



## Actividad | 1 | Base de datos para el sistema web

### Nombre del curso

---

Ingeniería en Desarrollo de Software



TUTOR: Aaron Salazar.

ALUMNO: Uziel de Jesús López Ornelas.

FECHA: 05 de febrero de 2026.

## **Tabla de Contenido**

<b>Introducción .....</b>	<b>1</b>
<b>Investigación de la actividad .....</b>	<b>1</b>
Restricciones.....	1
<b>Desarrollo .....</b>	<b>2</b>
Diagrama entidad-relación .....	2
Evidencia de tener SQL Server .....	3
Diagrama Base de datos.....	3
Creación de la base de datos.....	4
Creación de las tablas correspondientes .....	4
Ingreso de datos en las diferentes tablas con los 5 registros .....	5
Consulta de tablas a través de INNER JOINS .....	5
Link de GitHub.....	5
<b>Conclusión .....</b>	<b>6</b>
<b>Referencias.....</b>	<b>7</b>

## **Introducción**

Con esta actividad iniciaremos la materia de sistemas web II en esta tenemos una pequeña contextualización, una tienda llamada Sara necesita que un ingeniero en desarrollo de software realice una página web para manejar un carrete de compras, en específico una tienda que necesita un diseño para un sitio web e-commerce de microservicios. Principalmente se nos otorgan información que son las tablas, dentro de las tablas tienen sus atributos y de esos atributos se desplegarán llaves principales, llaves foráneas y relaciones. Para esto ocuparemos el software de SQL Server que será necesario instalarlo en nuestra PC, para comprender mejor todos estos términos necesitamos hacer una investigación acerca de las restricciones que maneja este software, por supuesto también tenemos que crear mediante las tablas un diagrama entidad de relación, este lo podemos crear en algún software de nuestro gusto ya sea diagrams.net o algún otro software. Dentro de SQL Server crearemos de igual manera un diagrama entidad relación. Ahora pasaremos al paso más importante; creación de las tablas y añadido de información con comandos que ya conocemos de acuerdo al software de SQL Server. Para realizar uno de los últimos pasos se tendría que consultar por medio de INNER JOINS las combinaciones de cierta información que nos proporciona la tabla para dar por finalizado generar el script de la actividad o de la base de datos para guardarla en un enlace de GitHub.

## **Investigación de la actividad**

### **Restricciones**

**NOT NULL:** se usa esta restricción para evitar insertar valores NULL en la columna especificada, es decir, un valor no aceptado.

**UNIQUE:** se utiliza para garantizar que no se inserte en valores duplicados en una columna específica o combinación de columnas que participen en la restricción UNIQUE y no formen parte de la PRIMARY KEY.

**PRIMARY KEY:** la columna o el conjunto de columnas que participan en PRIMARY KEY no pueden aceptar el valor NULL si esta se define en varias columnas. Todas las columnas en este valor tienen que ser únicas.

**FOREIGN KEY:** es una clave de SQL donde se utiliza para vincular dos tablas. La restricción identifica las relaciones de las tablas.

**CHECK:** se define en una columna o conjunto de columnas para limitar el rango de valores que se puede insertar en las columnas mediante una condición predefinida.

DEFAULT: proporciona un valor de columna predeterminado para las filas insertadas si no se especifica ningún valor para esta columna.

INDEX: se utiliza para crear y recuperar datos de la base de datos muy rápidamente.

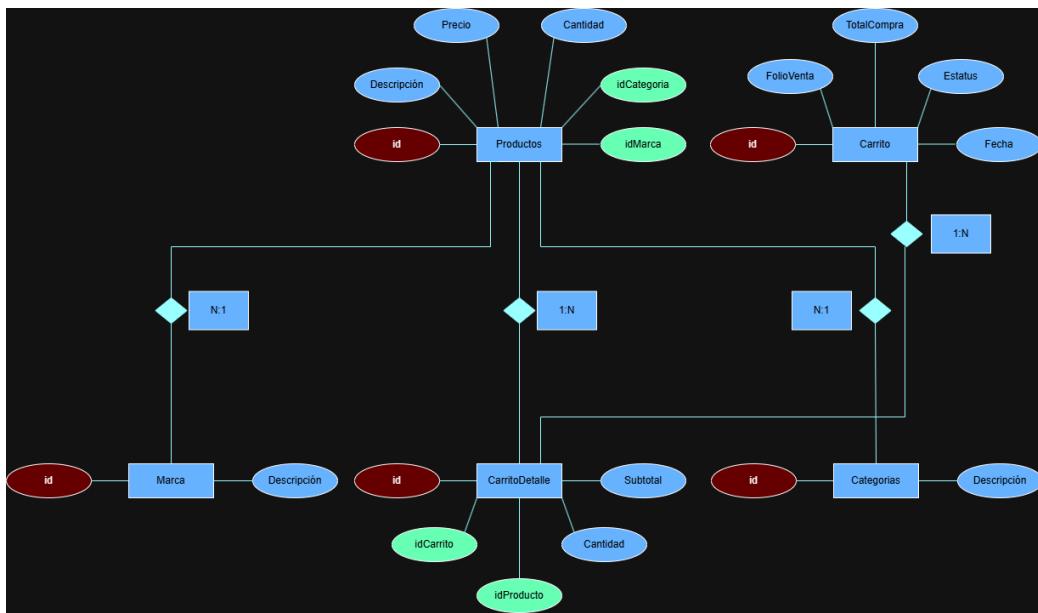
COLLATE: se usa para hacer referencia al nombre de la intercalación de SQL server o de la declaración de Windows que se va a aplicar la expresión definición de columna o definición de base de datos.

