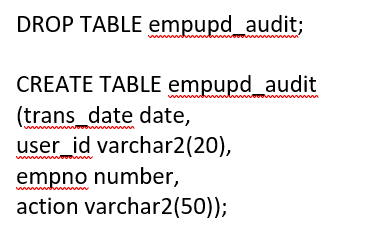
# Audit Trigger

This exercise is to demonstrate the use of triggers to trap errors and to update audit trail files.

In the first instance the trigger will not allow the user to reduce an employees salary.

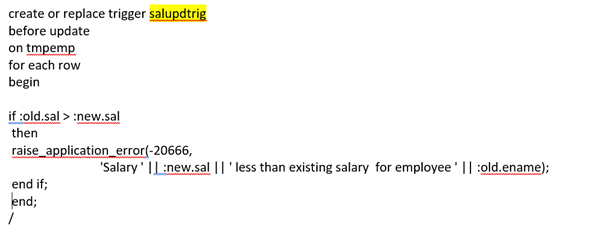
1. Let’s create\_audit table first, which we will use to audit trial table.



2. Now create a table called tmpemp as follows

**create table tmpemp as select \* from scott.emp;**

3. Type this trigger code **salupdtrig** and understand what it is doing.



4. Try updating a variety of salaries, both lower and higher than the current ones to see the effect. (

e.g.  **UPDATE tmpemp SET sal = nnnn WHERE empno = 'AAAA';**

6. Now create a trigger, by updating salupdtrig code to include **audit** of changes to the table, when salary added is higher than current salary. Note that there will be change in the old/new **sal** section.

7. Repeat 4 above, by trying to add salary less than current and salary higher than current.

9. Have a look at the entries in the empupd\_audit table. What values are there and why? Does this table get updated every time?

Important note: in week 6 we created a trigger tmp\_salary\_check , so if you enter a value that is less then range assigned for a particular job, you will trigger this trigger.

insert into tmpsalguide values ('CLERK', 5000, 15000);

insert into tmpsalguide values ('ANALYST', 10000, 20000);

insert into tmpsalguide values ('MANAGER', 20000, 40000);

insert into tmpsalguide values ('SALESMAN', 5000, 30000);

insert into tmpsalguide values ('PRESIDENT', 50000,'' );

However, if you insert the value that is within the range but higher than the current salary this piece of code will fire, and a new row of data will be inserted in the audit table.

insert into empupd\_audit values(sysdate,user,:old.empno,'salary update');