

Database Queries for Assignment-A6

Creating tables

```
CREATE TABLE Stud_Marks (
    roll INT,
    name VARCHAR(255),
    total_marks INT,
    PRIMARY KEY (roll)
);

CREATE TABLE Result (
    roll INT,
    name VARCHAR(255),
    class VARCHAR(255),
    FOREIGN KEY (roll) REFERENCES Stud_Marks (roll)
);
```

Inserting data

```
INSERT INTO Stud_Marks VALUES (1, 'Kshitij', 1400);
INSERT INTO Stud_Marks VALUES (2, 'Kalas', 500);
INSERT INTO Stud_Marks VALUES (3, 'Himanshu', 995);
INSERT INTO Stud_Marks VALUES (4, 'MEPA', 850);
INSERT INTO Stud_Marks VALUES (5, 'Macho', 900);
```

Procedure

```

CREATE OR REPLACE PROCEDURE proc_Grade (roll_no IN NUMBER) AS

-- declare section
p_roll Stud_Marks.roll%TYPE;
p_name Stud_Marks.name%TYPE;
p_total NUMBER;

BEGIN
SELECT roll, name, total_marks INTO p_roll, p_name, p_total FROM Stud_Marks WHERE roll = roll_no;

IF (p_total <= 1500 AND p_total >= 990) THEN
    INSERT INTO Result VALUES (p_roll, p_name, 'Distinction');
    DBMS_OUTPUT.PUT_LINE(p_name || ' (roll no. ' || p_roll || ') has been placed in the DISTINCTION category.');
ELSIF (p_total BETWEEN 900 AND 989) THEN
    INSERT INTO Result VALUES (p_roll, p_name, 'First Class');
    DBMS_OUTPUT.PUT_LINE(p_name || ' (roll no. ' || p_roll || ') has been placed in the FIRST CLASS.');
ELSIF (p_total BETWEEN 825 AND 899) THEN
    INSERT INTO Result VALUES (p_roll, p_name, 'Higher Second Class');
    DBMS_OUTPUT.PUT_LINE(p_name || ' (roll no. ' || p_roll || ') has been placed in the HIGHER SECOND CLASS.');
ELSE
    INSERT INTO Result VALUES (p_roll, p_name, 'Fail');
    DBMS_OUTPUT.PUT_LINE(p_name || ' (roll no. ' || p_roll || ') has FAILED.');
END IF;

EXCEPTION
WHEN NO_DATA_FOUND THEN
    DBMS_OUTPUT.PUT_LINE('No entry for this roll number in the Stud_Marks database.');

END proc_Grade;
/

```

Calling procedure

```

DECLARE
roll_no NUMBER;
BEGIN
roll_no := &roll_no; -- replace &roll_no with a number for Live SQL since it does not support user input.
proc_Grade(roll_no);
END;
/

```