

```
SQL> create table emp(empid number(3), empname varchar2(10), salary number(7));
```

Table created.

```
SQL> insert into emp values(3,'Sai Mohit',100000);
```

1 row created.

```
SQL> insert into emp values(4,'Adesh',90000);
```

1 row created.

```
SQL> insert into emp values(11,'Praneet',110000);
```

1 row created.

```
SQL> create or replace trigger trigaf
  2 after update or insert or delete on emp
  3 for each row
  4 begin
  5 if updating then
  6 dbms_output.put_line('TABLE IS UPDATED');
  7 elsif inserting then
  8 dbms_output.put_line('TABLE IS INSERTED');
  9 elsif deleting then
 10 dbms_output.put_line('TABLE IS DELETED');
 11 end if;
 12 end;
 13 /
```

Trigger created.

```
SQL> update emp set salary=200000 where empname='Sai Mohit';
```

1 row updated.

```
SQL> set serveroutput on
```

```
SQL> update emp set salary=200000 where empname='Sai Mohit';
TABLE IS UPDATED
```

1 row updated.

```
SQL> insert into emp values(5,'Rishit',95000);
TABLE IS INSERTED
```

1 row created.

```
SQL> delete from emp where empid=5;
TABLE IS DELETED
```

1 row deleted.

```
SQL> create or replace trigger trigbf
  2 before update or insert or delete on emp
```

```
3 for each row
4 begin
5 if updating then
6 dbms_output.put_line('TABLE IS UPDATED');
7 elsif inserting then
8 dbms_output.put_line('TABLE IS INSERTED');
9 elsif deleting then
10 dbms_output.put_line('TABLE IS DELETED');
11 end if;
12 end;
13 /
```

Trigger created.

```
SQL> insert into emp values(1,'Rishit',250000);
TABLE IS INSERTED
TABLE IS INSERTED
```

1 row created.

```
SQL> update emp set empid=18 where empname='Rishit';
TABLE IS UPDATED
TABLE IS UPDATED
```

1 row updated.

```
SQL> delete emp where empname='Rishit';
TABLE IS DELETED
TABLE IS DELETED
```

1 row deleted.

```
SQL> create table age (name varchar2(10), age number(3));
```

Table created.

```
SQL> insert into age values('Adesh',20);
```

1 row created.

```
SQL> insert into age values('Sai Mohit',21);
```

1 row created.

```
SQL> create trigger triggnew
2 after insert or update of age on person
3 for each row
4 begin
5 if(:new.age<0) then
6 dbms_output.put_line('INVALID AGE');
7 else
8 dbms_output.put_line('VALID AGE');
9 end if;
10 end;
```

Trigger created.

```
SQL> select * from person
2 /
```

NAME	AGE
Adesh	20
Sai Mohit	21

```
SQL> insert into person values('Random',18);
VALID AGE
```

1 row created.

```
SQL> insert into person values('Random',-18);
INVALID AGE
```

1 row created.

```
SQL> select * from person;
```

NAME	AGE
Adesh	20
Sai Mohit	21
Random	18
Random	-18

```
SQL> create trigger datacheck
2 after insert or update of age on person
3 for each row
4 begin
5 if(:new.age<0) then
6 raise_application_error(-20000,'NO NEGATIVE AGE ALLOWED');
7 end if;
8 end;
9 /
```

Trigger created.

```
SQL> insert into person values('Veer',10);
VALID AGE
```

1 row created.

```
SQL> insert into person values('Veer',-10);
insert into person values('Veer',-10)
*
```

ERROR at line 1:
ORA-20000: NO NEGATIVE AGE ALLOWED
ORA-06512: at "RA1911031010149.DATACHECK", line 3

ORA-04088: error during execution of trigger 'RA1911031010149.DATACHECK'

```
SQL> CREATE TABLE SRM_EMP(INAME VARCHAR2(10),ID NUMBER(5), SALARY NUMBER(10));
```

Table created.

```
SQL> CREATE TABLE SRM_SAL(INAME VARCHAR2(10),TOTALEMP NUMBER(5), TOTALSAL NUMBER(10));
```

Table created.

```
SQL> spool off
```