1. Utilizar un trigger llamado ins\_emple para controlar que en la empresa, sólo está permitido insertar en las tablas de la base de datos durante las horas de oficina normales, entre las 9:30 y las 17:30 horas, de lunes a viernes.

¿Es un trigger a nivel de tabla o de fila?

Usa del paquete DBMS\_STANDARD la función RAISE\_APPLICATION\_ERROR con el código -20001 para indicar que se ha intentado insertar fuera del horario.

Pruebas: Comprobar la hora antes de ejecutar. Si son mas de las 9:30, cambiar las condiciones del enunciado para estar fuera de la jornada laboral.

* **Antes** de compilar el trigger ejecutar la siguiente sentencia insert

INSERT INTO EMPLE VALUES (7934,'MUÑOZ','EMPLEADO',7782,'23/01/1982',169000,NULL,10);

¿Qué sucede?

* Compilar el trigger y volver a ejecutar la misma insert

INSERT INTO EMPLE VALUES (7934,'MUÑOZ','EMPLEADO',7782,'23/01/1982',169000,NULL,10);

NOTA: Recordar que 9:30 y 17:30 son cadenas, y que la función TO\_CHAR se puede aplicar a fechas. Comprueba si los días de las semanas los devuelve en ingles o castellano, para ello puedes usar la setencia:

SELECT TO\_CHAR (SYSDATE, 'DY') FROM DUAL;

Una vez probado, desactiva el trigger ins\_emple.

1. Un uso habitual de los triggers es el de auditoria. Crear un trigger para guardar en una tabla de control la fecha, el usuario de base de datos y el Tipo de Operación (insert, update o delete) que modificó la tabla EMPLE.

¿Es un trigger a nivel de tabla o de fila?

Considerar la tabla CTRL\_EMPLEADOS que está en el script tablas\_trigger.sql de ikas.

NOTA: Existe la función USER que informa del ID de usuario Oracle que tiene iniciada la sesión. Probar antes por ejemplo ejecutando la sentencia:

select USER from DUAL;

Probar el TRIGGER realizando las siguientes sentencias:

SELECT \* FROM emple;

INSERT INTO EMPLE VALUES (7934,'MUÑOZ','EMPLEADO',7782,'23/01/1982',169000,NULL,10);

select \* from Ctrl\_Empleados;

¿Qué sucede?

UPDATE EMPLE

SET SALARIO = 100

WHERE EMP\_NO = 8934;

select \* from Ctrl\_Empleados;

¿Qué sucede?

DELETE FROM EMPLE

WHERE EMP\_NO = 8934;

select \* from Ctrl\_Empleados;

¿Qué sucede?

INSERT INTO EMPLE VALUES (9934,'SARRALDE','EMPLEADO',7782,'23/01/1982',169000,NULL,10);

select \* from Ctrl\_Empleados;

¿Qué sucede?

UPDATE EMPLE

SET SALARIO = 100

WHERE EMP\_NO = 9934;

select \* from Ctrl\_Empleados;

¿Qué sucede?

DELETE FROM EMPLE

WHERE EMP\_NO = 9934;

select \* from Ctrl\_Empleados;

¿Qué sucede?

1. Crear un trigger para que no permita insertar empleados en el departamento "Ventas".

Usa del paquete DBMS\_STANDARD la función RAISE\_APPLICATION\_ERROR con el código -20500 para indicar que se ha intentado insertar un empleado en el departamento "Ventas".

¿Es un trigger a nivel de tabla o de fila?

Probar el TRIGGER.

Comprueba que el departamento “Ventas” es el código 30.

INSERT INTO EMPLE VALUES (7934,'SARRALDE','EMPLEADO',7782,'23/01/1982',169000,NULL,10)

¿Qué sucede?

INSERT INTO EMPLE VALUES (7934,'SARRALDE','EMPLEADO',7782,'23/01/1982',169000,NULL,30);

¿Qué sucede?

NOTA: Recordar que usando el Pseudo-Registro :new se registran los valores nuevos.

1. Crear un trigger para impedir que se aumente el salario de un empleado en más de un 20%.

Usa del paquete DBMS\_STANDARD la función RAISE\_APPLICATION\_ERROR con el código -20600 para indicar que se ha intentado aumentar el salario mas de un 20%.

NOTA: Recuerda que usando los Pseudo-Registros :old y :new se pueden registrar los valores antiguos y los nuevos. Recuerda la clausula WHEN a la hora de crear el trigger.

Realizar un ejemplo de uso del TRIGGER.

select \* from emple

where dept\_no=20

order by salario;

UPDATE EMPLE

SET SALARIO=SALARIO\*2

WHERE DEPT\_NO=20;

select \* from emple

where dept\_no=20

order by salario;

1. Existe la vista “EmpleadoDpto”en la que se recoge el numero y nombre del empleado junto con el nombre del departamento. Sobre esta vista están permitidas las operaciones DML.

Crear ahora un disparador, con el que en el caso de intentar hacer una inserción a esa vista, se compruebe:

* si no existe un departamento con ese nombre, se inserte en la tabla departamentos. El código del departamento será el máximo mas 10. Controlar el no sobrepasar el dominio de la columna dept\_no.
* se inserte el empleado en la tabla emple. Tener en cuenta que en la vista no están todos los campos de la tablas, por lo que el salario será 0, la fecha de alta el día en el que se realiza la inserción y en el resto habrá que insertar valores nulos.

Probar el trigger.

SELECT \* FROM DEPART;

SELECT \* FROM EmpleadoDpto;

INSERT INTO EmpleadoDpto VALUES (9998,'VISTA','NuevoDpto');

SELECT \* FROM DEPART;

SELECT \* FROM EmpleadoDpto

ORDER BY EMP\_NO;

SELECT \* FROM DEPART;

SELECT \* FROM EmpleadoDpto

ORDER BY EMP\_NO;

INSERT INTO EmpleadoDpto VALUES (9999,'VISTA2','Contabilidad');

SELECT \* FROM DEPART;

SELECT \* FROM EmpleadoDpto

ORDER BY EMP\_NO;

ROLLBACK;