

**DataBase Management System**  
**ASSIGNMENT**

**Name: Mehul jhunhunwala**

**Roll\_No: 20221428 Exam Roll: 22020570025**

**Subject: DBMS**

**Semester: 4th Semester**

**Date of Submission: 01/05/2024**

**Course: BSc (hons) Computer Science**  
**Ramanujan College**

**Given by: Kamlesh Raghuvanshi Sir**

- I. Create and use the following student-society database schema for a college to answer the given (sample) queries using the standalone SQL editor.

STUDENT	<u>Roll No</u>	StudentName	Course	DOB
	Char(6)	Varchar(20)	enumerate	Date

SOCIETY	<u>SocID</u>	SocName	MentorName	TotalSeats
	Char(6)	enumerate	Varchar(15)	Unsigned int

ENROLLMENT	<u>Roll No</u>	<u>SID</u>	DateOfEnrollment
	Char(6)	Char(6)	Date

Here Rollno (ENROLLMENT) and SID (ENROLLMENT) are foreign keys.

Underlined attributes make primary key of the respective table.

SocName, course is to be of Enum type

## 1.Creating The Student Table.

```

MySQL 8.0 Command Line Client - Unicode
Enter password: ****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 8.0.36 MySQL Community Server - GPL

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use harsh
Database changed
mysql> create table student(Roll_No char(6) Primary Key Not NULL,Student_Name varchar(20) Not Null,Course enum("Computer Science","Botany","Physics","Electronics"),Date_of_Birth Date Not Null);
Query OK, 0 rows affected (0.38 sec)
  
```

## Inserting Values in it:

```

mysql> insert into student values("AE1218","Harsh Arora","Computer Science","2003-08-27");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1206","Aditya Maurya","Computer Science","2003-10-06");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1244","Akshat Bhargav","Physics","2005-05-12");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1236","Mehul Jhunjhunwala","Computer Science","2003-04-04");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1214","Dhrub","Botany","2003-10-14");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1248","Nikil","Electronics","2003-07-22");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1236","Mosharraf Jameel","Computer Science","2003-12-25");
ERROR 1062 (23000): Duplicate entry 'AE1236' for key 'student.PRIMARY'
mysql> insert into student values("AE1227","Mosharraf Jameel","Computer Science","2003-12-25");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1201","Eichiro Oda","Physics","2003-01-01");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1269","Nico Robin","Computer Science","2003-02-08");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1258","Hardik Baniya","Electronics","2003-07-19");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1249","Sanji Vinsmoke","Physics","2003-03-02");
Query OK, 1 row affected (0.01 sec)

mysql> insert into student values("AE1275","Zoro Roronoa","Botany","2003-11-11");
Query OK, 1 row affected (0.01 sec)
  
```



## Updating Values in it:

```
mysql> update student set Course="Physics" where Roll_No="AE1237";
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update student set Date_of_Birth="2003-10-24" where Roll_No="AE1206";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update student set Roll_No="AE2890" where Student_Name="Nikil";
Query OK, 1 row affected (0.05 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select*from student;
```

Roll_No	Student_Name	Course	Date_of_Birth
AE1201	Eichiro Oda	Physics	2003-01-01
AE1206	Aditya Maurya	Computer Science	2003-10-24
AE1214	Dhrub	Botany	2003-10-14
AE1218	Harsh Arora	Computer Science	2003-08-27
AE1236	Mehul BhunjunWala	Computer Science	2003-04-04
AE1237	Mosharraf Jameel	Physics	2003-12-25
AE1244	Akshat Bhargav	Physics	2005-05-12
AE1249	Sanji Vinsmoke	Physics	2003-03-02
AE1258	Hardik Baniya	Electronics	2003-07-19
AE1269	Nico Robin	Computer Science	2003-02-08
AE1275	Zoro Roronoa	Botany	2003-11-11
AE2890	Nikil	Electronics	2003-07-22

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

## Altering values:

```
mysql> alter table student modify Student_Name varchar(30);
Query OK, 0 rows affected (0.23 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table student modify Course enum("Computer Science","Botany","Physics","Electronics","B.COM");
Query OK, 0 rows affected (0.02 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
Physics, Electronics" ); do this 1
mysql> alter table student modify Course enum("Computer Science","Botany","Physics","Electronics");
Query OK, 0 rows affected (0.22 sec)
Records: 0 Duplicates: 0 Warnings: 0
```



