

EVALUACION PROCESUAL HITO 2

BASE DE DATOS I

PRESENTA :
WILLIAM GARRI
ESCALANTE JIHUAÑA

SIS 6903691

EL ALTO, SEPTIEMBRE DE 2022



Microsoft®
SQL Server®



INDICE

I. PARTE TEORICA

- a) ¿Qué son las bases de datos?
- b) ¿A que se refiere cuando se habla de bases de datos relacionales?
- c) ¿Qué es el modelo entidad relación y/o diagrama entidad relación?
- d) ¿Cuáles son las figuras que representan a un diagrama entidad relación?
- e) ¿Qué es SQL Server y qué es SQL Server Management Studio?
- f) ¿Cómo se crea una base de datos?
- g) ¿Para qué sirve el comando USE?
- h) Crear una tabla cualquiera con 3 columnas y su primary key.
- i) Insertar 3 registros a la tabla creada anteriormente.
- j) ¿Cómo se elimina una tabla?

II. PARTE PRACTICA

- a) Crear el diseño para una UNIVERSIDAD
- b) Crear el diagrama Entidad Relación E-R para el ejercicio anterior.
- c) Crear la tabla universidad en base al diseño anterior.
- d) Agregar registros a la tabla creada anteriormente.
- e) Crear las tablas y 2 registros para cada tabla para el siguiente modelo ER.
- f) Crear el modelo entidad relación ER y su código SQL.

I. PARTE TEORICA



a).1

¿QUÉ SON LAS BASES DE DATOS?

- Se llama base de datos, o también banco de datos, a un conjunto de información perteneciente a un mismo contexto, ordenada de modo sistemático para su posterior recuperación, análisis y/o transmisión.
- Las bases de datos son el producto de la necesidad humana de almacenar la información, es decir, de preservarla contra el tiempo y el deterioro, para poder acudir a ella posteriormente.



a).2

TIPOS DE BASES DE DATOS

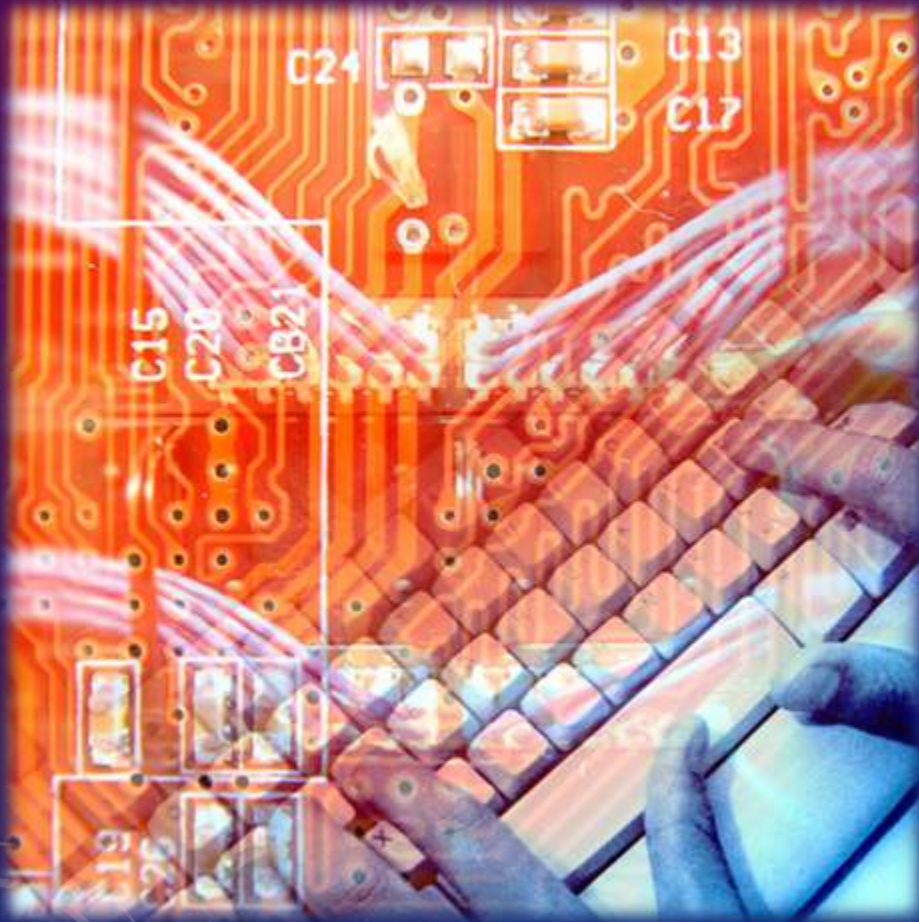


Según su variabilidad. Conforme a los procesos de recuperación y preservación de los datos, podemos hablar de:

- **Bases de datos estáticas.** Típicas de la inteligencia empresarial y otras áreas de análisis histórico, son bases de datos de sólo lectura, de las cuales se puede extraer información, pero no modificar la ya existente.
- **Bases de datos dinámicas.** Aparte de las operaciones básicas de consulta, estas bases de datos manejan procesos de actualización, reorganización, añadidura y borrado de información.

a).3

TIPOS DE BASES DE DATOS



Según su contenido. De acuerdo a la naturaleza de la información contenida, pueden ser:

- **Bibliográficas.** ordenado a partir de información clave como son los datos del autor, del editor, etc.
- **De texto completo.** Se manejan con textos históricos o documentales, cuya preservación debe ser a todo nivel y se consideran fuentes primarias.
- **Directorios.** Listados enormes de datos personalizados o de direcciones de correo electrónico, números telefónicos, etc.

b).1 ¿A QUE SE REFIERE CUANDO SE HABLA DE BASES DE DATOS RELACIONALES?

- Se refiere a que es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí.
- Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas.
- En una base de datos relacional, cada fila en una tabla es un registro con una ID única, llamada clave.

MOTORES DE BASES DE DATOS RELACIONALES

1. ORACLE

ORACLE

El motor relacional comercial más antiguo. Su creador, Larry Ellison, estuvo en el comité que definió SQL.



2. MICROSOFT SQL SERVER

Multiplataforma desde 2017. Son líderes en Business Intelligence (integrando más apps en el mismo paquete).

3. MYSQL

MySQL

Es el motor más usado en la web (y preferido por los CMS clásicos que usan PHP como WordPress, Drupal, Magento, etc.).



4. SQLITE

Es una base de datos embebida en el programa. Al estar integrado en todos los teléfonos se usa para almacenamiento interno de apps

5. MARIADB

MariaDB

Fork derivado de MySQL a partir de su compra por Oracle. Compatible con MySQL para poder cambiar un motor por otro.



6. POSTGRESQL

Inició como un proyecto universitario llamado INGRES, inspirado en Oracle. Usan funciones y triggers que MySQL no tuvo por años.

Aprende a diseñar y administrar bases de datos SQL en:

ed.team/cursos/sql



c).1

¿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", como personas, objetos o conceptos, se relacionan entre sí dentro de un sistema.
- Los diagramas ER a menudo se combinan con los diagramas de flujo de datos (DFD), que trazan el flujo de la información para procesos o sistemas.



c).2

LOS COMPONENTES Y LAS CARACTERÍSTICAS DE UN DIAGRAMA ER

- **Entidad**

Algo que se puede definir, como una persona, objeto, concepto u evento, que puede tener datos almacenados acerca de este.

