LAB ASSIGNMENT 3

Q1) Attributes Description of student table
1 Admission number Integer value
2 Name String of 20 characters long
3 Gender A single character
4 Date of birth Date type
5 Course String of 15 characters long
6 Family income Integer value
The SQL statement to create a table in MySQL to incorporate the details.

```
mysql> use LAB3;
Database changed
mysql> CREATE TABLE Student (
    ->     AdmissionNumber INT NOT NULL,
    ->     Name VARCHAR(20),
    ->     Gender CHAR(1),
    ->     DateOfBirth DATE,
    ->     Course VARCHAR(15),
    ->     FamilyIncome INT,
    ->     PRIMARY KEY (AdmissionNumber)
    -> );
Query OK, 0 rows affected (0.09 sec)

mysql> DESCRIBE STUDENT;
+-----------------+-------------+------+-----+---------+-------+
| Field           | Type        | Null | Key | Default | Extra |
+-----------------+-------------+------+-----+---------+-------+
| AdmissionNumber | int         | NO   | PRI | NULL    |       |
| Name            | varchar(20) | YES  |     | NULL    |       |
| Gender          | char(1)     | YES  |     | NULL    |       |
| DateOfBirth     | date        | YES  |     | NULL    |       |
| Course          | varchar(15) | YES  |     | NULL    |       |
| FamilyIncome    | int         | YES  |     | NULL    |       |
+-----------------+-------------+------+-----+---------+-------+
6 rows in set (0.01 sec)
```

1. Insert Sample data to the table

```
mysql> INSERT INTO Student (AdmissionNumber, Name, Gender, DateOfBirth, Course, FamilyIncome)
    -> VALUES
    -> (1, 'John', 'M', '1995-08-15', 'Science', 50000),
    -> (2, 'Jeni', 'F', '1998-04-22', 'Commerce', 60000),
    -> (3, 'Anil', 'M', '1997-11-10', 'Humanities', 55000),
    -> (4, 'Elizabeth', 'F', '1996-07-05', 'History', 48000);
Query OK, 4 rows affected (0.03 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM STUDENT;
+-----------------+-----------+--------+-------------+-----------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course    | FamilyIncome |
+-----------------+-----------+--------+-------------+-----------+--------------+
|               1 | John      | M      | 1995-08-15  | Science   |        50000 |
|               2 | Jeni      | F      | 1998-04-22  | Commerce  |        60000 |
|               3 | Anil      | M      | 1997-11-10  | Humanities|        55000 |
|               4 | Elizabeth | F      | 1996-07-05  | History   |        48000 |
+-----------------+-----------+--------+-------------+-----------+--------------+
4 rows in set (0.00 sec)
```

ii) Display name and course of students in the table student

```
mysql> SELECT Name, Course
    -> FROM Student;
+-----------+------------+
| Name      | Course     |
+-----------+------------+
| John      | Science    |
| Jeni      | Commerce   |
| Anil      | Humanities |
| Elizabeth | History    |
+-----------+------------+
4 rows in set (0.00 sec)
```

iii) Display the entire column values of a table

```
mysql> SELECT *
    -> FROM Student;
+-----------------+-----------+--------+-------------+-----------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course    | FamilyIncome |
+-----------------+-----------+--------+-------------+-----------+--------------+
|               1 | John      | M      | 1995-08-15  | Science   |        50000 |
|               2 | Jeni      | F      | 1998-04-22  | Commerce  |        60000 |
|               3 | Anil      | M      | 1997-11-10  | Humanities|        55000 |
|               4 | Elizabeth | F      | 1996-07-05  | History   |        48000 |
+-----------------+-----------+--------+-------------+-----------+--------------+
4 rows in set (0.00 sec)
```

iv) SQL statement to display the details of female students in the table.

```
mysql> SELECT *
    -> FROM Student
    -> WHERE Gender = 'F';
+-----------------+-----------+--------+-------------+----------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course   | FamilyIncome |
+-----------------+-----------+--------+-------------+----------+--------------+
|               2 | Jeni      | F      | 1998-04-22  | Commerce |        60000 |
|               4 | Elizabeth | F      | 1996-07-05  | History  |        48000 |
+-----------------+-----------+--------+-------------+----------+--------------+
2 rows in set (0.00 sec)
```