## 2. Creating Society Table:

```
mysql> create table society(Society_ID char(6) Primary Key Not NULL,Society_Name enum("Dwani","Dhun","PicFie","NSS"),Mentor_Name varchar(20) Not Null>Total_Seats int unsigned);
Query OK, 0 rows affected (0.08 sec)
```

## Inserting Values in it:

```
mysql> create table society(Society_ID char(6) Primary Key Not NULL,Society_Name enum("Dwani","Dhun","PicFie","NSS","Play_Them","Game_Fie","Watch_Club","Book_with_UVA","Entrapinur_LAB","NCC");
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> insert into society values("203457","Dwani","MR.Rahul","10");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203478","Dhun","MR.Sanjay","15");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203480","PicFie","MR.Singh","12");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203472","NSS","MRS.Khan","20");
Query OK, 1 row affected (0.01 sec)

mysql> alter table society modify Society_Name enum("Dwani","Dhun","PicFie","NSS","Play_Them","Game_Fie","Watch_Club","Book_with_UVA","Entrapinur_LAB","NCC");
Query OK, 0 rows affected (0.09 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> insert into society values("203499","Play_Them","MR.Bhargav","15");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203475","Game_Fie","MR.Chaudhary","24");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203488","Watch_Club","MR.Sadri","25");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203484","Book_with_UVA","MRS.Sharma","30");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203481","Entrapinur_LAB","MRS.Jinx","19");
Query OK, 1 row affected (0.01 sec)

mysql> insert into society values("203494","NCC","MRS.Ackerman","23");
Query OK, 1 row affected (0.01 sec)

mysql> select*from society;
```

Society_ID	Society_Name	Mentor_Name	Total_Seats
203457	Dwani	MR.Rahul	10
203472	NSS	MRS.Khan	20
203475	Game_Fie	MR.Chaudhary	24
203478	Dhun	MR.Sanjay	15
203480	PicFie	MR.Singh	12
203481	Entrapinur_LAB	MRS.Jinx	19
203484	Book_with_UVA	MRS.Sharma	30
203488	Watch_Club	MR.Sadri	25
203494	NCC	MRS.Ackerman	23
203499	Play_Them	MR.Bhargav	15

10 rows in set (0.00 sec)

mysql>

## Updating Values in it:

```
mysql> update society set Total_Seats="30" where Society_Name="NCC";
Query OK, 1 row affected (0.17 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update society set Mentor_Name="MR.Katakuri" where Society_Name="Dhun";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> select*from society;
```

Society_ID	Society_Name	Mentor_Name	Total_Seats
203457	Dwani	MR.Rahul	10
203472	NSS	MRS.Khan	20
203475	Game_Fie	MR.Chaudhary	24
203478	Dhun	MR.Katakuri	15
203480	PicFie	MR.Singh	12
203481	Entrapinur_LAB	MRS.Jinx	19
203484	Book_with_UVA	MRS.Sharma	30
203488	Watch_Club	MR.Sadri	25
203494	NCC	MRS.Ackerman	30
203499	Play_Them	MR.Bhargav	15

10 rows in set (0.00 sec)

## 3. Creating Enrollment Table:

```
mysql> create table enrollment(Roll_No char(6) primary key,Society_ID char(6) not null ,Date_of_Enrollment date);
Query OK, 0 rows affected (0.14 sec)

mysql> alter table enrollment add constraint sr1 foreign key(Roll_No) references student(Roll_No);
Query OK, 0 rows affected (0.17 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table enrollment add constraint s11 foreign key(Society_ID) references society(Society_ID);
Query OK, 0 rows affected (0.14 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

## Inserting values in it:

```
mysql> insert into enrollment values("AE1201","203484","2023-08-10");
Query OK, 1 row affected (0.00 sec)

mysql> insert into enrollment values("AE1214","203472","2023-08-23");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1244","203472","2023-09-22");
Query OK, 1 row affected (0.03 sec)

mysql> insert into enrollment values("AE1275","203481","2022-08-31");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1269","203481","2021-07-24");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE2890","203481","2021-09-12");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1206","203480","2022-07-02");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1218","203480","2022-05-09");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1236","203484","2023-08-27");
Query OK, 1 row affected (0.00 sec)

mysql> insert into enrollment values("AE1258","203499","2020-07-19");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1249","203481","2022-06-16");
Query OK, 1 row affected (0.01 sec)

mysql> insert into enrollment values("AE1237","203404","2023-08-22");
Query OK, 1 row affected (0.00 sec)

mysql> select*from enrollment;
+-----+
| Roll_No | Society_ID | Date_of_Enrollment |
+-----+
| AE1201 | 203484 | 2023-08-10 |
| AE1206 | 203480 | 2022-07-02 |
| AE1214 | 203472 | 2023-08-23 |
| AE1218 | 203480 | 2022-05-09 |
| AE1236 | 203484 | 2023-08-27 |
| AE1237 | 203404 | 2023-08-22 |
| AE1244 | 203472 | 2023-09-22 |
| AE1249 | 203481 | 2022-06-16 |
| AE1258 | 203499 | 2020-07-19 |
| AE1269 | 203481 | 2021-07-24 |
| AE1275 | 203481 | 2022-08-31 |
| AE2890 | 203481 | 2021-09-12 |
+-----+
12 rows in set (0.00 sec)
```

## Updating Values:

```
mysql> update enrollment set Date_of_Enrollment="2023-09-19" where Roll_No="AE1201";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update enrollment set Date_of_Enrollment="2022-09-24" where Roll_No="AE1237";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select*from enrollment;
+-----+
| Roll_No | Society_ID | Date_of_Enrollment |
+-----+
| AE1201 | 203484 | 2023-09-19 |
| AE1206 | 203480 | 2022-07-02 |
| AE1214 | 203472 | 2023-08-23 |
| AE1218 | 203480 | 2022-05-09 |
| AE1236 | 203484 | 2023-08-27 |
| AE1237 | 203404 | 2022-09-24 |
| AE1244 | 203472 | 2023-09-22 |
| AE1249 | 203481 | 2022-06-16 |
| AE1258 | 203499 | 2020-07-19 |
| AE1269 | 203481 | 2021-07-24 |
| AE1275 | 203481 | 2022-08-31 |
| AE2890 | 203481 | 2021-09-12 |
+-----+
12 rows in set (0.00 sec)
```



## Queries to do:

### 1. > Retrieve Names of Students Enrolled in Any Society:

Output:

```
mysql> select Student_Name from student inner join enrollment on student.Roll_No=enrollment.Roll_No;
+-----+
| Student_Name |
+-----+
| Elchiro Oda  |
| Aditya Maurya |
| Dhrub       |
| Harsh Arora  |
| Mehul Bhurchandala |
| Noshareef Jameel |
| Akshat Bhargav |
| Sanjiv Vinsmoke |
| Hardik Baniya |
| Nite Robin   |
| Zoro Roronoa |
| Nikhil      |
+-----+
12 rows in set (0.00 sec)
```

### 2. > Retrieve All Society

Names: Output:

```
mysql> select Society_Name from society;
+-----+
| Society_Name |
+-----+
| Dwanji       |
| NSS         |
| Game_Fie    |
| Dhun        |
| PicFie      |
| Entrapinur_LAB |
| Book_with_OVA |
| Watch_Club   |
| NCC         |
| Play_Them    |
+-----+
10 rows in set (0.00 sec)
```

### 3. >Retrieve Students Name Starting With Letter “A”.

**Output:**

```
mysql> select*from student where Student_Name like 'A%';
```

Roll_No	Student_Name	Course	Date_of_Birth
AE1206	Aditya Maurya	Computer Science	2003-10-24
AE1244	Akshat Bhargav	Physics	2005-05-12

2 rows in set (0.00 sec)

### 4. >Retrieve Students Name Studying In Courses “Computer Science” or “Electronics”.

**Output:**

