```
SQL> CREATE TABLE STUD_63 (NAME VARCHAR(10), ROLL NUMBER(2) PRIMARY KEY,
PHYSICS NUMBER(3), CHEMISTRY NUMBER(3), BIOLOGY NUMBER(3));

Table created.

SQL> DESC STUD_63;
 Name                                      Null?    Type
 ----------------------------------------- -------- ----------------------------
 NAME                                               VARCHAR2(10)
 ROLL                                      NOT NULL NUMBER(2)
 PHYSICS                                            NUMBER(3)
 CHEMISTRY                                          NUMBER(3)
 BIOLOGY                                            NUMBER(3)


SQL> INSERT INTO STUD_63
VALUES('&NAME',&ROLL,&PHYSICS,&CHEMISTRY,&BIOLOGY);
Enter value for name: DAVID
Enter value for roll: 1
Enter value for physics: 75
Enter value for chemistry: 64
Enter value for biology: 88
old   1: INSERT INTO STUD_63
VALUES('&NAME',&ROLL,&PHYSICS,&CHEMISTRY,&BIOLOGY)
new   1: INSERT INTO STUD_63 VALUES('DAVID',1,75,64,88)


SQL> SELECT * FROM STUD_63;

NAME            ROLL     PHYSICS           CHEMISTRY           BIOLOGY
---------- ---------- ---------- ---------- ---------- ----------
DAVID                1                 75                 64                 88
EMMY                 2                 78                 66                 54
FARDEEN              3                 98                 92                 94
GANESH               4                 74                 87                 61
JACOB                5                 65                 63                 72
MOHIT                6                 88                 84                 78

6 rows selected.
```

mark.sql

```
CREATE OR REPLACE PROCEDURE GRADE (N IN NUMBER, G OUT VARCHAR)

IS
TOTAL NUMBER:= 0;
R NUMBER;
M1 STUD_63.PHYSICS%TYPE;
M2 STUD_63.CHEMISTRY%TYPE;
M3 STUD_63.BIOLOGY%TYPE;

BEGIN
   SELECT PHYSICS,CHEMISTRY,BIOLOGY INTO M1,M2,M3 FROM STUD_63 WHERE
ROLL=N;
   TOTAL := M1+M2+M3;
   TOTAL := TOTAL/3;
   IF (TOTAL > 90) THEN
         G := 'A+';
   ELSIF (TOTAL > 80) THEN
         G := 'A';
   ELSIF (TOTAL > 70) THEN
         G := 'B+';
   ELSIF (TOTAL > 60) THEN
         G := 'B';
   ELSIF (TOTAL > 50) THEN
         G := 'C';
   ELSIF (TOTAL > 40) THEN
         G := 'D';
   ELSE
         G := 'FAILED';
   END IF;

   EXCEPTION
   WHEN NO_DATA_FOUND
   THEN dbms_output.put_line('Student not found');
END;
```

gradecalc.sql

```
DECLARE
   R NUMBER(2);
   G VARCHAR(5);
BEGIN
   R:=&roll_no;
   GRADE(R,G);
   if (G is not null)then
   dbms_output.put_line('Grade is :'||G);
   else
   dbms_output.put_line('Roll number not found:');
   end if;
end;
```

OUTPUT

```
SQL> @mark.sql
 35  /

Procedure created.

SQL> @gradecalc.sql
 13  /
Enter value for roll_no: 4
old   5:     R:=&roll_no;
new   5:     R:=4;
Grade is :B+

PL/SQL procedure successfully completed.

Commit complete.
SQL> @gradecalc.sql
 13  /
Enter value for roll_no: 10
old   5:     R:=&roll_no;
new   5:     R:=10;
Student not found
Roll number not found:

PL/SQL procedure successfully completed.

Commit complete.
```