

# BASE DE DATOS I

# TEMA: DEFENSA

## FINAL HITO-5

- ▶ INTEGRANTES:
- ▶ VICTOR HUGO QUISPE TORREZ
- ▶ SAUL ESCOBAR SERRANO
- ▶ BRAYAN JOSE VILLCA ICHUTA
- ▶ HEBER MOLLERICONA MIRANDA

# Aplicación a desarrollarse

```
CREATE DATABASE proyecto_hito5;

USE proyecto_hito5

CREATE TABLE biblioteca(id_biblioteca varchar (20) PRIMARY KEY,
nombre_biblioteca varchar (100) not null,
direccion varchar (50) not null,
horario_atencion varchar(30));

CREATE TABLE libro(id_libro int identity PRIMARY KEY,
nombre_libro varchar (50) not null,
seccion varchar (30) not null,
estado varchar (25) not null,
id_biblioteca varchar (20),
FOREIGN KEY (id_biblioteca) REFERENCES biblioteca(id_biblioteca));

CREATE TABLE usuario(id_usuario varchar(30) PRIMARY KEY,
nombre_usuario varchar (30) not null,
direccion_usuario varchar(50) not null,
celular int not null,
email varchar (30) not null,
id_libro int,
id_biblioteca varchar (20),
FOREIGN KEY (id_biblioteca) REFERENCES biblioteca(id_biblioteca),
FOREIGN KEY (id_libro) REFERENCES libro(id_libro));
```

El siguiente proyecto tiene el objetivo de aplicar los conocimientos adquiridos en el manejo de una biblioteca, realizando y priorizando el manejo de funciones y consultas como ejemplo de una base de datos de una biblioteca

# Objetivo

- Crear una base de datos que simule a un grupo de bibliotecas en donde en base a ciertos parámetros sea posible hallar al usuario, libro, la biblioteca correspondiente e incluso al genero del libro que uno quiere buscar.

```
CREATE TABLE genero_libros(id_genero varchar(50) PRIMARY KEY);

CREATE TABLE subgeneros(nombre_subgeneros varchar (50) ,
id_genero varchar(50),
id_libro int,
FOREIGN KEY (id_genero) REFERENCES genero_libros(id_genero),
Foreign Key (id_libro) REFERENCES libro(id_libro));

CREATE TABLE convenios(id_convenio varchar (30) Primary key,
nombre_institucion varchar(50),
tipo_convenio varchar(50),
id_biblioteca varchar(20),
id_usuario varchar (30),
FOREIGN KEY (id_biblioteca) REFERENCES biblioteca(id_biblioteca),
FOREIGN KEY (id_usuario) References usuario(id_usuario));
```

