PLSQL Assignment

****Q1.**** Create Table Students with following fields & Write 2 insert statements for it.

1. RollNo integer 10 => Auto Increment
2. FName string 50
3. LName string 50
4. DOB Date => Date of Birth
5. Add1 string 150
6. Add2 String 150
7. City String 100
8. State String 100
9. Pin String 10
10. Class String 10 => class in which student is studying

First field will be used as a unique identifier for the record.

Solution :

CREATE TABLE Students (

RollNo NUMBER(10) GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

FName VARCHAR2(50),

LName VARCHAR2(50),

DOB DATE,

Add1 VARCHAR2(150),

Add2 VARCHAR2(150),

City VARCHAR2(100),

State VARCHAR2(100),

Pin VARCHAR2(10),

Class VARCHAR2(10)

);

INSERT INTO Students (FName, LName, DOB, Add1, Add2, City, State, Pin, Class)

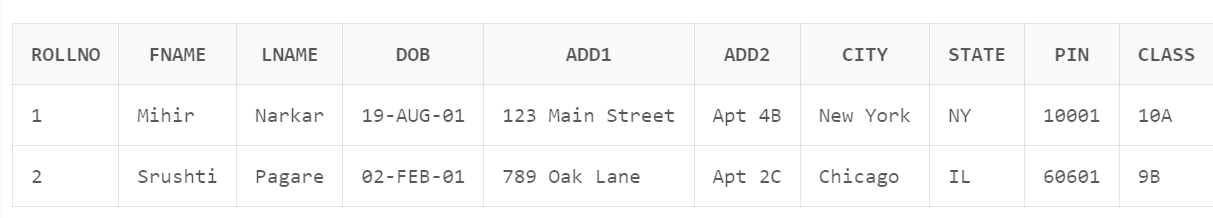
VALUES ('Mihir', 'Narkar', TO\_DATE('2001-08-19', 'YYYY-MM-DD'), '123 Main Street', 'Apt 4B', 'New York', 'NY', '10001', '10A');

INSERT INTO Students (FName, LName, DOB, Add1, Add2, City, State, Pin, Class)

VALUES ('Srushti', 'Pagare', TO\_DATE('2001-02-02', 'YYYY-MM-DD'), '789 Oak Lane', 'Apt 2C', 'Chicago', 'IL', '60601', '9B');

select \* from students;

**Output :**

****

****Q2.****Create table Subjects with following fields & Write 2 insert statements for it.

1. SubID integer 10 => Auto Increment
2. SubName String 100

First field will be used as a unique identifier for the record.

**Solution :**

CREATE TABLE Subjects (

SubID NUMBER(10) GENERATED ALWAYS AS IDENTITY PRIMARY KEY,

SubName VARCHAR2(100)

);

INSERT INTO Subjects (SubName)

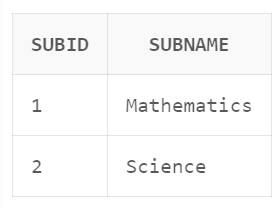
VALUES ('Mathematics');

INSERT INTO Subjects (SubName)

VALUES ('Science');

SELECT \* FROM Subjects;

**Output :**

****

**Q.3)**  Create a PL/SQL Procedure SP\_Exam\_result which should accept exam date as parameter and should create the records for each Student and each Subject in Exams table automatically assuming that all the exams were conducted on the same date. The MarksFrom field should be having value as 100. The MarksObtained field should have random values generated using following function.

ROUND(DBMS\_RANDOM.VALUE(30,100))

**Solution :**

CREATE OR REPLACE PROCEDURE SP\_Exam\_result (p\_exam\_date DATE) AS

BEGIN

FOR student\_record IN (SELECT RollNo FROM Students) LOOP

FOR subject\_record IN (SELECT SubID FROM Subjects) LOOP

INSERT INTO Exams (ExamDate, RollNo, SubID, MarksObtained, MarksFrom)

VALUES (p\_exam\_date, student\_record.RollNo, subject\_record.SubID, ROUND(DBMS\_RANDOM.VALUE(30, 100)), 100);

END LOOP;

END LOOP;

END SP\_Exam\_result;

