Transacciones

El proyecto Transacciones permite ejecutar varias sentencias SQL dentro de una transacción.

Pasos:

1. Crear un archivo .txt donde se encuentren todas las sentencias SQL a ejecutar, en el orden deseado, **menos los triggers.** Los triggers se manejan de una forma diferente al resto y en el caso de querer crear triggers se deben incluir aparte en un TXT llamado ‘triggers.txt’.

Estos se ejecutarán siempre al final de todo. Se pueden crear varios archivos de consultas, pero solo un archivo de triggers (por el momento). El archivo de los triggers debe estar ubicado en la misma carpeta que el ‘Transactions.exe’

1. Crear un archivo llamado ‘listado.txt’ en la misma carpeta que el ‘Transactions.exe’. En este se debe ingresar la ruta de los txt de sentencias, **sin incluir el de los triggers.** Los diferentes archivos deben separarse por un salto de línea.
2. Comprobar el archivo ‘app.ini’.
3. Abrir ‘Transactions.exe’ y presionar el botón ‘Comenzar transacción’. El formulario se encontrará deshabilitado y el ícono del cursor se modificará a un reloj de arena.
4. En caso de que la transacción se haya ejecutado con éxito se mostrará un mensaje al usuario con dicha confirmación. En caso de ejecutarse con errores se mostrará un cartel notificando que hubo un error y que se revisen las exceptions.

Consideraciones a tener en cuenta al armar los scripts:

* En ningún momento las sentencias deben llevar el comando **GO.** Ver el ejemplo de los triggers del siguiente punto.
* Al armar los scripts de los triggers, estos deben encontrarse separados por un salto de línea, por lo tanto si se encontrara un salto de línea dentro de una sentencia, solamente para mejorar la lectura del código o lo que fuere, este generaría error.

Ej:

*CREATE TRIGGER nombre1 ON tabla AFTER insert AS BEGIN sentencias\_sin\_salto\_de\_línea END*

*CREATE TRIGGER nombre2 ON tabla AFTER insert AS BEGIN sentencias\_sin\_salto\_de\_línea END*

*CREATE TRIGGER nombre3 ON tabla AFTER insert AS BEGIN sentencias\_sin\_salto\_de\_línea END*

* Comprobar el orden en que se desean ejecutar los scripts de acuerdo al archivo ‘listado.txt’.

Ej:

*PATH\SENTENCIA1.txt*

*PATH\SENTENCIA2.txt*

Errores comunes:

* Error de Timeout: Este error se resuelve dividiendo el código de las sentencias en 2 o X necesarias partes. El archivo triggers.txt no genera este tipo de exceptions, por ende el error proviene de los otros archivos txt. Luego de haber dividido el código se deben agregar las nuevas partes al archivo ‘listado.txt’.

Ej:

Dentro del archivo listado.txt tengo 1 script llamado transacción.txt, que luego de ejecutar el programa generó ese tipo de excepción. Abro el archivo transacción, selecciono la mitad de sentencias (**no cortar al medio una sentencia,** ej: si tengo un ‘select \* from zopt’ no poner nunca una mitad ‘select \*’ y en la otra ‘from zopt’), copio esa primer mitad en un nuevo txt y la segunda mitad en otro. Luego le modifico el nombre de los nuevos txt a Transacción1.txt (primera mitad) y a Transacción2.txt (segunda mitad). El número no es obligatorio incluirlo pero facilita la comprensión del código. Luego dentro del archivo listado.txt eliminaré la ruta del archivo Transacciones.txt e incluiré primero el archivo ‘Transacciones1.txt’ y luego ‘Transacciones2.txt’ con sus correspondientes rutas y separados por un salto de línea. Finalmente abro Transacciones.exe y corro nuevamente la transacción. En caso de generar otra vez el error se debe realizar todos los pasos con los 2 archivos (Transacciones1.txt y Transacciones2.txt) ya que por el momento no se puede determinar cuál de los 2 generó el error.

* El archivo RUTA\_DE\_LISTADO.TXT no se encuentra en RUTA\_DE\_TRANSACCIONES.EXE: Comprobar que el archivo listado.txt exista en la ruta del ejecutable.
* El archivo RUTA\_DEL\_ARCHIVO no existe. Verifique su existencia y si se encuentra correctamente escrito en listado.txt: No queda mucho por explicar.