v) Display the course From the table output should be displayed with there are no duplicate values.

```
mysql> SELECT DISTINCT Course
    -> FROM Student;
+------------+
| Course     |
+------------+
| Science    |
| Commerce   |
| Humanities |
| History    |
+------------+
4 rows in set (0.00 sec)
```

vi) Display name, course and monthly family income of only the Science group students whose income is below Rs. 25000.

```
mysql> SELECT Name, Course, FamilyIncome
    -> FROM Student
    -> WHERE Course = 'Science' AND FamilyIncome < 25000;
Empty set (0.00 sec)

mysql> INSERT INTO Student (AdmissionNumber, Name, Gender, DateOfBirth, Course, FamilyIncome)
    -> VALUES
    -> (5, 'Jayan', 'M', '1995-01-5', 'Science', 10000);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM STUDENT;
+-----------------+-----------+--------+-------------+------------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course     | FamilyIncome |
+-----------------+-----------+--------+-------------+------------+--------------+
|               1 | John      | M      | 1995-08-15  | Science    |        50000 |
|               2 | Jeni      | F      | 1998-04-22  | Commerce   |        60000 |
|               3 | Anil      | M      | 1997-11-10  | Humanities |        55000 |
|               4 | Elizabeth | F      | 1996-07-05  | History    |        48000 |
|               5 | Jayan     | M      | 1995-01-05  | Science    |        10000 |
+-----------------+-----------+--------+-------------+------------+--------------+
5 rows in set (0.00 sec)

mysql> SELECT Name, Course, FamilyIncome
    -> FROM Student
    -> WHERE Course = 'Science' AND FamilyIncome < 25000;
+-------+---------+--------------+
| Name  | Course  | FamilyIncome |
+-------+---------+--------------+
| Jayan | Science |        10000 |
+-------+---------+--------------+
1 row in set (0.00 sec)
```

vii) Write a query to display the name, course and monthly income of students studying In courses other than Science group (i.e.,the students of Commerce and Humanities Groups).

```
mysql> SELECT Name, Course, FamilyIncome
    -> FROM Student
    -> WHERE Course IN ('Commerce', 'Humanities');
+------+------------+--------------+
| Name | Course     | FamilyIncome |
+------+------------+--------------+
| Jeni | Commerce   |        60000 |
| Anil | Humanities |        55000 |
+------+------------+--------------+
2 rows in set (0.00 sec)
```

viii) Display a list of students whose monthly income falls in the range of Rs. 25000/- to Rs. 45000/-.

```
mysql> INSERT INTO Student (AdmissionNumber, Name, Gender, DateOfBirth, Course, FamilyIncome)
    -> VALUES(6, 'Rama', 'F', '1995-02-1', 'History', 30000);
Query OK, 1 row affected (0.01 sec)

mysql> SELECT * FROM STUDENT;
+-----------------+-----------+--------+-------------+------------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course     | FamilyIncome |
+-----------------+-----------+--------+-------------+------------+--------------+
|               1 | John      | M      | 1995-08-15  | Science    |        50000 |
|               2 | Jeni      | F      | 1998-04-22  | Commerce   |        60000 |
|               3 | Anil      | M      | 1997-11-10  | Humanities |        55000 |
|               4 | Elizabeth | F      | 1996-07-05  | History    |        48000 |
|               5 | Jayan     | M      | 1995-01-05  | Science    |        10000 |
|               6 | Rama      | F      | 1995-02-01  | History    |        30000 |
+-----------------+-----------+--------+-------------+------------+--------------+
6 rows in set (0.00 sec)

mysql> SELECT *
    -> FROM Student
    -> WHERE FamilyIncome BETWEEN 25000 AND 45000;
+-----------------+------+--------+-------------+---------+--------------+
| AdmissionNumber | Name | Gender | DateOfBirth | Course  | FamilyIncome |
+-----------------+------+--------+-------------+---------+--------------+
|               6 | Rama | F      | 1995-02-01  | History |        30000 |
+-----------------+------+--------+-------------+---------+--------------+
1 row in set (0.01 sec)
```

