SQL LAB -10

(Right Join , Self Join, Natural Join)

Submitted by: Lingutla Rajitha

AF ID : AF0366374

Batch ID : ANP-C7281

Lab: Use the same student management system Database and table from previous lab. Perform the following commands on the table Student and Enrollment.

```
mysql> use studentmanagementsystem;
Database changed
```

1. Assume a university where students can enroll in various courses.Now, write down a FULL outer JOIN query to retrieve the details.

```
mysql> SELECT *
    -> FROM Student s
    -> LEFT JOIN Enrollment e ON s.StudentID = e.StudentID
    -> UNION
    -> SELECT *
    -> FROM Student s
    -> RIGHT JOIN Enrollment e ON s.StudentID = e.StudentID;
```

StudentID	FirstName	LastName	DateOfBirth	Gender	Email	Phone	EnrollmentID	EnrollmentDate	StudentID	InstructorID	CourseID
1	John	Doe	2000-01-01	Male	john@email.com	123-456-7890	101	2023-01-15	1	1	1
1	John	Doe	2000-01-01	Male	john@email.com	123-456-7890	102	2023-01-16	1	2	2
2	Jane	Smith	2001-02-02	Female	jane@email.com	234-567-8901	103	2023-01-17	2	1	3
2	Jane	Smith	2001-02-02	Female	jane@email.com	234-567-8901	104	2023-01-18	2	2	4
3	Bob	Johnson	1999-03-03	Male	bob@email.com	345-678-9012	NULL	NULL	NULL	NULL	NULL
101	Jane	Smith	2000-01-01	Male	jane_Smith@example.com	9876543210	NULL	NULL	NULL	NULL	NULL
102	Ishitha	Iyer	2001-02-02	Female	Ishitha@gmail.com	9123456789	402	2023-01-02	102	302	202
103	Raman	Bhalla	2002-03-03	Male	Bhalla@gmail.com	9282726252	403	2023-01-03	103	303	203
104	Ruhi	Khan	2003-04-04	Female	Ruhi@gmail.com	9325649871	404	2023-01-04	104	304	204
105	Vidyuth	Sahay	2004-05-05	Male	Vidyuth@gmail.com	9563214789	405	2023-01-05	105	305	205

```
10 rows in set (0.04 sec)
```

2. Assume a university where students can enroll in various courses.Now, write down a Natural JOIN query to retrieve the details.

```
mysql> SELECT *
    -> FROM Student
    -> NATURAL JOIN Enrollment;
```

StudentID	FirstName	LastName	DateOfBirth	Gender	Email	Phone	EnrollmentID	EnrollmentDate	InstructorID	CourseID
1	John	Doe	2000-01-01	Male	john@email.com	123-456-7890	101	2023-01-15	1	1
1	John	Doe	2000-01-01	Male	john@email.com	123-456-7890	102	2023-01-16	2	2
2	Jane	Smith	2001-02-02	Female	jane@email.com	234-567-8901	103	2023-01-17	1	3
2	Jane	Smith	2001-02-02	Female	jane@email.com	234-567-8901	104	2023-01-18	2	4
102	Ishitha	Iyer	2001-02-02	Female	Ishitha@gmail.com	9123456789	402	2023-01-02	302	202
103	Raman	Bhalla	2002-03-03	Male	Bhalla@gmail.com	9282726252	403	2023-01-03	303	203
104	Ruhi	Khan	2003-04-04	Female	Ruhi@gmail.com	9325649871	404	2023-01-04	304	204
105	Vidyuth	Sahay	2004-05-05	Male	Vidyuth@gmail.com	9563214789	405	2023-01-05	305	205

8 rows in set (0.00 sec)

ChatGPT Exercise

Using ChatGPT generates SQL queries of the below problem .

Scenario 1: We have an "inventory" Table with following columns:ProductID,ProductName,StockQuantity and "sales" Table with following columns:SaleID,ProductID(foreign key),SaleQuantity,SaleDate.Now you need to use full outer join to generate a report that includes all products in the inventory and their sales, displaying NULL values for products that haven't been sold.Generate the chatGPT prompt for the above scenario.

```
SELECT i.ProductID, i.ProductName, i.StockQuantity, s.SaleQuantity, s.SaleDate
FROM inventory i
FULL OUTER JOIN sales s ON i.ProductID = s.ProductID;
```

This SQL query performs a FULL OUTER JOIN between the inventory and sales tables on the ProductID column. It retrieves details from both tables, displaying all products in the inventory along with their corresponding sales. If a product hasn't been sold, the corresponding columns from the sales table will contain NULL values.