| ***2º ASIR: ASGBD*** | **TAREA 04*:* TEMA 2 Acceso a la Información**  **EJERCICIOS DISPARADORES** | *Curso 2022-23*  *1ª Ev* |
| --- | --- | --- |
| *Nombre:* ***Marco Batista Calado*** | |

**Con las tablas EMPLE y DEPART**

1. Disparador para poner en mayúsculas el nombre y la localidad de un departamento cuando se dé de alta un nuevo departamento o se modifique.

**CREATE OR REPLACE TRIGGER EJ01**

**BEFORE INSERT OR UPDATE**

**ON DEPART**

**FOR EACH ROW**

**BEGIN**

**:NEW.DNOMBRE := UPPER(:NEW.DNOMBRE);**

**:NEW.LOC := UPPER(:NEW.LOC);**

**END;**

1. Se quiere mantener un histórico de los empleados borrados con: apellido del empleado borrado, fecha y hora y el usuario que lo borra. Crear para ello una tabla Historico\_Borrado\_Emple y realizar el trigger correspondiente.

**CREATE TABLE HISTORICO\_BORRADO\_EMPLE(**

**NOMBRE VARCHAR2(30),**

**FECHA DATE,**

**USUARIO VARCHAR2(30)**

**);**

**CREATE OR REPLACE TRIGGER EJ02**

**BEFORE DELETE ->aqui mejor after**

**ON EMPLE**

**FOR EACH ROW**

**BEGIN**

**INSERT INTO HISTORICO\_BORRADO\_EMPLE VALUES(:OLD.APELLIDO, SYSDATE,USER);**

**END;**

**PROBARLO:**

**DELETE EMPLE WHERE APELLIDO='SALA';**

**SELECT \* FROM HISTORICO\_BORRADO\_EMPLE;**

1. Escribir un disparador de base de datos que haga fallar cualquier operación de modificación del apellido o del número de un empleado.

**CREATE OR REPLACE TRIGGER EJ03**

**BEFORE UPDATE**

**ON EMPLE**

**FOR EACH ROW**

**BEGIN**

**IF UPDATING ('APELLIDO') THEN**

**RAISE\_APPLICATION\_ERROR(-20000, 'No se puede modificar apellido');**

**END IF;**

**IF UPDATING('DEPT\_NO') THEN**

**RAISE\_APPLICATION\_ERROR(-20000, 'No se puede modificar numero de empleado');**

**END IF;**

**END;**

***También podemos hacerlo con :NEW.APELLIDO <> :OLD.APELLIDO, que es más correcto porque te cercioras de que realmente se ha cambiado porque con updating igual lo que han actualizado es igual a lo anterior.***

1. Disparador que anula cualquier transacción que suponga una bajada de salario superior al 30%.

**CREATE OR REPLACE TRIGGER EJ04**

**BEFORE UPDATE**

**ON EMPLE**

**FOR EACH ROW**

**DECLARE**

**BAJADA NUMBER;**

**BEGIN**

**BAJADA:= :OLD.SALARIO\*0.3;**

**BAJADA:=:OLD.SALARIO-BAJADA;**

**IF :NEW.SALARIO<BAJADA THEN**

**RAISE\_APPLICATION\_ERROR(-20000, 'EL SALARIO ES DE UNA BAJADA MAYOR AL 30%');**

**END IF;**

**END;**

**Con las tablas de la Aplicación Clínica Veterinaria**

| **ANIMALES**  Ident  Nombre  Especie  Raza  Fecha Nacimiento  Peso  Sexo  Dni\_Dueño | **DUEÑOS**  DNI  Nombre  Direccion  Tfno Contacto  Alta\_Clinica  Cuota\_Mensual | **VISITAS**  Ident\_Animal  FH\_Visita  NumColegiado  Motivo  Diagnóstico  Precio | **VETERINARIOS**  NumColegiado  Nombre  Telefono |
| --- | --- | --- | --- |

Implementar los siguientes disparadores:

1. Al borrar un animal comprobar si tiene visitas anteriores a la fecha del sistema, en ese caso se muestra un error indicando que no se puede borrar.

**CREATE OR REPLACE TRIGGER EJ05**

**BEFORE DELETE**

**ON ANIMALES**

**FOR EACH ROW**

**DECLARE**

**CONTAR NUMBER;**

**BEGIN**

**SELECT COUNT(\*) INTO CONTAR FROM VISITAS WHERE IDENT\_ANIMAL=:OLD.IDENT\_ANIMAL AND FH\_VISITA < SYSDATE;**

**IF CONTAR >= 1 THEN**

**RAISE\_APPLICATION\_ERROR(-20000, 'No se puede borrar');**

**END IF;**

**END;**

NEW=OLD SI QUISIESEMOS PONERLO CON UN VER

1. Al dar de alta una visita o modificar el precio de una visita, si el dueño del animal tiene cuota mensual, cambiar el precio por el precio rebajado en un 10%.

**CREATE OR REPLACE TRIGGER EJ06**

**BEFORE INSERT OR UPDATE**

**ON VISITAS**

**FOR EACH ROW**

**DECLARE**

**CUANTOS NUMBER;**

**REBAJA NUMBER;**

**BEGIN**

**SELECT COUNT(\*) INTO CUANTOS**

**FROM DUEÑOS JOIN ANIMALES ON ANIMALES.DNI\_DUEÑO = DUEÑOS.DNI**

**WHERE CUOTA\_MENSUAL LIKE 'S' AND IDENT\_ANIMAL = :NEW.IDENT\_ANIMAL;**

**IF CUANTOS >= 1 THEN**

**REBAJA:= :NEW.PRECIO\*0.1;**

**REBAJA:= :NEW.PRECIO - REBAJA;**

**:NEW.PRECIO:=REBAJA;**

**END IF;**

**END;**

1. Cuando se da de alta un nuevo Dueño: si la dirección es nula poner en dirección el texto "Rellenar dirección".

**CREATE OR REPLACE TRIGGER EJ07**

**BEFORE INSERT**

**ON DUEÑOS**

**FOR EACH ROW**

**BEGIN**

**IF :NEW.DIRECCION IS NULL THEN**

**:NEW.DIRECCION:='Rellenar la dirección';**

**END IF;**

**END;**

1. Solo el usuario USUASIR puede dar de alta o borrar o modificar VETERINARIOS. **(DESDE SYSTEM, PQ TIENE LOS PERMISOS).**

**CREATE OR REPLACE TRIGGER EJ08**

**BEFORE UPDATE OR INSERT OR DELETE**

**ON VETERINARIOS**

**//FOR EACH ROW -> ESTO NO HACE FALTA PQ NO VOY A IR FILA POR FILA MIRANDO NUEVOS Y ANTIGUOS**

**BEGIN**

**IF USER <> 'USUASIR' THEN**

**RAISE\_APPLICATION\_ERROR(-20000, 'No eres el usuario USUASIR por lo que no puedes editar VETERINARIOS');**

**END IF;**

**END;**