- **Relación**





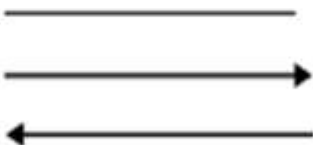
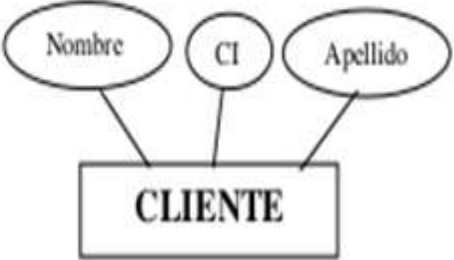

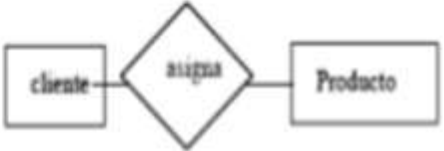
Cómo las entidades interactúan o se asocian entre sí. Piensa en las relaciones como si fueran verbos.

- **Atributo**

Una propiedad o característica de una entidad. A menudo se muestra como un óvalo o círculo.

d).1

¿CUÁLES SON
LAS FIGURAS
QUE
REPRESENTA
N A UN
DIAGRAMA
ENTIDAD
RELACIÓN?

DESCRIPCIÓN	SÍMBOLO	EJEMPLO
Rectángulos: representan conjuntos de Entidades.	Entidad 	
Elipses: representan atributos	Atributo 	
Líneas: conectan los atributos a los conjuntos de entidades, y los conjuntos de relaciones	Conexión 	
Rombos: representan relaciones.	Relación 	

e).1

¿QUÉ ES SQL SERVER Y QUÉ ES SQL SERVER MANAGEMENT STUDIO?

- ¿QUÉ ES MICROSOFT SQL SERVER?

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.



e).2

¿QUÉ ES SQL SERVER Y QUÉ ES SQL SERVER MANAGEMENT STUDIO?

- **¿PARA QUÉ SIRVE EXACTAMENTE MICROSOFT SQL SERVER?**

Microsoft SQL Server es ideal para almacenar toda la información deseada en bases de datos relacionales, como también para administrar dichos datos sin complicaciones, gracias a su interfaz visual y a las opciones y herramientas que tiene.

Es algo vital, especialmente en webs que tienen la opción de registrar usuarios para que inicien sesión.



e).3

¿QUÉ ES SQL SERVER Y QUÉ ES SQL SERVER MANAGEMENT STUDIO?

- ¿Qué es SQL Server Management Studio (SSMS)?

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 Data base.



e).4

¿QUÉ ES SQL SERVER Y QUÉ ES SQL SERVER MANAGEMENT STUDIO?

¿PARA QUÉ SE UTILIZA SQL SERVER MANAGEMENT STUDIO?

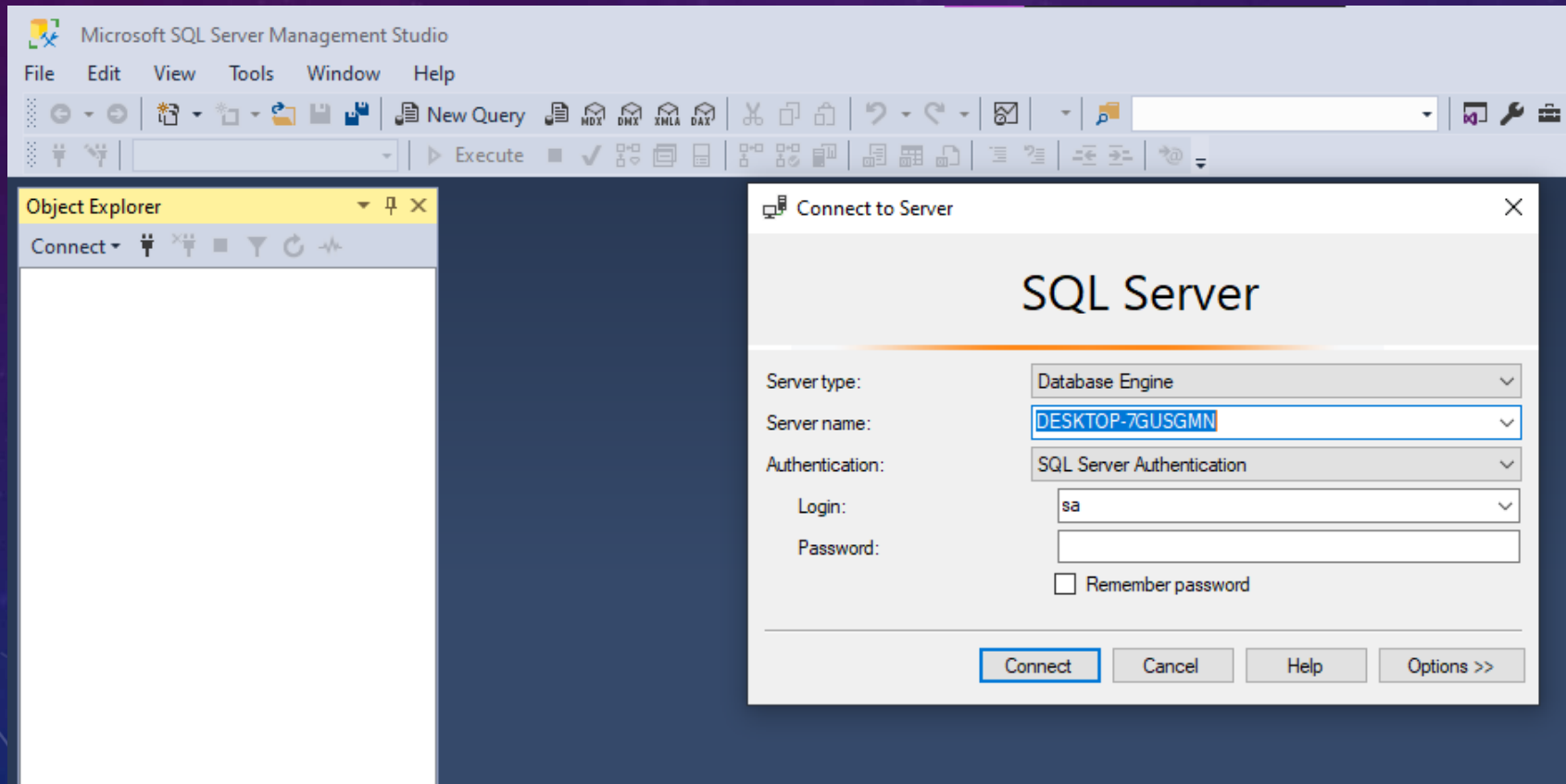
Se utiliza para las siguientes **actividades de la base de datos**, entre otras:

- Cree y modifique bases de datos rápidamente;
- Agregar y modificar objetos de base de datos, incluidas tablas y vistas;
- Prueba de objetos de base de datos con herramientas de prueba externas;
- Implementación de bases de datos;
- Ejecución de una query en bases de datos;
- Optimizar las bases de datos para mejorar el tiempo de respuesta;
- Gestionar bases de datos como copias de seguridad y restauración;
- Importar y exportar datos.

f).1

¿CÓMO SE CREA UNA BASE DE DATOS?

