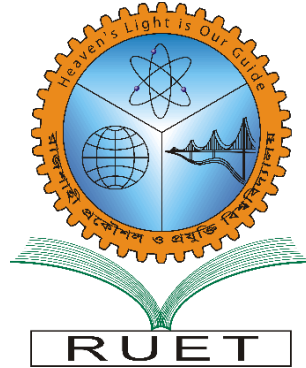


*“Heaven’s Light is Our Guide”*

Rajshahi University of Engineering & Technology, Rajshahi



Department of Electrical & Computer  
Engineering (ECE-2216)

Course Code: ECE-2216

Course Title: Database Systems Sessional

Experiment No: 02

Date of Submission: 01.10.2024

*Submitted To,*

**Oishi Jyoti**

Assistant Professor

Dept. Of ECE, RUET

*Submitted By,*

Name: Md. Fahmidun Nabi

Roll: 2110025

Session: 2021-2022

## Experiment No: 2

**Students Table**

student_id	student_name	age	GPA	department	year_of_admission	fees_paid	credits_earned	enrollment_status
1	Eleven	21	3.8	Engineering	2021	10000	120	active
2	Dustin	22	3.9	Science	2020	9000	110	active
3	Will	19	3.4	Business	2022	8500	95	active
4	Mike	23	3.7	Science	2021	9500	115	inactive
5	Max	20	3.5	Engineering	2020	12000	130	active
6	Eddie	22	4.0	Arts	2019	8000	140	active
7	Billy	24	2.9	Engineering	2022	5000	60	active
8	Alexei	25	3.2	Business	2018	7500	100	inactive
9	Steve	21	3.8	Science	2021	10500	120	active
10	Robin	20	3.6	Engineering	2022	11000	125	active
11	Lucas	18	2.7	Engineering	2023	4000	50	active
12	Nancy	23	3.9	Business	2019	9500	135	active

**Task:**

1. Find students who are older than 20 and have a GPA above the average GPA of all students
2. Find the top 5 students with the highest fees|paid, ordered by GPA (in descending order) as a tiebreaker
3. List students who belong to the "Engineering" department, have a GPA greater than 3.5, and are enrolled after 2020
4. Find students who are not active (i.e., enrollment\_status = 'inactive') and have not paid any fees (fees\_paid = 0)
5. Calculate the total fees paid and average GPA for each department, but only for departments with more than 10 students

**Objective:**

- To learn about the database system using XAMP.
- To know about SQL and NoSQL database system.
- To differentiate between DML and DDL in SQL.
- To learn about the query language to create tables and inputs in a database system.
- To apply queries to solve real life scenarios.

## Query and Output:

```
1 CREATE TABLE students(  
2     student_id VARCHAR(50) NOT NULL,  
3     student_name VARCHAR(50) NOT NULL,  
4     age VARCHAR(50) NOT NULL,  
5     GPA VARCHAR(50) NOT NULL,  
6     department VARCHAR(50) NOT NULL,  
7     year_of_admission VARCHAR(50) NOT NULL,  
8     fees_paid VARCHAR(50) NOT NULL,  
9     credits_earned VARCHAR(50) NOT NULL,  
10    enrollment_status VARCHAR(50) NOT NULL  
11 );  
12 INSERT INTO students(student_id,student_name,age,GPA,department,year_of_admission,fees_paid,credits_earned,enrollment_status)  
13 VALUES  
14 ('1','Eleven','21','3.8','Engineering','2021','10000','120','active'),  
15 ('2','Dustin','22','3.9','Science','2020','9000','110','active'),  
16 ('3','Will','19','3.4','Business','2022','8500','95','active'),  
17 ('4','Mike','23','3.7','Science','2021','9500','115','inactive'),  
18 ('5','Max','20','3.5','Engineering','2020','12000','130','active'),  
19 ('6','Eddie','22','4.0','Arts','2019','80000','140','active'),  
20 ('7','Billy','24','2.9','Engineering','2022','5000','60','active'),  
21 ('8','Alexei','25','3.2','Business','2018','7500','100','inactive'),  
22 ('9','Steve','21','3.8','Science','2021','10500','120','active'),  
23 ('10','Robin','20','3.6','Engineering','2022','11000','125','active'),  
24 ('11','Lucas','18','2.7','Engineering','2023','4000','50','active'),  
25 ('12','Nancy','23','3.9','Business','2019','9500','135','active');
```

```
SELECT `student_id`,`student_name` FROM `students` WHERE age>20 AND GPA > (SELECT AVG(GPA) FROM students);
```

student_id	student_name
1	Eleven
2	Dustin
4	Mike
6	Eddie
9	Steve
12	Nancy

```
SELECT `student_id`,`student_name` FROM students ORDER BY fees_paid DESC, GPA DESC LIMIT 5;
```

student_id	student_name	department	year_of_admission
1	Eleven	Engineering	2021
10	Robin	Engineering	2022

```
SELECT `student_id`,`student_name`,`fees_paid`,`enrollment_status` FROM students WHERE enrollment_status = 'inactive' AND fees_paid = 0;
```

student_id	student_name	fees_paid	enrollment_status
------------	--------------	-----------	-------------------

There is no real students that satisfy the conditions given. That's why there is not output in table section.

```
SELECT department,
       SUM(fees_paid) AS total_fees,
       AVG(gpa) AS average_gpa,
       COUNT(student_id) AS total_students
FROM Students
GROUP BY department HAVING COUNT(student_id) > 2;
```

department	total_fees	average_gpa	total_students
Business	25500	3.5	3
Engineering	42000	3.3	5
Science	29000	3.7999999999999994	3