



Tarea Hito 2

Base de Datos I

Unifranz

Sede el Alto

Hito2



¿Qué son las bases de datos?

Una base de datos es una recopilación organizada de información o datos estructurados, que normalmente se almacena de forma electrónica en un sistema informático. Normalmente, una base de datos está controlada por un Sistema de gestión de Base de datos (DBMS). En conjunto, los datos y el DBMS, junto con las aplicaciones asociadas a ellos, reciben el nombre de sistema de bases de datos, abreviado normalmente a simplemente base de datos.

Los datos de los tipos más comunes de bases de datos en funcionamiento actualmente se suelen utilizar como estructuras de filas y columnas en una serie de tablas para aumentar la eficacia del procesamiento y la consulta de datos. Así, se puede acceder, gestionar, modificar, actualizar, controlar y organizar fácilmente los datos. La mayoría de las bases de datos utilizan un lenguaje de consulta estructurada (SQL) para escribir y consultar datos.



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

- Una base de datos relacional es un tipo de base de datos que cumple con el modelo relacional. Así, según esta definición de base de datos relacional, se trata de una base de datos que almacena y da acceso a puntos de datos relacionados entre sí. El modelo relacional es una forma intuitiva y directa de representar datos sin necesidad de jerarquizarlos.
- Una base de datos relacional es, en esencia, un conjunto de tablas (o relaciones) formadas por filas (registros) y columnas (campos); así, cada registro (cada fila) tiene una ID única, denominada clave y las columnas de la tabla contienen los atributos de los datos. Cada registro tiene normalmente un valor para cada atributo, lo que simplifica la creación de relaciones entre los puntos de datos.
- De tal manera que una de las principales características de la base de datos relacional es que evitar la duplicidad de registros y a su vez garantizar la integridad referencial, es decir, que si se elimina uno de los registros, la integridad de los registros restantes no será afectada. Además, gracias a las claves se puede acceder de forma sencilla a la información y recuperarla en cualquier momento.

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

El modelo entidad relación es una herramienta que permite representar de manera simplificada los componentes que participan en un proceso de negocio y el modo en el que estos se relacionan entre sí.

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:** Los 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:** Con las 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.

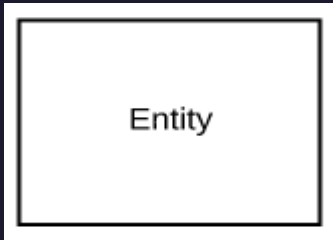
El diagrama entidad relación es la expresión gráfica del modelo entidad relación. En él las entidades se representan utilizando rectángulos, los atributos por medio de círculos o elipses y las relaciones como líneas que conectan las entidades que tienen algún tipo de vínculo. También es muy común el formato de diagrama en el que los atributos de una entidad aparecen listados en filas dentro del rectángulo que representa a esa entidad.

Además, es común que, en el modelo entidad-relación, los conectores que indican que dos entidades A y B están relacionadas entre sí tengan una apariencia gráfica diferente dependiendo del tipo de relación que exista entre ellas.

Los tipos de relaciones posibles entre dos entidades en un modelo entidad relación son:

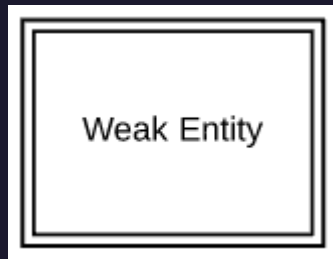
- **Relación uno a uno:** Un “individuo” de la entidad A solamente puede estar relacionado con un “individuo” de la entidad B, y ese “individuo” de la entidad B no puede estar relacionado con otros “individuos” de la entidad A. Por ejemplo, cada miembro de la entidad País se relaciona únicamente con 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.

¿Cuáles son las figuras que representan a un diagrama entidad relación?



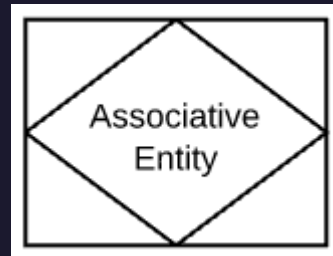
Entidad fuerte

Estas figuras son independientes de otras entidades y con frecuencia se les denomina entidades matriz ya que a menudo tienen entidades débiles que dependen de ellas. También tendrán una clave primaria, que distinga a cada suceso de la entidad.