AUTO\_INCREMENT: se utiliza para generar automáticamente un número único cuando se inserta un nuevo registro en la tabla.

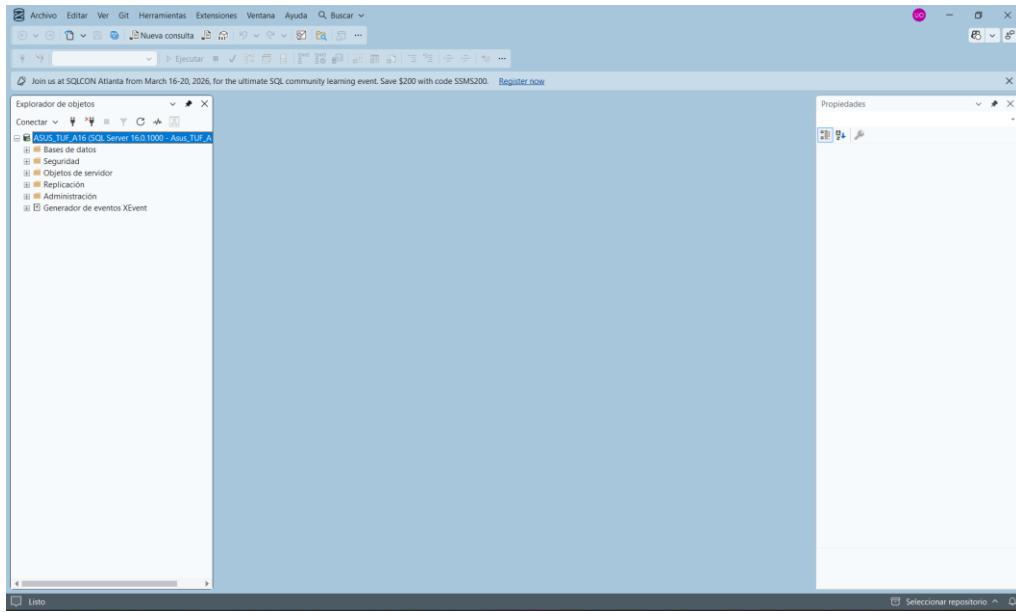
ENUM/SET: es un tipo de datos de cadena que permite restringir los valores de una columna a un conjunto predefinidos de opciones. Aparte de que también te permite seleccionar un valor o si se utiliza SET te permite seleccionar de cero, uno o varios valores.