```
mysql> select*from student where Course="Computer Science" or Course="Electronics";
```

Roll_No	Student_Name	Course	Date_of_Birth
AE1206	Aditya Maurya	Computer Science	2003-10-24
AE1218	Harsh Arora	Computer Science	2003-08-27
AE1236	Mehul JhunJhunwala	Computer Science	2003-04-04
AE1258	Hardik Baniya	Electronics	2003-07-19
AE1269	Nico Robin	Computer Science	2003-02-08
AE2890	Nikil	Electronics	2003-07-22

6 rows in set (0.00 sec)

### 5. >Retrieve Students Name Whose Roll No Starts With “A” and Ends With “6”,

**Output:**

```
mysql> select*from student where Roll_No like 'A%' and Roll_No like '%6';
```

Roll_No	Student_Name	Course	Date_of_Birth
AE1206	Aditya Maurya	Computer Science	2003-10-24
AE1236	Mehul JhunJhunwala	Computer Science	2003-04-04

2 rows in set (0.00 sec)

## 6. Find Society Details With More Than 'N' Seats And 'N' is Input By The User:

### Output:

```
mysql> select Society_Name from society where Total_Seats> #n ;
-> 15;
+-----+
| Society_Name |
+-----+
| NSS          |
| Game_Fie     |
| Entrapinur_LAB |
| Book_With_UVA |
| Watch_Club   |
| NCC          |
+-----+
6 rows in set (0.02 sec)
```

## 7. Update Society Table For A Mentor Name of a Specific Society:

### Output:

```
mysql> update society set Mentor_Name="MR.Katakuri" where Society_Name="Dhun";
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

## 8. Find Society Names Where More Than 3 Students Have Enrolled:

### Output:

```
mysql> SELECT Society_Name
-> from(select s2.Society_Name,count(*) as num_enrolled
-> FROM enrollment AS s1
-> INNER JOIN society AS s2 ON s1.Society_ID= s2.Society_ID
-> GROUP BY s2.Society_Name
-> ) AS society_counts
-> WHERE num_enrolled > 3;
+-----+
| Society_Name |
+-----+
| Entrapinur_LAB |
+-----+
1 row in set (0.00 sec)
```



## 9>Find Name of The Youngest Student Enrolled In Entrepreneurship\_Lab Society:

**Output:**

```
mysql> SELECT s.Student_Name
-> FROM student s
-> JOIN enrollment sm ON s.Roll_No = sm.Roll_No
-> WHERE sm.society_id = 203481
-> ORDER BY s.Date_of_Birth ASC
-> limit 1;
+-----+
| Student_Name |
+-----+
| Nico Robin   |
+-----+
1 row in set (0.00 sec)
```

## 10.>Find Name of The Most popular Society(number of people enrolled in it):

**Output:**

```
mysql> SELECT s.society_name, COUNT(m.roll_no) AS num_members
-> FROM society s
-> LEFT JOIN enrollment m ON s.Society_ID = m.Society_ID
-> GROUP BY s.society_name
-> ORDER BY num_members DESC
-> limit 1;
+-----+-----+
| society_name | num_members |
+-----+-----+
| Entrapinur_LAB | 4 |
+-----+-----+
1 row in set (0.00 sec)
```

## 11.>. Find the name of two least popular societies (on the basis of enrolled students).

**Output:**

```
mysql> SELECT Society_Name from society where Society_ID in (select Society_ID from enrollment group by Society_ID order by count(Roll_NO)) limit 2;
+-----+
| Society_Name |
+-----+
| NSS          |
| PicFie       |
+-----+
2 rows in set (0.00 sec)
```



## **12.> Find the student names who are not enrolled in any society.**

### **Output:**

```
mysql> SELECT Student_Name from student where Roll_No Not in (Select Distinct Roll_No from Enrollment);
+-----+
| Student_Name |
+-----+
| Zoro Roronoa |
+-----+
1 row in set (0.01 sec)
```

## **13.> Find the student names enrolled in at least two societies.**

### **Output:**

```
mysql> select Student_Name from student where Roll_No in (select Roll_No from enrollment group by Roll_No Having Count(Society_ID)>=2);
Empty set (0.00 sec)
```