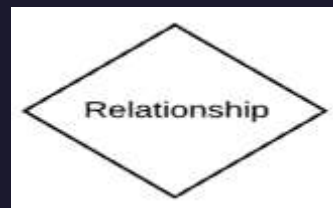
Entidad débil

Las entidades débiles dependen de algún otro tipo de entidad. No tienen claves primarias y no tienen significado en el diagrama sin su entidad matriz.



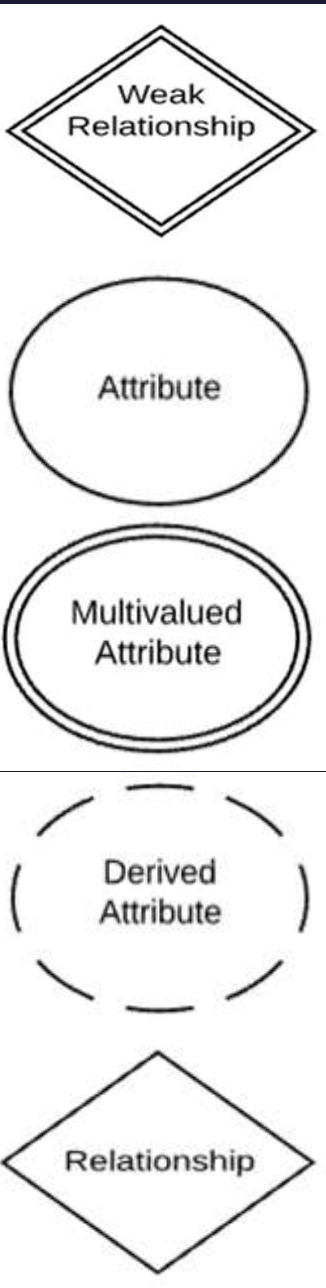
Entidad asociativa

Las entidades asociativas relacionan las instancias de varios tipos de entidades. También contienen atributos que son específicos a la relación entre esas instancias de entidades.



Relación

Las relaciones son asociaciones entre dos o más entidades.



Relación débil

Las relaciones débiles son conexiones entre una entidad débil y su propietario.

Atributo

Los atributos son las características de una entidad, una relación de muchos a muchos, o una relación de uno a uno.

Atributo de varios valores

Los atributos de valores múltiples son aquellos que pueden tomar más de un valor.

Atributo derivado

Los atributos derivados son atributos cuyos valores se pueden calcular a partir de valores de atributos relacionados.

Relación

Las relaciones son asociaciones entre dos o más entidades.

¿Qué es SQL Server y qué es SQL Server Management Studio?

SQL Server es un sistema de gestión de bases de datos relacionales, o RDBMS, desarrollado y comercializado por Microsoft.

Al igual que otro software RDBMS, SQL Server está construido sobre SQL, un lenguaje de programación estándar para interactuar con las bases de datos relacionales. El servidor SQL está vinculado a Transact-SQL, o T-SQL.

Durante más de 20 años SQL Server funcionó exclusivamente en el entorno Windows. En 2016, Microsoft lanzó la versión para instalar Linux, a partir de la versión SQL Server 2017 que estuvo disponible en octubre de 2016 ya se puede ejecutar tanto en Windows como en Linux.

SQL Server Management Studio (SSMS para abreviar) es un entorno de desarrollo integrado para administrar cualquier infraestructura SQL. Se utiliza para acceder, administrar, configurar y desarrollar todos los componentes de [SQL Server](#) y SQL Database. Microsoft lo ha optimizado a lo largo de los años y es un programa de administración de servidores y bases de datos muy popular.



¿Cómo se crea una base de datos?

Lo más normal es que tu hosting te ofrezca la posibilidad de crear las bases de datos que necesites, aunque hay hostings que ponen una limitación en el número de bases de datos que puedes crear.

Para que no te veas limitado en ese aspecto (y en otros) es conveniente contratar un buen hosting que te permita crear tantas como necesites, como es el caso de estos dos.

