



BASE DE DATOS I

# MANEJO DEL MODELO ENTIDAD RELACIÓN-ER

ESTUDIANTE: Univ. Karen Alisson Chuquimia Mamani

# SECCIÓN: MANEJO DE CONCEPTOS.

## 1. ¿Qué son las bases de datos?

Es un almacén sobre personas, productos, pedidos u otras cosas y nos permite guardar grandes cantidades de información de forma organizada. También es una herramienta para recopilar y organizar información.

## 2. ¿A que se refiere cuando se habla de bases de datos relacionales?

Es una recopilación de elementos de datos con relaciones predefinidas entre ellos. Estos elementos se organizan como un conjunto de tablas con columnas (campos) y filas (registros). Las tablas se utilizan para guardar información sobre los objetos que se van a representar en la base de datos.

## 3. ¿Qué es el modelo entidad relación y/o diagrama entidad relación?

Un diagrama entidad-relación, también conocido como modelo entidad relación o ERD, es un tipo de diagrama de flujo que ilustra cómo las "entidades".

## SECCIÓN: MANEJO DE CONCEPTOS.

4. ¿Cuáles son las figuras que representan a un diagrama entidad relación? Explique cada una de ellas.

El modelo entidad relación tiene tres elementos principales:

- **Entidades:** El modelo contará con una entidad por cada uno de los componentes del proceso de negocio. Así, en un negocio de venta de suscripciones a revistas, podemos tener entidades “Cliente”, “Dirección”, “Factura”, “Producto”, o “Incidencias”, entre otras.
- **Atributos:** Componente fundamental de cada modelo entidad-relación, nos permiten describir las propiedades que tiene cada entidad. “Nombre”, “Primer Apellido”, “Segundo Apellido”, “Fecha de nacimiento”, “Género” o “Segmento de valor” serán atributos de la entidad “Cliente”.
- **Relaciones:** Se establecen vínculos entre parejas de entidades. Cada “Cliente” tendrá una “Dirección” de envío en la que recibirá la suscripción, podrá estar suscrito a uno o varios “Productos”, y recibirá una “Factura” con la periodicidad acordada.

## SECCIÓN: MANEJO DE CONCEPTOS.

### 5. ¿Qué es SQL Server y qué es SQL Server Management Studio?

Microsoft SQL Server es uno de los principales sistemas de gestión de bases de datos relacional del mercado que presta servicio a un amplio abanico de aplicaciones de software destinadas a la inteligencia empresarial y análisis sobre entornos corporativos.

SQL Server Management Studio (SSMS) es la interfaz de usuario cliente preferida y oficial con la cual se puede manejar, configurar, desplegar, actualizar y administrar una instancia SQL Server

### 6. ¿Cómo se crea una base de datos?

Se introduce `CREATE DATABASE` nombre;

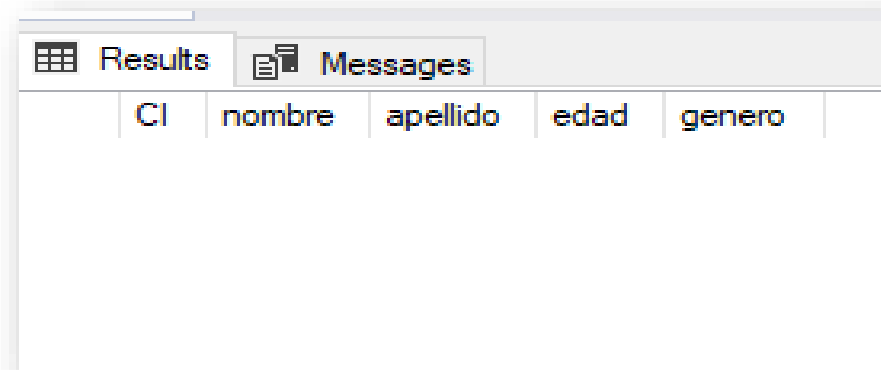
# SECCIÓN: MANEJO DE CONCEPTOS.

7. ¿Para qué sirve el comando USE?

El USE nos permite conectar con la creación de la base de datos.

8. Crear una tabla cualquiera con 3 columnas y su primary key.

```
CREATE TABLE PERSONA(  
    CI varchar(50) PRIMARY KEY,  
    nombre varchar(50),  
    apellido varchar(50),  
    edad integer,  
    genero varchar(50)  
);  
  
SELECT * FROM persona;
```



The screenshot shows a database interface with two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with five columns: 'CI', 'nombre', 'apellido', 'edad', and 'genero'. The table is currently empty.