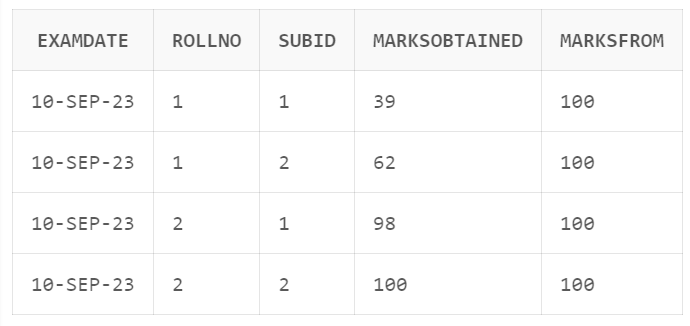
BEGIN

SP\_Exam\_result(TO\_DATE('2023-09-10', 'YYYY-MM-DD')); -- Replace with your desired exam date

END;

select \* from exams;

**Output :**

****

****Q4.****Create a PL/SQL Procedure SP\_Exam\_result which should accept exam date as parameter and should create the records for each Student and each Subject in Exams table automatically assuming that all the exams were conducted on the same date. The MarksFrom field should be having value as 100. The MarksObtained field should have random values generated using following function.

ROUND(DBMS\_RANDOM.VALUE(30,100))

**Solution :**

CREATE OR REPLACE PROCEDURE SP\_Exam\_result(p\_exam\_date DATE) AS

BEGIN

FOR student\_rec IN (SELECT RollNo FROM Students) LOOP

FOR subject\_rec IN (SELECT SubID FROM Subjects) LOOP

-- Check if a record already exists with the same combination of values

BEGIN

INSERT INTO Exams (ExamDate, RollNo, SubID, MarksObtained, MarksFrom)

VALUES (p\_exam\_date, student\_rec.RollNo, subject\_rec.SubID, ROUND(DBMS\_RANDOM.VALUE(30, 100)), 100);

EXCEPTION

WHEN DUP\_VAL\_ON\_INDEX THEN

-- Handle the exception (record already exists)

NULL;

END;

END LOOP;

END LOOP;

COMMIT;

END SP\_Exam\_result;

BEGIN

SP\_Exam\_result(TO\_DATE('2023-09-10', 'YYYY-MM-DD')); -- Replace with your desired exam date

END;

// Retrieve all exam results for a specific exam date:

SELECT \* FROM Exams

WHERE ExamDate = TO\_DATE('2023-09-10', 'YYYY-MM-DD');

**Output :** 

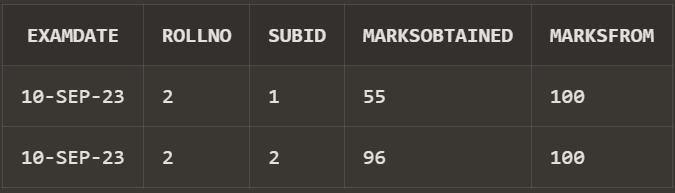
// Retrieve exam results for a specific student for a given exam date:

SELECT \* FROM Exams

WHERE ExamDate = TO\_DATE('2023-09-10', 'YYYY-MM-DD')

AND RollNo = 2;

**Output :**



****Q5.****Using SP\_Exam\_result create exam results for exams date 31-Aug-2018, 30-Nov-2018, 28-Feb-2019 and 15-May-2019. Write calls to SP\_Exam\_result procedure.

**Solution :**

BEGIN

SP\_Exam\_result(TO\_DATE('2018-08-31', 'YYYY-MM-DD'));

END;

BEGIN

SP\_Exam\_result(TO\_DATE('2018-11-30', 'YYYY-MM-DD'));

END;

BEGIN

SP\_Exam\_result(TO\_DATE('2019-02-28', 'YYYY-MM-DD'));

END;