ix) Retrieve the details of students in Commerce and Humanities groups(Using IN operator).

```
mysql> SELECT *
    -> FROM Student
    -> WHERE Course IN ('Commerce', 'Humanities');
+-----------------+------+--------+-------------+------------+--------------+
| AdmissionNumber | Name | Gender | DateOfBirth | Course     | FamilyIncome |
+-----------------+------+--------+-------------+------------+--------------+
|               2 | Jeni | F      | 1998-04-22  | Commerce   |        60000 |
|               3 | Anil | M      | 1997-11-10  | Humanities |        55000 |
+-----------------+------+--------+-------------+------------+--------------+
2 rows in set (0.00 sec)
```

x) Display the name of students whose name start with Letter A.

```
mysql> SELECT Name
    -> FROM Student
    -> WHERE Name LIKE 'A%';
+------+
| Name |
+------+
| Anil |
+------+
1 row in set (0.00 sec)
```

xi) Display the details of students in the alphabetical order of their names.

```
mysql> SELECT *
    -> FROM Student
    -> ORDER BY Name;
+-----------------+-----------+--------+-------------+------------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course     | FamilyIncome |
+-----------------+-----------+--------+-------------+------------+--------------+
|               3 | Anil      | M      | 1997-11-10  | Humanities |        55000 |
|               4 | Elizabeth | F      | 1996-07-05  | History    |        48000 |
|               5 | Jayan     | M      | 1995-01-05  | Science    |        10000 |
|               2 | Jeni      | F      | 1998-04-22  | Commerce   |        60000 |
|               1 | John      | M      | 1995-08-15  | Science    |        50000 |
|               6 | Rama      | F      | 1995-02-01  | History    |        30000 |
+-----------------+-----------+--------+-------------+------------+--------------+
6 rows in set (0.01 sec)
```

xii) Display the details according to their monthly family income. Result from the highest income to the lowest.

```
mysql> SELECT *
    -> FROM Student
    -> ORDER BY FamilyIncome DESC;
+-----------------+-----------+--------+-------------+------------+--------------+
| AdmissionNumber | Name      | Gender | DateOfBirth | Course     | FamilyIncome |
+-----------------+-----------+--------+-------------+------------+--------------+
|               2 | Jeni      | F      | 1998-04-22  | Commerce   |        60000 |
|               3 | Anil      | M      | 1997-11-10  | Humanities |        55000 |
|               1 | John      | M      | 1995-08-15  | Science    |        50000 |
|               4 | Elizabeth | F      | 1996-07-05  | History    |        48000 |
|               6 | Rama      | F      | 1995-02-01  | History    |        30000 |
|               5 | Jayan     | M      | 1995-01-05  | Science    |        10000 |
+-----------------+-----------+--------+-------------+------------+--------------+
6 rows in set (0.00 sec)
```

xiii) Display the name and family income of Science group students according to the descending order of income.

```
mysql> SELECT Name, FamilyIncome
    -> FROM Student
    -> WHERE Course = 'Science'
    -> ORDER BY FamilyIncome DESC;
+-------+--------------+
| Name  | FamilyIncome |
+-------+--------------+
| John  |        50000 |
| Jayan |        10000 |
+-------+--------------+
2 rows in set (0.00 sec)
```

xiv) Display the name and course of the students who have the highest family monthly income.

```
mysql> SELECT Name, Course
    -> FROM Student
    -> WHERE FamilyIncome = (SELECT MAX(FamilyIncome) FROM Student);
+------+----------+
| Name | Course   |
+------+----------+
| Jeni | Commerce |
+------+----------+
1 row in set (0.03 sec)
```

Q2.

