

Práctica de TRIGGERS

1. Crear un TRIGGER BEFORE DELETE sobre la tabla EMPLOYEES que

impida borrar un registro si su JOB_ID es algo relacionado con CLERK

```
create or replace trigger impedirBorrarClerk before
delete on employees FOR EACH ROW
begin
if :old.job_id like('%CLERK%') then
raise_application_error(-20001, 'No se pueden eliminar empleados de tipo CLERK');
end if;
end;
```

```
select* from employees;
delete from employees where job_id like('%CLERK');
```

Error que empieza en la línea: 36 del comando -

```
delete from employees where job_id like('%CLERK')
```

Error en la línea de comandos : 36 Columna : 13

Informe de error -

Error SQL: ORA-20001: No se pueden eliminar empleados de tipo CLERK

ORA-06512: en "HR.IMPEDIRBORRARCLERK", línea 3

ORA-04088: error durante la ejecución del disparador 'HR.IMPEDIRBORRARCLERK'

2. Crear una tabla denominada AUDITORIA con las siguientes columnas:

CREATE TABLE AUDITORIA (

USUARIO VARCHAR(50),

FECHA DATE,

SALARIO_ANTIGUO NUMBER,

SALARIO_NUEVO NUMBER);

3. Crear un TRIGGER BEFORE INSERT de tipo STATEMENT, de forma que



cada vez que se haga un INSERT en la tabla REGIONS guarde una fila

en la tabla AUDITORIA con el usuario y la fecha en la que se ha hecho el

INSERT

```
create or replace trigger insertarEnAuditoria after insert ON regions
BEGIN
insert into auditoria values(USER,SYSDATE,0,0);
END;
```

```
INSERT INTO REGIONS VALUES(8,'TURKMENISTAN');
SELECT * FROM AUDITORIA;
```

Script x	Resultado de la Consulta x		
  SQL Todas las Filas Recuperadas: 1 en 0.01 segundos			
USUARIO	FECHA	SALARIO_ANTIGUO	SALARIO_NUEVO
HR	15/10/23	0	0

4. Realizar otro trigger BEFORE UPDATE de la columna SALARY de tipo EACH ROW.

- Si la modificación supone rebajar el salario el TRIGGER debe disparar un RAISE_APPLICATION_FAILURE “no se puede bajar un salario”.
- Si el salario es mayor debemos dejar el salario antiguo y el salario nuevo en la tabla AUDITORIA.

```
create or replace trigger insertarEnAuditoriaCambioSalario BEFORE UPDATE ON EMPLOYEES for each row
BEGIN
IF :old.salary>:new.salary then
raise_application_error(-20011, 'no se puede bajar un salario.');
```

```
elsif :old.salary<:new.salary then
insert into auditoria values(USER,SYSDATE,:old.salary,:new.salary);
end if;
END;
```

```
update employees set salary=25000 where employee_id=100;
```

```
update employees set salary=24000 where employee_id=100;
```

```
select * from employees;
```

Salida de Script x Resultado de la Consulta x

Tarea terminada en 0.04 segundos

1 fila actualizadas.

Error que empieza en la línea: 64 del comando -

```
update employees set salary=24000 where employee_id=100
```

Error en la línea de comandos : 64 Columna : 8

Informe de error -

Error SQL: ORA-20011: no se puede bajar un salario.

ORA-06512: en "HR.INSERTARENAUDITORIACAMBIOSSALARIO", línea 3

ORA-04088: error durante la ejecución del disparador 'HR.INSERTARENAUDITORIACAMBIOSSALARIO'

5. Crear un TRIGGER BEFORE INSERT en la tabla DEPARTMENTS que al insertar un departamento compruebe que el código no esté repetido y luego que si el LOCATION_ID es NULL le ponga 1700 y si el MANAGER_ID es NULL le ponga 200

```
insert into departments (department_id,department_name) values('500','Departamento de futbol siuuu');
select * from departments;
```

Salida de Script x

Resultado de la Consulta x

SQL | Todas las Filas Recuperadas: 28 en 0.004 segundos

DEPARTMENT_ID	DEPARTMENT_NAME	MANAGER_ID	LOCATION_ID
19	190 Contracting	(null)	1700
20	200 Operations	(null)	1700
21	210 IT Support	(null)	1700
22	220 NOC	(null)	1700
23	230 IT Helpdesk	(null)	1700
24	240 Government Sales	(null)	1700
25	250 Retail Sales	(null)	1700
26	260 Recruiting	(null)	1700
27	270 Payroll	(null)	1700
28	500 Departamento de futbol siuuu	200	1700

```
insert into departments (department_id,department_name) values('210','Departamento de futbol siuuu');
select * from departments;
```

Salida de Script x	Resultado de la Consulta x
Tarea terminada en 0.048 segundos	
1 fila insertadas.	
<p>Error que empieza en la línea: 89 del comando -</p> <p>insert into departments (department_id,department_name) values('210','Departamento de futbol siuuu')</p> <p>Error en la línea de comandos : 89 Columna : 13</p> <p>Informe de error -</p> <p>Error SQL: ORA-20069: El código de departamento ya está en uso</p> <p>ORA-06512: en "HR.CHEKEARNULOSENDEPARTAMENTOS", línea 8</p> <p>ORA-04088: error durante la ejecución del disparador 'HR.CHEKEARNULOSENDEPARTAMENTOS'</p>	