**EXCEPCIONES:**

1. Crear una SELECT (no un cursor explícito) que devuelva el nombre de un

empleado pasándole el EMPLOYEE\_ID en el WHERE,

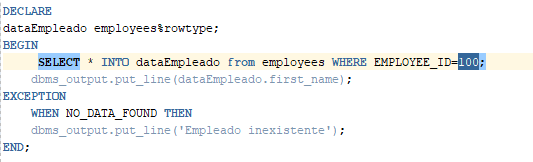
• Comprobar en primer lugar que funciona pasando un empleado

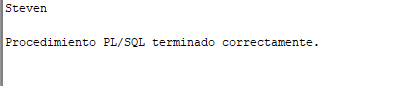
existente

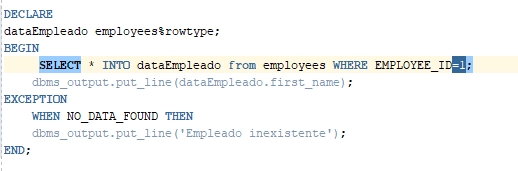
• Pasar un empleado inexistente y comprobar que genera un error

• Crear una zona de EXCEPTION controlando el NO\_DATA\_FOUND

para que pinte un mensaje como “Empleado inexistente”







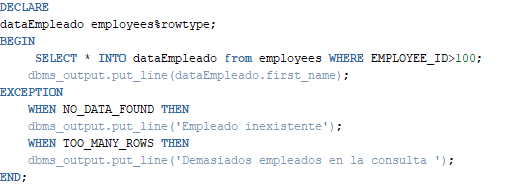


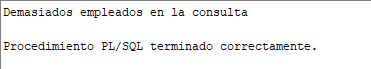
1. Modificar la SELECT para que devuelva más de un empleado, por ejemplo

poniendo EMPLOYEE\_ID> 100. Debe generar un error. Controlar la

excepción para que genere un mensaje como “Demasiados empleados

en la consulta”





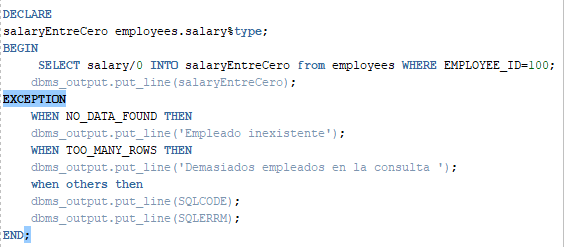
1. Modificar la consulta para que devuelva un error de división por CERO,

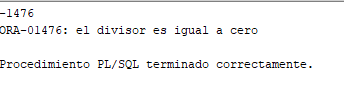
por ejemplo, vamos a devolver el salario en vez del nombre y lo dividimos

por 0. En este caso, en vez de controlar la excepción directamente,

creamos una sección WHEN OTHERS y pintamos el error con SQLCODE

y SQLERR





1. El error -00001 es clave primaria duplicada.
2. Aunque ya existe una predefinida (DUP\_VAL\_ON\_INDEX) vamos

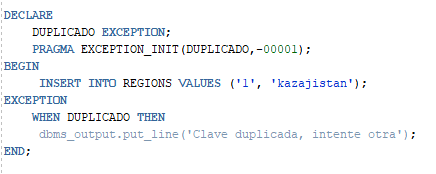
a crear una excepción no -predefinida que haga lo mismo.

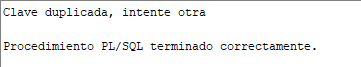
1. Vamos a usar la tabla REGIONS para hacerlo más fácil
2. Usamos PRAGMA EXCEPTION\_INIT y creamos una excepción

denominada “duplicado”.

1. Cuando se genere ese error debemos pintar “Clave duplicada,

intente otra”.





**Práctica con EXCEPCIONES DE USUARIO**

1- Crear una Excepción personalizada denominada

CONTROL\_REGIONES.

• Debe dispararse cuando al insertar o modificar una región

queramos poner una clave superior a 200. Por ejemplo usando una

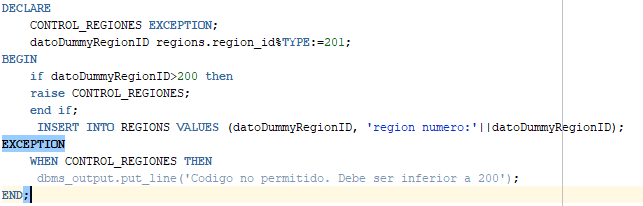
variable con ese valor.

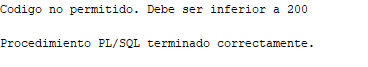
• En ese caso debe generar un texto indicando algo así como

“Codigo no permitido. Debe ser inferior a 200”

• Recordemos que las excepciones personalizadas deben

dispararse de forma manual con el RAISE.





**Práctica con RAISE\_APPLICATION\_ERROR**

1. Modificar la practica anterior para disparar un error con RAISE\_APPLICATION

en vez de con PUT\_LINE.

a. Esto permite que la aplicación pueda capturar y gestionar el error que

devuelve el PL/SQL