## **14.> Find society names In which maximum students are enrolled .**

### **Output:**

```
mysql> select Society_Name from society where Society_ID in (select Society_ID from enrollment group by Society_ID order by count(Roll_No) limit 1;
+-----+
| Society_Name |
+-----+
| NSS          |
+-----+
1 row in set (0.00 sec)
```

**15. Find names of all students who have enrolled in any society and society names in which at least one student has enrolled .**

**Output:**

```
mysql> select Student_Name As Name from student where Roll_No in (Select Roll_No From enrollment) union select Society_Name As SName from society where Society_ID in (select Society_ID from enrollment);
```

Name
Dhrub
Akshat Bhargav
Aditya Maurya
Harsh Arora
Sanji Vinsmoke
Nico Robin
Nikil
Eichiro Oda
Mehul Jhunjhunwala
Mosharraf Jameel
Hardik Baniya
NSS
PicFie
Entrapinur_LAB
Book_With_UVA
NCC
Play_Them

17 rows in set (0.00 sec)

**16. Find names of students who are enrolled in any of the three societies 'NSS', 'PicFie' and 'Play\_Them'.**

**Output:**

```
mysql> select distinct Student_Name from student where Roll_No in (Select Roll_No from enrollment where Society_ID in (Select Society_ID from society where Society_Name in ("NSS","Dwani","PicFie")));
```

Student_Name
Dhrub
Akshat Bhargav
Aditya Maurya
Harsh Arora

4 rows in set (0.00 sec)



**17. Find society names such that its mentor has a name with 'Singh' in it.**

**Output:**

```
mysql> select Society_Name from society where Mentor_Name like "%Singh%";
+-----+
| Society_Name |
+-----+
| PicFie       |
+-----+
1 row in set (0.01 sec)
```

**18. Find the society names in which the number of enrolled students is only 10% of its capacity.**

**Output:**

```
mysql> SELECT s.Society_Name
-> FROM society s
-> LEFT JOIN (
->   SELECT Society_ID, COUNT(Roll_No) AS con
->   FROM enrollment
->   GROUP BY Society_ID
-> ) e ON s.Society_ID = e.Society_ID
-> WHERE e.con is NULL or e.con<0.001 * s.Total_Seats;
+-----+
| Society_Name |
+-----+
| Dwani        |
| Game_Fie     |
| Dhun         |
| Watch_Club   |
+-----+
4 rows in set (0.00 sec)
```



## 19. Display the vacant seats for each society.

### Output:

```
mysql> SELECT Society_Name,Total_seats -(Select count(Roll_No) from enrollment where enrollment.Society_ID=society.Society_ID) as Vacant_Seats from society;
```

Society_Name	Vacant_Seats
Dwani	11
NSS	20
Game_Fie	26
Dhun	17
PicFie	11
Entrapinur_LAB	18
Book_With_UVA	31
Watch_Club	28
NCC	32
Play_Them	16

10 rows in set (0.00 sec)

## 20. Increment Total Seats of each society by 10%.

### Output:

```
mysql> update society Set Total_Seats=(Total_Seats+Total_Seats*(10/100));
Query OK, 10 rows affected (0.03 sec)
Rows matched: 10  Changed: 10  Warnings: 0

mysql> select*from society;
```

Society_ID	Society_Name	Mentor_Name	Total_Seats
203457	Dwani	MR. Rahul	11
203472	NSS	MRS. Khan	22
203475	Game_Fie	MR. Chaudhary	26
203478	Dhun	MR. Katakuri	17
203480	PicFie	MR. Singh	13
203481	Entrapinur_LAB	MRS. Jinx	21
203484	Book_With_UVA	MRS. Sharma	33
203488	Watch_Club	MR. Sadri	28
203494	NCC	MRS. Ackerman	33
203499	Play_Them	MR. Bhargav	17

10 rows in set (0.00 sec)



