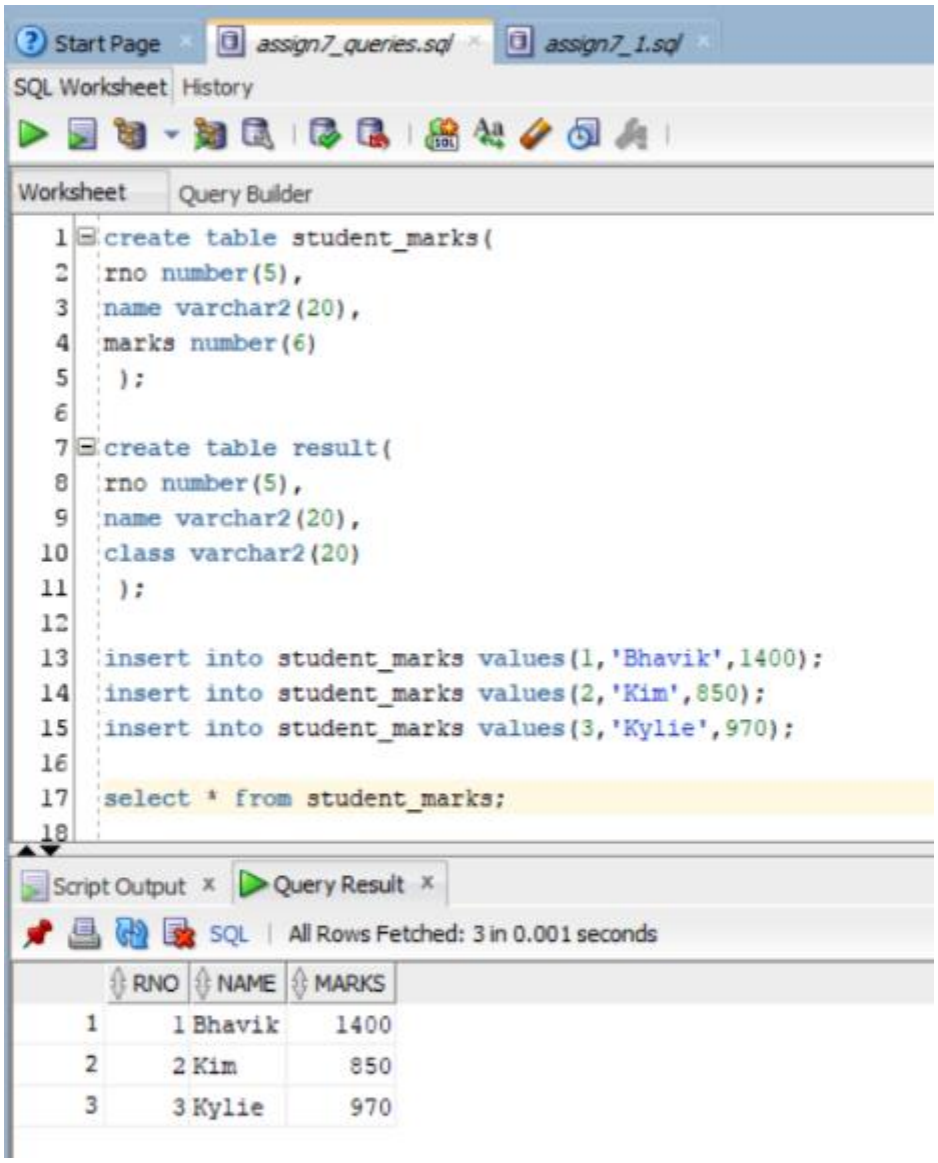
# ASSIGNMENT 7

Name: Bhavik Ransubhe
Class: TE (B) COMP
ROLL NO :39055

*******************************************************************

## 1) Stored Procedure without parameters :-

*Creating tables (tables will remain same for all ) and inserting values:*

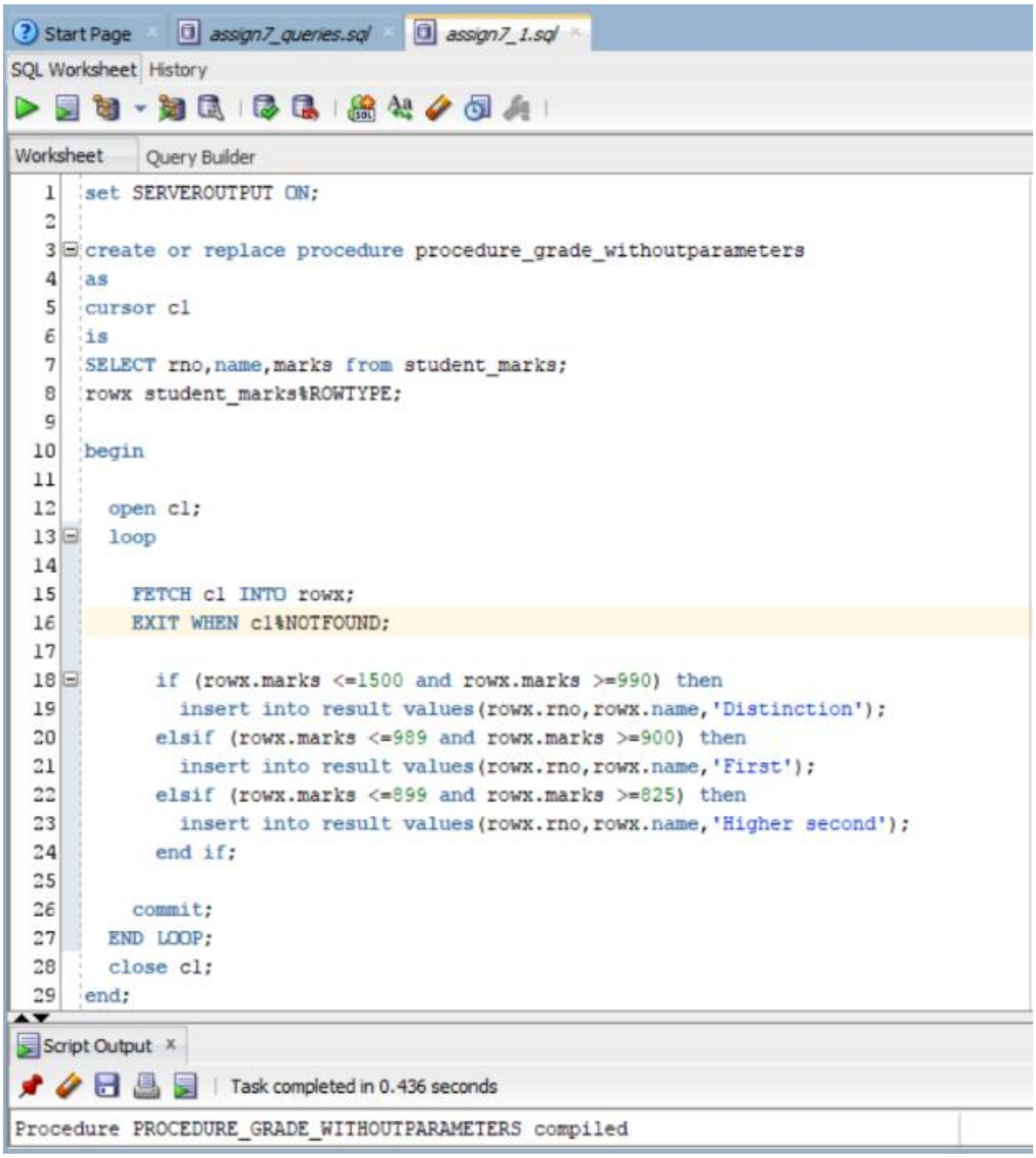


```
create table student_marks(
rno number(5),
name varchar2(20),
marks number(6)
 );

create table result(
rno number(5),
name varchar2(20),
class varchar2(20)
 );

insert into student_marks values(1,'Bhavik',1400);
insert into student_marks values(2,'Kim',850);
insert into student_marks values(3,'Kylie',970);

select * from student_marks;
```

All Rows Fetched: 3 in 0.001 seconds

|  | RNO | NAME | MARKS |
|---|---|---|---|
| 1 | 1 | Bhavik | 1400 |
| 2 | 2 | Kim | 850 |
| 3 | 3 | Kylie | 970 |

