**Este diseño se corresponde con el video de apoyo número 6**

El procedimiento de eliminar usuario solo debe ejecutarse si el usuario se ha creado por error. Si el usuario ya tiene registradas operaciones a su nombre (ej. Una compra) no debemos eliminarlo ya que se perdería la relación con otras tablas.

USE BDSISTEMAVENTAS;

GO

CREATE PROC SP\_ELIMINARUSUARIO (

@Idusuario int,

@Documento varchar(50),

--Parámetros de salida resultado de operar con el registro usuario

@Respuesta bit output, --controla 1=se ha realizado operación 0=no se ha realizado la operacion

@Mensaje varchar(500) output)

AS

BEGIN

SET @Respuesta=0

SET @Mensaje=''

DECLARE @PasoRegla bit = 1

if exists(SELECT \* FROM COMPRA INNER JOIN USUARIO ON COMPRA.IdUsuario=USUARIO.IdUsuario

WHERE USUARIO.IdUsuario= @Idusuario)

begin

set @Respuesta=0

SET @PasoRegla=0;

set @Mensaje=@Mensaje+'No se puede eliminar porque este usuario tiene operaciones de compra registradas.\n'

end

if exists(SELECT \* FROM VENTA INNER JOIN USUARIO ON VENTA.IdUsuario=USUARIO.IdUsuario

WHERE USUARIO.IdUsuario= @Idusuario)

begin

SET @PasoRegla=0

set @Respuesta=0

set @Mensaje=@Mensaje+'No se puede eliminar porque este usuario tiene operaciones de venta registradas.\n'

end

if (@PasoRegla = 1)

begin

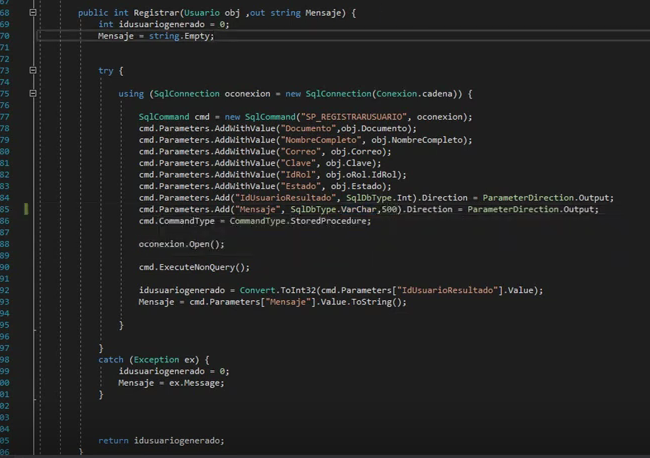
DELETE USUARIO where IdUsuario=@IdUsuario

SET @Respuesta=1

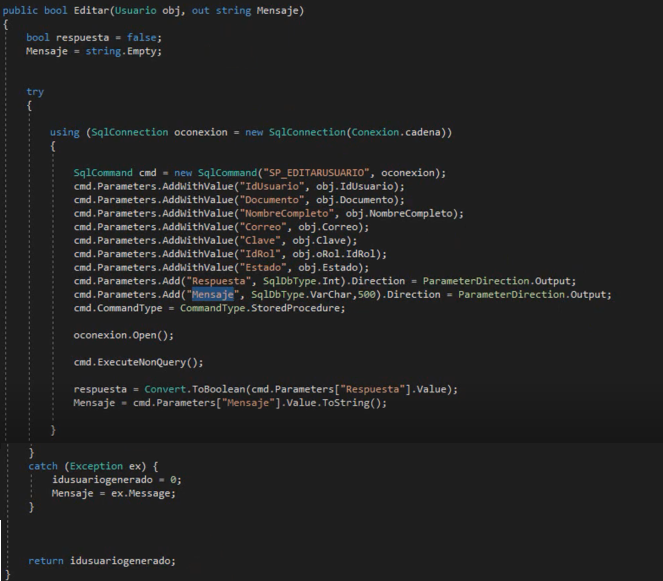
end

END

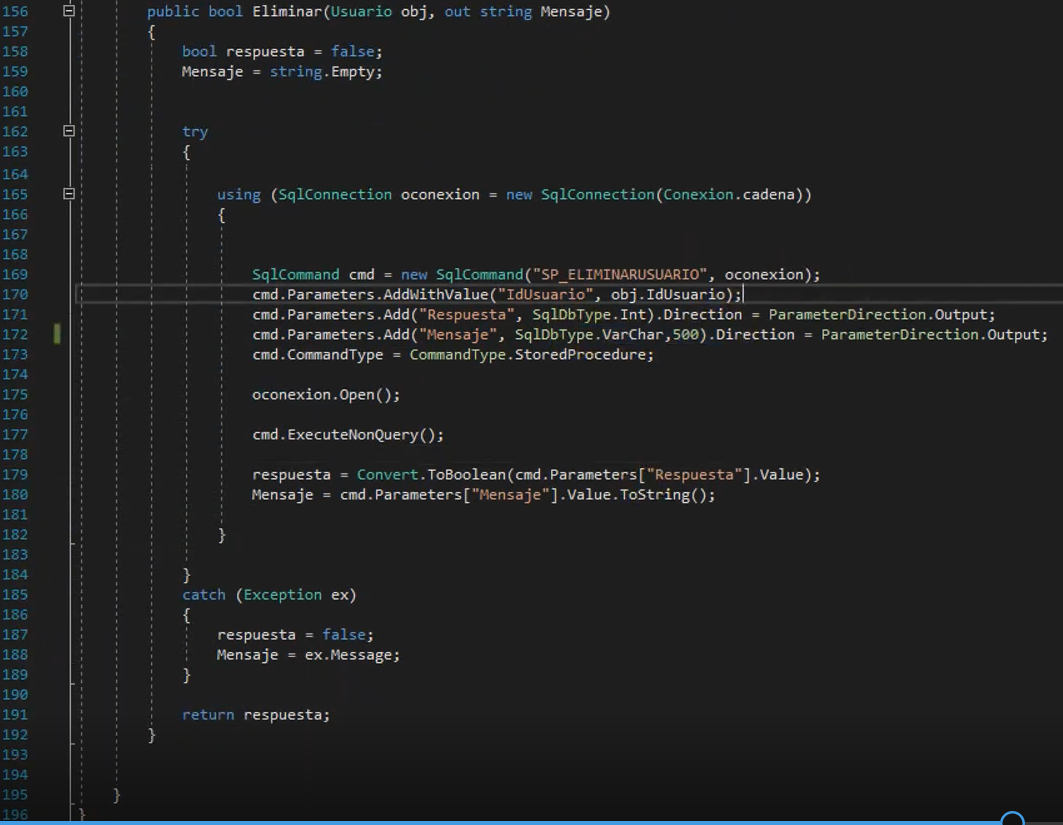
Para llamar a los procedimientos almacenados que tenemos en nuestra base de datos modificamos nuestra clase CD\_Usuario creando un nuevo método Registras que va a devolver un entero, el idusuario. Recibirá un objeto usuario y tendrá un parámetro de salida de igual forma que funciona en el procedimiento SQL.