## 21. Add the enrollment fees paid ('yes'/'No') field in the enrollment table.

### Output:

```
mysql> alter table enrollment add Fees_Paid enum("YES","NO") Default "NO";
Query OK, 0 rows affected (0.03 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> update enrollment set Fees_Paid =YES where Roll_NO=AE1201;
ERROR 1054 (42S22): Unknown column 'AE1201' in 'where clause'
mysql> update enrollment set Fees_Paid =YES where Roll_NO='AE1201';
ERROR 1054 (42S22): Unknown column 'YES' in 'field list'
mysql> update enrollment set Fees_Paid ='YES' where Roll_NO='AE1201';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update enrollment set Fees_Paid ='YES' where Roll_NO='AE1249';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update enrollment set Fees_Paid ='YES' where Roll_NO='AE1258';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update enrollment set Fees_Paid ='YES' where Roll_NO='AE2890';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update enrollment set Fees_Paid ='YES' where Roll_NO='AE1214';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> select*from enrollment;
```

Roll_No	Society_ID	Date_of_Enrollment	Fees_Paid
AE1201	203484	2023-09-29	YES
AE1206	203480	2022-07-02	NO
AE1214	203472	2023-08-23	YES
AE1218	203480	2022-05-09	NO
AE1236	203484	2023-08-27	NO
AE1237	203494	2022-09-24	NO
AE1244	203472	2023-09-22	NO
AE1249	203481	2022-06-16	YES
AE1258	203499	2020-07-19	YES
AE1269	203481	2021-07-24	NO
AE2890	203481	2021-09-12	YES

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



## 22. Update date of enrollment of society id '203484' to '2018-01-15', '203480' to current date and '203499' to '2018-01-02'.

### Output:

```
mysql> update enrollment set Date_of_Enrollment='2018-01-15' where Society_ID='203484';
Query OK, 2 rows affected (0.02 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> update enrollment set Date_of_Enrollment=curdate() where Society_ID='203480';
Query OK, 2 rows affected (0.01 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> update enrollment set Date_of_Enrollment=2018-01-02 where Society_ID='203499';
ERROR 1292 (22007): Incorrect date value: '2015' for column 'Date_of_Enrollment' at row 1
mysql> update enrollment set Date_of_Enrollment=2018-01-10 where Society_ID='203499';
ERROR 1292 (22007): Incorrect date value: '2007' for column 'Date_of_Enrollment' at row 1
mysql> update enrollment set Date_of_Enrollment='2018-01-10' where Society_ID='203499';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select*from enrollment;
```

Roll_No	Society_ID	Date_of_Enrollment	Fees_Paid
AE1201	203484	2018-01-15	YES
AE1206	203480	2024-03-03	NO
AE1214	203472	2023-08-23	YES
AE1218	203480	2024-03-03	NO
AE1236	203484	2018-01-15	NO
AE1237	203494	2022-09-24	NO
AE1244	203472	2023-09-22	NO
AE1249	203481	2022-06-16	YES
AE1258	203499	2018-01-10	YES
AE1269	203481	2021-07-24	NO
AE2890	203481	2021-09-12	YES

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



## 23. Create a view to keep track of society names with the total number of students enrolled in it.

### Output:

```
mysql> create view society_members as select Society_Name,(select count(*) from enrollment where enrollment.Society_ID=society.Society_ID) as Total_Students from society;
Query OK, 0 rows affected (0.01 sec)

mysql> select*from Total_student;
ERROR 1146 (42S02): Table 'harsh.total_student' doesn't exist
mysql> select*from Total_students;
ERROR 1146 (42S02): Table 'harsh.total_students' doesn't exist
mysql> select*from society_members;
```

Society_Name	Total_Students
Dwani	0
NSS	2
Game_Fie	0
Dhun	0
PicFie	2
Entrapinur_LAB	3
Book_With_UVA	2
Watch_Club	0
NCC	1
Play_Them	1

```
10 rows in set (0.01 sec)
```

## 24. Find student names enrolled in all the societies.

### Output:

```
mysql> select Student_Name from student s join enrollment e on s.Roll_No = e.Roll_No join society on society.Society_ID=e.Society_ID group by e.Roll_No having count(*) in (select count(*) from Society);
Empty set (0.03 sec)
```





## 25. Count the number of societies with more than 2 students enrolled in it.

### Output:

```
mysql> select count(*) as Number_of_Societies from society where Society_ID in (select Society_ID from enrollment group by Society_ID having count(Roll_No)>2);
+-----+
| Number_of_Societies |
+-----+
| 1 |
+-----+
1 row in set (0.00 sec)
```

