

PROYECTO FINAL BASE DE DATOS

FARMACIA



Docente: Ing. William Roddy Barra Paredes

Nombre completo: Ludwing Antoni Vargas Ibarra

Nombre Grupo: Nacho Libre

Grupo: 7

Semestre: II/2021

Github URL:

[https://github.com/LToni18/BaseDeDatos1/tree/main/
Avance_Proyecto](https://github.com/LToni18/BaseDeDatos1/tree/main/Avance_Proyecto)

1. Introducción.

¿Por qué se eligió este tema para el proyecto?

- Debido a que este programa se puede llegar a implementar en un futuro en un negocio de manera personal.
- Para ello se deberá de implementar mejoras al programa para que se tenga un uso correcto del mismo.

Conceptos breves

¿Qué es SQL Server Management Studio y qué puedo hacer con él?

- 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.



¿Qué es DataGrip?

- DataGrip es el entorno de base de datos multimotor. Si el DBMS tiene un controlador de JDBC, puede conectarse a él mediante DataGrip. Este proporciona introspección de base de datos y varios instrumentos para crear o modificar objetos para los motores compatibles.



¿Qué es lo que se utilizara para su realización?

Se utilizan los siguientes programas para base de datos:

- Microsoft SQL Server Management Studio.
- DataGrip.

Se utiliza el siguiente programa para programación:

- Microsoft Visual Studio.

2. Diseño de la base de Datos.

2.1 Análisis y definición de Tablas.

Nombre de la Tabla	Descripción
Medicamentos	Almacena datos de los medicamentos que serán puestos a la venta
Inventario	Esta tabla se relaciona con la tabla medicamentos para ayudar al control de la venta y cuantos medicamentos quedan por vender al igual que los que ingresan a la farmacia.
Personal	Almacena todos los datos básicos del personal al igual que su usuario y contraseña para el login de esa manera tendremos todos los datos requeridos para nuestro personal al momento de dar un servicio
Facturación	Almacena los datos de las facturas de una farmacia y controla la venta, cuantos, cuales medicamentos se han llevado de la farmacia los clientes a su vez de incluir los datos del personal que lo atendió.

2.2 Diseño de la Base de Datos.

2.2.1 Código SQL de las tablas.

Nombre de la Tabla	Código
Medicamentos	<pre>CREATE TABLE Medicamentos(id_medicamento int PRIMARY KEY, nombre_medicamento VARCHAR(50), cantidad int, precio_venta int); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (875,'Paracetamol',15,3); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (321,'Ibuprofeno',20,5); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (953,'Aspirina',40,2); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (124,'Omeprazol',10,4); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (532,'Salbutamol',32,7); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta) VALUES (163,'Lanzoprasol',26,3); INSERT INTO Medicamentos(id_medicamento,nombre_medicamento,cantidad,precio_venta)</pre>

	<pre> cio_venta) VALUES (267,'Resfrianex',43,5); </pre>
Inventario	<pre> CREATE TABLE Inventario(id_inventario int PRIMARY KEY, id_medicamento int, nombre_medicamento VARCHAR(50), cantidad int, precio_compra int, precio_venta int, FOREIGN KEY (id_medicamento) REFERENCES Medicamentos(id_medicamento)); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (1,875,'Paracetamol',50,2,3); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (2,321,'Ibuprofeno',30,4,5); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (3,953,'Aspirina',42,1,2); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (4,124,'Omeprazol',37,3,4); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (5,532,'Salbutamol',13,5,7); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (6,163,'Lanzoprasol',9,2,3); INSERT INTO Inventario(id_inventario, id_medicamento, nombre_medicamento, cantidad, precio_compra, precio_venta) VALUES (7,267,'Resfrianex',27,3,5); </pre>
Personal	<pre> CREATE TABLE Personal(id_personal int PRIMARY KEY, nombre_completo VARCHAR(100), genero VARCHAR(1), edad int, CI int); INSERT INTO Personal(id_personal,nombre_completo,genero,edad,CI) VALUES (123,'Ludwing Vargas','M',22,1002); INSERT INTO Personal(id_personal,nombre_completo,genero,edad,CI) VALUES (568,'Alejandro Vargas','M',20,2054); INSERT INTO Personal(id_personal,nombre_completo,genero,edad,CI) VALUES (954,'Yeami Sanchez','F',19,5684); INSERT INTO Personal(id_personal,nombre_completo,genero,edad,CI) VALUES (347,'Lucero Chambi','F',21,8462); </pre>
Facturación	<pre> CREATE TABLE Facturacion(id_factura int PRIMARY KEY, id_medicamento int, nombre_cliente VARCHAR(50), fecha_venta DATE, precio_venta int, </pre>