STUDENT relation

AdmNo	Roll	Name	Batch	Marks	Result
101	24	Sachin	Science	480	EHS
102	14	Rahul	Commerce	410	EHS
103	4	Fathima	Humanities	200	NHS
104	12	Mahesh	Commerce	180	NHS
105	24	Nelson	Humanities	385	EHS
106	8	Joseph	Commerce	350	EHS
107	24	Shaji	Humanities	205	NHS
108	2	Bincy	Science	300	EHS

```
mysql> CREATE TABLE STUDENT_RELATION (
    ->     AdmNo INT,
    ->     Roll INT,
    ->     Name VARCHAR(20),
    ->     Batch VARCHAR(20),
    ->     Marks INT,
    ->     Result VARCHAR(3)
    -> );
Query OK, 0 rows affected (0.09 sec)

mysql> Describe STUDENT_RELATION;
+--------+-------------+------+-----+---------+-------+
| Field  | Type        | Null | Key | Default | Extra |
+--------+-------------+------+-----+---------+-------+
| AdmNo  | int         | YES  |     | NULL    |       |
| Roll   | int         | YES  |     | NULL    |       |
| Name   | varchar(20) | YES  |     | NULL    |       |
| Batch  | varchar(20) | YES  |     | NULL    |       |
| Marks  | int         | YES  |     | NULL    |       |
| Result | varchar(3)  | YES  |     | NULL    |       |
+--------+-------------+------+-----+---------+-------+
6 rows in set (0.00 sec)
```

```
mysql> INSERT INTO STUDENT_RELATION (AdmNo, Roll, Name, Batch, Marks, Result) VALUES
    -> (101, 24, 'Sachin', 'Science', 480, 'EHS'),
    -> (102, 14, 'Rahul', 'Commerce', 410, 'EHS'),
    -> (103, 4, 'Fathima', 'Humanities', 200, 'NHS'),
    -> (104, 12, 'Mahesh', 'Commerce', 180, 'NHS'),
    -> (105, 24, 'Nelson', 'Humanities', 385, 'EHS'),
    -> (106, 8, 'Joseph', 'Commerce', 350, 'EHS'),
    -> (108, 24, 'Shaji', 'Humanities', 205, 'NHS'),
    -> (109, 2, 'Bincy', 'Science', 300, 'EHS');
Query OK, 8 rows affected (0.02 sec)
Records: 8  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM  STUDENT_RELATION;
+-------+------+---------+------------+-------+--------+
| AdmNo | Roll | Name    | Batch      | Marks | Result |
+-------+------+---------+------------+-------+--------+
|   101 |   24 | Sachin  | Science    |   480 | EHS    |
|   102 |   14 | Rahul   | Commerce   |   410 | EHS    |
|   103 |    4 | Fathima | Humanities |   200 | NHS    |
|   104 |   12 | Mahesh  | Commerce   |   180 | NHS    |
|   105 |   24 | Nelson  | Humanities |   385 | EHS    |
|   106 |    8 | Joseph  | Commerce   |   350 | EHS    |
|   108 |   24 | Shaji   | Humanities |   205 | NHS    |
|   109 |    2 | Bincy   | Science    |   300 | EHS    |
+-------+------+---------+------------+-------+--------+
8 rows in set (0.00 sec)
```

A. Display all the students who are eligible for higher studies.

```
mysql> SELECT * FROM STUDENT_RELATION WHERE Result = 'EHS';
+-------+------+--------+------------+-------+--------+
| AdmNo | Roll | Name   | Batch      | Marks | Result |
+-------+------+--------+------------+-------+--------+
|   101 |   24 | Sachin | Science    |   480 | EHS    |
|   102 |   14 | Rahul  | Commerce   |   410 | EHS    |
|   105 |   24 | Nelson | Humanities |   385 | EHS    |
|   106 |    8 | Joseph | Commerce   |   350 | EHS    |
|   109 |    2 | Bincy  | Science    |   300 | EHS    |
+-------+------+--------+------------+-------+--------+
5 rows in set (0.00 sec)
```

B. Display all the students in the Commerce batch who are failed.

```
mysql> SELECT * FROM STUDENT_RELATION WHERE Batch = 'Commerce' AND Result = 'NHS';
+-------+------+--------+----------+-------+--------+
| AdmNo | Roll | Name   | Batch    | Marks | Result |
+-------+------+--------+----------+-------+--------+
|   104 |   12 | Mahesh | Commerce |   180 | NHS    |
+-------+------+--------+----------+-------+--------+
1 row in set (0.00 sec)
```