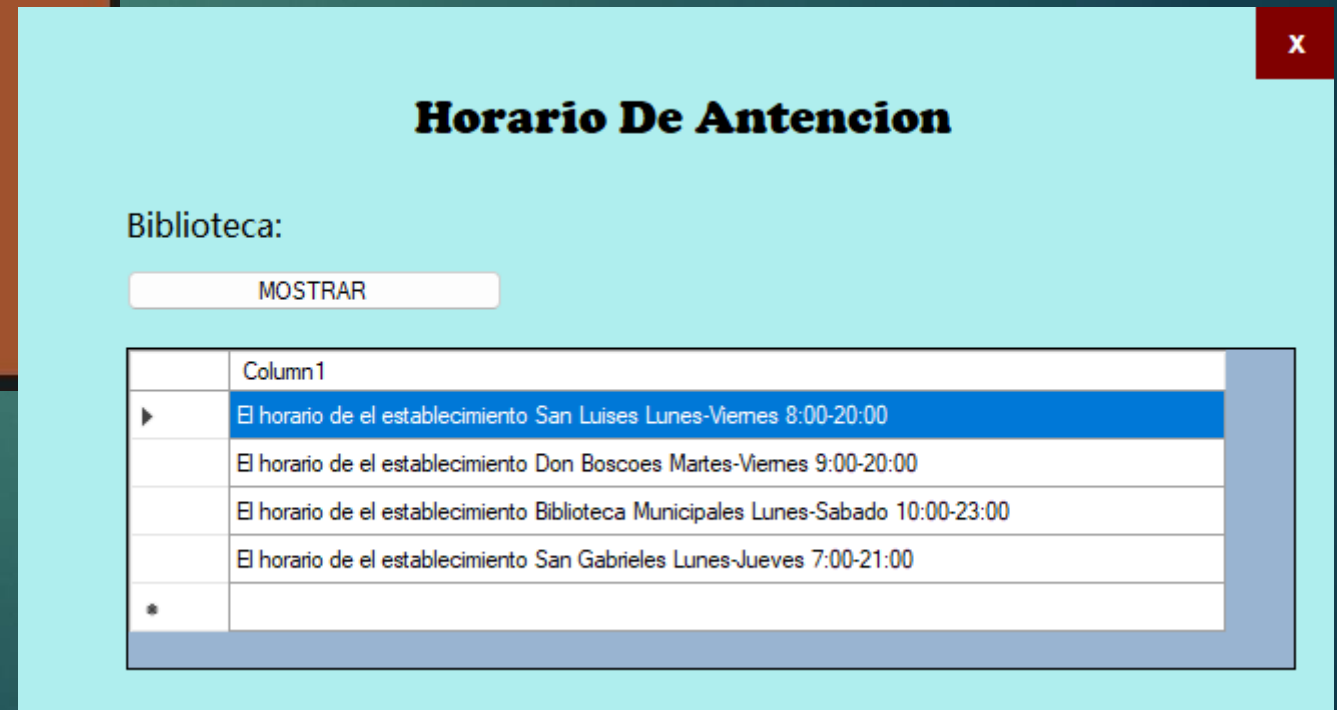
```
Insert into biblioteca(id_biblioteca,nombre_biblioteca,direccion,horario_atencion)
VALUES('bib-111','San Luis','Av. 6 de Agosto','Lunes-Viernes 8:00-20:00'),
('bib-222','Don Bosco','Av Civica','Martes-Viernes 9:00-20:00'),
('bib-333','Biblioteca Municipal','Plaza del estudiante','Lunes-Sabado 10:00-23:00'),
('bib-444','San Gabriel','Av. Busch','Lunes-Jueves 7:00-21:00');

Insert into libro(nombre_libro,seccion,estado,id_biblioteca)
Values ('La metamorfosis','sec-111','Disponible','bib-111'),
('Las mil y una noches','sec-222','Ocupado','bib-222'),
('Enciclopedia de Bolivia','sec-333','Disponible','bib-222'),
('Harry Potter','sec-111','Ocupado','bib-111'),
('Introduccion a C#','sec-333','Disponible','bib-333'),
('Administracion de empresas','sec-333','Ocupado','bib-444'),
('Calculo Integral y Diferencial','sec-222','Disponible','bib-444'),
('Algebra II','sec-111','Disponible','bib-333');

Insert into usuario(id_usuario,nombre_usuario,direccion_usuario,celular,email,id_libro,id_biblioteca) Values
('usu-111','Pedro Ramirez','Av 6 de Agosto',77789623,'Rami@gmail.com',6,'bib-444'),
('usu-222','Luis Gonzales','Nuevos Horizontes',77789624,'Luis@gmail.com',7,'bib-444'),
('usu-333','Noelia Fatima','Av Civica',77789625,'Noelia@gmail.com',2,'bib-222'),
('usu-444','Mario Garcia','Achumani',77789626,'Mario@gmail.com',4,'bib-111'),
('usu-555','Lola Lopez','Irpavi',77789627,'Lola@gmail.com',4,'bib-111'),
('usu-666','Tristan Condori','16 de Julio',77789628,'Tristan@gmail.com',1,'bib-333'),
('usu-777','Melissa Diaz','Av Arce',77789629,'Melissa@gmail.com',5,'bib-333'),
('usu-888','Ramiro Quispe','Av 6 de Marzo',77789630,'Ramiro@gmail.com',3,'bib-222');

Insert Into genero_libros(id_genero)Values
('Narrativo'),('Docente'),('Cantador'),('Actuador'),('Bello');
```

# Diseño en Visual Studio



## Solicitar Libro

x

Nombre:

Id\_Biblioteca:

Id\_Libro:

Email:

Direccion:

¿Tienes un convenio?