## Code:

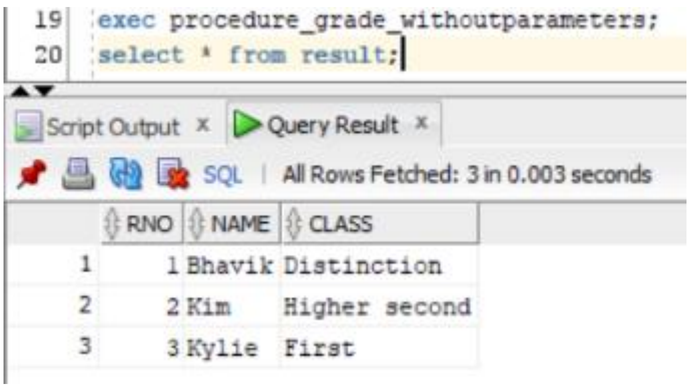


```
set SERVEROUTPUT ON;

create or replace procedure procedure_grade_withoutparameters
as
cursor c1
is
SELECT rno,name,marks from student_marks;
rowx student_marks%ROWTYPE;

begin

  open c1;
  loop

    FETCH c1 INTO rowx;
    EXIT WHEN c1%NOTFOUND;

      if (rowx.marks <=1500 and rowx.marks >=990) then
        insert into result values(rowx.rno,rowx.name,'Distinction');
      elsif (rowx.marks <=989 and rowx.marks >=900) then
        insert into result values(rowx.rno,rowx.name,'First');
      elsif (rowx.marks <=899 and rowx.marks >=825) then
        insert into result values(rowx.rno,rowx.name,'Higher second');
      end if;

    commit;
  END LOOP;
  close c1;
end;
```

Task completed in 0.436 seconds

```
Procedure PROCEDURE_GRADE_WITHOUTPARAMETERS compiled
```

**Output:**



```
exec procedure_grade_withoutparameters;
select * from result;
```

Script Output × Query Result ×

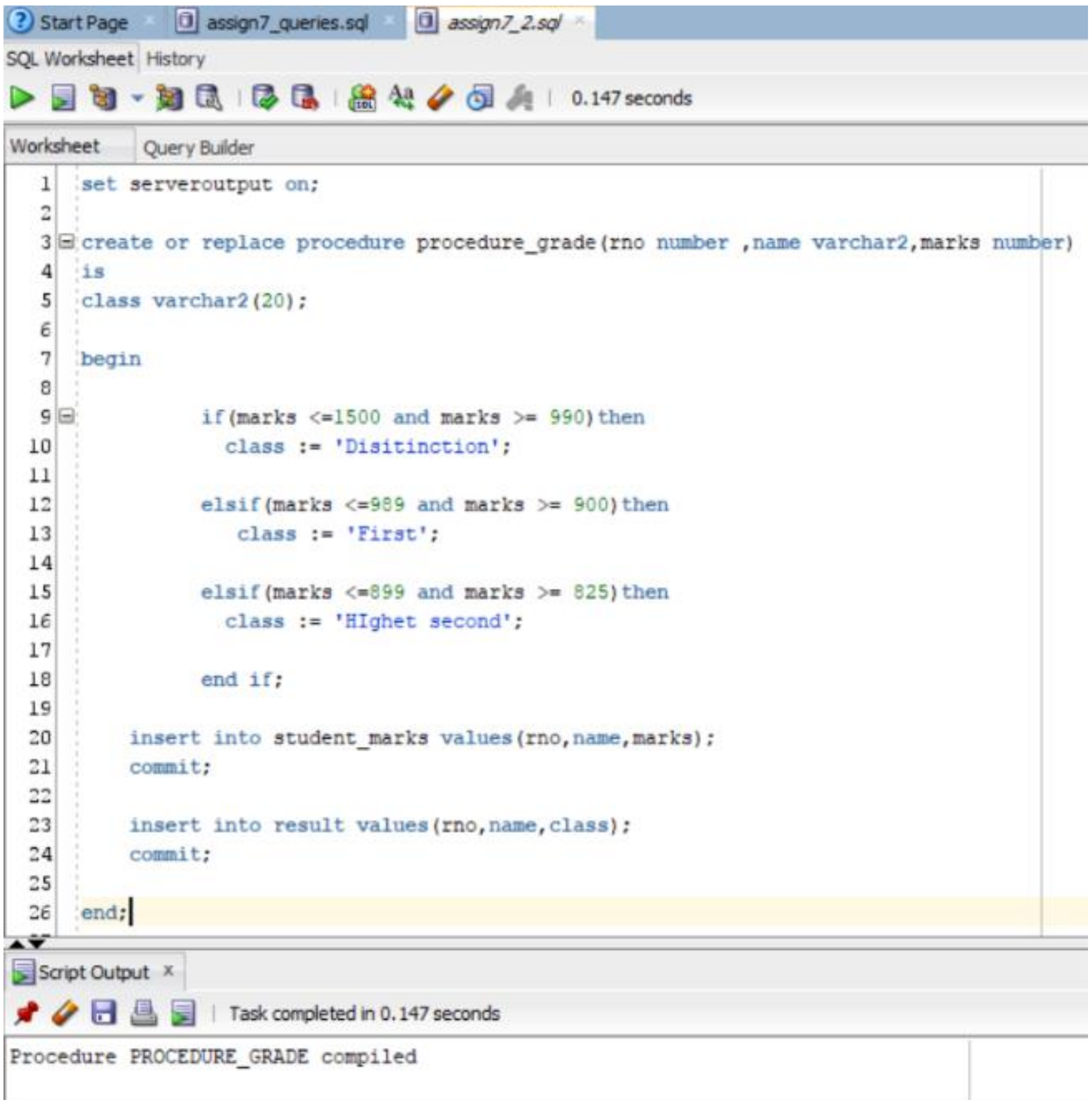
SQL | All Rows Fetched: 3 in 0.003 seconds

```
    RNO NAME   CLASS
1     1 Bhavik Distinction
2     2 Kim    Higher second
3     3 Kylie  First
```

