DBMS Lab Assignment 6

```
SQL> CREATE TABLE STUD_MARKS(NAME VARCHAR(20), TOTAL_MARKS NUMBER(4));

Table created.

SQL> CREATE TABLE RESULT(ROLL NUMBER(2), NAME VARCHAR(2), CLASS VARCHAR(20));

Table created.

SQL> SET SERVEROUTPUT ON
SQL> CREATE OR REPLACE PROCEDURE ADD_MARKS(SNAME IN VARCHAR, MARKS IN
NUMBER) AS
2  BEGIN
3  INSERT INTO STUD_MARKS VALUES(SNAME, MARKS);
4  END;
5  /

Procedure created.

SQL> EXECUTE ADD_MARKS('A', 995);

PL/SQL procedure successfully completed.

SQL> EXECUTE ADD_MARKS('B', 950);

PL/SQL procedure successfully completed.

SQL> EXECUTE ADD_MARKS('C', 880);

PL/SQL procedure successfully completed.

SQL> EXECUTE ADD_MARKS('D', 810);

PL/SQL procedure successfully completed.

SQL> SELECT * FROM STUD_MARKS;

NAME                 TOTAL_MARKS
-------------------- -----------
A                    995
B                    950
C                    880
```

```
D                    810

SQL> CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN
NUMBER) AS
  2  GRADE VARCHAR(40);
CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN NUMBER)
AS
             *
ERROR at line 1:
ORA-00922: missing or invalid option


SQL> CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN
NUMBER) AS GRADE VARCHAR(40); BEGIN
  2  IF MARKS BETWEEN 990 AND 1500 THEN GRADE:='DIST';
CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN NUMBER)
AS GRADE VARCHAR(40); BEGIN
             *
ERROR at line 1:
ORA-00922: missing or invalid option


SQL> CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN
NUMBER) AS GRADE VARCHAR(40); BEGIN
2  IF MARKS BETWEEN 990 AND 1500 THEN GRADE:='DIST'
3  /
CREATE OR REPLACE PROCEDUREPROC_GRADE(SNAME IN VARCHAR, MARKS IN NUMBER)
AS GRADE VARCHAR(40); BEGIN
             *
ERROR at line 1:
ORA-00922: missing or invalid option


SQL> CREATE OR REPLACE PROCEDURE PROC_GRADE(SNAME IN VARCHAR, MARKS IN
NUMBER) AS GRADE VARCHAR(40);
2     BEGIN
3     IF MARKS BETWEEN 990 AND 1500 THEN
4     GRADE := 'DIST';
5     ELSIF MARKS BETWEEN 900 AND 989 THEN
6     GRADE := 'FIRST CLASS';
7     ELSIF MARKS BETWEEN 825 AND 899 THEN
8     GRADE := 'HIGHER SECOND CLASS';
9     ELSE
```

```
 10  GRADE := 'NOT CATEGORIZED';
 11  END IF;
 12  INSERT INTO RESULT(NAME, CLASS) VALUES(SNAME, GRADE);
 13  END;
 14  /

Procedure created.

SQL> BEGIN
  2  FOR S IN(SELECT NAME,TOTAL_MARKS FROM STUD_MARKS) LOOP
  3  PROC_GRADE(S.NAME,S.TOTAL_MARKS);
  4  END LOOP;
  5  END;
  6  /

PL/SQL procedure successfully completed.

SQL> SELECT * FROM RESULT;

      ROLL NA CLASS
---------- -- --------------------
           A  DIST
           B  FIRST CLASS
           C  HIGHER SECOND CLASS
           D  NOT CATEGORIZED

SQL> EXECUTE PROC_GRADE('E', 1000);

PL/SQL procedure successfully completed.

SQL> SELECT * FROM RESULT;

      ROLL NA CLASS
---------- -- --------------------
           A  DIST
           B  FIRST CLASS
           C  HIGHER SECOND CLASS
           D  NOT CATEGORIZED
           E  DIST

SQL>
```