C. Display all the students in the batch Science or Commerce.

```
mysql> SELECT * FROM STUDENT_RELATION WHERE Batch IN ('Science', 'Commerce');
+-------+------+--------+----------+-------+--------+
| AdmNo | Roll | Name   | Batch    | Marks | Result |
+-------+------+--------+----------+-------+--------+
|   101 |   24 | Sachin | Science  |   480 | EHS    |
|   102 |   14 | Rahul  | Commerce |   410 | EHS    |
|   104 |   12 | Mahesh | Commerce |   180 | NHS    |
|   106 |    8 | Joseph | Commerce |   350 | EHS    |
|   109 |    2 | Bincy  | Science  |   300 | EHS    |
+-------+------+--------+----------+-------+--------+
5 rows in set (0.00 sec)
```

D. To Display admission number and name of students who are Eligible for Higher Studies.

```
mysql> SELECT AdmNo, Name FROM STUDENT_RELATION WHERE Result = 'EHS';
+-------+--------+
| AdmNo | Name   |
+-------+--------+
|   101 | Sachin |
|   102 | Rahul  |
|   105 | Nelson |
|   106 | Joseph |
|   109 | Bincy  |
+-------+--------+
5 rows in set (0.00 sec)
```

E. To Display name and marks of those students in the Humanities batch who are Not eligible for

Higher Studies.

```
mysql> SELECT Name, Marks FROM STUDENT_RELATION WHERE Batch = 'Humanities' AND Result = 'NHS';
+---------+-------+
| Name    | Marks |
+---------+-------+
| Fathima |   200 |
| Shaji   |   205 |
+---------+-------+
2 rows in set (0.00 sec)
```

Q3

artists Example Input:

artist_id	artist_name	label_owner
101	Ed Sheeran	Warner Music Group
120	Drake	Warner Music Group
125	Bad Bunny	Rimas Entertainment

songs Example Input:

song_id	artist_id	name
55511	101	Perfect
45202	101	Shape of You
22222	120	One Dance
19960	120	Hotline Bling

Do the following

```
mysql> CREATE TABLE ARTISTS (
    ->     artist_id INT,
    ->     artist_name VARCHAR(20),
    ->     label_owner VARCHAR(20)
    -> );
Query OK, 0 rows affected (0.07 sec)

mysql> CREATE TABLE ARTISTS (
    ->     artist_id INT,
    ->     artist_name VARCHAR(20),
    ->     label_owner VARCHAR(20)
    -> );
ERROR 1050 (42S01): Table 'artists' already exists
mysql> INSERT INTO ARTISTS (artist_id, artist_name, label_owner) VALUES
    -> (101, 'Ed Sheeran', 'Warner Music Group'),
    -> (120, 'Drake', 'Warner Music Group'),
    -> (125, 'Bad Bunny', 'Rimas Entertainment');
Query OK, 3 rows affected (0.02 sec)
Records: 3  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM ARTISTS;
+-----------+-------------+---------------------+
| artist_id | artist_name | label_owner         |
+-----------+-------------+---------------------+
|       101 | Ed Sheeran  | Warner Music Group  |
|       120 | Drake       | Warner Music Group  |
|       125 | Bad Bunny   | Rimas Entertainment |
+-----------+-------------+---------------------+
3 rows in set (0.00 sec)
```

```
mysql> CREATE TABLE SONGS (
    ->     song_id INT,
    ->     artist_id INT,
    ->     name VARCHAR(255)
    -> );
Query OK, 0 rows affected (0.09 sec)

mysql>
mysql> INSERT INTO SONGS (song_id, artist_id, name) VALUES
    -> (55511, 101, 'Perfect'),
    -> (45202, 101, 'Shape of You'),
    -> (22222, 120, 'One Dance'),
    -> (19960, 120, 'Hotline Bling');
Query OK, 4 rows affected (0.03 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> SELECT * FROM sONGS;
+---------+-----------+---------------+
| song_id | artist_id | name          |
+---------+-----------+---------------+
|   55511 |       101 | Perfect       |
|   45202 |       101 | Shape of You  |
|   22222 |       120 | One Dance     |
|   19960 |       120 | Hotline Bling |
+---------+-----------+---------------+
4 rows in set (0.00 sec)
```

