**SELECT \* FROM CLIENTES\_BANCA  
  
¿Que pasa cuando deseas realizar esta consulta?**

Muestra todos los registros de la table.

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

En la primera sesión si permite realizar la consulta, por lo cual muestra los registros, inluyendo estos últimos.

En la segunda sesión al hacer falta en la transacción el COMMIT como tal no ha terminado, por lo cual no permite que en la otra sesión se pueda ejecutar la consulta.

**Explica por qué ocurre dicho evento.**

"001" "Manuel Rios Maldonado" 9000.00

El sistema si me regreso el registro con esa clave.

**¿Qué ocurrió y por qué?**

Solo me mostró los que se registrarón al inicio, debido a que hicimos un rollback, es decir no se realizó ninguna consulta de la transacción.

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

Para verificar el número de error, es decir el 0 ES una ejecución exitosa, distinto a 0, sería algún timpo de error  
  
**¿Qué hace la transacción?**

Inserta nuevos clientes, después verifica si se realizo exitosamente, si es así realiza el commit de caso contrario realiza un ROLLBACK.  
  
**¿Hubo alguna modificación en la tabla? Explica qué pasó y por qué.**

No, ya que el tercer registro, ya tiene esa numero de cuenta, no puede realizarce ya que viola la llave primaria, por lo cual hay un error, por lo tanto aunque las dos anteriores esté correctas estás las va a revertir.

**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.**

CREATE PROCEDURE REGISTRAR\_RETIRO\_CAJERO

@noCuenta varchar(5),

@monto numeric(10,2)

AS

BEGIN TRANSACTION PRUEBA1

INSERT INTO OperacionesBancarias VALUES(@noCuenta, 'A', GETDATE(), @monto);

UPDATE ClientesBanca SET saldo = saldo - @monto WHERE NoCuenta = @noCuenta;

IF @@ERROR = 0

COMMIT TRANSACTION PRUEBA1

ELSE

BEGIN

PRINT 'A transaction needs to be rolled back'

ROLLBACK TRANSACTION PRUEBA1

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.**

CREATE PROCEDURE REGISTRAR\_DEPOSITO\_VENTANILLA @uNoCuenta varchar(5),

@uMonto numeric(10,2)

AS

BEGIN TRANSACTION DEPOSITO  
INSERT INTO OperacionesBancarias VALUES (@uNoCuenta,’B’,GETDATE(),@uMonto);

UPDATE ClientesBanca SET SALDO = SALDO + @uMonto;

IF @@ERROR = 0  
COMMIT TRANSACTION DEPOSITO  
ELSE  
BEGIN  
PRINT 'A transaction needs to be rolled back'  
ROLLBACK TRANSACTION PRUEBA5  
END

GO