Para crear una base de datos, tienes que acceder al panel de tu hosting y buscar la sección “Bases de datos”.

Haz clic en “MySQL Bases de datos”.

Se abre una ventana en la que puedes crear o eliminar las bases de datos que necesites, así como tantos usuarios como necesites también.

Cada base de datos necesita tener un usuario con privilegios de administrador para poder acceder a ella, modificarla o eliminarla.

¿Para qué sirve el comando USE?

- El comando USE le indica al MySQL que use la base de datos nombreBD como la base de datos por defecto para los comandos que se utilicen a continuación. La base de datos nombreBD continuará siendo la base de datos por defecto hasta el final de la sesión o hasta que se ejecute de nuevo el USE con el nombre de otra base de datos.
- Sintaxis:
- Mysql> USE nomBD;
- Ejemplo:
- mysql> USE mysql;



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

```
create database universidad_unifranz2  
  
use universidad_Unifranz2;  
  
create table Estudiante3(  
    cod_est varchar(20) primary key,  
    nombre varchar(20),  
    apellido varchar(20)  
);
```

¿Cómo se elimina una tabla?

```
);  
  
insert  
Estudiante3(cod_est,nombre,apellido)  
values('SIS-13372398','CARLOS','FLORES PAUCARA');  
  
select*  
from Estudiante3;  
drop table Estudiante3;
```

Insertar 3 registros a la tabla creada

```
insert  
Estudiante3(cod_est,nombre,apellido)  
values('SIS-13372398','CARLOS','FLORES PAUCARA');  
  
select*  
from Estudiante3;  
  
create table materia(  
    cod_mat varchar(20) primary key,  
    nombre varchar (20),  
    aula integer  
);
```

00 %

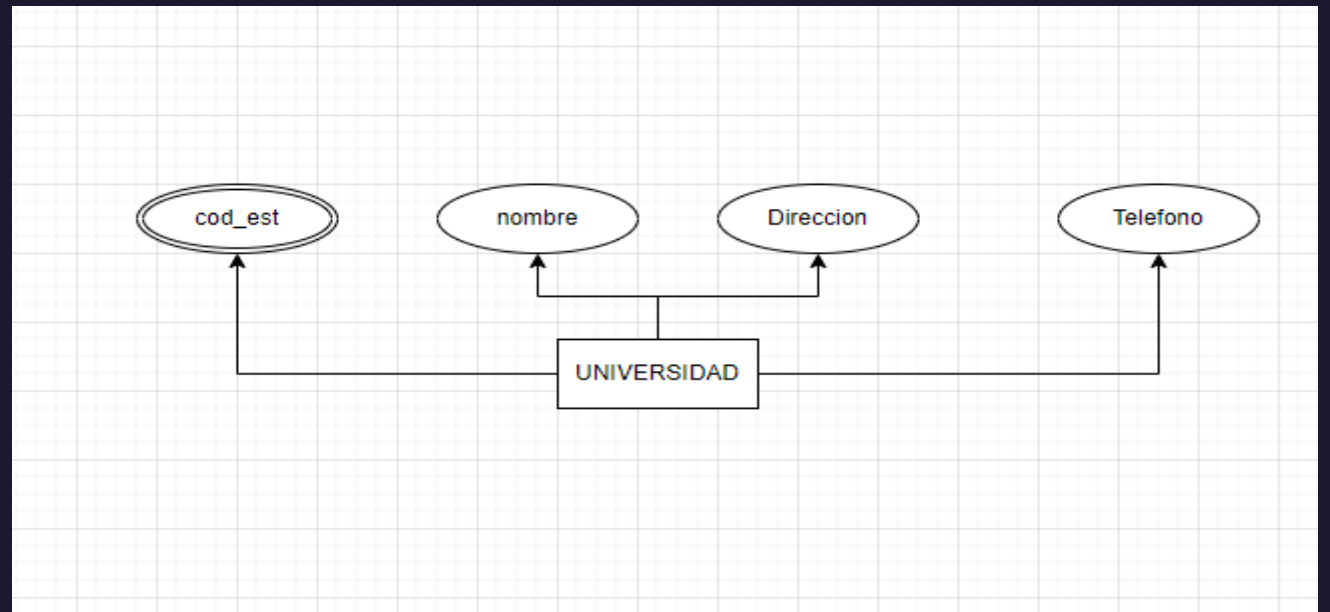
Results Messages

	cod_est	nombre	apellido
1	SIS-13372398	CARLOS	FLORES PAUCARA

Crear el diseño para una UNIVERSIDAD.

```
create table universidad(  
  cod_est varchar(20) primary key,  
  nombre varchar(20),  
  direccion varchar(23),  
  telefono varchar(23)
```

Crear el diagrama Entidad Relación E-R para el ejercicio anterior.