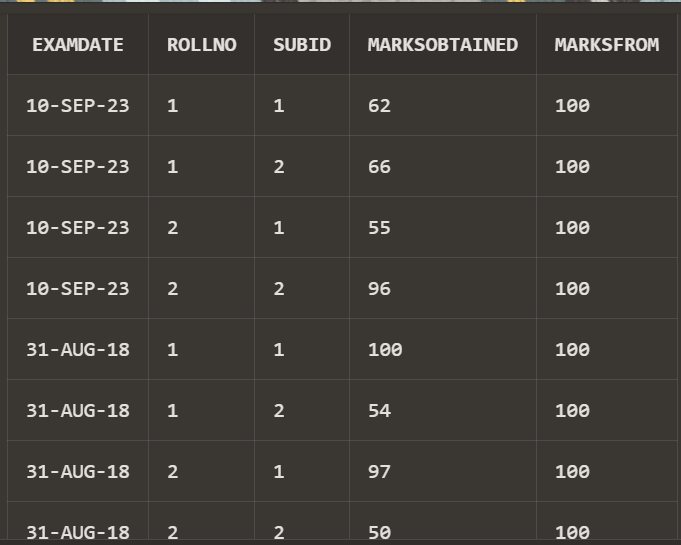
BEGIN

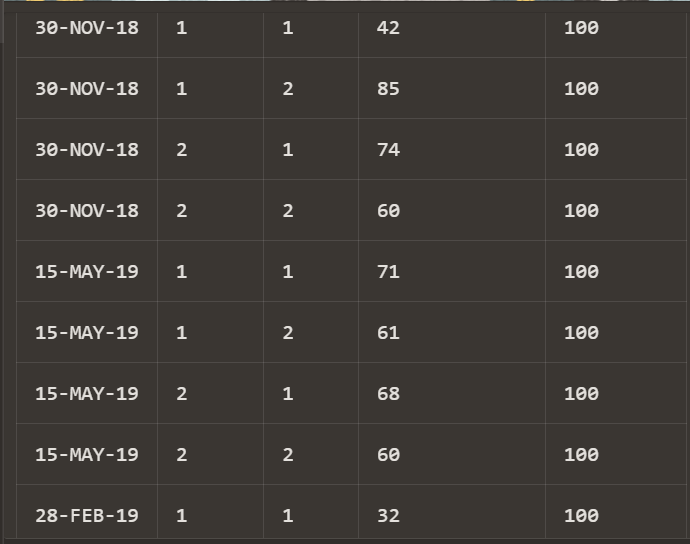
SP\_Exam\_result(TO\_DATE('2019-05-15', 'YYYY-MM-DD'));

END;

Select \* from exams ;

**Output :**

****

****

****Q6.**** Write SQL query to generate output as given below for the students by adding up all the marks for the Student for each Subject

**Solution :**

SELECT

s.RollNo AS "Roll No.",

s.FName || ' ' || s.LName AS "Student Full Name",

'Mathematics' AS "Subject Name",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks1",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks2",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks3",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks4",

SUM(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Total",

ROUND((SUM(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) / 400) \* 100, 2) AS "%"

FROM

Students s

JOIN

Exams e ON s.RollNo = e.RollNo

JOIN

Subjects sub ON e.SubID = sub.SubID

WHERE

sub.SubName = 'Mathematics'

GROUP BY

s.RollNo, s.FName, s.LName

UNION ALL

SELECT

s.RollNo AS "Roll No.",

s.FName || ' ' || s.LName AS "Student Full Name",

'Science' AS "Subject Name",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks1",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks2",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks3",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks4",

SUM(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Total",

ROUND((SUM(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) / 400) \* 100, 2) AS "%"

FROM

Students s

JOIN

Exams e ON s.RollNo = e.RollNo

JOIN

Subjects sub ON e.SubID = sub.SubID

WHERE

sub.SubName = 'Science'

