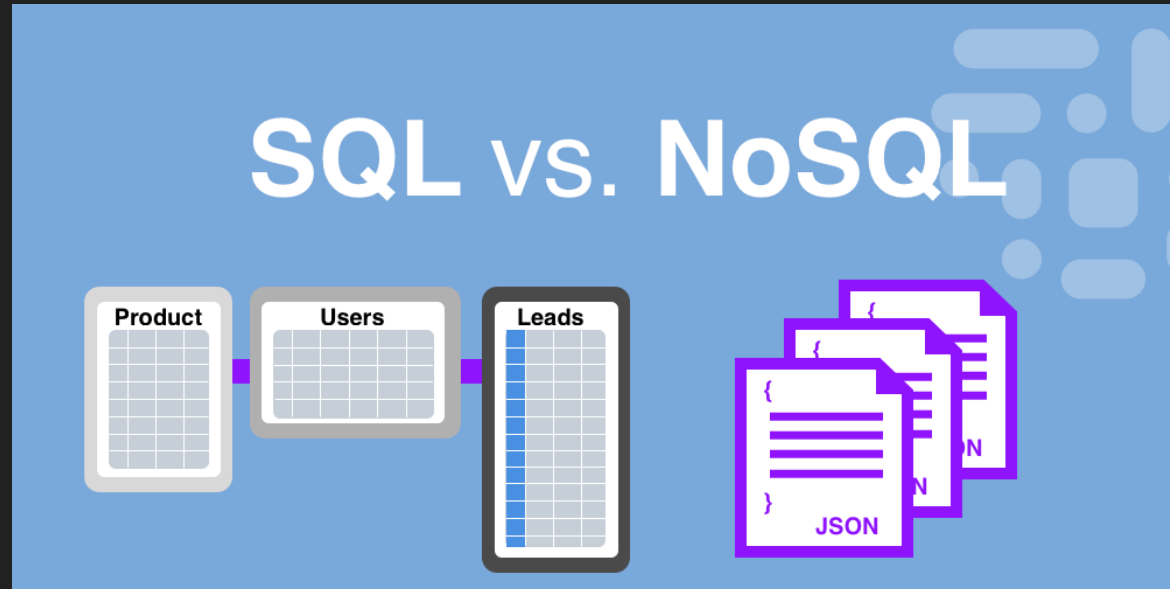
Muestre un ejemplo del uso de JOINS.

En este ejemplo, estamos uniendo las tablas Sales.SalesOrderDetail y Production.Product. Las tablas tienen el alias siguiente: SOD para Sales.SalesOrderDetail y P para Production.Product. La lógica JOIN se basa en registros coincidentes en las columnas SOD.ProductID y P.ProductID. Los registros se filtran devolviendo únicamente los registros con el SOD.UnitPrice mayor que 1000. Finalmente, el conjunto de resultados se devuelve en orden con el más caro primero según la cláusula ORDER BY y sólo los 100 productos más altos según la cláusula TOP.

```
SELECT TOP 100 P.ProductID, P.Name, P.ListPrice, P.Size,  
P.ModifiedDate, SOD.UnitPrice, SOD.UnitPriceDiscount,  
SOD.OrderQty, SOD.LineTotal FROM Sales.SalesOrderDetail SOD INNER  
JOIN Production.Product P ON SOD.ProductID = P.ProductID WHERE  
SOD.UnitPrice > 1000 ORDER BY SOD.UnitPrice DESC
```

Qué es SQL y NoSQL.

SQL permite combinar de forma eficiente diferentes tablas para extraer información relacionada, mientras **que NoSQL** no lo permite o muy limitadamente. **NoSQL** permite distribuir grandes cantidades de información; mientras **que SQL** facilita distribuir bases de datos relacionales.



Las siglas de **ISO** se corresponden con “International Organization for Standardization” o “Organización Internacional de Normalización”. Se trata de un órgano cuya principal función es la de crear normas de carácter internacional.



Quien creo el modelo entidad relación o mas conocido como E-R

Peter Chen (también **conocido** como Peter Pin-Shan Chen) actualmente se desempeña como miembro de la facultad de la Universidad Carnegie Mellon ubicada en Pittsburgh y se le atribuye el desarrollo del **modelo ER** para el diseño de bases de datos en los 70.