CI	nombre	apellido	edad	genero
----	--------	----------	------	--------

# SECCIÓN: MANEJO DE CONCEPTOS.

9. Insertar 3 registros a la tabla creada anteriormente.

```
SELECT * FROM persona;  
  
INSERT INTO persona (CI,nombre,apellido,edad,genero)  
VALUES ('12345601','Nombre1','Apellido1',10,'genero1');  
  
INSERT INTO persona (CI,nombre,apellido,edad,genero)  
VALUES ('12345602','Nombre2','Apellido2',1,'genero2');  
  
INSERT INTO persona (CI,nombre,apellido,edad,genero)  
VALUES ('12345603','Nombre3','Apellido3',11,'genero3');
```

Results		Messages			
	CI	nombre	apellido	edad	genero
1	12345601	Nombre1	Apellido1	10	genero1
2	12345602	Nombre2	Apellido2	1	genero2
3	12345603	Nombre3	Apellido3	11	genero3

10.¿Cómo se elimina una tabla?

Se ingresa **DROP TABLE** nombre;





# SECCIÓN: PARTE PRÁCTICA

---

# SECCIÓN: PARTE PRÁCTICA

I. Crear el diseño para una UNIVERSIDAD.

- Analizar qué cosas debería de tener como atributos una universidad.
- Adjuntar la imagen que resuelve el problema.

UNIVERSIDAD

Nombre VARCHAR(50) PRIMARY KEY

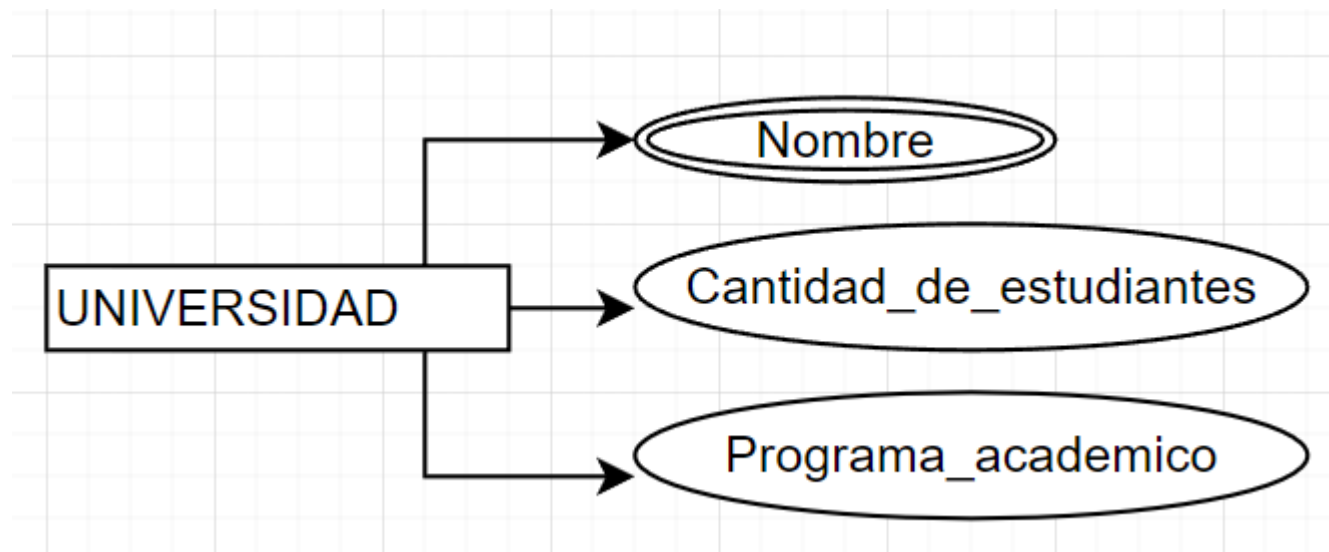
Cantidad\_de\_estudiantes INTEGER

Programa\_académico VARCHAR(50)