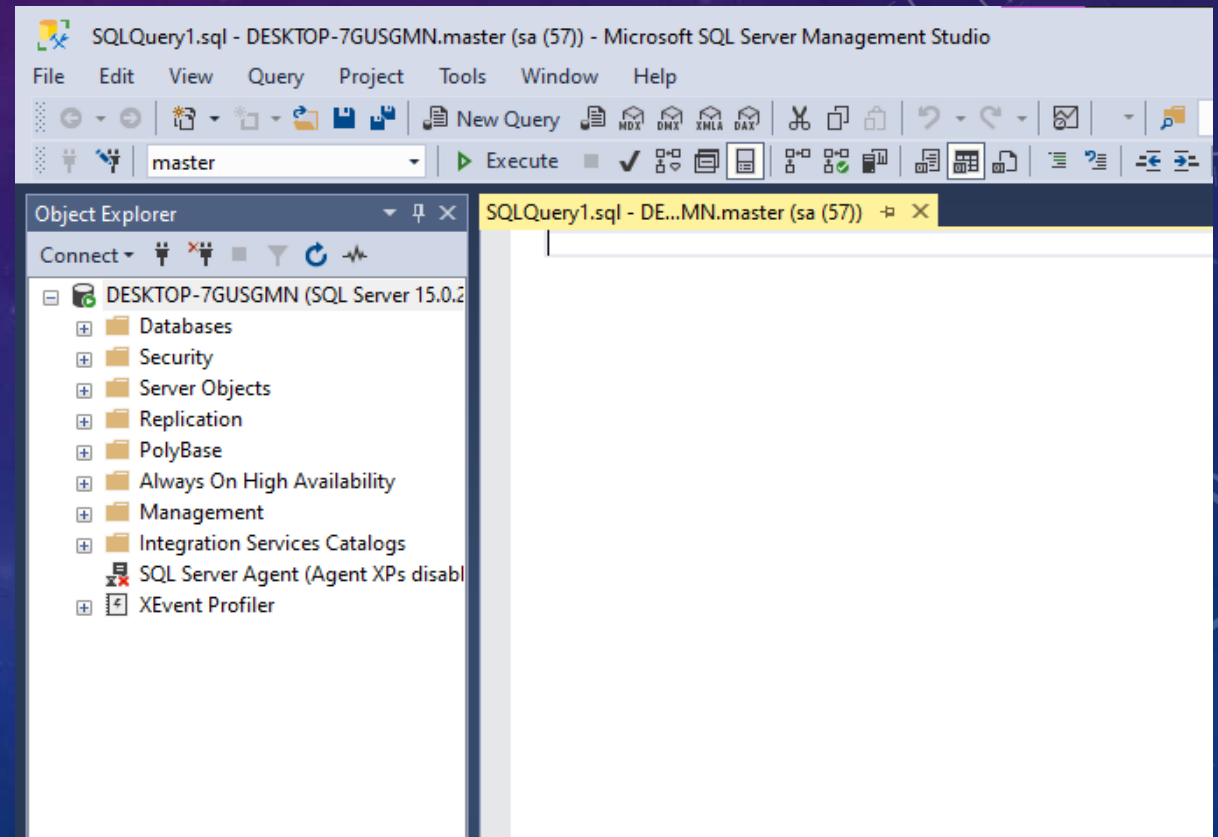
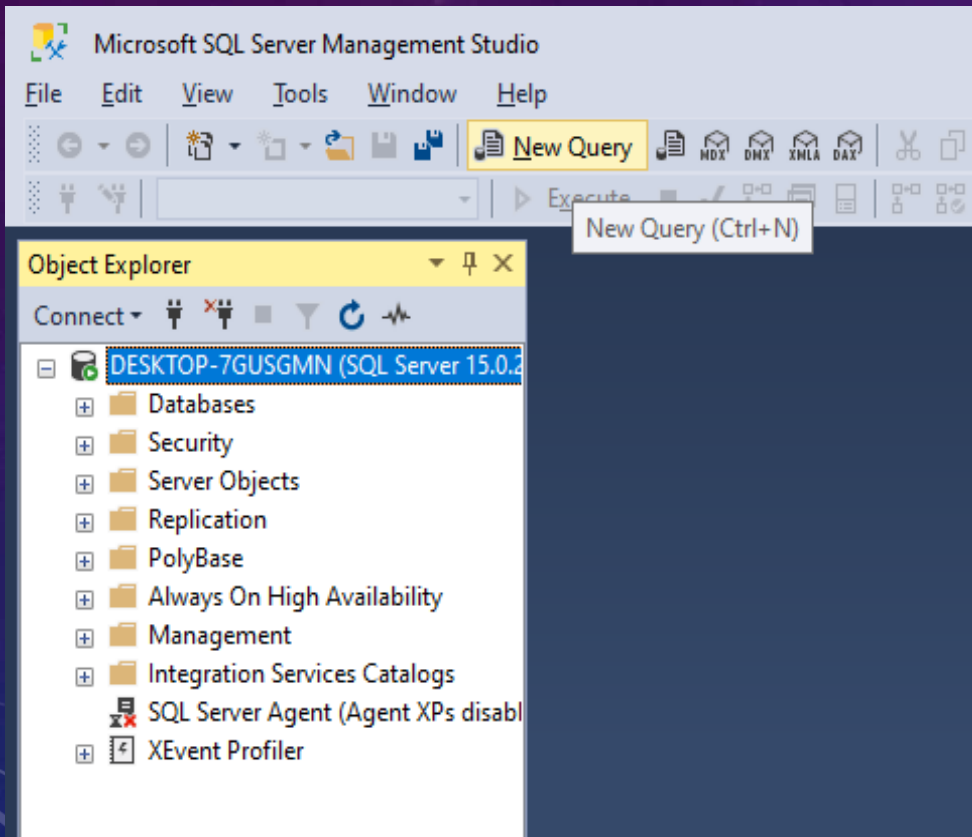
PASO 1: ABRIMOS LO QUE ES EL PROGRAMA



f).2

¿CÓMO SE CREA UNA BASE DE DATOS?

