SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 2500

sita 2 3000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> create table emp\_log

2 (

3 essn int,

4 new\_sal number(8,2)

5 );

Table created.

SQL> commit;

Commit complete.

SQL> create or replace trigger sal\_change

2 after update of sal on emp

3 for each row

4 begin

5 insert into emp\_log values(:new.emp.ssn,:new.emp.sal);

6 end;

7 /

Warning: Trigger created with compilation errors.

SQL> show errors;

Errors for TRIGGER SAL\_CHANGE:

LINE/COL ERROR

-------- -----------------------------------------------------------------

2/28 PLS-00049: bad bind variable 'NEW.EMP'

2/41 PLS-00049: bad bind variable 'NEW.EMP'

SQL> create or replace trigger sal\_change

2 after update of sal on emp

3 for each row

4 begin

5 insert into emp\_log values(:new.ssn,:new.sal);

6 end;

7 /

Trigger created.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 2500

sita 2 3000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> select \* from emp\_log;

no rows selected

SQL> update emp set sal=10000 where ssn=1;

1 row updated.

SQL> select \* from emp\_log;

ESSN NEW\_SAL

---------- ----------

1 10000

SQL> select \* from emp\_log;

ESSN NEW\_SAL

---------- ----------

1 10000

SQL> alter table emp\_log add old\_sal number(8,2);

Table altered.

SQL> select \* from emp\_log;

ESSN NEW\_SAL OLD\_SAL

1 10000

SQL> drop trigger sal\_change;

Trigger dropped.

SQL> create or replace trigger sal\_change

2 after update of sal on emp

3 for each row

4 begin

5 insert into emp\_log values(:new.ssn,:new.sal,:old.sal);

6 end;

7 /

Trigger created.

SQL> select \* from emp;

ENAME SSN SAL

ram 1 10000

sita 2 3000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> select \* from emp\_log;

ESSN NEW\_SAL OLD\_SAL

---------- ---------- ----------

1 10000

SQL> update emp set sal=30000 where ssn=2;

1 row updated.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> select \* from emp\_log;

ESSN NEW\_SAL OLD\_SAL

---------- ---------- ----------

1 10000

2 30000 3000

SQL> create or replace trigger t

2 before insert or delete or update on emp

3 begin

4 case

5 when inserting then

6 dbms\_output.put\_line('inserted');

7 when deleting then

8 dbms\_output.put\_line('deleted');

9 when updating then

10 dbms\_output.put\_line('updated');

11 end case;

12 end;

13 /

Trigger created.

SQL> set serveroutput on;

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> delete from emp where ssn=8;

deleted

1 row deleted.

SQL> create or replace trigger t

2 before insert or delete or update on emp

3 begin

4 case

5 when inserting then

6 dbms\_output.put\_line('inserted');

7 when deleting then

8 dbms\_output.put\_line('deleted');

9 when updating then

10 dbms\_output.put\_line('updated');

11 end case;

12 end;

13 /

Trigger created.

SQL> set serveroutput on;

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

ram 8 6000

8 rows selected.

SQL> delete from emp where ssn=8;

deleted

1 row deleted.

SQL> create or replace trigger t

2 after insert or delete or update on emp

3 begin

4 case

5 when inserting then

6 dbms\_output.put\_line('inserted');

7 when deleting then

8 dbms\_output.put\_line('deleted');

9 when updating then

10 dbms\_output.put\_line('updated');

11 end case;

12 end;

13 /

Trigger created.

SQL> delete from emp where ssn=7;

deleted

1 row deleted.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

6 rows selected.

SQL> rollback;

Rollback complete.

SQL> create table deleted\_emp

2 (

3 dename varchar(30),

4 dssn int,

5 dsal number(8,2)

6 );

Table created.

SQL> create or replace trigger del\_emp

2 after delete on emp

3 begin

4 insert into deleted\_emp values(:old.ename,:old.ssn,:old.sal);

5 select \* from deleted\_emp;

6 end;

7 /

create or replace trigger del\_emp

\*

ERROR at line 1:

ORA-04082: NEW or OLD references not allowed in table level triggers

SQL> create or replace trigger del\_emp

2 after delete on emp

3 for each row

4 begin

5 insert into deleted\_emp values(:old.ename,:old.ssn,:old.sal

6 select \* from deleted\_emp;

7 end;

8 /

Warning: Trigger created with compilation errors.

SQL> show errors;

Errors for TRIGGER DEL\_EMP:

LINE/COL ERROR

-------- -----------------------------------------------------------------

2/2 PL/SQL: SQL Statement ignored

3/2 PL/SQL: ORA-00917: missing comma

SQL> create or replace trigger del\_emp

2 after delete on emp

3 for each row

4 begin

5 insert into deleted\_emp values(:old.ename,:old.ssn,:old.sal);

6 end;

7 /

Trigger created.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

sameeksha 7 4250

7 rows selected.

SQL> select \* from deleted\_emp;

no rows selected

SQL> delete from emp where ssn=7;

deleted

1 row deleted.

SQL> select \* from deleted\_emp;

DENAME DSSN DSAL

------------------------------ ---------- ----------

sameeksha 7 4250

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

6 rows selected.

SQL> alter table deleted\_emp add date\_deleted date;

Table altered.

SQL> rollback;

Rollback complete.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

6 rows selected.

SQL> create or replace trigger del\_emp

2 after delete on emp

3 for each row

4 begin

5 insert into deleted\_emp values(:old.ename,:old.ssn,:old.sal,sysdate);

6 end;

7 /

Trigger created.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 5000

sangeeta 6 2700

6 rows selected.