## SECCIÓN: PARTE PRÁCTICA

2. Crear el diagrama Entidad Relación E-R para el ejercicio anterior.
- Adjuntar la imagen del diagrama generado.

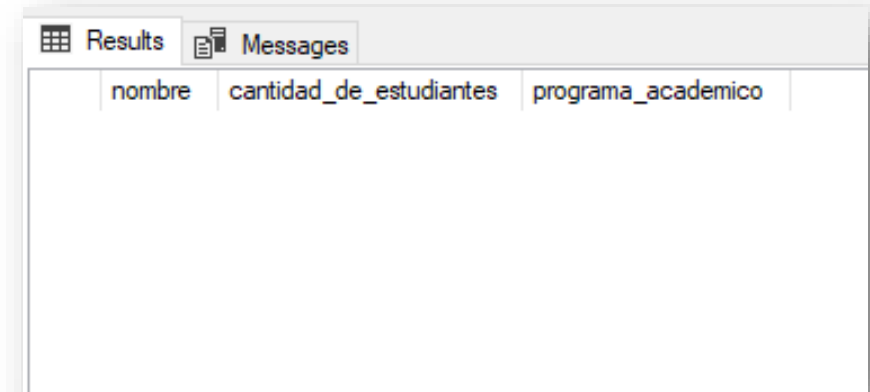


## SECCIÓN: PARTE PRÁCTICA

3. Crear la tabla universidad en base al diseño anterior.

- Crear una base de datos de nombre **Hito2Tarea**
- Crear la tabla **universidad** en la base de datos creada.
- Adjuntar la consulta SQL (imagen).

```
CREATE DATABASE Hito2Tarea;  
USE Hito2Tarea;  
CREATE TABLE universidad(  
    nombre VARCHAR (30) PRIMARY KEY,  
    cantidad_de_estudiantes INTEGER,  
    programa_academico VARCHAR (30),  
);  
SELECT * FROM universidad;
```



The screenshot shows a database interface with two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying the structure of a table with three columns: 'nombre', 'cantidad\_de\_estudiantes', and 'programa\_academico'.

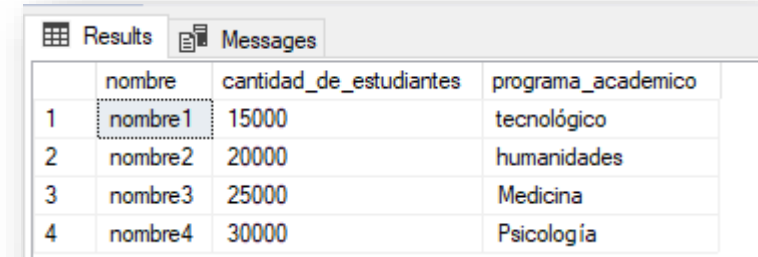
nombre	cantidad_de_estudiantes	programa_academico
--------	-------------------------	--------------------

# SECCIÓN: PARTE PRÁCTICA

4. Agregar registros a la tabla creada anteriormente.

- Agregar 4 registros a la tabla creada.
- Adjuntar la consulta SQL (imagen)

```
SELECT * FROM universidad;  
  
INSERT INTO universidad (nombre,cantidad_de_estudiantes,programa_academico)  
VALUES ('nombre1',15000,'tecnológico');  
  
INSERT INTO universidad(nombre,cantidad_de_estudiantes,programa_academico)  
VALUES ('nombre2',20000,'humanidades');  
  
INSERT INTO universidad (nombre,cantidad_de_estudiantes,programa_academico)  
VALUES ('nombre3',25000,'Medicina');  
  
INSERT INTO universidad(nombre,cantidad_de_estudiantes,programa_academico)  
VALUES ('nombre4',30000,'Psicología');
```



The screenshot shows a database interface with two tabs: 'Results' and 'Messages'. The 'Results' tab is active, displaying a table with the following data:

	nombre	cantidad_de_estudiantes	programa_academico
1	nombre1	15000	tecnológico
2	nombre2	20000	humanidades
3	nombre3	25000	Medicina
4	nombre4	30000	Psicología

