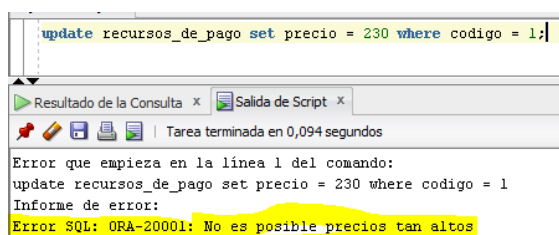


Sesión 14 – Ejercicios de repaso

1.- Crea una función TIENENPORTALES que reciba el nombre de un portal y devuelva el número de versiones que los usuarios tienen descargadas de ese portal (es decir, contar el número de registros de la tabla TIENEN para ese portal). Si no existe ninguno devolverá -1. Si el portal no existe devolverá -99.

2.- Crea un procedimiento almacenado MULTIPLICAR_PRECIOS que reciba por parámetro el código de un recurso, cuente cuantos accesos ha tenido (tabla CONTROL_ACCESO) y si ha tenido más de 2, multiplique por dos su precio en la tabla RECURSO_PAGO. No es necesario controlar si el recurso existe o no en la BD o si es de pago o no. Recuerda que un UPDATE que no hace nada no provoca excepción.

3.- Crea un disparador que impida mediante un "raise" que se puedan tener recursos de pago en la BD con precios superiores a 200 (al insertar o al modificar). Además, ese mismo disparador debe asegurarse que si se intenta insertar o modificar un precio de un recurso por debajo de 10, en la BD quedará el precio como 10.



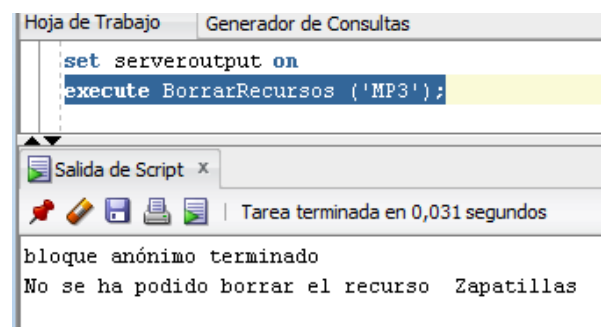
```
update recursos_de_pago set precio = 230 where codigo = 1;
```

Resultado de la Consulta x Salida de Script x

Tarea terminada en 0,094 segundos

Error que empieza en la línea 1 del comando:
update recursos_de_pago set precio = 230 where codigo = 1
Informe de error:
Error SQL: ORA-20001: No es posible precios tan altos

4.- Crear un procedimiento almacenado BORRARRECURSOS que reciba el nombre de un formato, abra un cursor con todos los recursos de pago de ese formato, y para cada uno de ellos intente borrarlos de la tabla RECURSO_PAGO. Si se produce cualquier error (por ejemplo, que el recurso tenga accesos, etc.) se sacará un mensaje indicando "no se ha podido borrar el recurso nombre_recurso" y la ejecución no se detendrá.



Hoja de Trabajo Generador de Consultas

```
set serveroutput on  
execute BorrarRecursos ('MP3');
```

Salida de Script x

Tarea terminada en 0,031 segundos

bloque anónimo terminado
No se ha podido borrar el recurso Zapatillas

5.- Crea un disparador que evite (provocando un raise) que se borren imágenes con fecha de modificación anterior a 01/05/2019 (mira en los PDF de teoría cual es la función que devuelve el updatetime) y que evite también (provocando otro raise) que se pueda cambiar el mimetype de una imagen que tenga mime image/png por image/jpeg.

6.- Crea una función que reciba como parámetro un mimeType y nos devuelva la suma en Ks de las imágenes de ese mimeType.

