Ex.No.11 03.05.2025	TRIGGERS

AIM:

To implement and demonstrate the use of database triggers to perform and control insert, update, and delete function.

CREATE TABLE:

SQL>create table student(s_id number primary key, name varchar2(10), department varchar2(10));

Table created.

INSERT VALUES TO TABLE:

SQL> insert into student values (1, 'Kabesh', 'IT');

1 row created.

SQL> insert into student values (2, 'Kamalesh', 'ECE');

1 row created.

SQL> insert into student values (3, 'Jegan', 'EEE');

1 row created.

SQL>create table audit(s_id number, action_time date, action_type varchar2(10));

Table created.

SQL> create or replace trigger trg_add_student

after insert on student

for each row

begin

insert into audit(s_id, action_time, action_type)

values(:new.s_id, sysdate, 'insert');

end;

/

trigger created.

```
SQL> insert into student values (4, 'Kabesh', 'IT');
1 row created.

SQL> select * from audit;

 S_ID                ACTION_TIME       ACTION_TYPE
----------           -------------------        -------------------
        4                 06-MAY-25              INSERT

SQL> create or replace trigger trg_update_student
after update on student
for each row
begin
   insert into audit(s_id, action_time, action_type)
   values(:new.s_id, sysdate, 'update');
end;
/
Trigger created.

SQL> update student set department = 'FT' where s_id = 2;
1 row updated.

SQL> select * from audit;

S_ID              ACTION_TIME            ACTION_TYPE
----------        -------------------            ----------------
        4                 06-MAY-25                 INSERT
        2                 06-MAY-25                 UPDATE
```

```
SQL> create or replace trigger trg_delete_student
after delete on student
for each row
begin
   insert into audit(s_id, action_time, action_type)
   values(:old.s_id, sysdate, 'delete');
end;
/
Trigger created.

SQL> delete from student where s_id = 3;
1 row deleted.

SQL> select * from audit;

S_ID        ACTION_TIME      ACTION_TYPE
----------  ----------------  ---------------
       4     06-MAY-25         INSERT
       2     06-MAY-25         UPDATE
       3     06-MAY-25         DELETE
```

EXAMPLE 1

INSERT, UPDATE, DELETE ON STUDENT TABLE

```
SQL> create or replace trigger trg_all_actions
after insert or update or delete on student
for each row
begin
   if inserting then
```

```
        insert into audit(s_id, action_time, action_type)
        values(:new.s_id, sysdate, 'insert');
    elsif updating then
        insert into audit(s_id, action_time, action_type)
        values(:new.s_id, sysdate, 'update');
    elsif deleting then
        insert into audit(s_id, action_time, action_type)
        values(:old.s_id, sysdate, 'delete');
    end if;
end;
/
Trigger created.

SQL> insert into student values (5, 'Sanjay', 'IT');
1 row created.
SQL> update student set department = 'EEE' where s_id = 1;
1 row created.
SQL> delete from student where s_id = 2;
1 row deleted.


SQL> select * from audit;

S_ID          ACTION_TIME     ACTION_TYPE
----------    -----------     -----------
      4           06-MAY-25     INSERT
      2           06-MAY-25     UPDATE
      3           06-MAY-25     DELETE
      5           06-MAY-25     INSERT
      5           06-MAY-25     INSERT
```

1	06-MAY-25	UPDATE
1	06-MAY-25	UPDATE
2	06-MAY-25	DELETE
2	06-MAY-25	DELETE

9 rows selected.

EXAMPLE 2

PREVENT NULL VALUE FOR DEPARTMENT

```
SQL> create or replace trigger trg_prevent_null_dept
before update on student
for each row
begin
   if :new.department is null then
      raise_application_error(-20002, 'department cannot be set to null.');
   end if;
end;
/
Trigger created

SQL> update student set department = null where s_id = 1;
ERROR at line 1:
ORA-20002: Department cannot be set to NULL.
ORA-06512: at "SYSTEM.TRG_PREVENT_NULL_DEPT", line 3
ORA-04088: error during execution of trigger 'SYSTEM.TRG_PREVENT_NULL_DEPT'
```

CONTENTS	MARKS ALLOTED	MARKS OBTAINED
Aim,Algorithm,SQL,PL/SQL	30	
Execution and Result	20	
Viva	10	
Total	60	

RESULT

The experiment effectively demonstrated the use of database triggers in enforcing business rules and automatically maintaining audit trails.