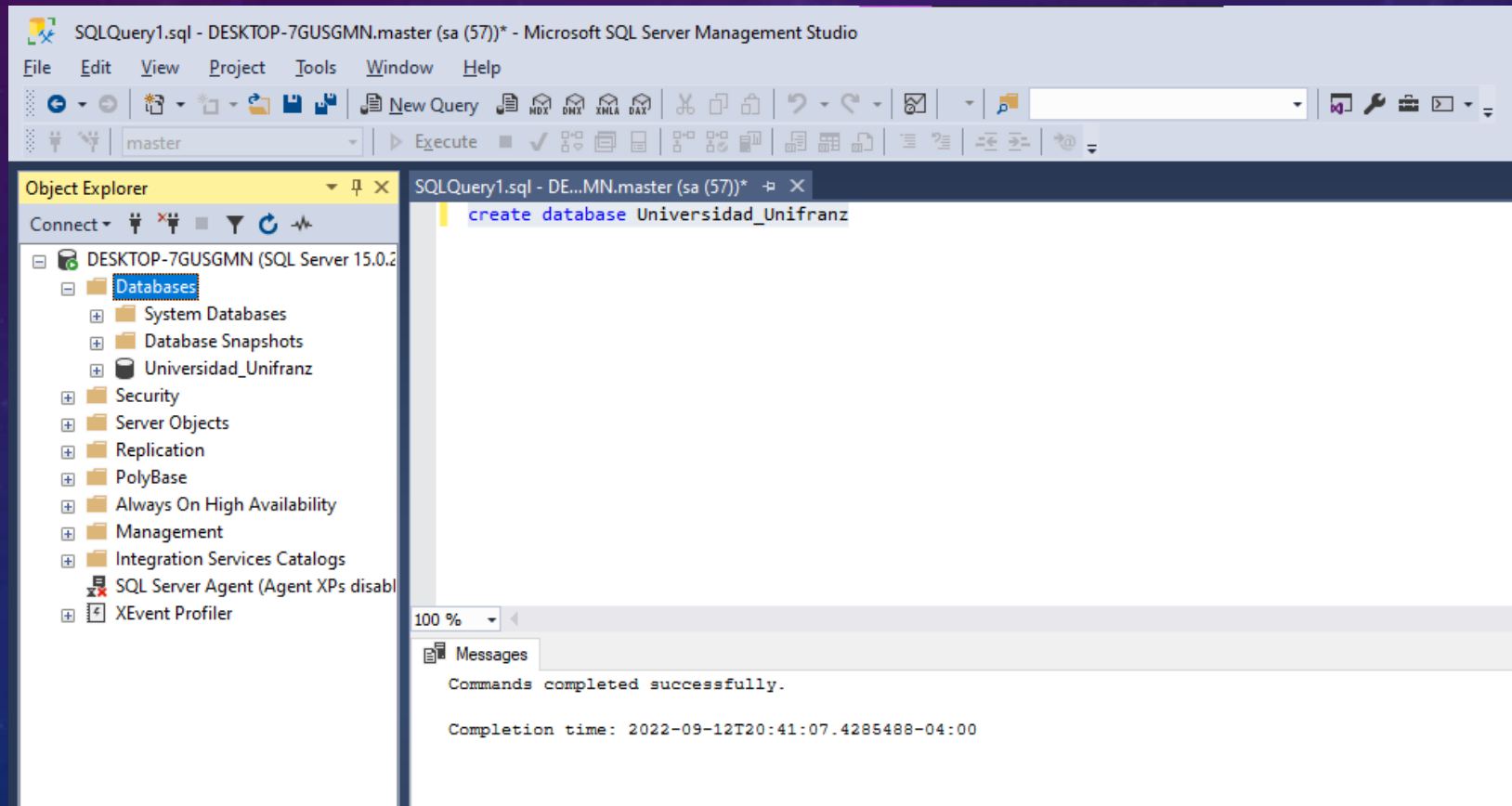
PASO 2: ABRIMOS NUEVO DOCUMENTO



f).3

¿CÓMO SE CREA UNA BASE DE DATOS?

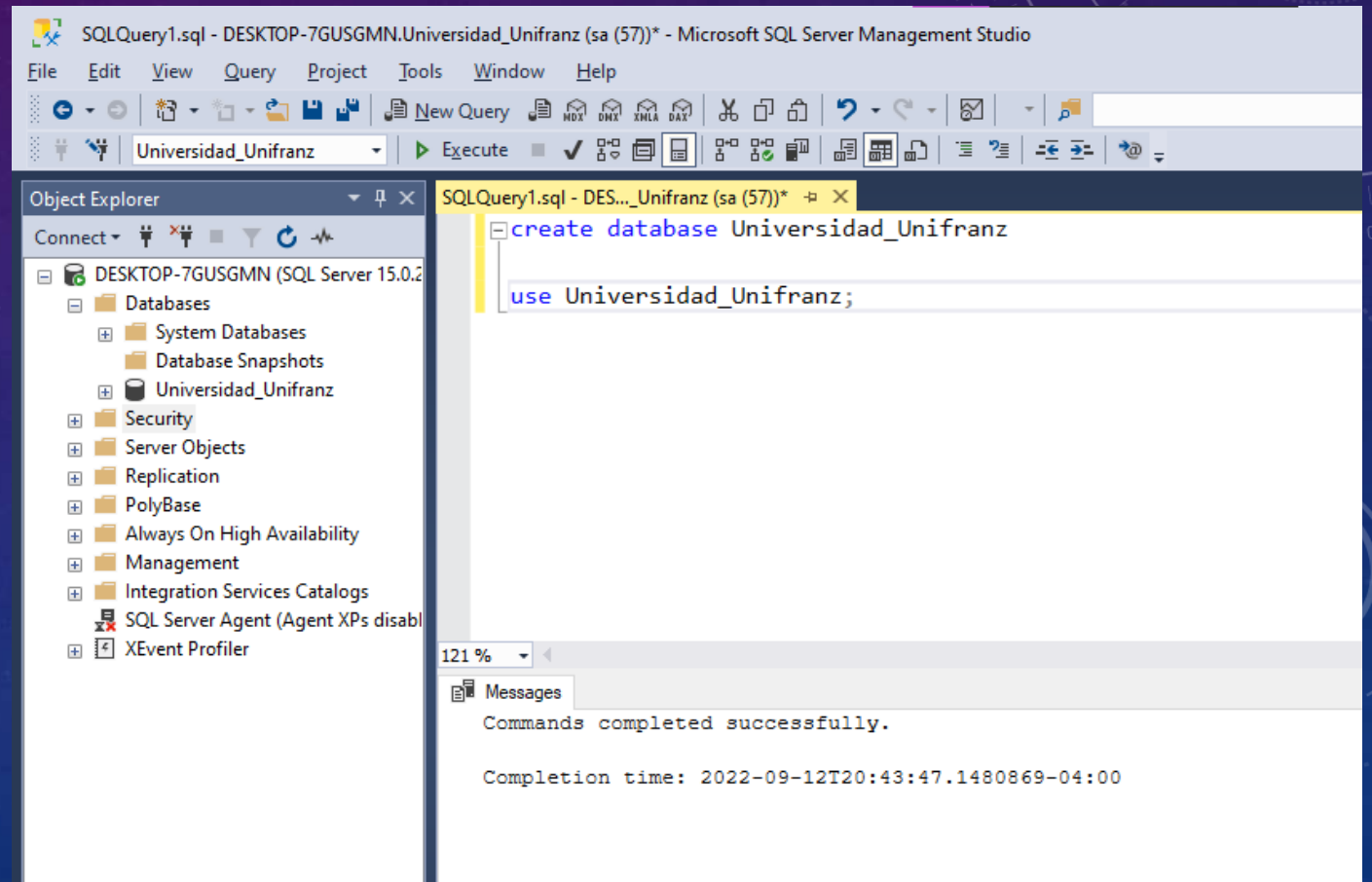
PASO 3: ESCRIBIMOS EL COMANDO PARA CREAR UNA BASE DE DATOS “CRÉATE DATABASE” Y LUEGO COLOCAMOS EL NOMBRE DE LA BASE DE DATOS Y PULSAMOS F5 PARA EJECUTARLO



g).1

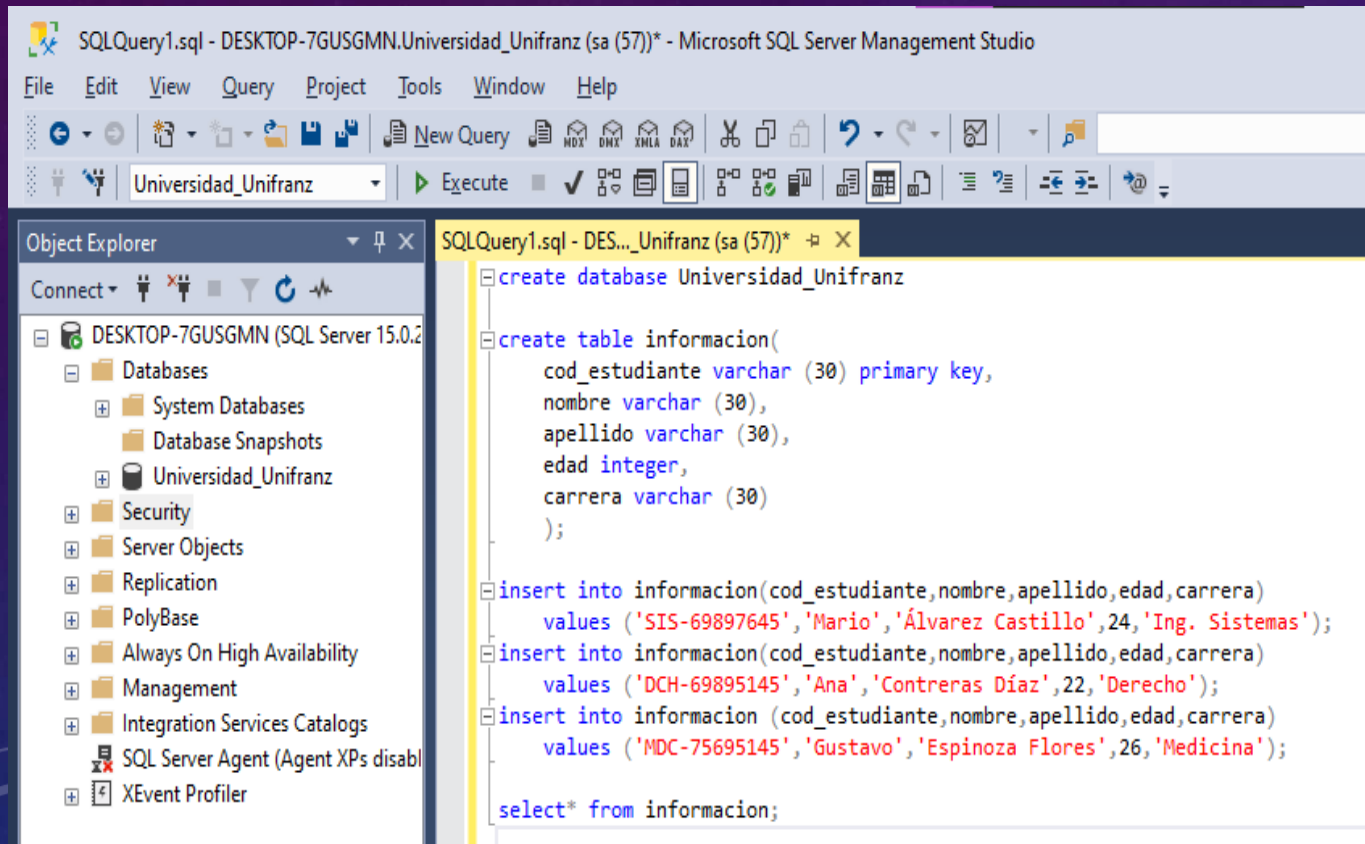
¿PARA QUÉ SIRVE EL COMANDO USE?