Crear una función que permita sumar 3 números .

```
CREATE OR ALTER FUNCTION mi_funcion_que_suma_tres_numeros(@par1
INT, @par2 INT,@par3 INT)
RETURNS INTEGER
AS
BEGIN
    --logica
    RETURN
    (@par1)+(@par2)+(@par3);
END;

SELECT dbo.mi_funcion_que_suma_tres_numeros(5,5, 10);
```

Manejo de consultas

Mostrar los productos(Nombre y stock) con stock mayor igual a 10

```
69  
70 ✓ SELECT pro.nombre,pro.stock  
71 FROM productos AS pro  
72 WHERE pro.stock >= 10;  
73
```

Output x PROCESUAL.dbo.productos x

1 row v < > | ↺ ■ + - ↻ 🔍 ↑ T

	nombre	stock
1	refrigerador	15

Mostrar el nombre del producto y la categoría de los productos pertenecen a la categoría de "electrodomesticos".

```
73
74 --Mostrar el nombre del producto y la categoría de los productos pertenecen a la categoría de electrodomesticos.
75
76 ✓ SELECT pro.nombre,pro.categoria,C.tipo
77 FROM productos AS pro
78 INNER JOIN categorias c on c.id_categoria = pro.categoria
79 WHERE pro.categoria = 1;
```

Output × Result 151 ×

< < 2 rows > > | ↺ ■ ★

	nombre	categoria	tipo
1	refrigerador	1	electrodomesticos
2	microonda	1	electrodomesticos

Que productos(nombre) tiene el pedido con id igual a = 1.

```
80
81  --Que productos (nombre) tiene el pedido con id igual a = 1
82
83  ✓ SELECT nombre AS count
84     from productos,pedido
85     WHERE id_pedido = 1;
86
```

Output × count:varchar(100) ×

3 rows

	count
1	refrigerador
2	microonda
3	los vengadores

Cuantos(count) productos tiene el pedido con id igual a = 2.

```
88
89 ✓ SELECT COUNT(id_producto) AS count
90     from productos,pedido
91     WHERE id_producto = 2;
92
```

Output × count:int ×

1 row ▾

	count
1	2

Crear una función que permita sumar 3 números.

```
93  --Crear una función que permita sumar 3 números.
94
95  CREATE OR ALTER FUNCTION mi_funcion_que_suma_tres_numeros(@par1 INT, @par2 INT, @par3 INT)
96  RETURNS INTEGER
97  AS
98  BEGIN
99      --logica
100     RETURN
101     (@par1)+(@par2)+(@par3);
102 END;
103
104 ✓ SELECT dbo.mi_funcion_que_suma_tres_numeros(5,5, 10);
105
```

Output × dbo.mi_funcion_que_s..._numeros(5,5, 10):int ×

1 row ▾ ↺ ■ ★

<anonymous> ▾

1	20
---	----

Crear una función que permita restar 3 números

```
106  --Crear una función que permita restar 3 números.
107
108  CREATE OR ALTER FUNCTION mi_funcion_que_reste_tres_numeros(@par1 INT, @par2 INT,@par3 INT)
109  RETURNS INTEGER
110  AS
111  BEGIN
112      --logica
113      RETURN
114      (@par1)-(@par2)-(@par3);
115  END;
116
117  ✓ SELECT dbo.mi_funcion_que_reste_tres_numeros(5,5, 10);
118
```

Output ×

dbo.mi_funcion_que_r..._numeros(5,5, 10):int ×

1 row ▾

↺ ↻ ⏏

<anonymous> ▾

1

-10

Cómo unificaría en una sola función el ejercicio 3.5 y 3.7 (los dos anteriores).

```
120
121 CREATE OR ALTER FUNCTION funcion_para_suma_y Resta_de_tres_numeros (@operacion VARCHAR(10),@par1 INT, @par2 INT,@par3 INT)
122 RETURNS INT
123 AS
124 BEGIN
125     DECLARE @respuesta INT
126     IF (@operacion) = 'suma'
127     BEGIN
128         SET @respuesta = (@par1)+(@par2)+(@par3);
129     END
130     ELSE IF (@operacion) = 'resta'
131     BEGIN
132         SET @respuesta = (@par1)-(@par2)-(@par3);
133     END
134     RETURN @respuesta;
135 END;
136
137 ✓ SELECT dbo.funcion_para_suma_y Resta_de_tres_numeros('resta',5,5,10);
```

Output ×

1 row ▾

<anonymous> ▾

1	-10
---	-----