Si,  
Continuar

No,  
Solicitar

## Libros Disponibles

x

**Buscar:**

En la Biblioteca:

Por Genero De Libro

Por ID Libro

Buscar

Mostrar Todo

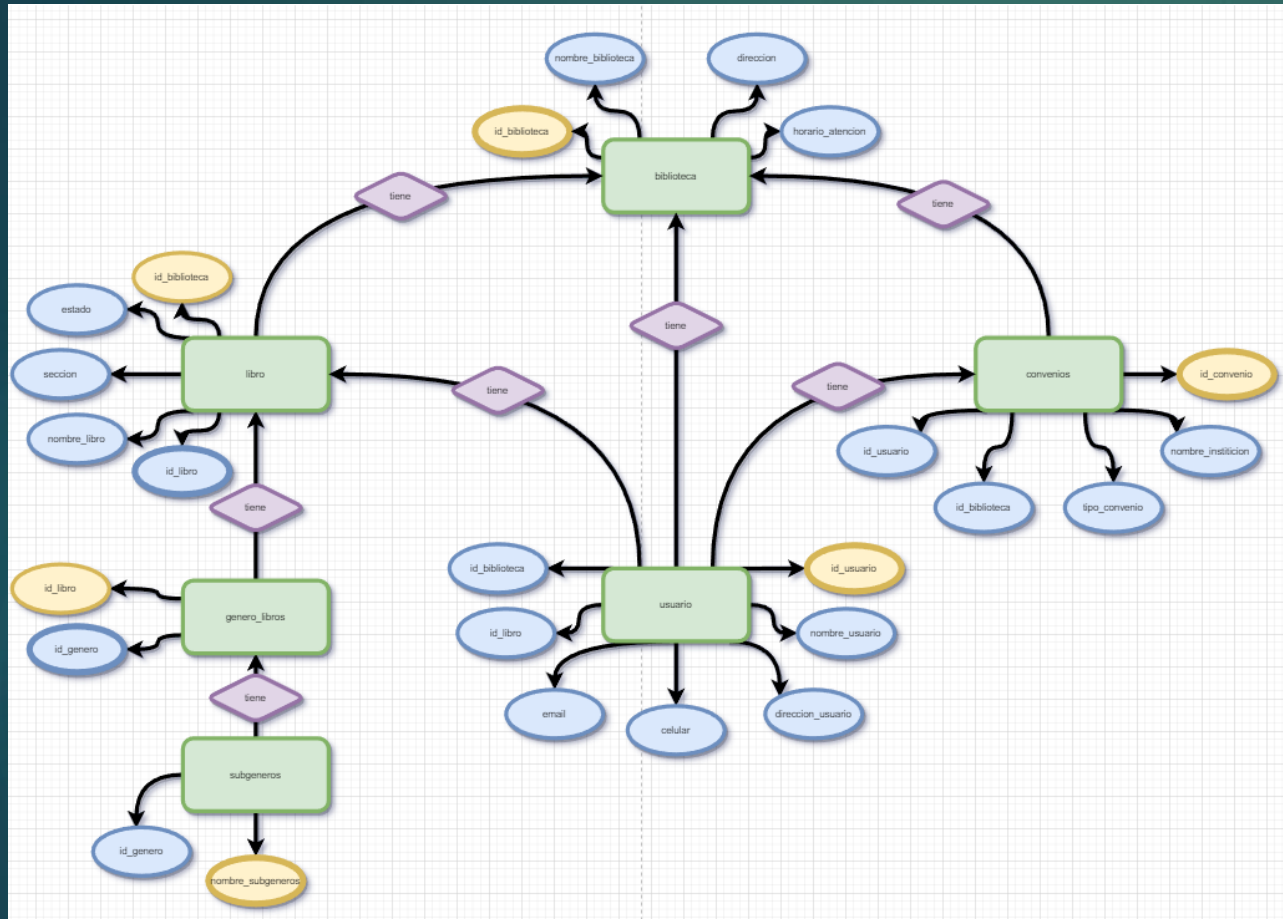


# Análisis y diseño

- El diseño en el editor SQL es el siguiente:



# Diseño Entidad Relación



- Se trata de la representación de la base de datos en el programa draw.io donde se nos muestra cada entidad y sus atributos y el como se relacionan unas entre otras, gracias a las herramientas que nos ofrecen.