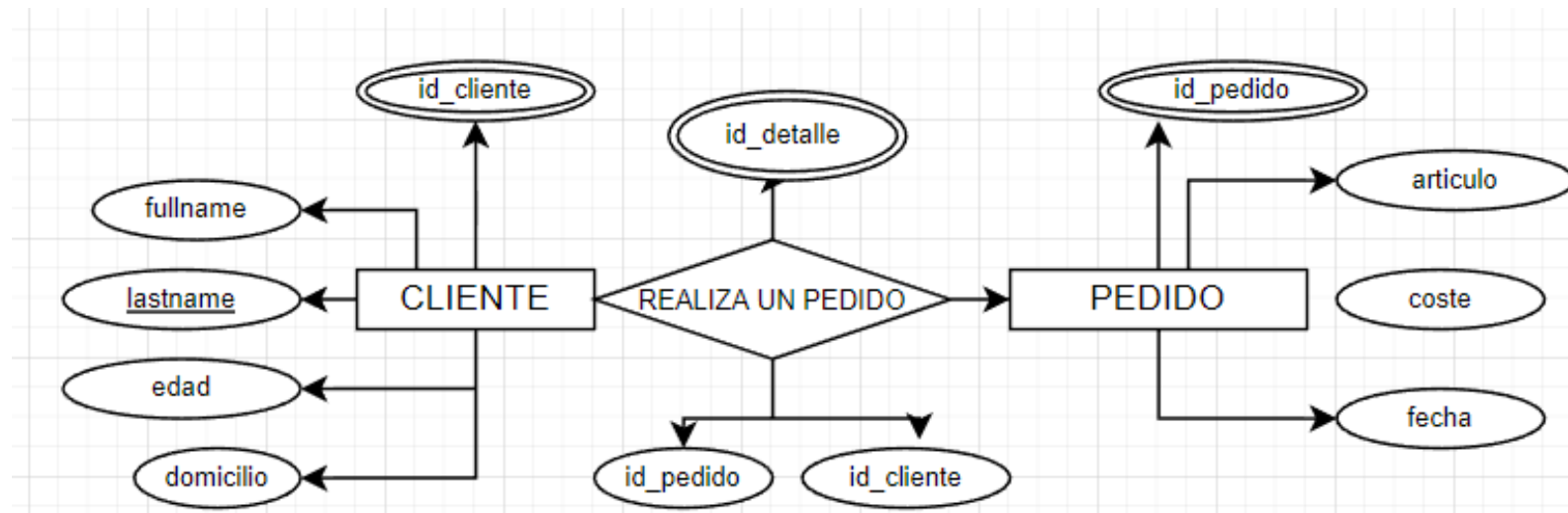
## SECCIÓN: PARTE PRÁCTICA

15. Crear las tablas y 2 registros para cada tabla para el siguiente modelo ER.

- Diseño.
- Se **sugiere** crear una base de datos de nombre **POLLOS\_COPA** y en ella crear las tablas:

- **cliente**
- **detalle\_pedido**
- **pedido**

- Adjuntar el código SQL generado.



```
CREATE DATABASE POLLOS_COPA;
USE POLLOS_COPA;
```

```
CREATE TABLE cliente(
id_cliente VARCHAR(30) PRIMARY KEY,
fullname VARCHAR(30),
lastname VARCHAR(30),
edad INTEGER,
domicilio VARCHAR(15)
);
INSERT INTO cliente (id_cliente,fullname,lastname,edad,domicilio)
VALUES ('ID_01', 'fullname1', 'lastname1', 15, 'El Alto');
INSERT INTO cliente (id_cliente,fullname,lastname,edad,domicilio)
VALUES ('ID_02', 'fullname2', 'lastname2', 16, 'El Alto0');
```

```
CREATE TABLE detalle_pedido(
id_detalle_pedido VARCHAR(30) PRIMARY KEY,
id_cliente VARCHAR(30),
id_pedido VARCHAR(30),
);
```

```
INSERT INTO
detalle_pedido(id_detalle_pedido,id_cliente,id_pedido)
VALUES ('ID_00000','ID_00','ID_001');
```

```
INSERT INTO
detalle_pedido(id_detalle_pedido,id_cliente,id_pedido)
VALUES ('ID_00001','ID_01','ID_010');
```

```
CREATE TABLE pedido(
id_pedido VARCHAR(30) PRIMARY KEY,
articulo VARCHAR(50),
costo INTEGER,
fecha INTEGER,
);
```

```
INSERT INTO pedido(id_pedido,articulo,costo,fecha)
VALUES ('ID_011','Art.1',100,01-02-2000);
INSERT INTO pedido(id_pedido,articulo,costo,fecha)
VALUES ('ID_012','Art.2',200,02-03-2001);
```

```
SELECT *FROM pedido;
SELECT *FROM detalle_pedido;
SELECT *FROM cliente;
```