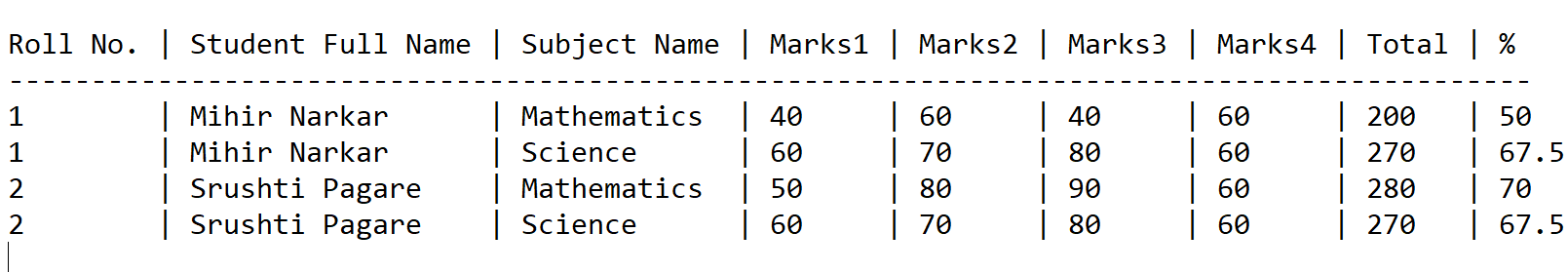
GROUP BY

s.RollNo, s.FName, s.LName

ORDER BY

"Roll No.", "Subject Name";

**Output :**

****

****Q7.****Store the above out put in a table called Results.

**Solution :**

CREATE TABLE Results AS

SELECT

s.RollNo AS "Roll No.",

s.FName || ' ' || s.LName AS "Student Full Name",

sub.SubName AS "Subject Name",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks1",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks2",

MAX(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) AS "Marks3",

MAX(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Marks4",

SUM(CASE WHEN sub.SubName = 'Mathematics' THEN e.MarksObtained ELSE NULL END) +

SUM(CASE WHEN sub.SubName = 'Science' THEN e.MarksObtained ELSE NULL END) AS "Total",

ROUND((SUM(CASE WHEN sub.SubName IN ('Mathematics', 'Science') THEN e.MarksObtained ELSE NULL END) / 400) \* 100, 2) AS "%"

FROM

Students s

JOIN

Exams e ON s.RollNo = e.RollNo

JOIN

Subjects sub ON e.SubID = sub.SubID

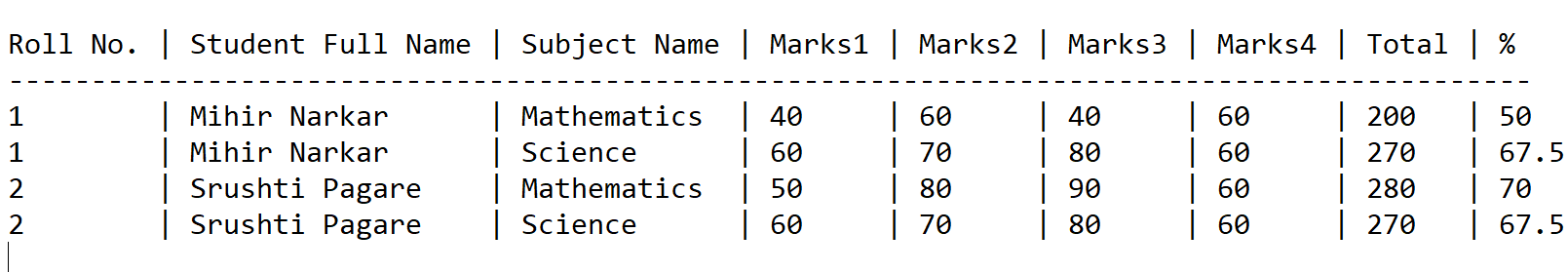
WHERE

sub.SubName IN ('Mathematics', 'Science')

GROUP BY

s.RollNo, s.FName, s.LName, sub.SubName;

Select \* from results;



****Q8.****Write a query to find out the total highest mark scorer for each subject.

**Solution :**

WITH SubjectScores AS (

SELECT

sub.SubName AS "Subject Name",

s.RollNo AS "Roll No.",

s.FName || ' ' || s.LName AS "Student Full Name",

e.MarksObtained AS "Marks"

FROM

Students s

JOIN

Exams e ON s.RollNo = e.RollNo

JOIN

Subjects sub ON e.SubID = sub.SubID

)

SELECT

"Subject Name",

"Roll No.",

"Student Full Name",

"Marks"

FROM

(

SELECT

"Subject Name",

"Roll No.",

"Student Full Name",

"Marks",

RANK() OVER (PARTITION BY "Subject Name" ORDER BY "Marks" DESC) AS Rank

FROM

SubjectScores

)

WHERE

Rank = 1;

**Output:**