Crear la tabla universidad en base al diseño anterior.

```
create database universidad_unifranz0
use universidad_Unifranz0;
create table universidad(
    cod_est varchar(20) primary key,
    nombre varchar(20),
    direccion varchar(23),
    telefono varchar(23)
)
```

Agregar registros a la tabla creada anteriormente.

```
insert into
universidad(cod_est,nombre,direccion,telefono)
values ('SIS-13372398','Carlos','Achocalla','63060578');
insert into
universidad(cod_est,nombre,direccion,telefono)
values ('ICO-1527363','Kevin','Villa_Bolivar-D','76347343');
SELECT*
FROM Universidad;
```

100 %

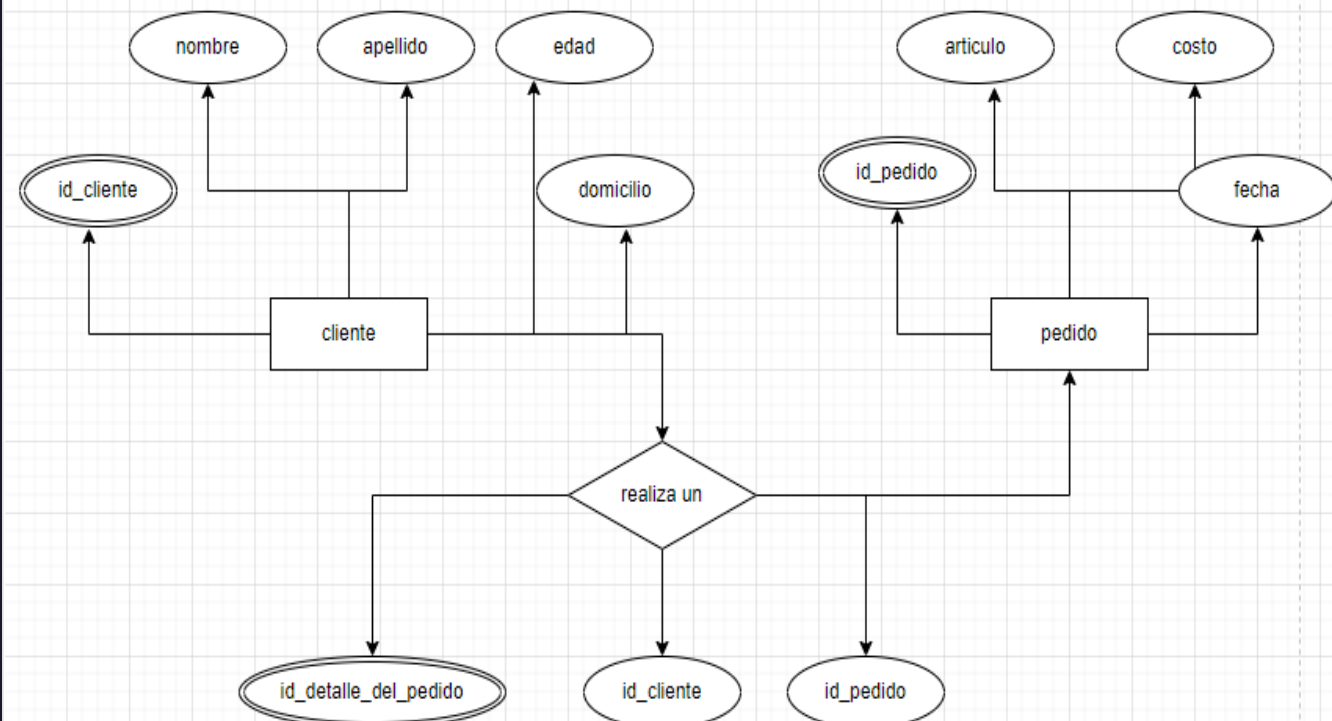
Results Messages

	cod_est	nombre	direccion	telefono
1	ICO-1527363	Kevin	Villa_Bolivar-D	76347343
2	SIS-13372398	Carlos	Achocalla	63060578

