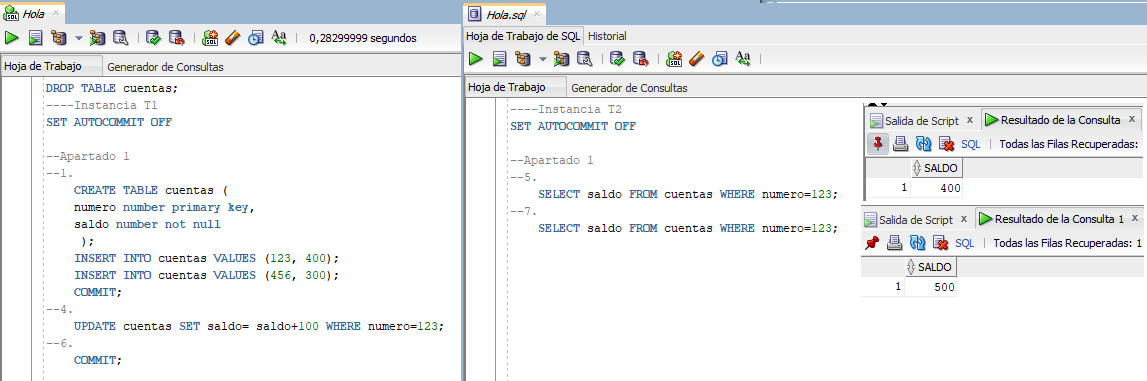
**Práctica 6: Transacciones**

Juan Carlos Llamas Núñez 3º DG

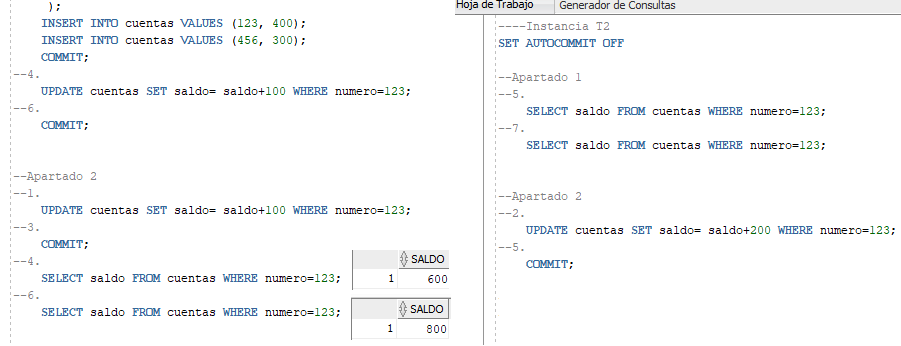
**Apartado 1. Bloqueos (Select)**

Tras ejecutar la quinta instrucción obtenemos que el saldo de la cuenta 123 es 400 y tras la ejecución de la instrucción 7 el saldo es 500. Esto sucede porque hasta que no termina la transacción de T1 no queda registrado la actualización.



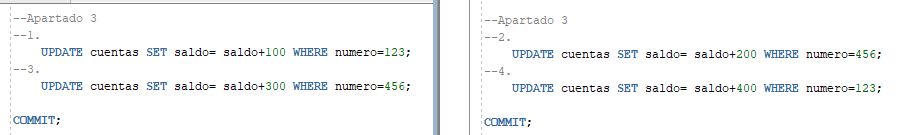
**Apartado 2. Bloqueos (Update)**

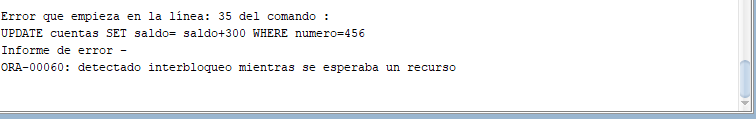
Tras ejecutar la segunda instrucción el programa deja de responder y no ejecuta dicha acción. Eso sucedo porque hay una transacción que está accediendo a ese mismo recurso. Cuando ejecutamos la tercera instrucción y cerramos la primera transacción desde la segunda transacción se ejecuta el proceso que había quedado pendiente. Al ejecutar la cuarta instrucción obtenemos un saldo de 600 y tras confirmar la transacción en la instrucción 5 se tiene un saldo de 800 en la consulta de la instrucción 6.