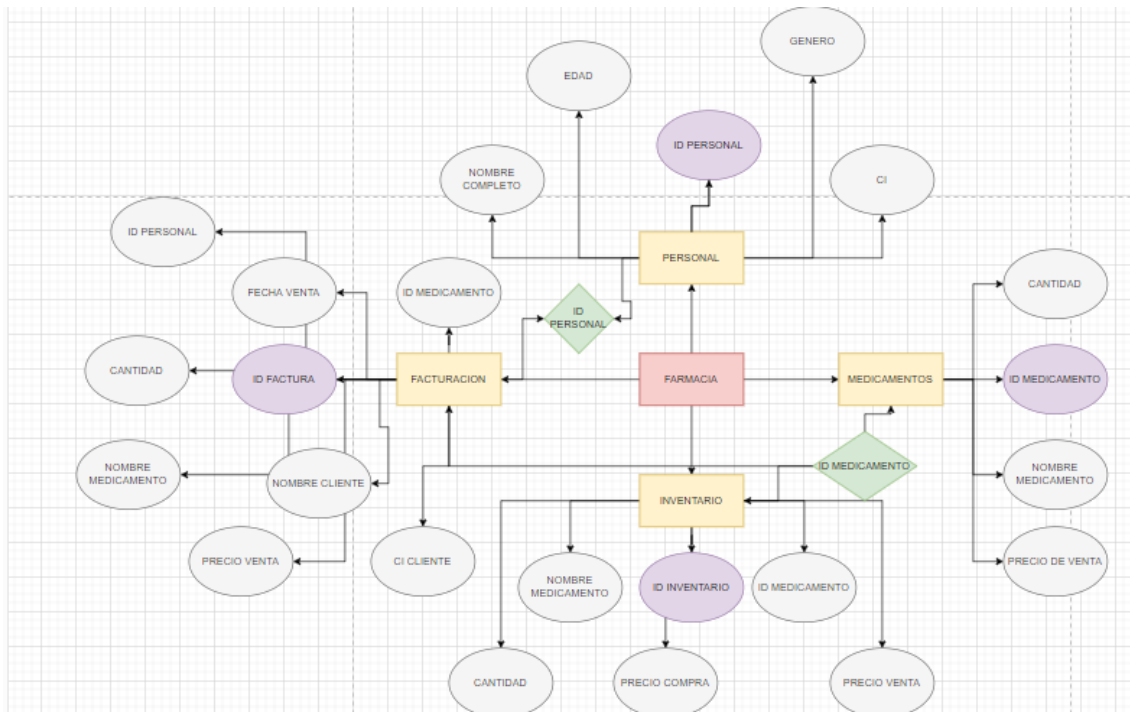
```

nombre_medicamento VARCHAR(50),
cantidad int,
id_personal int,
CI_cliente int,
FOREIGN KEY (id_medicamento) REFERENCES
Medicamentos(id_medicamento),
FOREIGN KEY (id_personal) REFERENCES
Personal(id_personal)
);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (567,875,'Pedro
Gonzales',13/07/2021,3,'Paracetamol',4,123,5876);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (961,321,'Aylin
Vargas',1/01/2019,5,'Ibuprofeno',10,568,7534);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (258,953,'Jhamil
Cuentas',23/03/2021,2,'Aspirina',2,347,9647);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (741,124,'Jesusa
Ibarra',21/10/2020,4,'Omeprazol',5,954,1234);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (852,532,'Fabricio
Mayta',17/05/2021,7,'Salbutamol',11,123,4826);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (963,163,'Katherine
Gareca',14/08/2020,3,'Lanzoprasol',6,954,3716);
INSERT INTO
Facturacion(id_factura,id_medicamento,nombre_cliente,fecha_
venta,precio_venta,nombre_medicamento,cantidad,id_personal,
CI_cliente)
VALUES (437,267,'Yesica
Rivera',30/11/2021,5,'Resfrianex',5,954,1867);

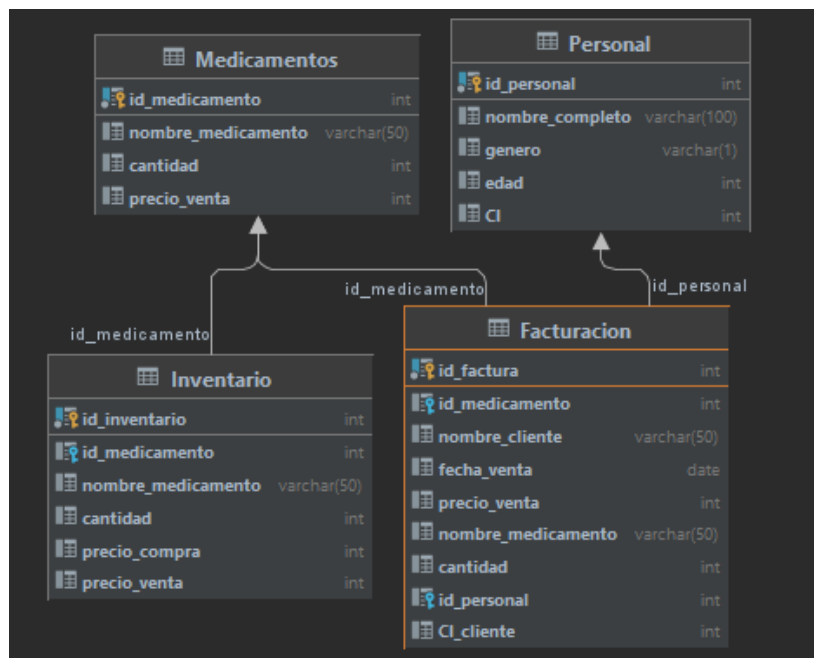
```

