

INSTRUCCIONES

- Ejecuta antes el script BDejemplo.sql para empezar con la BD limpia
- Abre el SqlDeveloper una vez (lo llamamos Transacción SESION 1)
- Abre el SqlDeveloper otra vez (lo llamamos Transacción SESION2)
- Ejecuta **en el orden indicado** cada instrucción
- Estudia el resultado y el cambio en la BD

SESION 1	SESION 2
===== PROBAR PROPIEDADES	ACID =====
-- ESCENARIO - dos sesiones: -- usan el aislamiento por defecto: ¿cual?	
commit ; set autocommit off	
	Commit; set autocommit off
select * from cliente where DNI = '00000001';	
SELECT dbms_transaction.local_transaction_id FROM dual;	
	SELECT dbms_transaction.local_transaction_id FROM dual;
update cliente set nombrec = 'transac-1' where DNI = '00000001';	
SELECT dbms_transaction.local_transaction_id FROM dual;	
	select * from cliente where DNI = '00000001';
	SELECT dbms_transaction.local_transaction_id FROM dual;
select * from cliente where DNI = '00000001';	
	update cliente set nombrec = 'trans---2' where DNI = '00000001'; ➔ Se queda esperando
commit	
	➔ Actualiza la fila
	select * from cliente where DNI = '00000001';
select * from cliente where DNI = '00000001';	
SELECT dbms_transaction.local_transaction_id FROM dual;	
	commit
select * from cliente where DNI = '00000001';	
	select * from cliente

	-- permite: esta fila no está bloqueada select saldo from TieneT where dni = '00000004' and numt = '30000300'; -- devuelve el valor actualizado: 40299
-- paso 5 -- session 1 update TieneT set saldo = saldo - 1 where dni = '00000003' and numt ='30000002'; -- permite: la bloqueó esta sesión select saldo from TieneT where dni = '00000003' and numt ='30000002'; -- devuelva el valor actualizado por ella: 29	
	-- paso 6 -- session 2 select * from compras where dni= '00000003' and numt = '30000002'; -- filas: la inicial y la añadida por esta sesión 00000003 30000002 1 501 tienda7 3 00000003 30000002 200 501 sesion2:una 1
-- paso 7 -- session 1 INSERT INTO Compras VALUES ('00000003', '30000002',111, 0501,'sesion1:??',1); -- Error SQL: ORA-08177: no se puede serializar el acceso para esta transacción -- porque está "serializable" y -- sesion 2 actualizó antes COMPRAS eso bloquea el bloque entero afectado select * from compras where dni= '00000003' and numt = '30000002'; -- responde , pero no ve la fila nueva de la sesion 2 00000003 30000002 1 501 tienda7 3	
	---- paso 8 -- session 2 update TieneT set saldo = saldo - 1 where dni = '00000005' and numt ='50000030'; -- fila que existía antes de empezar las dos sesiones 00000005 50000030 901 500 -- da ORA-08177: no se puede serializar el acceso para esta transacción CAUSA: por se serializable (no por bloqueada)

	Commit Borrar y crear BDEjemplo - da error de tabla Cliente (y otras) ORA-00054: recurso ocupado y obtenido con NOWAIT especificado o timeout vencido → la otra trans. tiene bloqueada tabla con claves ajenas en Cliente
commit	
	Borrar y crear BDEjemplo - lo ejecuta bien
-- ESCENARIO: dos sesiones serializables: -- sesión 1 bloquea SOLO una fila con select .. TieneT ... for update -- sesion 2 bloquea SOLO otra fila select .. TieneT ... for update set transaction isolation level serializable;	
	Commit set transaction isolation level serializable;
- paso 1 -- session 1 select saldo from TieneT where dni = '00000003' and numt ='30000002' for update; -- solo bloquea esa fila = 30	
	- paso 2 -- session 2 select saldo from TieneT where dni = '00000003' and numt ='30000002' for update; -- se queda esperando (cancelar a mano) select saldo from TieneT where dni = '00000005' and numt = '50000030' for update; -- da 500 -- solo está bloqueada la otra fila en sesion1 update TieneT set saldo = saldo - 1 where dni = '00000005' and numt ='50000030'; -- queda saldo= 499 -- deja actualizar porque está bloqueada otra fila en la sesión 1 -- la fila seleccionada con select . . . for update
update TieneT set saldo = saldo - 1 where dni = '00000003' and numt ='30000002'; -- ejecuta: 1 filas actualizadas -> da 29	

