TRIGGER

Aim:

- To understand Trigger
- Implement Trigger

Prerequisite: Oracle, SQL

Outcome: Students will be able to create trigger for a desired event.

Theory:

Stored Program which are automatically fired or executed when some event occur such as DML or DDL or DB operation.

Syntax:

```
Create or Replace Trigger trigger-namge
Before| After | Instead of Insert | update | Delete
Of colName
On tableName
Referencing old as O new as N
For each row
When (condition)
Declare
        ------
Begin
        ------
Exception
        ------
End;
```

Example:

```
Create or Replace Trigger display_salary_change

Before Insert or update or Delete

On tableName

Referencing old as new as

For each row

When (new.id>0)

Declare

        salary_difference number;

Begin

        salary_difference:= :new.salary - :old.salary;

        dbms_output.put_line(‘salary difference is: ‘ || salary_difference)
```

End;

Practice Exercise

1	Create following tables student(id number,name varchar2(25),programme varchar2(25))
2	Insert following data in student table: Id name programme 1 rohan diploma 2 anu MA 3 robert diploma 4 tom btech 5 sunny diploma
3	Create a trigger which fires for every update done on programme attribute and print the old and new value of program.

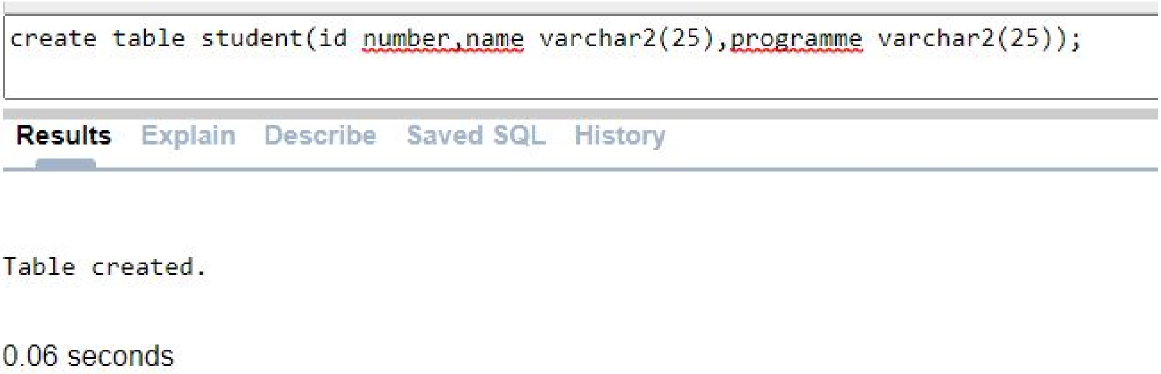
Instructions:

1. Write and execute the query in Oracle SQL server/ SQL* Plus.
2. Paste the snapshot of the output in input & output section.

Part B

Code and Output:

1.

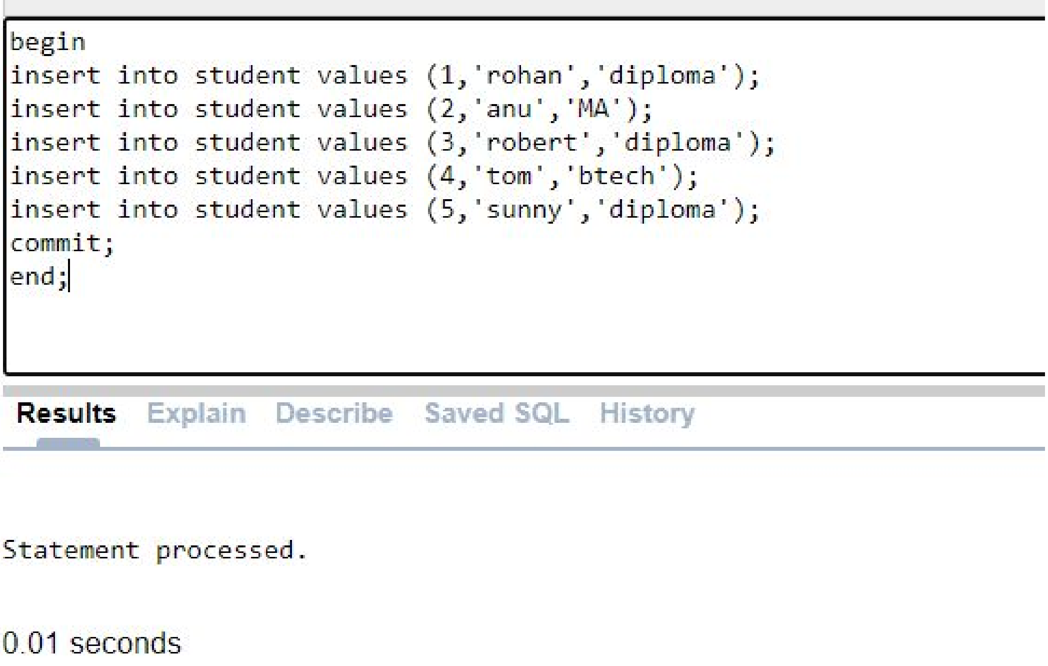


```
create table student(id number,name varchar2(25),programme varchar2(25));
```

```
Table created.

0.06 seconds
```

2.

