Named PL/SQL Block: PL/SQL Stored Procedure and Stored Function.

Write a Stored Procedure namely proc_Grade for the categorization of student. If marks scoredby students in examination is <=1500 and marks>=990 then student will be placed in distinction category if marks scored are between 989 and 900 category is first class, if marks 899 and 825 category is Higher Second Class. Write a PL/SQL block to use procedure created with above requirement. Stud_Marks(name, total_marks) Result(Roll,Name, Class)

**Step 1: Creating the Required Tables**

First, let's assume you have the following tables in your database:

sql

Copy code

```sql
CREATE TABLE Stud_Marks (

  Name VARCHAR2(100),

  Total_Marks NUMBER

);


CREATE TABLE Result (

  Roll NUMBER GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,

  Name VARCHAR2(100),

  Class VARCHAR2(50)

);
```

**Step 2: Creating the Stored Procedure**

Next, we will create the stored procedure proc_Grade that will categorize students based on their marks.

sql

Copy code

```sql
CREATE OR REPLACE PROCEDURE proc_Grade IS

  CURSOR student_cursor IS

    SELECT Name, Total_Marks FROM Stud_Marks;


  v_name Stud_Marks.Name%TYPE;

  v_marks Stud_Marks.Total_Marks%TYPE;

  v_category VARCHAR2(50);

BEGIN

  FOR student_record IN student_cursor LOOP
```

```
        v_name := student_record.Name;

        v_marks := student_record.Total_Marks;


     -- Categorize based on marks
     IF v_marks >= 990 AND v_marks <= 1500 THEN

        v_category := 'Distinction';
     ELSIF v_marks >= 900 AND v_marks < 990 THEN

        v_category := 'First Class';
     ELSIF v_marks >= 825 AND v_marks < 900 THEN

        v_category := 'Higher Second Class';
     ELSE

        v_category := 'Not Categorized'; -- For marks below 825
     END IF;


     -- Insert result into the Result table
     INSERT INTO Result (Name, Class) VALUES (v_name, v_category);
   END LOOP;


   COMMIT;  -- Commit the transaction
EXCEPTION
   WHEN OTHERS THEN
     ROLLBACK; -- Rollback in case of any error
     DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
END proc_Grade;
/
```

**Explanation of the Stored Procedure**

1. **Cursor Declaration:**

   o   A cursor student_cursor is declared to fetch names and total marks from the Stud_Marks table.

2. **Loop through Cursor:**

   o   A loop iterates over each record in the cursor.

- o It retrieves the name and total marks of each student.

3. **Categorization Logic:**

   - o The marks are checked against the given criteria to determine the student's category.

   - o The appropriate category is assigned to v_category.

4. **Inserting Results:**

   - o The student's name and category are inserted into the Result table.

5. **Error Handling:**

   - o In case of any errors, the transaction is rolled back, and an error message is displayed.

## Step 3: Invoking the Stored Procedure

Finally, we can create a PL/SQL block to call the proc_Grade procedure.

sql

Copy code

```
BEGIN

  -- Assume some sample data is already present in Stud_Marks table

  INSERT INTO Stud_Marks (Name, Total_Marks) VALUES ('Alice', 1200);

  INSERT INTO Stud_Marks (Name, Total_Marks) VALUES ('Bob', 850);

  INSERT INTO Stud_Marks (Name, Total_Marks) VALUES ('Charlie', 950);

  INSERT INTO Stud_Marks (Name, Total_Marks) VALUES ('David', 700);


  -- Call the procedure to categorize students

  proc_Grade;


  -- Display the results

  FOR rec IN (SELECT * FROM Result) LOOP

    DBMS_OUTPUT.PUT_LINE('Roll: ' || rec.Roll || ', Name: ' || rec.Name || ', Class: ' || rec.Class);

  END LOOP;

END;

/
```

## Explanation of the PL/SQL Block

1. **Inserting Sample Data:**

- o   Sample data is inserted into the Stud_Marks table to demonstrate the procedure's functionality.

2. **Calling the Procedure:**

   - o   The proc_Grade procedure is called to categorize the students based on their marks.

3. **Displaying Results:**

   - o   A loop is used to display the results from the Result table, showing the roll number, name, and category of each student.

## Notes

- Make sure you have appropriate privileges to create procedures and tables in your Oracle database environment.

- You may need to enable DBMS_OUTPUT to see the output when running the PL/SQL block in a tool like SQL*Plus or SQL Developer.

This complete setup allows you to categorize student marks effectively while demonstrating the use of stored procedures and PL/SQL blocks in Oracle.

4o mini