*****************************************************************

**2.Stored Procedure with parameters :-**

## Code:



Start Page | assign7_queries.sql | assign7_2.sql

SQL Worksheet History

0.147 seconds

Worksheet Query Builder

```
set serveroutput on;

create or replace procedure procedure_grade(rno number ,name varchar2,marks number)
is
class varchar2(20);

begin

          if(marks <=1500 and marks >= 990)then
            class := 'Disitinction';

          elsif(marks <=989 and marks >= 900)then
             class := 'First';

          elsif(marks <=899 and marks >= 825)then
            class := 'HIghet second';

          end if;

    insert into student_marks values(rno,name,marks);
    commit;

    insert into result values(rno,name,class);
    commit;

end;
```

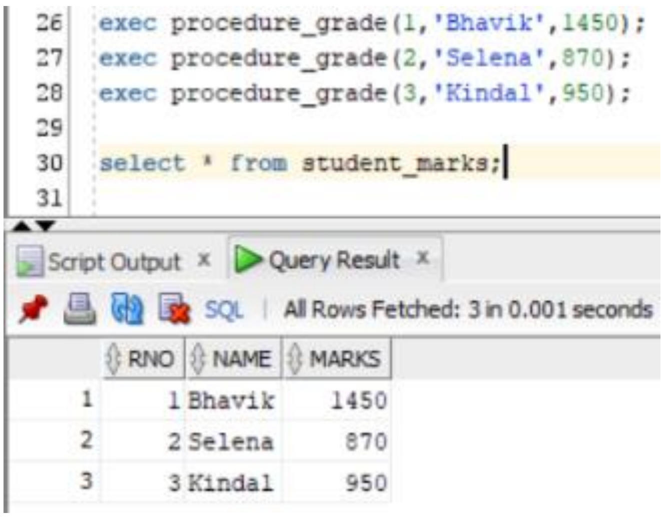
Script Output ×

Task completed in 0.147 seconds

```
Procedure PROCEDURE_GRADE compiled
```

## Output:

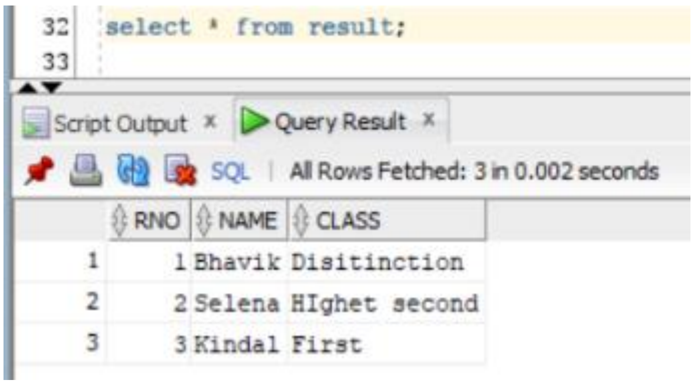
*Executing procedure with parameters :*



```
exec procedure_grade(1,'Bhavik',1450);
exec procedure_grade(2,'Selena',870);
exec procedure_grade(3,'Kindal',950);

select * from student_marks;
```