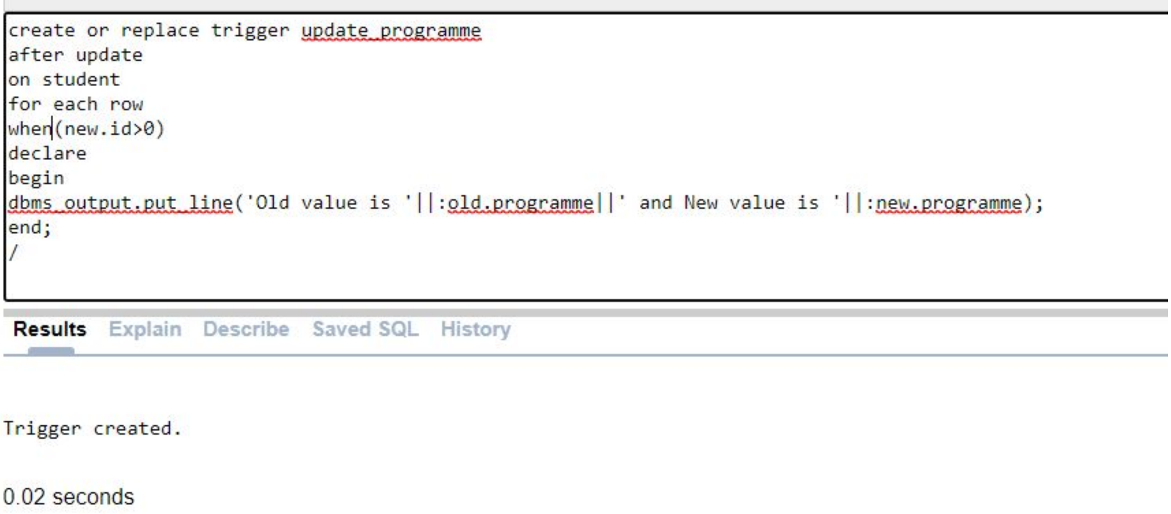


```
begin
insert into student values (1,'rohan','diploma');
insert into student values (2,'anu','MA');
insert into student values (3,'robert','diploma');
insert into student values (4,'tom','btech');
insert into student values (5,'sunny','diploma');
commit;
end;
```

```
Statement processed.

0.01 seconds
```

3.

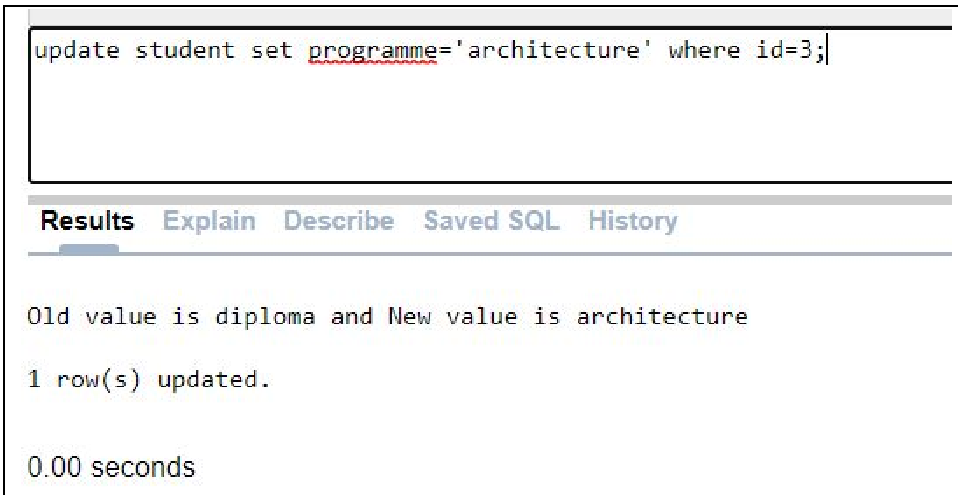


```
create or replace trigger update_programme
after update
on student
for each row
when(new.id>0)
declare
begin
dbms_output.put_line('Old value is '||:old.programme||' and New value is '||:new.programme);
end;
/
```

```
Trigger created.

0.02 seconds
```

update student set programme='architecture' where id=3;



```
update student set programme='architecture' where id=3;
```

```
Old value is diploma and New value is architecture

1 row(s) updated.
```

0.00 seconds

Observation & Learning:

Understood the use of triggers and implemented pl/sql triggers.

Conclusion:

Learned and executed pl/sql triggers .