- se utiliza para designar una base externa como base de datos actual, en otras palabras, la base a la cual se dirigirán las próximas consultas SQL en el proceso actual.



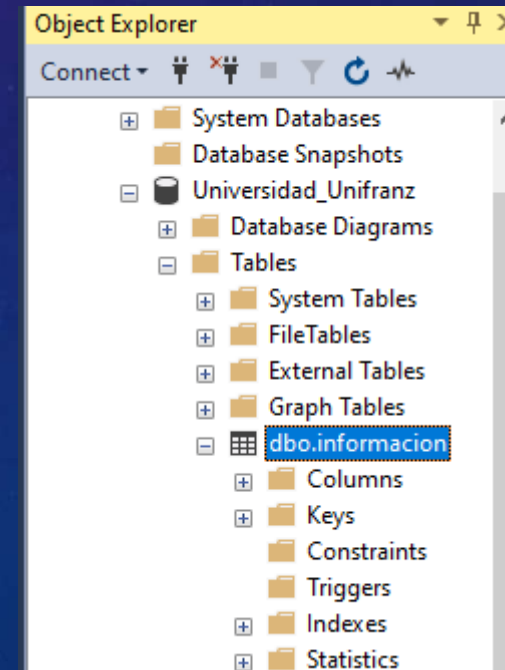
h).1

CREAR UNA TABLA CUALQUIERA CON 3 COLUMNAS Y SU PRIMARY KEY.



The screenshot shows the 'Results' tab in SQL Server Management Studio. It displays a table with 6 columns: 'cod_estudiante', 'nombre', 'apellido', 'edad', and 'carrera'. The table contains 3 rows of data:

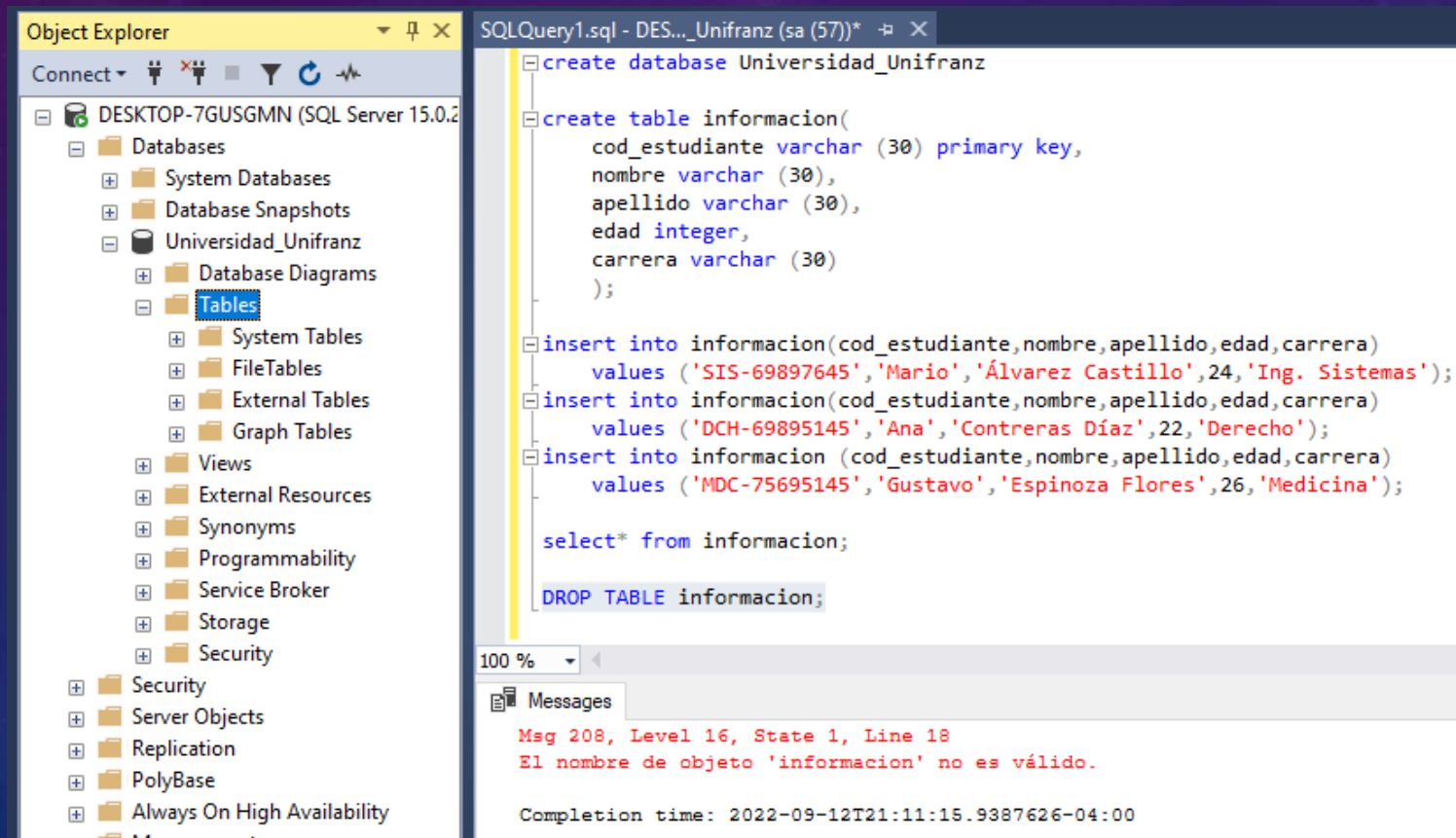
	cod_estudiante	nombre	apellido	edad	carrera
1	DCH-69895145	Ana	Contreras Diaz	22	Derecho
2	MDC-75695145	Gustavo	Espinoza Flores	26	Medicina
3	SIS-69897645	Mario	Alvarez Castillo	24	Ing. Sistemas



h).1

¿CÓMO SE ELIMINA UNA TABLA?