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

```
create database Pollos_Cop
use Pollos_Cop;
--Creando tabla de datos del cliente
create table cliente(
    id_cliente integer primary key,
    nombre varchar (20),
    apellido varchar (20),
    edad integer,
    domicillio varchar (20),
);
--Creando tabla de datos de realiza un
create table realiza_un(
    id_detalle_pedido integer primary key,
    id_cliente integer ,
    id_pedido integer,
);
--Creando tabla de datos de pedido
create table pedido1(
    id_pedido integer primary key,
    articulo varchar (20),
    costo float,
    fecha integer,
);
insert into cliente (id_cliente, Nombre, Apellido, Edad, domicillio)
values (13372398, 'Luis Mateo', 'Rodrigues', 20, 'tilata')
insert into cliente (id_cliente, Nombre, Apellido, Edad, domicillio)
values (4514140, 'Enrique Luis', 'Lopez', 19, 'viacha')
select*
```

```
insert into realiza_un(id_detalle_pedido,id_cliente,id_pedido)
values (2323365,13372398,2235623556)
insert into realiza_un(id_detalle_pedido,id_cliente,id_pedido)
values (2225366,13872353,2223247834)
select*
from realiza_un
insert into pedido1(id_pedido,articulo, costo, fecha)
values (2325433, 'pollo_broaster', 30, 23/05/12)
insert into pedido1(id_pedido,articulo, costo, fecha)
values (54545354, 'valde_pollo', 120, 22/08/18)
select*
from pedido1
```



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

```
create database compra_de_vehiculos1;
use compra_de_vehiculos1;

--Creando tabla de datos del comprador
create table comprador (

CI varchar (30) primary key,
Nombre varchar (50),
Apellido varchar (30),
Edad integer

);

--Creando tabla de datos del vehiculo
create table Vehiculo (

Placa varchar (10) Primary key,
Color varchar (20),
Marca varchar (30),
Año integer

);

--Creando tabla de datos de la distribuidora
create table Distribuidora(

NIT varchar (10) Primary key,
Nombre varchar (20),
Ubicacion varchar (30),

);
```

```
--Creando tabla de datos de la entidad relacion compra
create table Compra (
Numero_de_compra integer primary key,
CI varchar (30),
Placa varchar (10),
NIT varchar (10),
foreign key (CI) references Comprador (CI),
foreign key (Placa) references Vehiculo (Placa),
foreign key (NIT) references Distribuidora (NIT),

);

--Añadiendo datos na la tabla comprador
insert into comprador (CI, Nombre, Apellido,Edad)
values ('7014140lp', 'Luis Mateo','Rodriguez',20)
insert into comprador (CI, Nombre, Apellido,Edad)
values ('4514140cbb', 'Enrique Luis','Lopez',19)
insert into comprador (CI, Nombre, Apellido,Edad)
values ('7014175or', 'Jose','Ubaldo',31)
insert into comprador (CI, Nombre, Apellido,Edad)
values ('7014870lp', 'Cristina','Casas',25)
insert into comprador (CI, Nombre, Apellido,Edad)
values ('2414140scz', 'Angel','Casas',20)

--Ver datos de la tabla comprador
select *
from comprador

--Añadiendo datos na la tabla comprador
insert into Vehiculo (Placa,Color,Marca,Año)
values('7894PJB','Negro','Suzuki',2021)
```