También crearemos el método editar que nos va a devolver un booleno de respuesta y que ejecutará el procedimiento SPEDITARUSURIO.

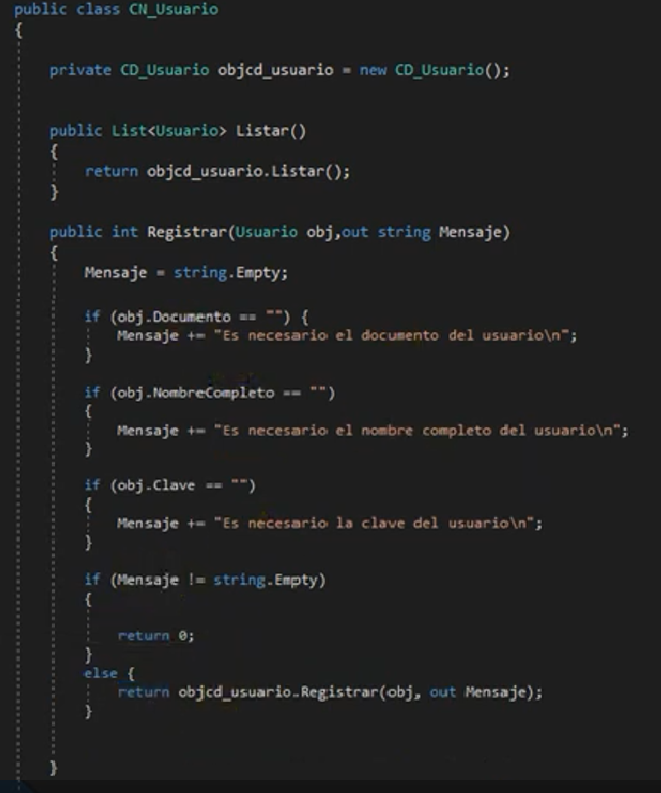


Y por último el método de eliminar usuario que igualmente devolverá una respuesta booleana

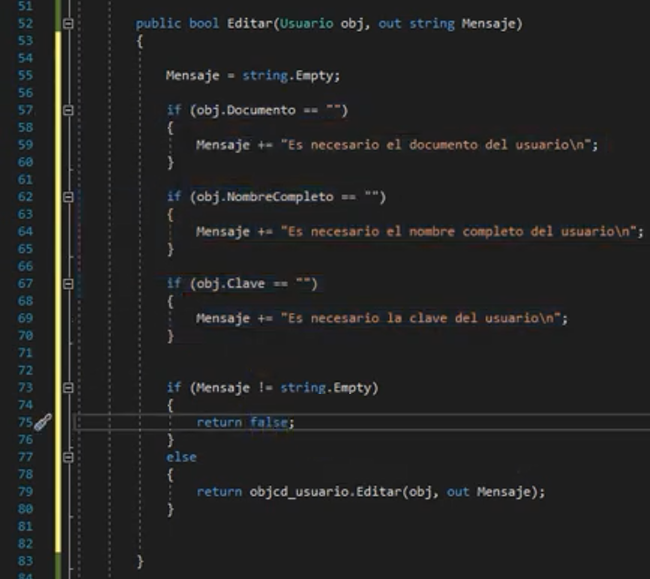


Ahora desde nuestra capa de negocio vamos a realizar el puente entre la capa de datos y la capa de presentación. La capa de negocio debe comprobar que se cumplen las reglas. Agregaremos algunas validaciones. También se pueden añadir en la capa de presentación pero en siguiendo el modelo por capas debemos agregar las reglas de negocio.

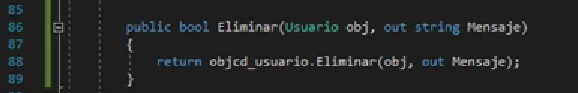
El método de registrar no debería ejecutarse si no ha pasado las reglas



De igual modo el método Editar realizará sus controles:



Por último, incorporamos el método de eliminar a nuestra capa de negocio:



Regresamos a nuestro formulario de usuario de la capa de presentación. En el evento de click del botón guardar eliminamos (o comentamos) lo que habíamos programado y generamos el código correcto con acceso a la bbdd. El botón guardar dará de alta un nuevo usuario si no existe y modificará un usuario si ya existe.

