

NAME – Sayan Paul
ROLL - 21CS8056
REG - 21U10209
ASSIGNMENT - 4

1. Retrieve the names of students who are enrolled in either the Computer Science or Mathematics departments. (In)

```
[mysql> select s.sname
[      -> from student s
[      -> join programme p on s.pcode = p.pcode
[      -> join department d on p.dcode = d.dcode
[      -> where d.dname in ('Computer Science Department','Mathematics Department');
+-----+
| sname |
+-----+
| Navneet |
| Devanjana |
| Oishik |
| Ritabrata Das |
| Vedang (Honors) |
| Jaydeep |
| Vivek Gupta |
+-----+
7 rows in set (0.00 sec)
```

2. Find all courses with credit values between 3 and 5, both inclusive. (Between)

```
[mysql> select title,credit from course where credit between 3 and 5;
+-----+-----+
| title | credit |
+-----+-----+
| DSA | 5.00 |
| DLD | 5.00 |
| OS | 5.00 |
| DAA | 5.00 |
| Power Systems | 4.00 |
| control systems | 5.00 |
| VLSI | 4.00 |
| Basic Electronics | 5.00 |
| Antenna | 3.00 |
| Thermodynamics | 4.00 |
| Fluid Mechanics | 5.00 |
| Machine Design | 5.00 |
| Heat Transfer | 3.00 |
| Control Engineering | 4.00 |
| Network Analysis | 4.00 |
| Discrete Mathematics | 5.00 |
| Fuzzy Logic | 5.00 |
+-----+-----+
17 rows in set (0.00 sec)
```

3. Retrieve the names of students whose addresses contain the word 'pur'. (Like)

```
[mysql> select sname
[      -> from student
[      -> where address like '%pur%';
```

| sname |
|-----------------|
| Navneet |
| Vedang (Honors) |
| Suman Karmakar |
| Prince Maurya |

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

4. List all unique addresses of students. (Distinct)

```
[mysql> select distinct address  
-> from student;
```

| address |
|-------------------------|
| Jaipur,Rajasthan |
| Mumbai |
| Kolkata |
| Kashipur |
| Siliguri |
| Murshidabad,West Bengal |
| Delhi |
| Lucknow,Uttar Pradesh |
| Howrah |
| Purulia,West Bengal |
| Asansol,West Bengal |
| Kanpur |
| Ranchi,Jharkhand |

13 rows in set (0.00 sec)

5. Retrieve the names of students who achieved the top 5 grades in the 'ComputerScience' department. (Limit, Offset)

```
[mysql> select s.sname
[   -> from student s
[   -> join student_course sc on s.rollno = sc.rollno
[   -> join course c on sc.coursenum = c.coursenum
[   -> join department d on c.dcode = d.dcode
[   -> where d.dname = 'Computer Science Department'
[   -> order by sc.grade desc
[   -> limit 5;
```

```
+-----+
| sname          |
+-----+
| Ritabrata Das  |
| Devanjana      |
| Vedang (Honors)|
| Vedang (Honors)|
| Devanjana      |
+-----+
```

5 rows in set (0.00 sec)

6. Identify all teachers whose department affiliation is known (Not NULL).

```
[mysql> select tname  
[      -> from teacher  
[      -> where dcode is not null;
```

| tname |
|------------------|
| Suvrojit Das |
| Tanmoy De |
| Rajat Mahapatra |
| Mousami Saha |
| Subhrata Nandi |
| Nirmal Baran Hui |
| Sankar Mahato |
| Aritro De |
| Asit Baran Puri |
| Animesh Patari |
| Bibhash Sen |
| Durbal Mandal |
| Dipesh Das |
| Lakshmi Kant Dey |
| Pinaki Pal |

15 rows in set (0.00 sec)

7. Create a view named "TopStudents" that displays the names and total grades of students who scored above 90% in total. (View)

```
mysql> CREATE VIEW TopStudents AS
-> SELECT s.sname, SUM(sc.grade) AS TotalGrade
-> FROM student_course sc
-> INNER JOIN student s ON sc.rollno = s.rollno
-> GROUP BY s.rollno
-> HAVING AVG(sc.grade) > 90;
Query OK, 0 rows affected (0.00 sec)
```

```
[mysql> select * from TopStudents;
+-----+-----+
| sname          | TotalGrade |
+-----+-----+
| Vedang (Honors) |          453 |
| Richa Roy (Honors) |          283 |
| Satyam (Honors) |          295 |
+-----+-----+
3 rows in set (0.00 sec)
```

8. Delete the "TopStudents" view from the database. (Drop)

```
[mysql> drop view TopStudents;
Query OK, 0 rows affected (0.00 sec)
```

9. Remove all records from the "Grades" table without deleting the table structure. (Truncate)

```
[mysql> truncate table student_course;
Query OK, 0 rows affected (0.01 sec)

[mysql> select * from student_course;
Empty set (0.00 sec)
```