A. Combines info from the artists AND songs table.

```
mysql> SELECT *
    -> FROM ARTISTS
    -> CROSS JOIN SONGS ON ARTISTS.artist_id = SONGS.artist_id;
+-----------+-------------+--------------------+---------+-----------+---------------+
| artist_id | artist_name | label_owner        | song_id | artist_id | name          |
+-----------+-------------+--------------------+---------+-----------+---------------+
|       101 | Ed Sheeran  | Warner Music Group |   55511 |       101 | Perfect       |
|       101 | Ed Sheeran  | Warner Music Group |   45202 |       101 | Shape of You  |
|       120 | Drake       | Warner Music Group |   22222 |       120 | One Dance     |
|       120 | Drake       | Warner Music Group |   19960 |       120 | Hotline Bling |
+-----------+-------------+--------------------+---------+-----------+---------------+
4 rows in set (0.00 sec)
```

B. Returns only the rows with matching values from both tables.

```
mysql> SELECT *FROM ARTISTS INNER JOIN SONGS ON ARTISTS.artist_id = SONGS.artist_id;
+-----------+-------------+--------------------+---------+-----------+---------------+
| artist_id | artist_name | label_owner        | song_id | artist_id | name          |
+-----------+-------------+--------------------+---------+-----------+---------------+
|       101 | Ed Sheeran  | Warner Music Group |   55511 |       101 | Perfect       |
|       101 | Ed Sheeran  | Warner Music Group |   45202 |       101 | Shape of You  |
|       120 | Drake       | Warner Music Group |   22222 |       120 | One Dance     |
|       120 | Drake       | Warner Music Group |   19960 |       120 | Hotline Bling |
+-----------+-------------+--------------------+---------+-----------+---------------+
4 rows in set (0.00 sec)
```

C. Returns all the rows from the left table and the matching rows from the right table.

```
mysql> SELECT ARTISTS.artist_id, ARTISTS.artist_name, ARTISTS.label_owner, SONGS.song_id, SONGS.name
    -> FROM ARTISTS
    -> LEFT JOIN SONGS ON ARTISTS.artist_id = SONGS.artist_id;
+-----------+-------------+--------------------+---------+---------------+
| artist_id | artist_name | label_owner        | song_id | name          |
+-----------+-------------+--------------------+---------+---------------+
|       101 | Ed Sheeran  | Warner Music Group |   45202 | Shape of You  |
|       101 | Ed Sheeran  | Warner Music Group |   55511 | Perfect       |
|       120 | Drake       | Warner Music Group |   19960 | Hotline Bling |
|       120 | Drake       | Warner Music Group |   22222 | One Dance     |
|       125 | Bad Bunny   | Rimas Entertainment |    NULL | NULL          |
+-----------+-------------+--------------------+---------+---------------+
5 rows in set (0.00 sec)
```

D. Returns all the rows from the right table and the matching rows from the left table.

```
mysql> SELECT ARTISTS.artist_id, ARTISTS.artist_name, ARTISTS.label_owner, SONGS.song_id, SONGS.name
    -> FROM ARTISTS
    -> RIGHT JOIN SONGS ON ARTISTS.artist_id = SONGS.artist_id;
+-----------+-------------+--------------------+---------+---------------+
| artist_id | artist_name | label_owner        | song_id | name          |
+-----------+-------------+--------------------+---------+---------------+
|       101 | Ed Sheeran  | Warner Music Group |   55511 | Perfect       |
|       101 | Ed Sheeran  | Warner Music Group |   45202 | Shape of You  |
|       120 | Drake       | Warner Music Group |   22222 | One Dance     |
|       120 | Drake       | Warner Music Group |   19960 | Hotline Bling |
+-----------+-------------+--------------------+---------+---------------+
4 rows in set (0.00 sec)
```