**Michelle Fernanda Anguiano Juárez – A01209529**

**Lab 23**

**Revisa el contenido de la tabla clientes\_banca desde la ventana que inicializaste como la segunda sesión.**

**SELECT** *\** **FROM** CLIENTES\_BANCA

* **¿Que pasa cuando deseas realizar esta consulta?**

Despliega el contenido de Clientes\_banca, el mismo de la sesión 1

**Te habrás percatado que falta el COMMIT TRANSACTION**

**Revisa el contenido de la tabla clientes\_banca desde la ventana que inicializaste como la primera sesión con la siguiente consulta.**

**SELECT** *\** **FROM** CLIENTES\_BANCA

**Revisa el contenido de la tabla clientes\_banca desde la ventana que inicializaste como la segunda sesión.**

**SELECT** *\** **FROM** CLIENTES\_BANCA

* **¿Qué pasa cuando deseas realizar esta consulta?**

Al consultar la primera sesión se queda esperando porque faltó el commit transaction. En la segunda sesión si despliega el contenido de la tabla.

**Intenta con la siguiente consulta desde la segunda sesión.**

**SELECT** *\** **FROM** CLIENTES\_BANCA **where NoCuenta**=**'001'**

* **Explica por qué ocurre dicho evento.**

Porque bloquea por registro, no por tabla

**Por último, regresa a la ventana que mantiene activa tu primera sesión, agrega el siguiente comando a la pantalla y ejecútalo.**

**ROLLBACK TRANSACTION** PRUEBA2

**Revisa nuevamente el contenido de la tabla clientes\_banca desde la ventana que inicializaste como la segunda sesión.**

**SELECT** *\** **FROM** CLIENTES\_BANCA

* **¿Qué ocurrió y por qué?**

Despliega solo los primeros registros porque deshizo la transacción de la segunda sesión.

**Posteriormente revisa si las tablas de clientes\_banca y movimientos sufrieron algún cambio, es decir, si dio de alta el registro que se describe en la transacción y su actualización.**

Sí se dio de alta el registro con éxito

* **¿Para qué sirve el comando @@ERROR revisa la ayuda en línea?**

Para cachar un error en caso de que exista alguno y determinar qué hacer. En este caso si no hay errores hace un commit a la transacción, si hay un error imprime un mensaje y cancela la transacción con rollback.

* **¿Explica qué hace la transacción?**

La transacción inserta tres registros

* **¿Hubo alguna modificación en la tabla? Explica qué pasó y por qué.**

No hubo ninguna modificación. Como ya existía un registro en Clientes\_Banca con la llave primaria 001, se detectó el error y se hizo un rollback a la transacción.

**Crea las siguientes transacciones dentro de un store procedure:**

* **Una transacción que registre el retiro de una cajero. nombre del store procedure REGISTRAR\_RETIRO\_CAJERO que recibe 2 parámetros en NoCuenta y el monto a retirar.**

**IF** *EXISTS* (  
 **SELECT name FROM** sysobjects  
 **WHERE name** = **'REGISTRAR\_RETIRO\_CAJERO' AND type** = **'P'**)  
 **DROP PROCEDURE** *REGISTRAR\_RETIRO\_CAJERO***GO  
CREATE PROCEDURE** *REGISTRAR\_RETIRO\_CAJERO* @NoCuenta **varchar**(5),  
 @Monto **numeric**(10,2)  
**AS  
 BEGIN TRANSACTION** Retiro  
 **INSERT INTO** Movimientos **VALUES**(@NoCuenta,**'A'**,*GETDATE*(),@Monto);  
 **UPDATE** CLIENTES\_BANCA **SET SALDO** = **SALDO** -@Monto  
 **WHERE NoCuenta**=@NoCuenta  
 **IF** *@@ERROR* = 0  
 **COMMIT TRANSACTION** Retiro  
 **ELSE  
 BEGIN  
 PRINT 'A transaction needs to be rolled back'  
 ROLLBACK TRANSACTION** Retiro  
 **END  
GO**

* **Una transacción que registre el deposito en ventanilla. Nombre del store procedure REGISTRAR\_DEPOSITO\_VENTANILLA que recibe 2 parámetros en NoCuenta y el monto a depositar.**

**IF** *EXISTS* (  
 **SELECT name FROM** sysobjects  
 **WHERE name** = **'REGISTRAR\_DEPOSITO\_CAJERO' AND type** = **'P'**)  
 **DROP PROCEDURE** *REGISTRAR\_DEPOSITO\_CAJERO***GO  
CREATE PROCEDURE** *REGISTRAR\_DEPOSITO\_CAJERO* @NoCuenta **varchar**(5),  
 @Monto **numeric**(10,2)  
**AS  
 BEGIN TRANSACTION** Deposito  
 **INSERT INTO** Movimientos **VALUES**(@NoCuenta,**'B'**,*GETDATE*(),@Monto);  
 **UPDATE** CLIENTES\_BANCA **SET SALDO** = **SALDO** +@Monto  
 **WHERE NoCuenta**=@NoCuenta  
 **IF** *@@ERROR* = 0  
 **COMMIT TRANSACTION** Deposito  
 **ELSE  
 BEGIN  
 PRINT 'A transaction needs to be rolled back'  
 ROLLBACK TRANSACTION** Deposito  
 **END  
GO**