## Desarrollo

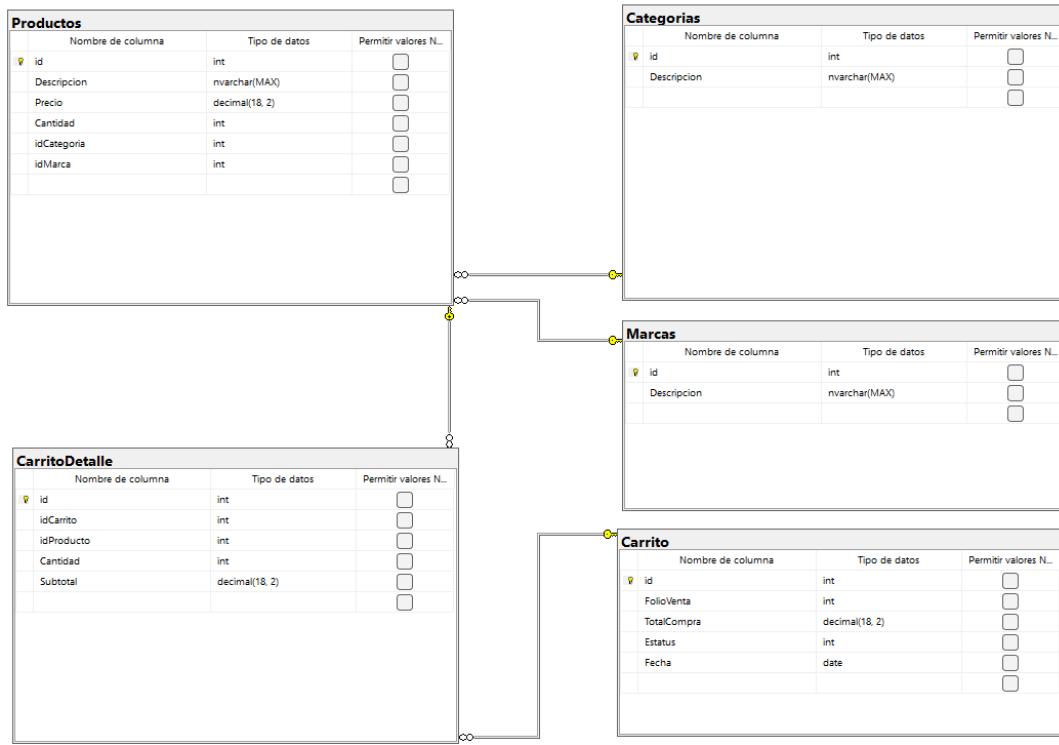
### Diagrama entidad-relación



## Evidencia de tener SQL Server



## Diagrama Base de datos



## Creación de la base de datos

```
SQLQuery1.s...asust (69)* ✘ ×
1   CREATE DATABASE Tienda_Sara
2
3   USE Tienda_Sara|
```

## Creación de las tablas correspondientes

```
CREATE TABLE Categorias (
    Id INT PRIMARY KEY IDENTITY (1,1),
    Descripcion VARCHAR (100) NOT NULL,
);|
```

```
CREATE TABLE Marca (
    Id INT PRIMARY KEY IDENTITY (1,1),
    Descripcion VARCHAR (100) NOT NULL,
);|
```

```
CREATE TABLE Productos (
    Id INT PRIMARY KEY IDENTITY (1,1),
    Descripcion VARCHAR (250) NOT NULL,
    Precio DECIMAL (10,2) NOT NULL,
    Cantidad INT NOT NULL,
    IdCategoria INT,
    IdMarca INT,
    CONSTRAINT FK_Productos_Categorias FOREIGN KEY (IdCategoria) REFERENCES Categorias(Id),
    CONSTRAINT FK_Productos_Marca FOREIGN KEY (IdMarca) REFERENCES Marca(Id),
);
```

```
CREATE TABLE Carrito (
    Id INT PRIMARY KEY IDENTITY (1,1),
    FolioVenta VARCHAR (50) UNIQUE,
    TotalCompra DECIMAL (10,2) DEFAULT 0,
    Estatus VARCHAR (20),
    Fecha DATETIME DEFAULT GETDATE(),
);
```

```
CREATE TABLE CarritoDetalle (
    Id INT PRIMARY KEY IDENTITY (1,1),
    IdCarrito INT,
    IdProducto INT,
    Cantidad INT NOT NULL,
    Subtotal DECIMAL (10,2) NOT NULL,
    CONSTRAINT FK_CarritoDetalle_Carrito FOREIGN KEY (IdCarrito) REFERENCES Carrito(Id),
    CONSTRAINT FK_CarritoDetalle_Productos FOREIGN KEY (IdProducto) REFERENCES Productos(Id),
);|
```

## Ingreso de datos en las diferentes tablas con los 5 registros

```

INSERT INTO Categorias (Descripcion) VALUES
('Laptop'), ('Procesador'), ('Almacenamiento'), ('Accesorios'), ('Teclado');

INSERT INTO Marca (Descripcion) VALUES
('Asus'), ('AMD'), ('Samsung'), ('Gamecraft'), ('Novotek');

INSERT INTO Productos (Descripcion, Precio, Cantidad, IdCategoria, IdMarca) VALUES
('Laptop Asus TUF A16', 19000.00, 10, 1,1),
('AMD Ryzen 7535HS', 6500.00, 15, 2,2),
('SSD Samsung 1TB', 2500.00, 20, 3,3),
('Mouse RGB Gamecraft', 1200.00, 15, 4,4),
('Teclado GAMER Novotek', 700.00, 10, 5,5);

INSERT INTO Carrito (FolioVenta, TotalCompra, Estatus, Fecha) VALUES
('V-001', 20200.00, 'Pagado', '2026-02-13 23:34'),
('V-002', 6500.00, 'Pendiente', '2026-02-13 00:15'),
('V-003', 2500.00, 'Cancelado', '2026-02-13 02:56'),
('V-004', 1900.00, 'Pendiente', '2026-02-13 03:00'),
('V-005', 19000.00, 'Pagado', '2026-02-13 1:56');

INSERT INTO CarritoDetalle (IdCarrito, IdProducto, Cantidad, Subtotal) VALUES
(1,1,1, 19000.00),
(1,4,1, 1200.00),
(2,2,1, 6500.00),
(3,3,1, 2500.00),
(4,4,1, 1200.00);