# Manejo de funciones

```
CREATE FUNCTION libros_disponibles(@esta varchar(25))
Returns integer as
Begin
Declare @result int =0;
SELECT @result=count(lib.id_libro)
From libro as lib
WHERE lib.estado=@esta;
Return @result;
End;

Select dbo.libros_disponibles('Disponible')

Create FUNCTION horario(@nombre varchar (100))
returns varchar (100) as
begin
Declare @nom varchar(100)
Select @nom= concat('El horario es de ',bib.horario_atencion)
From biblioteca as bib
WHERE bib.nombre_biblioteca=@nombre
RETURN @nom;
END;

SELECT dbo.horario('Don Bosco')
```

- ❑ **Definición:**
- ❑ Como sabemos el editor SQL cuenta con algunas funciones agregadas (max, min, count, avg, sum) donde nos permiten efectuar acciones, pero devolviéndonos un valor único agregado. Eso nos lleva a que debemos crear nosotros una función en donde guiándonos en ciertos parámetros, nos sea posible obtener valores enteros o en cadena.

```

Create FUNCTION obtenerConvenio(@usu varchar(30),@bib varchar(20))
RETURNS VARCHAR (100) AS
BEGIN
DECLARE @resultado varchar(100);
Select @resultado=Concat('El usuario ',usu.nombre_usuario,' tiene convenio en ',bib.nombre_biblioteca+' de tipo '+con.tipo_convenio)
FROM usuario as usu
Inner JOIN biblioteca as bib on bib.id_biblioteca=usu.id_biblioteca
Inner JOIN convenios as con on con.id_biblioteca=bib.id_biblioteca
WHERE usu.id_usuario=@usu and bib.id_biblioteca=@bib
return @resultado;
end;

SELECT dbo.obtenerConvenio('usu-111','bib-444')

```

```

CREATE FUNCTION obtenerLibro(@idbiblio varchar(20),@gene varchar(50),@idlib int)
Returns varchar(50) as
begin
Declare @libro varchar(50);
Select @libro= CONCAT('Los libros disponibles son; ',lib.nombre_libro)
FROM libro as lib
Inner join biblioteca as bib on bib.id_biblioteca=lib.id_biblioteca
Inner JOIN subgeneros as sub on sub.id_libro=lib.id_libro
WHERE bib.id_biblioteca=@idbiblio and sub.nombre_subgeneros=@gene and lib.id_libro=@idlib
RETURN @libro;
END;

SELECT dbo.obtenerLibro('bib-333','Educativo',8)

```

# Consultas SQL

- Realizaremos algunas consultas SQL con el manejo de Inner Join, donde Inner Join nos permita relacionar 2 o mas tablas gracias a que estas comparten un atributo en común.

```
--CONSULTA SQL que muestre el nombre de usuario, el tipo de convenio y el nombre de la
--biblioteca que pertenezca al convenio='con-111' y la biblioteca='bib-111'
SELECT usu.nombre_usuario, con.tipo_convenio, bib.nombre_biblioteca
FROM usuario as usu
Inner JOIN convenios as con on con.id_usuario=usu.id_usuario
inner join biblioteca as bib on bib.id_biblioteca=usu.id_biblioteca
WHERE con.id_convenio='con-111' and bib.id_biblioteca='bib-111'

--Consulta que muestra el genero del libro y su nombre
SELECT gen.id_genero, lib.nombre_libro
FROM genero_libros as gen
Inner JOIN subgeneros as sub on gen.id_genero=sub.id_genero
Inner JOIN libro as lib on lib.id_libro=sub.id_libro
WHERE sub.id_genero='Docente'
```

```
--Consulta que muestra el email y el nombre del libro que pertenecen a la biblioteca ='bib-444'  
SELECT usu.email, lib.nombre_libro  
FROM usuario as usu  
Inner JOIN biblioteca as bib on bib.id_biblioteca=usu.id_biblioteca  
Inner JOIN libro as lib on lib.id_libro=usu.id_libro  
WHERE bib.id_biblioteca='bib-333'
```

```
--Consulta que muestra el nombre el tipo de convenio, la direccion del usuario que pertenezcan  
--a una biblioteca, y el nombre del libro que pertenezca a una seccion de dicha biblioteca  
SELECT lib.nombre_libro,con.tipo_convenio,usu.dirreccion_usuario  
FROM libro as lib  
INNER JOIN convenios as con on con.id_biblioteca=lib.id_biblioteca  
INNER JOIN usuario as usu on usu.id_biblioteca=lib.id_biblioteca  
WHERE lib.seccion='sec-222' and lib.id_biblioteca='bib-444'
```