## 26. Add column Mobile number in student table with default value '9999999999'.

### Output:

```
mysql> alter table student add Mobile_Number int(10) unsigned default 9999999999;
ERROR 1067 (42000): Invalid default value for 'Mobile_Number'
mysql> alter table student add Mobile_Number int(10) unsigned default 9999999999;
Query OK, 0 rows affected, 1 warning (0.05 sec)
Records: 0 Duplicates: 0 Warnings: 1

mysql> update student set Mobile_Number='9876543210' where Roll_No='AE1269';
ERROR 1264 (22003): Out of range value for column 'Mobile_Number' at row 1
mysql> update student set Mobile_Number='987654321' where Roll_No='AE1269';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update student set Mobile_Number='987678902' where Roll_No='AE1201';
Query OK, 1 row affected (0.02 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update student set Mobile_Number='9870000000' where Roll_No='AE1218';
ERROR 1264 (22003): Out of range value for column 'Mobile_Number' at row 1
mysql> update student set Mobile_Number='9870000000' where Roll_No='AE1218';
Query OK, 1 row affected (0.01 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select* from student;
+-----+-----+-----+-----+-----+
| Roll_No | Student_Name | Course | Date_of_Birth | Mobile_Number |
+-----+-----+-----+-----+-----+
| AE1201 | Eichiro Oda | Physics | 2003-01-01 | 987678902 |
| AE1206 | Aditya Maurya | Computer Science | 2003-10-24 | 999999999 |
| AE1214 | Dhruv | Botany | 2003-10-14 | 999999999 |
| AE1218 | Harsh Arora | Computer Science | 2003-08-27 | 987000000 |
| AE1236 | Mehul JhunJhunWala | Computer Science | 2003-04-04 | 999999999 |
| AE1237 | Mosharraf Jameel | Physics | 2003-12-25 | 999999999 |
| AE1244 | Akshat Bhargav | Physics | 2005-05-12 | 999999999 |
| AE1249 | Sanji Vinsmoke | Physics | 2003-03-02 | 999999999 |
| AE1258 | Hardik Baniya | Electronics | 2003-07-19 | 999999999 |
| AE1269 | Nico Robin | Computer Science | 2003-02-08 | 987654321 |
| AE1275 | Zoro Roronoa | Botany | 2003-11-11 | 999999999 |
| AE2890 | Nikil | Electronics | 2003-07-22 | 999999999 |
+-----+-----+-----+-----+-----+
12 rows in set (0.00 sec)
```



**27. Find the total number of students whose age is > 20 years.**

**Output:**

```
mysql> select count(Student_Name) as Number_of_Students_Age from student where timestampdiff(year,Date_of_Birth,curdate())>20;
+-----+
| Number_of_Students_Age |
+-----+
| 3 |
+-----+
1 row in set (0.01 sec)
```

**28. Find names of students who are born in 2003 and are enrolled in at least one society.**

**Output:**

```
mysql> select Student_Name from student where year(Date_of_Birth)=2003 and Roll_No in (select Roll_No from enrollment);
+-----+
| Student_Name |
+-----+
| Dhrub        |
| Aditya Maurya |
| Harsh Arora  |
| Sanji Vinsmoke |
| Nico Robin   |
| Nikil        |
| Eichiro Oda  |
| Mehul JhunJhunWala |
| Mosharraaf Jameel |
| Hardik Baniya |
+-----+
10 rows in set (0.01 sec)
```



## 29. Count all societies whose name starts with 'N' and ends with 'S' and at least 1 students are enrolled in the society.

