```
CREATE TABLE stud_marks1 (
   roll_no INT PRIMARY KEY,
   name VARCHAR2(20) NOT NULL,
   total_marks INT NOT NULL
);


CREATE TABLE result1 (
   roll_no INT,
   name VARCHAR2(20),
   grade VARCHAR2(20)
);

INSERT INTO stud_marks1 VALUES (1, 'Alice', 1000);
INSERT INTO stud_marks1 VALUES (2, 'Bob', 950);
INSERT INTO stud_marks1 VALUES (3, 'Charlie', 880);
INSERT INTO stud_marks1 VALUES (4, 'Dave', 800);

COMMIT; -- Save the inserted data

CREATE OR REPLACE PROCEDURE proc_grade (
   temp IN INT,
   p_roll_no OUT stud_marks1.roll_no%TYPE,
   p_name OUT stud_marks1.name%TYPE,
   p_total OUT stud_marks1.total_marks%TYPE
) AS
BEGIN
   -- Select the student's details based on roll number
   SELECT name, total_marks, roll_no INTO p_name, p_total, p_roll_no
   FROM stud_marks1
   WHERE roll_no = temp;

   -- Insert grade based on total marks
   IF p_total <= 1500 AND p_total >= 990 THEN
      INSERT INTO result1 VALUES (p_roll_no, p_name, 'distinction');
   ELSIF p_total <= 989 AND p_total >= 900 THEN
      INSERT INTO result1 VALUES (p_roll_no, p_name, 'first class');
   ELSIF p_total <= 899 AND p_total >= 825 THEN
```

```
      INSERT INTO result1 VALUES (p_roll_no, p_name, 'HSC');
   ELSE
      INSERT INTO result1 VALUES (p_roll_no, p_name, 'fail');
   END IF;

EXCEPTION
   WHEN no_data_found THEN
      DBMS_OUTPUT.PUT_LINE('Roll number ' || temp || ' not found');
END;
/

DELETE FROM result1;
COMMIT;

DECLARE
   p_roll_no stud_marks1.roll_no%TYPE;
   p_name stud_marks1.name%TYPE;
   p_total stud_marks1.total_marks%TYPE;
BEGIN
   proc_grade(1, p_roll_no, p_name, p_total); -- Execute for roll_no 1
   proc_grade(2, p_roll_no, p_name, p_total); -- Execute for roll_no 2
   proc_grade(3, p_roll_no, p_name, p_total); -- Execute for roll_no 3
   proc_grade(4, p_roll_no, p_name, p_total); -- Execute for roll_no 4
END;
/


SELECT * FROM result1;
```