**Apartado 3. Bloqueos (Deadlock)**

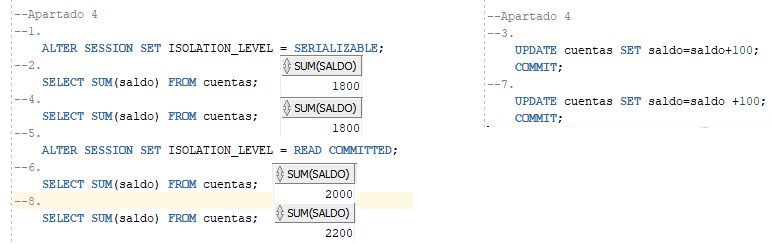
En la instrucción 3 se bloquea la transacción 1 porque T2 ya está accediendo a ese recurso y en la instrucción 4 se bloquea la transacción 2 porque la transacción 1 ya está accediendo a ese recurso. Tras un breve instante de tiempo, en la transacción 1 aparece un mensaje de detección de interbloqueo.





**Apartado 4. Niveles de aislamiento**

Podemos observar en este apartado la diferencia entre el nivel de aislamiento READ COMITTED y SERIALIZABLE. Cuando seleccionamos el nivel SERIALIZABLE se produce un aislamiento completo de las transacciones y no se permite a otras transacciones insertar, actualizar o borrar datos que estamos utilizando en nuestra transacción. De esta manera se limita el acceso concurrente a recursos compartidos. En la segunda instrucción obtenemos una suma de saldos (saldo a partir de ahora) de 1800. En la cuarta instrucción el saldo sigue siendo de 1800 ya que, como hemos cambiado el nivel de aislamiento a SERIALIZABLE, la transacción de la instancia T2 no se ejecutará hasta que se comprometa la transacción que sigue abierta. En la instrucción 6 obtenemos un saldo de 2000 ya que, tras cambiar el nivel de aislamiento a READ COMITTED, han quedado recogidos los cambios de la transacción de la instancia 2. Por último, tras ejecutar la instrucción 8 el saldo es de 2200 porque ha quedado recogida la actualización hecha en la transacción de la instancia T2.



Solamente me ha dado tiempo a llegar hasta aquí porque he tenido problemas con la conexión de la base de datos después de que aparentemente se hubieran solucionado y todo fuera bien durante un buen rato. Independientemente, los apretados que he realizado me han servido para comprender mejor como trabaja SQL con las transacciones.

A continuación, pego el texto plano del código de ambas instancias:

DROP TABLE cuentas;

----Instancia T1

SET AUTOCOMMIT OFF

--Apartado 1

--1.

CREATE TABLE cuentas (

numero number primary key,

saldo number not null

);

INSERT INTO cuentas VALUES (123, 400);

INSERT INTO cuentas VALUES (456, 300);

COMMIT;

--4.

UPDATE cuentas SET saldo= saldo+100 WHERE numero=123;

--6.

COMMIT;

--Apartado 2

--1.

UPDATE cuentas SET saldo= saldo+100 WHERE numero=123;

--3.

COMMIT;

--4.

SELECT saldo FROM cuentas WHERE numero=123;

--6.

SELECT saldo FROM cuentas WHERE numero=123;

--Apartado 3

--1.

UPDATE cuentas SET saldo= saldo+100 WHERE numero=123;

--3.

UPDATE cuentas SET saldo= saldo+300 WHERE numero=456;

COMMIT;

--Apartado 4

--1.

ALTER SESSION SET ISOLATION\_LEVEL = SERIALIZABLE;

--2.

SELECT SUM(saldo) FROM cuentas;

--4.

SELECT SUM(saldo) FROM cuentas;

--5.

ALTER SESSION SET ISOLATION\_LEVEL = READ COMMITTED;

--6.

SELECT SUM(saldo) FROM cuentas;

--8.

SELECT SUM(saldo) FROM cuentas;

----Instancia T2

SET AUTOCOMMIT OFF

--Apartado 1

--5.

SELECT saldo FROM cuentas WHERE numero=123;

--7.

SELECT saldo FROM cuentas WHERE numero=123;

--Apartado 2

--2.

UPDATE cuentas SET saldo= saldo+200 WHERE numero=123;

--5.

COMMIT;

--Apartado 3

--2.

UPDATE cuentas SET saldo= saldo+200 WHERE numero=456;

--4.

UPDATE cuentas SET saldo= saldo+400 WHERE numero=123;

COMMIT;

--Apartado 4

--3.

UPDATE cuentas SET saldo=saldo+100;

COMMIT;

--7.

UPDATE cuentas SET saldo=saldo +100;

COMMIT;