### Output:

```
mysql> select count(*) as Society_NAME from society where Society_Name like "N%S" and Society_ID in (select Society_ID from enrollment group by Society_ID having count(Roll_No) >= 2);
```

Society_NAME
1

1 row in set (0.00 sec)

## 30. Display the following information:

- Society name
- Mentor name
- Total Capacity
- Total Enrolled
- Unfilled Seats

### Output:

```
mysql> select Society_Name,Mentor_Name,Total_Seats,count(Roll_No) as Total_Enrolled,Total_Seats-count(Roll_No) as Unfilled_Seats from society left join enrollment on enrollment.Society_ID = society.Society_ID group by Mentor_Name,Society_Name,Total_Seats;
```

Society_Name	Mentor_Name	Total_Seats	Total_Enrolled	Unfilled_Seats
Dwani	MR.Rahul	11	0	11
NSS	MRS. Khan	22	2	20
Game_Fie	MR. Chaudhary	26	0	26
Dhun	MR. Katakuri	17	0	17
PicFie	MR. Singh	13	2	11
Entrapinur_LAB	MRS. Jinx	21	3	18
Book_With_UVA	MRS. Sharma	33	2	31
Watch_Club	MR. Sadri	28	0	28
NCC	MRS. Ackerman	33	1	32
Play_Them	MR. Bhargav	17	1	16

10 rows in set (0.00 sec)

**Q1.Do the following database administration commands:  
create user, create role, grant privileges to a role, revoke  
privileges from a role, create index:**

```
mysql> CREATE USER DemonBoyHash IDENTIFIED BY '6063';  
Query OK, 0 rows affected (0.27 sec)  
  
mysql> CREATE ROLE DBMS;  
Query OK, 0 rows affected (0.02 sec)  
  
mysql> GRANT SELECT, INSERT, UPDATE ON most_runs TO DemonBoyHash;  
Query OK, 0 rows affected (0.01 sec)
```

```
mysql> REVOKE INSERT, UPDATE ON most_runs FROM DemonBoyHash;  
Query OK, 0 rows affected (0.02 sec)
```

```
mysql> CREATE INDEX myindex ON society (Society_ID);  
Query OK, 0 rows affected (0.21 sec)  
Records: 0 Duplicates: 0 Warnings: 0
```

## Q2.Execute any five queries given in Q 1 of practical list on society database through a high-level language using ODBC connection.

```
import mysql.connector

conn = mysql.connector.connect(
    host="localhost",
    database="harsh",
    user="root",
    password="6063"
)

cursor = conn.cursor()

queries = [
    "SELECT * FROM Student where Student_Name like '%A';",
    "SELECT * FROM Student WHERE Course IN ('Computer Science', 'Electronics');",
    "SELECT Student_Name From Student where Roll_No Not In (Select Roll_No from Enrollment);",
    "SELECT Society_Name from Society Where Mentor_Name like '%Singh';",
    "SELECT Count(Student_Name) as Number_of_Students_Age from Student where timestampdiff(year,Date_of_Birth,curdate())>20;"
]

for query in queries:
    cursor.execute(query)
    rows = cursor.fetchall()
    for row in rows:
        print(row)
    print() # Empty Line between queries

# Close the cursor and the connection
cursor.close()
conn.close()
```

```
('AE1201', 'Eichiro Oda', 'Physics', datetime.date(2003, 1, 1), 987678902)
('AE1206', 'Aditya Maurya', 'Computer Science', datetime.date(2003, 10, 24), 999999999)
('AE1218', 'Harsh Arora', 'Computer Science', datetime.date(2003, 8, 27), 987000000)
('AE1236', 'Mehul JhunJhunWala', 'Computer Science', datetime.date(2003, 4, 4), 999999999)
('AE1258', 'Hardik Baniya', 'Electronics', datetime.date(2003, 7, 19), 999999999)
('AE1275', 'Zoro Roronoa', 'Botany', datetime.date(2003, 11, 11), 999999999)

('AE1206', 'Aditya Maurya', 'Computer Science', datetime.date(2003, 10, 24), 999999999)
('AE1218', 'Harsh Arora', 'Computer Science', datetime.date(2003, 8, 27), 987000000)
('AE1236', 'Mehul JhunJhunWala', 'Computer Science', datetime.date(2003, 4, 4), 999999999)
('AE1258', 'Hardik Baniya', 'Electronics', datetime.date(2003, 7, 19), 999999999)
('AE1269', 'Nico Robin', 'Computer Science', datetime.date(2003, 2, 8), 987654321)
('AE2890', 'Nikil', 'Electronics', datetime.date(2003, 7, 22), 999999999)

('Zoro Roronoa',)

('PicFie',)

(4,)
```