PARA ELIMINAR UNA TABLA DEBEMOS DE COLOCAR EL SIGUIENTE COMANDO “DROP TABLE” Y POSTERIORMENTE EL NOMBRE DE LA TABLA CREADA ANTERIORMENTE



The screenshot displays the SQL Server Enterprise Manager interface on the left and a SQL Query window on the right. In the Enterprise Manager, the 'Tables' folder under the 'Universidad_Unifranz' database is selected. The SQL Query window shows a script that creates a database, a table named 'informacion', inserts three rows of data, and then attempts to drop the table. The script is as follows:

```
create database Universidad_Unifranz

create table informacion(
    cod_estudiante varchar (30) primary key,
    nombre varchar (30),
    apellido varchar (30),
    edad integer,
    carrera varchar (30)
);

insert into informacion(cod_estudiante,nombre,apellido,edad,carrera)
values ('SIS-69897645','Mario','Álvarez Castillo',24,'Ing. Sistemas');
insert into informacion(cod_estudiante,nombre,apellido,edad,carrera)
values ('DCH-69895145','Ana','Contreras Díaz',22,'Derecho');
insert into informacion (cod_estudiante,nombre,apellido,edad,carrera)
values ('MDC-75695145','Gustavo','Espinoza Flores',26,'Medicina');

select* from informacion;

DROP TABLE informacion;
```

Below the script, the 'Messages' pane shows an error message:

```
Msg 208, Level 16, State 1, Line 18
El nombre de objeto 'informacion' no es válido.
```

The completion time is noted as 2022-09-12T21:11:15.9387626-04:00.

I. PARTE PRACTICA



a).1

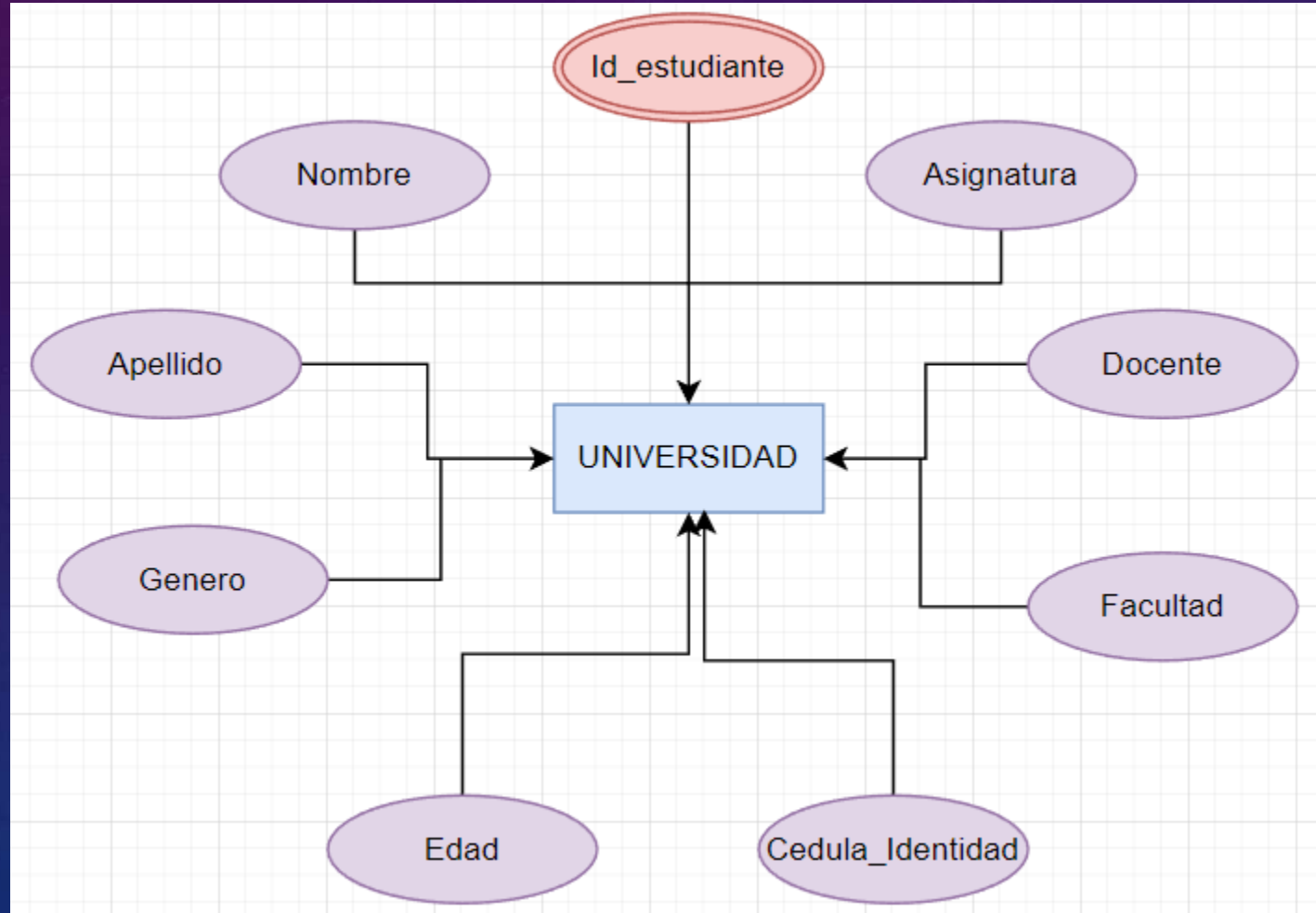
CREAR EL DISEÑO PARA UNA “UNIVERSIDAD”

UNIVERSIDAD

- **id_estudiante** (varchar = string, Llave primaria y unica)
- **Nombre** (varchar = string)
- **Apellido** (varchar = string)
- **Genero** (varchar = string)
- **Edad** (integer)
- **Cedula_identidad** (integer)
- **Facultad** (varchar = string)
- **Docente** (varchar = string)
- **Asignatura** (varchar = string)