<pre> select saldo from TieneT where dni = '00000005' and numt = '50000030'; -- ejecuta, pero da 500 (valor inicial) update TieneT set saldo = saldo - 1 where dni = '00000005' and numt = '50000030'; -- se queda esperando, está bloqueada en sesion2 </pre>	
	commit
<pre> Error SQL: ORA-08177: no se puede serializar el acceso para esta transacción: -- por ser serializable no permite actualizar, Aunque ha liberado la fila la sesion2 -- no aborta la trans select saldo from TieneT where dni = '00000005' and numt = '50000030'; -- sigue viendo valor inicial 500 commit </pre>	
<pre> -- ESCENARIO: dos sesiones read committed: -- sesión 1 bloquea SOLO una fila con select .. TieneT ... for update -- sesión 2 bloquea SOLO otra fila select .. TieneT ... for update commit; set transaction isolation level read committed; </pre>	
	<p>Borrar y crear BDEjemplo - lo ejecuta bien</p> <p>commit set transaction isolation level read committed;</p>
<pre> - paso 1 -- session 1 select saldo from TieneT where dni = '00000003' and numt = '30000002' for update; -- solo bloquea esa fila = 30 </pre>	
	<pre> - paso 2 -- session 2 select saldo from TieneT where dni = '00000005' and numt = '50000030' for update; -- da 500 -- deja actualizar porque está bloqueada otra fila en la sesión 1 -- la fila seleccionada con select ... for update update TieneT </pre>

	set saldo = saldo - 1 where dni = '00000005' and numt = '50000030'; -- da 499
select saldo from TieneT where dni = '00000005' and numt = '50000030'; -- da 500	
	commit
select saldo from TieneT where dni = '00000005' and numt = '50000030'; -- da 499 sin salir de su trans, por ser read committed y la sesion2 ha hecho commit	
	select saldo from TieneT where dni = '00000003' and numt = '30000002' for update; -- espera (fila bloqueada en session1)
commit	
	-- ejecuta y da el resultado= 30 Commit
-- ESCENARIO: dos sesiones read committed: -- sesión 1 bloquea tabla exclusive Lock table Tienet in exclusive mode select . . TieneT . . . for update -- sesion 2 bloquea SOLO otra fila select . . TieneT . . . for update set transaction isolation level read committed;	
	set transaction isolation level read committed;
Lock table Tienet in exclusive mode;	
	update TieneT set saldo = saldo - 1 where dni = '00000005' and numt = '50000030'; -- se queda esperando (cancelo a mano) Lock table Tienet in share mode; -- se queda esperando (cancelo a mano)
update TieneT set saldo = saldo - 1 where dni = '00000003' and numt = '30000002'; -- actualizado Commit;	

-- termina transacción -- empieza otra	
	update Tienet set saldo = saldo - 1 where dni = '00000005' and numt = '50000030'; -- ejecuta : actualiza: 499 Lock table Tienet in share mode;
Lock table Tienet in share mode; -- Se queda esperando (cancelo a mano) porque sesion2 hizo un update que bloque la tabla	
	update Tienet set saldo = saldo - 1 where dni = '00000003' and numt = '30000002'; -- ejecuta : 28 Commit -- termina transacción -- empieza otra Lock table Tienet in share mode;
Lock table Tienet in share mode; -- ejecuta el lock Lock table Tienet in exclusive mode; -- se queda esperando	