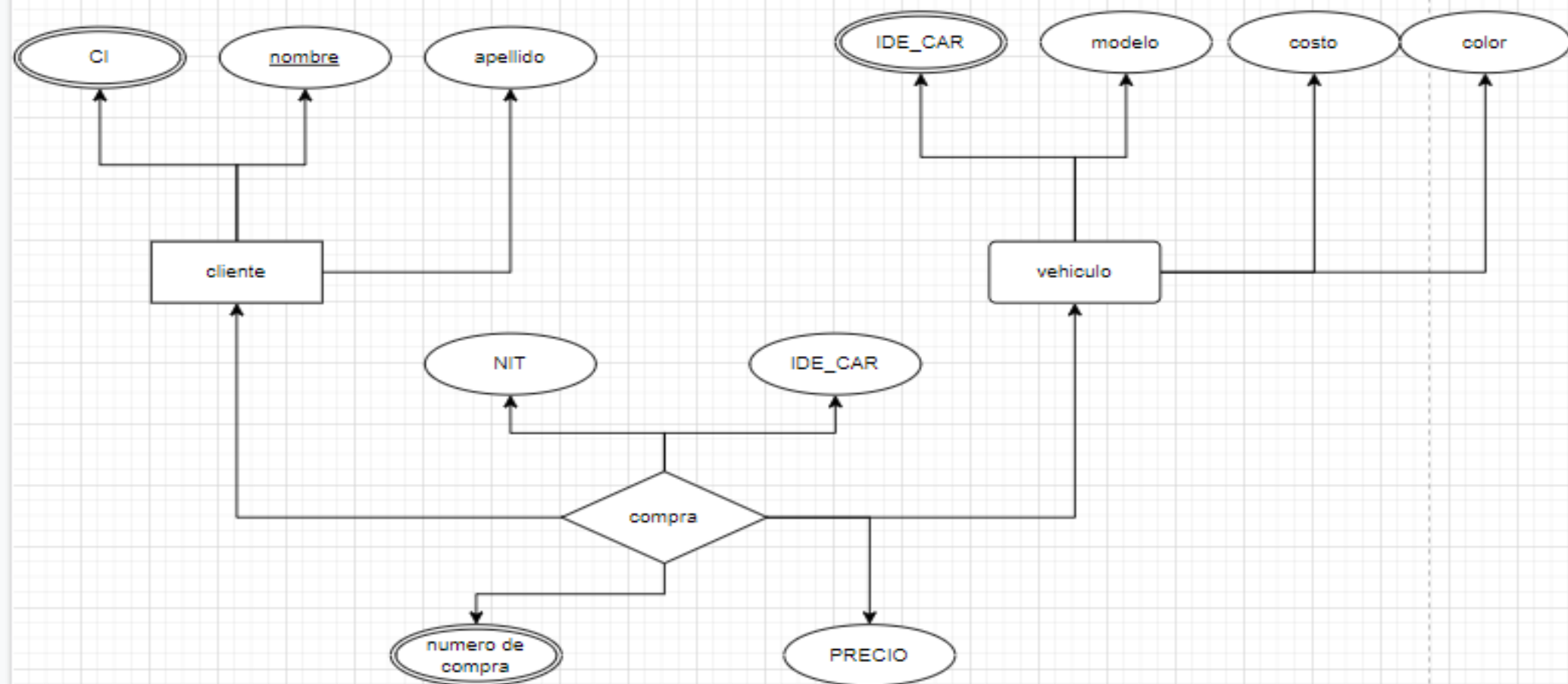
```
--Añadiendo datos na la tabla comprador
insert into Vehiculo (Placa,Color,Marca,Año)
values('7894PJB','Negro','Suzuki',2021)
insert into Vehiculo (Placa,Color,Marca,Año)
values('78750PB','Azul','Nissan',2015)
insert into Vehiculo (Placa,Color,Marca,Año)
values('784242JB','Blanco','Toyota',2020)
insert into Vehiculo (Placa,Color,Marca,Año)
values('756234PRTB','Rojo','Porsche',2018)
insert into Vehiculo (Placa,Color,Marca,Año)
values('1JB','Salmon','Chevrolet',2022)

--Ver datos de la tabla vehiculo
select *
from Vehiculo

--Añadiendo datos de la tabla distribuidora
insert into Distribuidora(NIT,Nombre,Ubicacion)
values ('7981325PL','Patito','La Paz')

--Ver datos de la tabla distribuidora
select*
from Distribuidora

--Ver datos tabla entidad relacion compra
select *
from Compra
```



Gracias

Atte. Carlos Daniel Flores Paucara

SISI3372398

BASE DE DATOS I

