

# Test de ingreso - Programación

## Realizar un ABM/CRUCD

Crear una base de datos llamada "TestCrud". En esta base de datos vas a tener que crear la/s estructura/s que te permitan tener a varios usuarios y distintos tipos de usuarios, ej un usuario visitante y un administrador.

Luego realizar un ABM o CRUD (Create, Read, Update, Delete) para poder darle de alta a estos usuarios, modificarlos o eliminarlos

Por último una pantalla de Login que te envíe a distintas URL según el tipo de usuario.

Por ejemplo, si, se loguea un usuario visitante => te envía a una pantalla que dice "Hola".

Si se loguea un administrador => lo dirige a la pantalla del ABM.

Para esto se debe utilizar .NET Core

## Generar todos los scripts necesarios de DB para resolver los siguientes puntos.

Se necesitan crear las siguientes tablas:

1) Crear una tabla de alquiler de películas y otra de ventas, que te permita registrar qué usuario compró/alquiló dicha película, a que precio y en qué momento

Se necesitan generar Stored Procedures que te permitan:

1) Crear usuarios, cuyo documento no exista actualmente en la base de datos, en caso de existir, debería devolver un mensaje de error, en caso contrario insertarlo

2) Crear/Borrar/Modificar películas (Borrar es poner en 0 el stock de ventas y alquileres)

3) Crear géneros

4) Asignar géneros a películas, verificar que la película no tenga asignada el género previamente.

5) Alquilar y Vender películas

6) Obtener las películas en stock para alquiler

7) Obtener las películas en stock para vender

8) Alquilar película

9) Vender película

10) Devolver película

11) Ver películas que no fueron devueltas y que usuarios la tienen

12) Ver qué películas fueron alquiladas por usuario y cuánto pagó y que día

13) Ver todas las películas, cuantas veces fueron alquiladas y cuanto se recaudo por ellas

\*/

```
CREATE TABLE tRol
(
    cod_rol INT IDENTITY PRIMARY KEY
    ,txt_desc VARCHAR(500)
    ,sn_activo INT
)
GO
INSERT INTO trol VALUES ( 'Administrador',-1)
INSERT INTO trol VALUES ( 'Cliente', -1)
GO
CREATE TABLE tUsers
(cod_usuario INT PRIMARY KEY IDENTITY, txt_user VARCHAR(50), txt_password
VARCHAR(50),txt_nombre VARCHAR(200), txt_apellido VARCHAR(200), nro_doc
VARCHAR(50), cod_rol INT, sn_activo INT
, CONSTRAINT fk_user_rol FOREIGN KEY (cod_rol) REFERENCES tRol(cod_rol)
)
GO
INSERT INTO tUsers VALUES ( 'Admin', 'PassAdmin123', 'Administrador', 'Test', '1234321', -1,-1)
INSERT INTO tUsers VALUES ('userTest', 'Test1', 'Ariel', 'ApellidoConA', '12312321', 1, -1)
INSERT INTO tUsers VALUES ('userTest2', 'Test2', 'Bernardo', 'ApellidoConB', '12312322', 1, -1)
INSERT INTO tUsers VALUES ('userTest3', 'Test3', 'Carlos', 'ApellidoConC', '12312323', 1, -1)
GO
CREATE TABLE tPelicula (cod_pelicula INT PRIMARY KEY IDENTITY, txt_desc
VARCHAR(500), cant_disponibles_alquiler INT, cant_disponibles_venta INT,
precio_alquiler NUMERIC(18,2), precio_venta NUMERIC(18,2))
GO
INSERT INTO tPelicula VALUES ('Duro de matar III', 3, 0,1.5,5.0)
INSERT INTO tPelicula VALUES ('Todo Poderoso', 2,1,1.5,7.0)
INSERT INTO tPelicula VALUES ('Stranger than fiction', 1,1,1.5,8.0)
INSERT INTO tPelicula VALUES ('OUIJA', 0,2,2.0,20.50)
GO
CREATE TABLE tGenero (cod_genero INT PRIMARY KEY IDENTITY, txt_desc
VARCHAR(500) )
INSERT INTO tGenero VALUES('Acción')
INSERT INTO tGenero VALUES('Comedia')
INSERT INTO tGenero VALUES('Drama')
INSERT INTO tGenero VALUES('Terror')
GO
CREATE TABLE tGeneroPelicula (cod_pelicula INT, cod_genero INT
, PRIMARY KEY(cod_pelicula, cod_genero)
, CONSTRAINT fk_genero_pelicula FOREIGN KEY(cod_pelicula) REFERENCES
tpelicula(cod_pelicula)
```

```
, CONSTRAINT fk_pelicula_genero FOREIGN KEY(cod_genero) REFERENCES  
tGenero(cod_genero))  
GO  
INSERT INTO tGeneroPelicula VALUES(1,1)  
INSERT INTO tGeneroPelicula VALUES(2,2)  
INSERT INTO tGeneroPelicula VALUES(3,2)  
INSERT INTO tGeneroPelicula VALUES(3,3)  
INSERT INTO tGeneroPelicula VALUES(4,4)  
GO
```