



# Concurrencia y recuperabilidad

Se está realizando una prueba de ABD a los alumnos de un curso. Durante la realización de la misma, uno de los profesores que se encarga de impartir la asignatura en las titulaciones de grado mira por encima del hombro de un alumno y ve que ha insertado una tupla en una tabla RESULTADOS para responder una pregunta. La solución propuesta le resulta curiosa y decide estudiarla con detenimiento. El examen está tranquilo, no hay apenas preguntas por parte de los alumnos, por lo que se va a su ordenador y ejecuta la sentencia SELECT correspondiente para ver la respuesta que ha dado ese alumno a esa respuesta.

Perplejo, observa que la sentencia SELECT no devuelve ningún resultado cuando él mismo ha visto cómo se hacía el insert con éxito. Mira hacia el puesto donde está el alumno en cuestión y ve que está leyendo otra pregunta, no escribe nada en su ordenador. El profesor vuelve a ejecutar la misma sentencia SELECT y ahora sí que sale...



1. ¿A que se debe que la primera vez no saliera nada? ¿Qué ocurrió para que después sí que saliera?

A que el alumno no había realizado un commit.



2. ¿Cómo se llama el hecho de que la ejecución de 2 sentencias SELECT iguales devuelvan resultados distintos en la misma transacción?

Hacer una lectura errónea



3. ¿Es posible evitar esto último? Es decir, ¿existe una forma de asegurarnos de que 1 sentencia ejecutada 2 veces en la misma transacción devuelva exactamente el mismo valor?

Si, realizando un commit antes de hacer la consulta.

El profesor decide bloquear la tabla de resultados para que ningún alumno pueda insertar nuevos datos mientras resuelve el misterio.



4. ¿Es posible bloquear una tabla entera para que no se puedan insertar datos? ¿Cómo se llama ese tipo de bloqueo? Escriba una sentencia que bloquee una tabla denominada RESULTADO (creada previamente con los campos que desee) para que ningún usuario pueda modificarla. ¿Cuándo y cómo se desbloquea?

Si. Se llama bloqueo de tabla.

```
LOCK TABLE RESULTADO IN EXCLUSIVE MODE;
```

Se desbloquea realizando un rollback o un commit.

Como el profesor no da con la solución decide auditar lo que está ocurriendo en la base de datos.



5. ¿Cual es la instrucción para auditar las inserciones en la tabla RESULTADO? ¿Cómo puede el profesor obtener los resultados de la auditoría?

```
AUDIT INSERT TABLE ON RESULTADOS;
```

```
SELECT * FROM DBA_OBJ_AUDIT_OPTS WHERE OBJECT_NAME = 'RESULTADO';
```

Para hacer una prueba de la auditoría, el profesor inserta una fila en la tabla resultados. Cuando va a borrarla se equivoca y no pone clausula WHERE en la instrucción DELETE, por lo que borra todas las filas de la tabla. Nervioso, se dispone a hacer un rollback, pero como está usando el SQL Developer se confunde de botón y pulsa COMMIT. No solo ha borrado las respuestas a la prueba actual, sino ¡todas las respuestas de todos los alumnos!.

Mientras una gota fría recorre lentamente su cara, recuerda la tecnología flashback.



6. ¿Qué opciones hay para recuperar la información perdida utilizando la tecnología flashback? ¿Qué requisitos se deben cumplir? Y si en lugar de borrar todas las filas de la tabla hubiera hecho un DROP TABLE, ¿habría solución?

Se pueden recuperar los datos cambiados de la tabla.

Hay que tener la tecnología activada, configurando un fast recovery area y poner la BD en modo ARCHIVELOG. Y luego activar FLASHBACK DATABASE: ALTER DATABASE FLASHBACK ON;

Si, flashback también permite recuperar tablas borradas.

Cuando termina la prueba decide hacer una copia de seguridad de los datos del esquema.



## 7. ¿Qué opciones de backup tiene? ¿Cuál sería el más sencillo?

Tiene tres opciones:

- RMAN → backup físico
- Backup manual → backup físico
- Utilizando Data Pump Export → backup lógico

El más sencillo sería RMAN, debido a que hacerlo manual se tardaría más y para utilizar Data Pump se requiere utilizar Data Pump Import.