SistemaVentas - Transacciones - Caso Factura / Detalle Factura

**## 1. Ejecutar la base de datos contenida en la carpeta database**

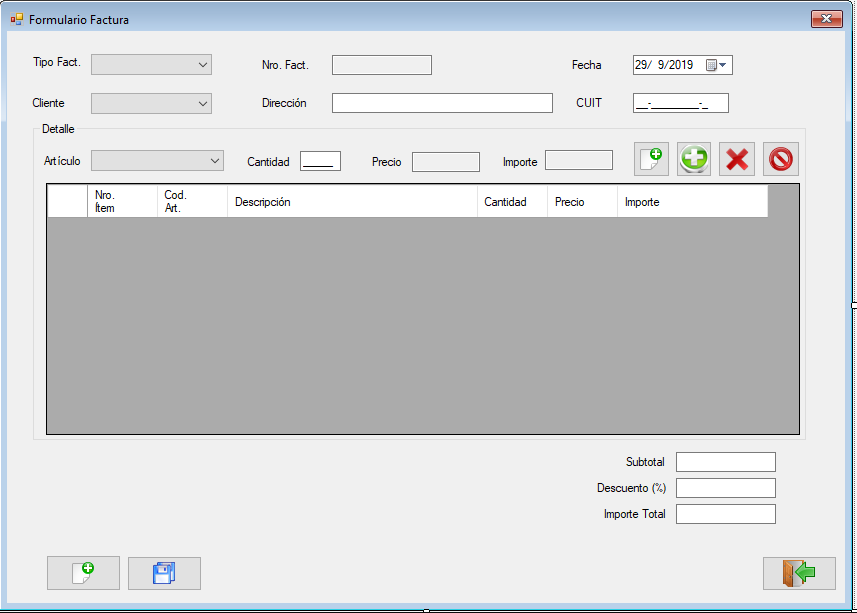
Cambiar el string de conexion

**## 2. Temas**

Iniciar la solución que se encuentra en la carpeta src/SistemaVentas.sln

Selecciones cada uno de los combos.

Se provee un ejemplo para actualizar bajo transacción una factura.



**\*\*Solución\*\***

Para llevar a cabo una transacción, siga estos pasos.

1. Llame al método BeginTransaction del objeto SqlConnection para marcar el comienzo de la transacción. El método BeginTransaction devuelve una referencia a la transacción. Esta referencia se asigna a los objetos SqlCommand que están inscritos en la transacción.

2. Asigne el objeto Transaction a la propiedad Transaction del objeto SqlCommand que se va a ejecutar. Si el comando se ejecuta en una conexión con una transacción activa y el objeto Transaction no se ha asignado a la propiedad Transaction del objeto Command, se inicia una excepción.

3. Ejecute los comandos necesarios.

4. Llame al método Commit del objeto SqlTransaction para completar la transacción, o al método Rollback para finalizarla. Si la conexión se cierra o elimina antes de que se hayan ejecutado los métodos Commit o Rollback, la transacción se revierte.

## 3. Versionar en GitHub los cambios locales (add / commit / push)

3.0. \*\*pull\*\* para traer los cambios de otro compañero.

> A continuación vamos a crear el \*\*commit\*\* y subir los cambios al servidor GitHub.

3.1. \*\*Status\*\*. Verificamos los cambios pendientes de versionar.

$ git status

3.2. \*\*Add\*\*. Agregamos todos los archivos nuevos no versionados.

$ git add -A

3.3. \*\*Commit\*\*. Generamos un commit con todos los cambios y agregamos un comentario.

3.4. \*\*Push\*\*. Enviamos todos los commits locales a GitHub

$ git push

3.5. \*\*Status\*\*. Verificar que no quedaron cambios pendientes de versionar

$ git status