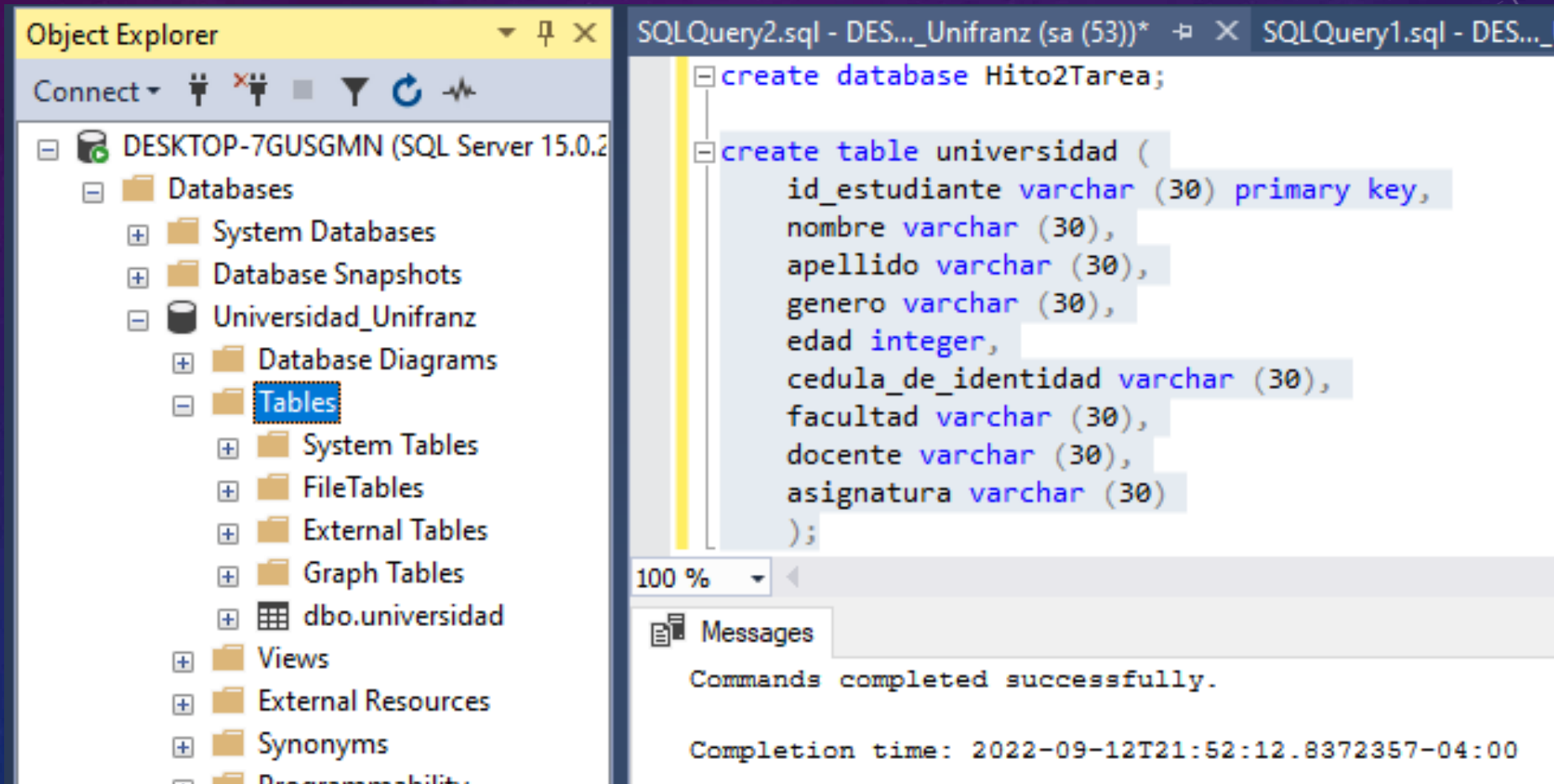
b).1

CREAR EL DIAGRAMA ENTIDAD RELACIÓN E-R PARA EL EJERCICIO ANTERIOR



c).1

CREAR LA TABLA UNIVERSIDAD EN BASE AL DISEÑO ANTERIOR.



The screenshot displays the SQL Server Enterprise Manager interface. On the left, the 'Object Explorer' pane shows the 'Unifranz' database selected, with the 'Tables' folder expanded. The main window shows the 'SQLQuery2.sql' script with the following SQL code:

```
create database Hito2Tarea;  
  
create table universidad (  
    id_estudiante varchar (30) primary key,  
    nombre varchar (30),  
    apellido varchar (30),  
    genero varchar (30),  
    edad integer,  
    cedula_de_identidad varchar (30),  
    facultad varchar (30),  
    docente varchar (30),  
    asignatura varchar (30)  
);
```

The 'Messages' pane at the bottom indicates that the commands were completed successfully.

Commands completed successfully.

Completion time: 2022-09-12T21:52:12.8372357-04:00

d).1

AGREGAR REGISTROS A LA TABLA CREADA ANTERIORMENTE

SQLQuery2.sql - DES..._Unifranz (sa (53))* X SQLQuery1.sql - DES..._Unifranz (sa (57))*

```
insert into universidad (id_estudiante,nombre,apellido,genero,edad,cedula_de_identidad,facultad,docente,asignatura)
values ('SIS-74646923','gerardo','apaza quispe','masculino',20,74646923,'Ingenieria','ing. Jose Tomas Morilla','calculo II');
insert into universidad (id_estudiante,nombre,apellido,genero,edad,cedula_de_identidad,facultad,docente,asignatura)
values ('DCH-64840923','jose','cordon hervas','masculino',23,64840923,'Ciencias','doc. Federico Barrera','Constitucional II');
insert into universidad (id_estudiante,nombre,apellido,genero,edad,cedula_de_identidad,facultad,docente,asignatura)
values ('MDC-68987923','Maria','Seoane Pascual','femenino',25,68987923,'Ciencias de la Salud humana','doc. Angel Laguna','Bioquímica Médica');
insert into universidad (id_estudiante,nombre,apellido,genero,edad,cedula_de_identidad,facultad,docente,asignatura)
values ('SIS-77489823','Aroa','Tapia Rovira','femenino',22,77489823,'Ingenieria','ing. Silvia Villanueva','Programacion II');

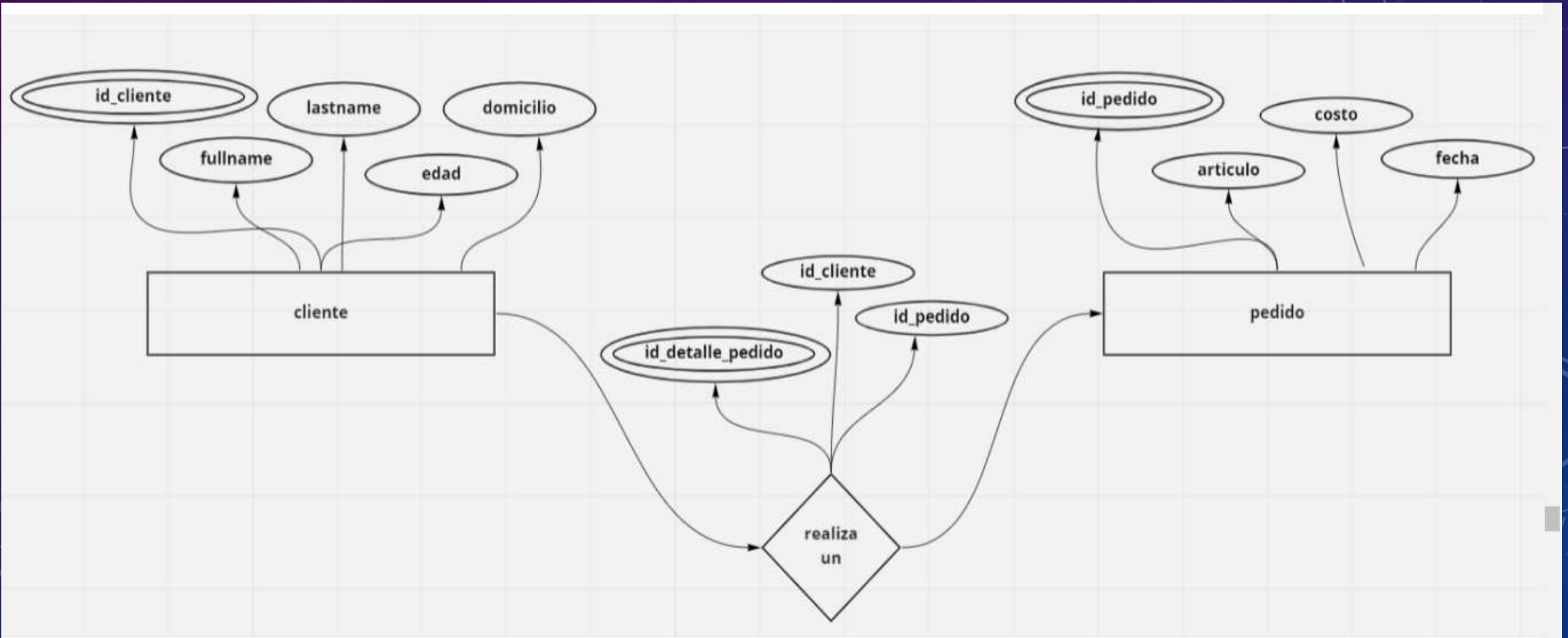
select* from universidad;
```

100 %

Results Messages

	id_estudiante	nombre	apellido	genero	edad	cedula_de_identidad	facultad	docente	asignatura
1	DCH-64840923	jose	cordon hervas	masculino	23	64840923	Ciencias	doc. Federico Barrera	Constitucional II
2	MDC-68987923	Maria	Seoane Pascual	femenino	25	68987923	Ciencias de la Salud humana	doc. Angel Laguna	Bioquimica Medica
3	SIS-74646923	gerardo	apaza quispe	masculino	20	74646923	Ingenieria	ing. Jose Tomas Morilla	calculo II
4	SIS-77489823	Aroa	Tapia Rovira	femenino	22	77489823	Ingenieria	ing. Silvia Villanueva	Programacion II

e).1 **CREAR LAS TABLAS Y 2 REGISTROS PARA CADA TABLA PARA EL SIGUIENTE MODELO ER.**



e).2 CREAR LAS TABLAS Y 2 REGISTROS PARA CADA TABLA PARA EL SIGUIENTE MODELO ER.

