

SQL MODULE

LAB - 5

Mohammad Faraz Shaik

AF0366704

Questions

Lab 1: Use the Student management system Database and table from our previous lab and write a sql query to achieve the below scenario.

Assume you are managing a university database that tracks student enrollments in various courses. You have two tables, "Student" and "Enrollment". The goal is to retrieve information about each student's ID, first name, last name, and their enrollment details, including the enrollment ID and the associated course ID.

Hint: Use inner join to retrieve data.

Submission:

Create an SQL script file containing your solutions for all tasks (queries). Name the file "lab_assignment1.sql" Provide comments above each query to indicate the query's purpose.

ChatGPT Exercise

Using ChatGPT generates SQL queries of the below problem .

Scenario 1: Imagine you have tables for students and courses. Use an inner join to generate a list of all possible student-course combinations, displaying the student name and course name.

We have a "Student" table with the following a
columns: StudentId, FirstName, LastName and "Course" table with the following a
columns: CourseId, CourseName and Enrollment table with the following a

columns:EnrollmentID,StudentID(Foreign key),CourseID(Foreign Key).You want to use inner join to generate a list of all possible student-course combinations.Generate the ChatGPT prompt for the above scenario.

ASSIGNMENT -1&2



```
mysql> describe student;
```

Field	Type	Null	Key	Default	Extra
stuId	int	NO	PRI	NULL	
firstname	varchar(30)	NO		NULL	
lastname	varchar(30)	NO		NULL	
DOB	date	NO		NULL	
gender	char(1)	NO		NULL	
email	varchar(50)	NO	UNI	NULL	
ph	varchar(10)	YES	UNI	NULL	

```
7 rows in set (0.00 sec)
```

```
mysql> describe enrollment;
```

Field	Type	Null	Key	Default	Extra
EnrollmentId	int	NO	PRI	NULL	
StudentId	int	NO	MUL	NULL	
CourseId	int	NO		NULL	

```
3 rows in set (0.00 sec)
```



```
mysql> SELECT
-> s.StuId,
-> s.FirstName,
-> s.LastName,
-> e.EnrollmentId,
-> e.CourseId
-> FROM
-> Student s
-> INNER JOIN
-> Enrollment e ON s.StuId = e.StudentId;
```

StuId	FirstName	LastName	EnrollmentId	CourseId
1	Rohit	Sharma	1	101
2	virat	kohli	2	102
3	mahi	dhoni	3	103
4	yuvraj	singh	4	104
1	Rohit	Sharma	5	105

```
5 rows in set (0.00 sec)
```



```
mysql> CREATE TABLE Course_new (
-> CourseId INT NOT NULL PRIMARY KEY,
-> CourseName VARCHAR(100) NOT NULL
-> );
Query OK, 0 rows affected (0.24 sec)
```



```
mysql> INSERT INTO Course (CourseId, CourseName) VALUES
    -> (101, 'Mathematics'),
    -> (102, 'Physics'),
    -> (103, 'Chemistry'),
    -> (104, 'Biology'),
    -> (105, 'Computer Science');
ERROR 1062 (23000): Duplicate entry '101' for key 'course.PRIMARY'
mysql> INSERT INTO Course_New (CourseId, CourseName) VALUES
    -> (101, 'Mathematics'),
    -> (102, 'Physics'),
    -> (103, 'Chemistry'),
    -> (104, 'Biology'),
    -> (105, 'History');
Query OK, 5 rows affected (0.14 sec)
Records: 5 Duplicates: 0 Warnings: 0
```



```
mysql> SELECT
    -> s.FirstName,
    -> s.LastName,
    -> c.CourseName
    -> FROM
    -> Student s
    -> INNER JOIN
    -> Enrollment e ON s.StuId = e.StudentId
    -> INNER JOIN
    -> Course_new c ON e.CourseId = c.CourseId;
+-----+-----+-----+
| FirstName | LastName | CourseName |
+-----+-----+-----+
| Rohit    | Sharma   | Mathematics |
| virat    | kohli    | Physics     |
| mahi     | dhoni    | Chemistry   |
| yuvraj   | singh    | Biology     |
| Rohit    | Sharma   | History     |
+-----+-----+-----+
5 rows in set (0.00 sec)
```