|  | RNO | NAME | MARKS |
|---|---|---|---|
| 1 | 1 | Bhavik | 1450 |
| 2 | 2 | Selena | 870 |
| 3 | 3 | Kindal | 950 |

*Final output :*



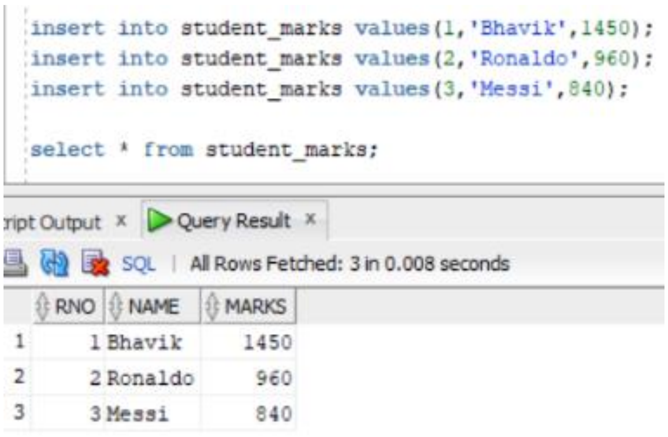
```
select * from result;
```

|  | RNO | NAME | CLASS |
|---|---|---|---|
| 1 | 1 | Bhavik | Disitinction |
| 2 | 2 | Selena | HIghet second |
| 3 | 3 | Kindal | First |

*********************************************************************************

## 3.Stored function:-

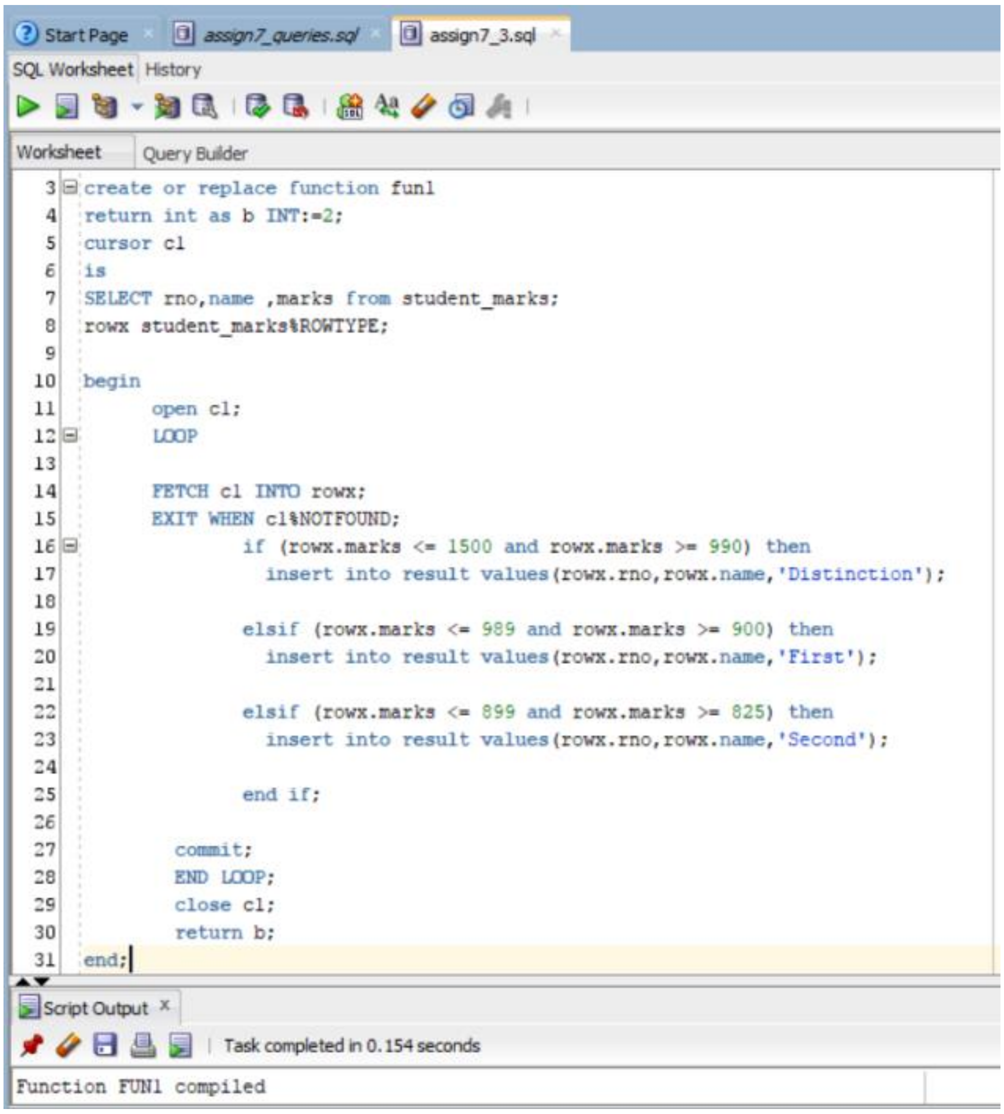
*Inserting values in table :*



```
insert into student_marks values(1,'Bhavik',1450);
insert into student_marks values(2,'Ronaldo',960);
insert into student_marks values(3,'Messi',840);

select * from student_marks;
```