```

## Consulta de tablas a través de INNER JOINS

```

SELECT
    P.Descripcion AS Productos,
    M.Descripcion AS Marca,
    C.Descripcion AS Categorias,
    P.Precio
FROM Productos P
JOIN Marca M ON P.IdMarca = M.Id
JOIN Categorias C ON P.IdCategoria = C.Id;

```

	Productos	Marca	Categorias	Precio
1	Laptop Asus TUF A16	Asus	Laptop	19000.00
2	AMD Ryzen 7535HS	AMD	Procesador	6500.00
3	SSD Samsung 1TB	Samsung	Almacenamiento	2500.00
4	Mouse RGB Gamecraft	Gamecraft	Accesorios	1200.00
5	Teclado GAMER Novotek	Novotek	Teclado	700.00

## Link de GitHub

[https://github.com/UZLOP984/Sistemas\\_Web\\_II.git](https://github.com/UZLOP984/Sistemas_Web_II.git)

## Conclusión

En esta primera actividad aprendí a analizar primeramente lo que es el diseño para la creación de una base de datos, No es simplemente entrar software y empezar a programar, antes de eso es importante crear componentes necesarios para realizar y encajar nuestra base de datos, eso quiere decir que analizando la información crearemos bocetos de las tablas, de las relaciones, de las llaves principales, de las llaves foráneas, qué comandos podemos utilizar y sobre todo qué información va a ir estructurada en dichas tablas. De acuerdo a que ya se tienen todo estructurado ahora pasamos a la acción esto es analizarlo y traspasarlo a SQL Server, colocar la sintaxis correcta y de ahí gestionar la información para que esta sea correcta y no tenga errores y si los hay pues entonces sería corregirlos para que pase por ese proceso de calidad que utilicen todos los softwares. El proceso es entretenido, pero sobre todo tenemos que estar de acuerdo en tener una concentración necesaria para no cometer errores ni dejar nada fuera porque si se vincula con alguna base de datos externa o algún programa se puede tener errores.

## Referencias

W3Schools.com. (s. f.). [https://www.w3schools.com/sql/sql\\_constraints.asp](https://www.w3schools.com/sql/sql_constraints.asp)

Yaseen, A. (2019, 16 diciembre). *Restricciones en SQL Server: SQL NOT NULL, UNIQUE y SQL PRIMARY KEY*. SQL Shack - Articles About Database Auditing, Server Performance, Data Recovery, And More. <https://www.sqlshack.com/es/restricciones-en-sql-server-sql-not-null-unique-y-sql-primary-key/>

Google Search. (s. f.).

[https://www.google.com/search?q=restricci%C3%B3n+sql+collate&oq=restricci%C3%B3n+sql+collate&gs\\_lcp=EgZjaHJvbWUyBggAEEUYOTIHCAEQIRigATIHCAIQIRigATIHCAMQIRifBTIHCAQQIRifBTIHCAUQIRifBTIHCAQYQIRifBTIHCAcQIRifBdIBCTEwMTY5ajBqN6gCCLACAfEFNddRUtk8XB8&sourceid=chrome&ie=UTF-8](https://www.google.com/search?q=restricci%C3%B3n+sql+collate&oq=restricci%C3%B3n+sql+collate&gs_lcp=EgZjaHJvbWUyBggAEEUYOTIHCAEQIRigATIHCAIQIRigATIHCAMQIRifBTIHCAQQIRifBTIHCAUQIRifBTIHCAQYQIRifBTIHCAcQIRifBdIBCTEwMTY5ajBqN6gCCLACAfEFNddRUtk8XB8&sourceid=chrome&ie=UTF-8)

Markingmyname. (s. f.). *COLLATE (Transact-SQL) - SQL server*. Microsoft Learn.

<https://learn.microsoft.com/es-es/sql/t-sql/statements/collations?view=sql-server-ver17>

W3Schools.com. (s. f.-b). [https://www.w3schools.com/sql/sql\\_autoincrement.asp](https://www.w3schools.com/sql/sql_autoincrement.asp)

GeeksforGeeks. (2025, 23 julio). *ENUM in MySQL*. GeeksforGeeks.

<https://www.geeksforgeeks.org/sql/enumerator-enum-in-mysql/>