Results

Messages

	id_pedido	articulo	costo	fecha
1	ID_011	Art.1	100	-2001
2	ID_012	Art.2	200	-2002

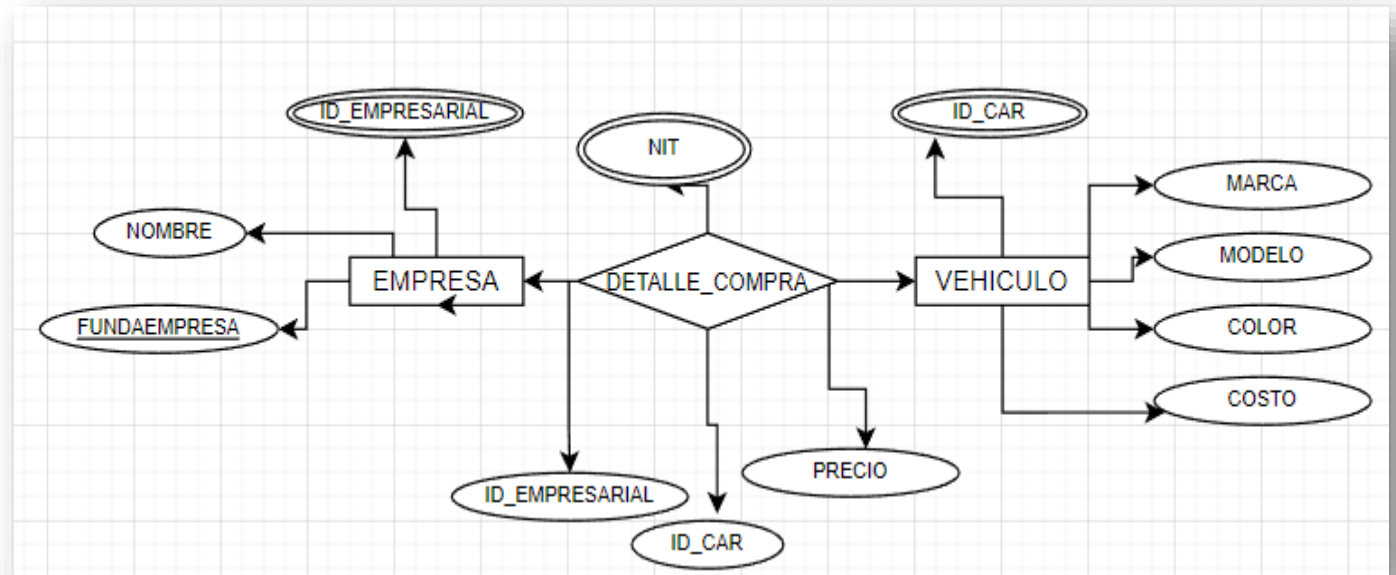
	id_detalle_pedido	id_cliente	id_pedido
1	ID_00000	ID_00	ID_001
2	ID_00001	ID_01	ID_010

	id_cliente	fullname	lastname	edad	domicilio
1	ID_01	fullname1	lastname1	15	El Alto
2	ID_02	fullname2	lastname2	16	El Alto0

## SECCIÓN: PARTE PRÁCTICA

6. Crear el modelo entidad relación ER y su código SQL.

- El contexto de análisis es:
  - Una empresa compra vehículos.
- Adjuntar el diagrama Entidad Relación **ER** (imagen)
- Adjuntar el código SQL generado.
- **Sugerencia:** Podría crear las entidades
  - empresa
  - detalle\_compra
  - vehículos



```
CREATE DATABASE Hito01;  
USE Hito01;
```

```
CREATE TABLE EMPRESA(  
ID_EMPRESARIAL VARCHAR(30) PRIMARY KEY,  
nombre varchar (20),  
FundaEmpresa varchar(20)  
);
```

```
CREATE TABLE VEHICULO(  
IDE_CAR varchar(100) PRIMARY KEY,  
modelo varchar(100),  
color varchar(100),  
marca varchar(100),  
costo integer  
);
```

```
CREATE TABLE DET_COMPRA(  
NIT integer PRIMARY KEY,  
ID_EMPRESARIAL varchar (30),  
IDE_CAR varchar(100),  
precio integer,  
FOREIGN KEY (ID_EMPRESARIAL) REFERENCES  
EMPRESA(ID_EMPRESARIAL),  
FOREIGN KEY (IDE_CAR) REFERENCES  
VEHICULO(IDE_CAR)  
);
```

```
SELECT * FROM EMPRESA;  
SELECT * FROM VEHICULO;  
SELECT * FROM DET_COMPRA;
```



**¡GRACIAS  
POR SU  
ATENCION!**

12 DE SEPTIEMBRE DEL 2022