A QUI TENEMOS 2 TABLAS “CLIENTE Y PEDIDO” CON SUS REGISTROS

```
create database POLLOS_COPA;

create table cliente (
    id_cliente varchar (30) primary key,
    nombre varchar (30),
    apellido varchar (30),
    edad integer,
    domicilio varchar (30)
);

insert into cliente (id_cliente,nombre,apellido,edad,domicilio)
values ('KHA7TAFG','Roxana','Melian Barrera',34,'086 Mosciski Mountains');

select* from cliente;
```

```
create table pedido (
    id_pedido varchar (30) primary key,
    articulo varchar (30),
    costo integer,
    fecha varchar (30)
);

insert into pedido (id_pedido,articulo,costo,fecha)
values ('HBSW7T6G','4 presas',30,'18-de-agosto');

select* from pedido
```

e).2 CREAR LAS TABLAS Y 2 REGISTROS PARA CADA TABLA PARA EL SIGUIENTE MODELO ER.

Y EN ESTE CUANDRO VEMOS LA RELACION ENTIDAD DE LAS 2 TABLAS ANTERIORES

```
create table detalle_pedido (  
    id_detalle_pedido varchar (30) primary key,  
    id_cliente varchar (30),  
    id_pedido varchar (30),  
    foreign key (id_cliente) references cliente (id_cliente),  
    foreign key (id_pedido) references pedido (id_pedido)  
);  
  
insert into detalle_pedido (id_detalle_pedido,id_cliente,id_pedido)  
values ('JDI948HDW','KHA7TAFG','HBSW7T6G');  
  
select* from detalle_pedido;
```

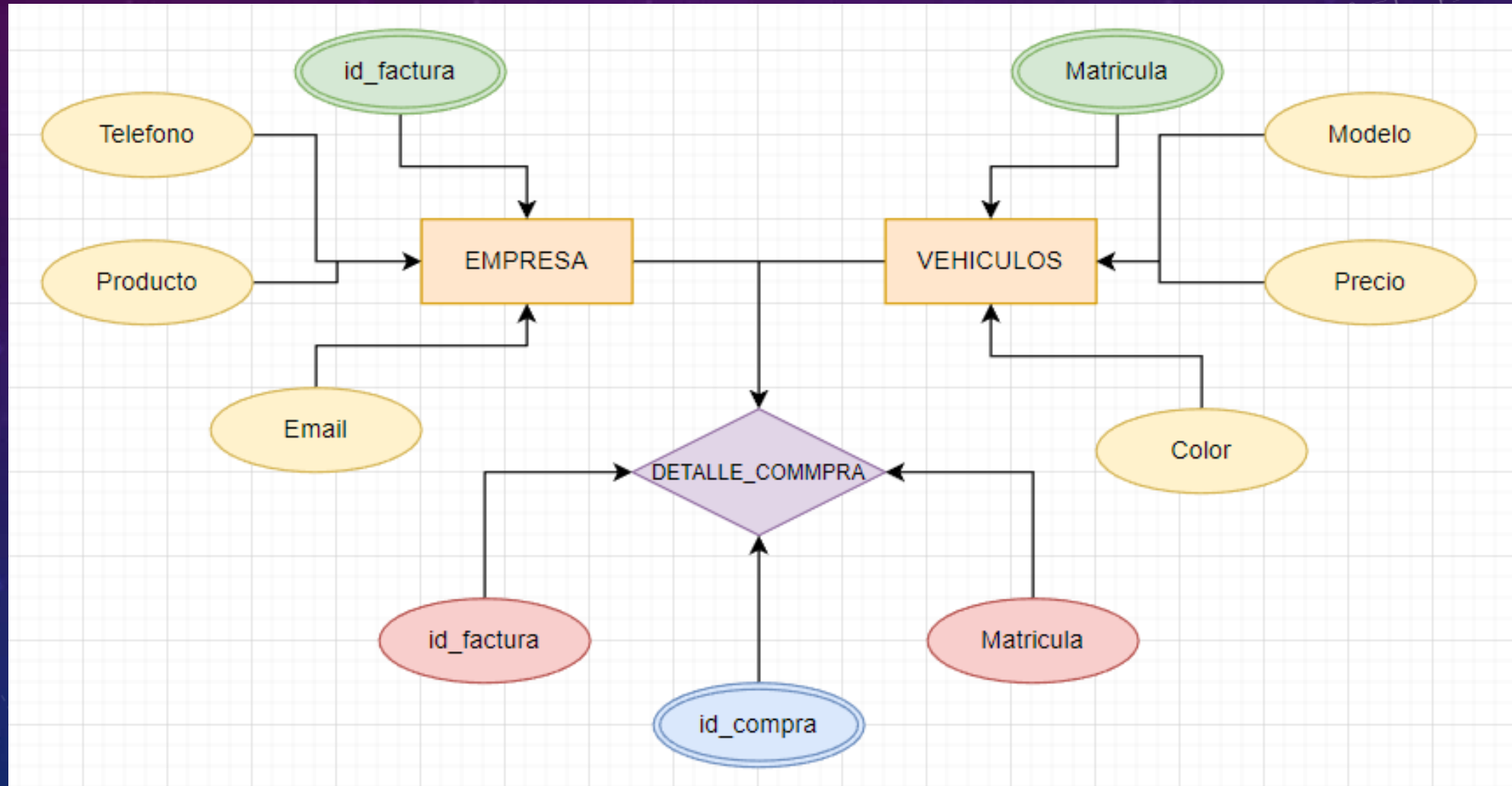
100 %

Results Messages

	id_detalle_pedido	id_cliente	id_pedido
1	JDI948HDW	KHA7TAFG	HBSW7T6G

f).1

CREAR EL MODELO ENTIDAD RELACIÓN ER Y SU CÓDIGO SQL.



f).2 CREAR EL MODELO ENTIDAD RELACIÓN ER Y SU CÓDIGO SQL.

A QUI SE VE LAS DOS TABLAS CON SUS RESPECTIVOS REGISTRO DE DOS ENTIDADES

```
create database universidad;

create table empresa (
    id_factura varchar (30) primary key,
    telefono integer,
    producto varchar (30),
    email varchar (30)
);

insert into empresa (id_factura,telefono,producto,email)
values ('KF3-F54V',78465342,'accesorios','will.esc.jih@gmail.com');

select* from empresa;
```

```
create table vehiculo (
    matricula varchar (30) primary key,
    modelo varchar (30),
    precio integer,
    color varchar (30)
);

insert into vehiculo(matricula,modelo,precio,color)
values ('5421-TFD','COROLLA-12AS',23000,'Azul Marino');

select* from vehiculo;
```

f).2

CREAR EL MODELO ENTIDAD RELACIÓN ER Y SU CÓDIGO SQL.

Y EN ESTE CUADRO YA PODEMOS VER LO QUE ES LA RELACION ENTRE LAS DOS TABLAS ANTERIORES

```
create table detalle_compra (  
    id_compra varchar (30) primary key,  
    id_factura varchar (30),  
    matricula varchar (30),  
    foreign key (id_factura) references empresa (id_factura),  
    foreign key (matricula) references vehiculo (matricula)  
);  
  
insert into detalle_compra (id_compra,id_factura,matricula)  
values ('KJ8DY-WMKJ0','KF3-F54V','5421-TFD');  
  
select* from detalle_compra;
```

100 %

Results Messages

	id_compra	id_factura	matricula
1	KJ8DY-WMKJ0	KF3-F54V	5421-TFD

**GRACIAS
POR SU
ATENCIÓN**