|  | RNO | NAME | MARKS |
|---|---|---|---|
| 1 | 1 | Bhavik | 1450 |
| 2 | 2 | Ronaldo | 960 |
| 3 | 3 | Messi | 840 |

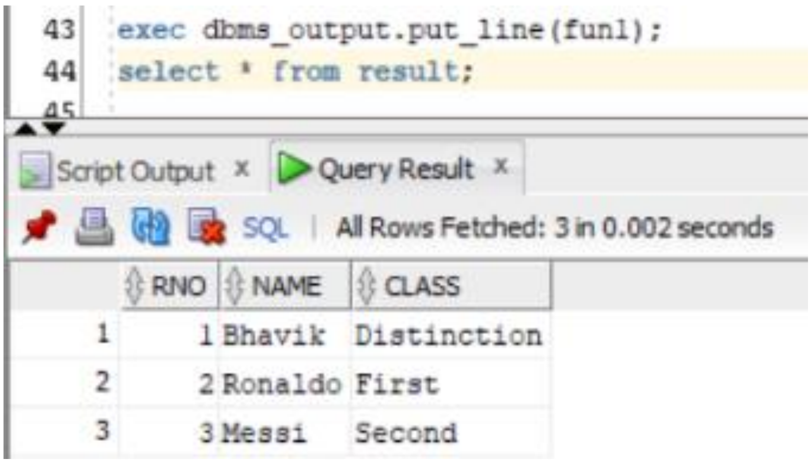
**Code:**



```
create or replace function fun1
return int as b INT:=2;
cursor c1
is
SELECT rno,name ,marks from student_marks;
rowx student_marks%ROWTYPE;

begin
        open c1;
        LOOP

        FETCH c1 INTO rowx;
        EXIT WHEN c1%NOTFOUND;
                if (rowx.marks <= 1500 and rowx.marks >= 990) then
                  insert into result values(rowx.rno,rowx.name,'Distinction');

                elsif (rowx.marks <= 989 and rowx.marks >= 900) then
                  insert into result values(rowx.rno,rowx.name,'First');

                elsif (rowx.marks <= 899 and rowx.marks >= 825) then
                  insert into result values(rowx.rno,rowx.name,'Second');

                end if;

          commit;
          END LOOP;
          close c1;
          return b;
end;
```

```
Task completed in 0.154 seconds
Function FUN1 compiled
```

**Output:**



```
exec dbms_output.put_line(fun1);
select * from result;
```

```
All Rows Fetched: 3 in 0.002 seconds
```

|   | RNO | NAME | CLASS |
|---|---|---|---|
| 1 | 1 | Bhavik | Distinction |
| 2 | 2 | Ronaldo | First |
| 3 | 3 | Messi | Second |

*******************************************************************************************

END

*******************************************************************************************