2.2.2 Modelo entidad relación de la Base de Datos ER

<https://drive.google.com/file/d/1tVfQ52xlyhTzZ7KipGlxbBF1K36Jgx2/view?usp=sharing>



2.2.3 Modelo lógico de la Base de Datos.



2.2.4 Búsquedas, funciones, vistas, etc.

```

3
SELECT inv.cantidad
FROM Inventario as inv
WHERE inv.id_medimento=124;
GO

SELECT fa.nombre_medimento,DATENAME (DW, fecha_venta)+' ', '+
DATENAME(DAY, fecha_venta)+' ', '
+DATENAME(MONTH, fecha_venta)+' ',
'+DATENAME(YEAR, fecha_venta)
FROM Facturacion as fa

```

```

GO

SELECT fa.nombre_medicamento
FROM Facturacion as fa
WHERE nombre_cliente='Yesica Rivera'
GO

SELECT
id_inventario,id_medicamento,nombre_medicamento,cantidad,precio_compra,precio_venta
FROM Inventario;
GO

CREATE OR ALTER FUNCTION valida_password (@passwordInfo
VARCHAR(100))
RETURNS VARCHAR(100)
AS
BEGIN
    DECLARE @respuesta VARCHAR(100) = '';
    IF LEN(@passwordInfo) >= 8
        BEGIN
            SET @respuesta = 'Password correcto!';
        END
    ELSE
        BEGIN
            SET @respuesta = 'Password incorrecto!';
        END
    RETURN @respuesta;
END;
SELECT dbo.valida_password('Yeami954');
GO

SELECT id_personal,nombre_completo,genero,edad,CI
FROM Personal;
GO

SELECT COUNT(per.edad)
FROM Personal as per
WHERE per.edad>=18;
GO

CREATE OR ALTER VIEW cantidad_inventario AS
SELECT inv.nombre_medicamento AS nombre_medicamento, inv.cantidad
as cantidad, cantidades=
    CASE
        WHEN inv.cantidad > 1 AND inv.cantidad <= 20 THEN
            'CANTIDAD BAJA'

        WHEN inv.cantidad > 20 AND inv.cantidad <= 40 THEN
            'CANTIDAD MEDIANA'

        WHEN inv.cantidad > 40 AND inv.cantidad <= 60 THEN
            'CANTIDAD ALTA'

        ELSE 'CANTIDAD DESCONOCIDA'
    END
FROM Inventario AS inv;

SELECT ca.*
FROM cantidad_inventario as ca;
GO

```

```

CREATE OR ALTER VIEW precios AS
    SELECT inv.precio_venta, precio =
        CASE
            WHEN inv.precio_venta > 1 AND inv.precio_venta >= 4 THEN
                'BARATO'
            WHEN inv.precio_venta > 4 AND inv.precio_venta >= 7 THEN
                'CARO'
            ELSE 'Sin precio'
        END
    FROM Inventario AS inv;

SELECT pr.*
FROM precios AS pr;
GO

SELECT id_medimento, nombre_medimento, cantidad, precio_venta
FROM Medicamentos;
GO

```

3. Conclusión

Finalmente se puede definir este proyecto como uno muy funcional, cabe recalcar que se deben de corregir algunos errores y hacer mejoras dentro del código, para el cual se debe de implementar para su funcionamiento.

También se debe de corregir la presentación para que sea mas atractivo a la vista.

Notas a considerarse:

En github crear una nueva carpeta de nombre PROYECTO_FINAL.

Los documentos que deben estar presente en github son los siguientes.

- Informe (documento solicitado en esta documentación)
- Un archivo excel(u otro) con todos los participantes (el título del archivo debe ser el nombre del grupo)
- Diagrama entidad relación (Imagen).
- Diagrama modelo lógico (Imagen).
- Archivo .sql con todo el código SQL de la Base de Datos.