SQL> select \* from deleted\_emp;

DENAME DSSN DSAL DATE\_DELE

------------------------------ ---------- ---------- ---------

sameeksha 7 4250

SQL> delete from emp where ssn=6;

deleted

1 row deleted.

SQL> select \* from deleted\_emp;

DENAME DSSN DSAL DATE\_DELE

------------------------------ ---------- ---------- ---------

sameeksha 7 4250

sangeeta 6 2700 01-JAN-06

SQL> create table salup

2 (

3 ssn int,

4 old\_sal number(8,2),

5 new\_sal number(8,2),

6 account varchar(50)

7 );

Table created.

SQL> create or replace trigger tu

2 after update of sal on emp

3 for each row

4 begin

5 insert into salup values(:new.ssn,:old.sal,:new.sal,user);

6 end;

7 /

Trigger created.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 50000

SQL> update emp set sal=1000 where ssn=1;

1 row updated.

SQL> select \* from emp;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 1000

sita 2 30000

geeta 3 1250

ramesh 4 3250

rakesh 5 50000

SQL> select \* from salup;

SSN OLD\_SAL NEW\_SAL ACCOUNT

1 10000 1000 SCOTT

SQL> create table te

2 (

3 ename varchar(30),

4 ssn int primary key,

5 sal number(8,2)

6 );

Table created.

SQL> create table salup

2 (

3 ssn int,

4 acc varchar(50),

5 new\_sal number(8,2),

6 old\_sal number(8,2),

7 dofupdation date);

Table created.

SQL> insert into te values('ram',1,10000);

1 row created.

SQL> select \* from te;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

SQL> create or replace trigger ts

2 after update of sal on te

3 for each row

4 begin

5 insert into salup values(:new.ssn,user,:new.sal,:old.sal,sysdate);

6 end;

7 /

Trigger created.

SQL> select \* from salup;

no rows selected

SQL> update te set sal=1000 where ssn=1;

1 row updated.

SQL> select \* from salup;

SSN ACC NEW\_SAL OLD\_SAL DOFUPDATI

1 SCOTT 1000 10000 14-OCT-14

SQL> create or replace trigger t

2 after update of sal on te

3 for each row

4 declare

5 d salup%rowtype;

6 cursor c is select \* from salup;

7 begin

8 insert into salup values(:new.ssn,user,:new.sal,:old.sal,sysdate);

9 open c;

10 loop

11 fetch c into d;

12 exit when c%notfound;

13 dbms\_output.put\_line('newssn='||d.ssn);

14 dbms\_output.put\_line('username='||d.acc);

15 dbms\_output.put\_line('newsalary='||d.new\_sal);

16 dbms\_output.put\_line('oldsalary='||d.old\_sal);

17 dbms\_output.put\_line('date of modification='||d.dofupdation);

18 end loop;

19 close c;

20 end;

21 /

Trigger created.

SQL> select \* from te;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 1000

set serveroutput on;

SQL> select \* from te;

ENAME SSN SAL

------------------------------ ---------- ----------

ram 1 10000

SQL> select \* from salup;

SSN ACC NEW\_SAL OLD\_SAL DOFUPDATI

1 SCOTT 10000 1000 14-OCT-14

SQL> update te set sal=1000 where ssn=1;

newssn=1

username=SCOTT

newsalary=10000

oldsalary=1000

date of modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=1000

oldsalary=10000

date of modification=14-OCT-14

1 row updated.

SQL> select \* from salup;

SSN ACC NEW\_SAL OLD\_SAL DOFUPDATI

1 SCOTT 10000 1000 14-OCT-14

1 SCOTT 1000 10000 14-OCT-14

SQL> update te set sal=2000 where ssn=1;

newssn=1

username=SCOTT

newsalary=10000

oldsalary=1000

date of modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=1000

oldsalary=10000

date of modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=2000

oldsalary=1000

date of modification=14-OCT-14

1 row updated.

SQL> select \*from salup;

SSN ACC NEW\_SAL OLD\_SAL DOFUPDATI

1 SCOTT 10000 1000 14-OCT-14

1 SCOTT 1000 10000 14-OCT-14

1 SCOTT 2000 1000 14-OCT-14

SQL> create or replace trigger t

2 after update of sal on te

3 for each row

4 declare

5 d salup%rowtype;

6 cursor c is select \* from salup;

7 begin

8 insert into salup values(:new.ssn,user,:new.sal,:old.sal,sysdate);

9 open c;

10 loop

11 fetch c into d;

12 exit when c%notfound;

13

dbms\_output.put\_line('newssn='||d.ssn||chr(10)||'username='||d.acc||chr(10)||'newsalary='|

|d.new\_sal||chr(10)||'oldsalary='||d.old\_sal||chr(10)||'date of

modification='||d.dofupdation);

14 end loop;

15 close c;

16 end;

17 /

Trigger created.

SQL> update te set sal=3000 where ssn=1;

newssn=1

username=SCOTT

newsalary=10000

oldsalary=1000

date of

modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=1000

oldsalary=10000

date of

modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=2000

oldsalary=1000

date of

modification=14-OCT-14

newssn=1

username=SCOTT

newsalary=3000

oldsalary=2000

date of

modification=14-OCT-14

1 row updated.

SQL> select \* from salup;

SSN ACC NEW\_SAL OLD\_SAL DOFUPDATI

1 SCOTT 10000 1000 14-OCT-14

1 SCOTT 1000 10000 14-OCT-14

1 SCOTT 2000 1000 14-OCT-14

1 SCOTT 3000 2000 14-OCT-14