"Heaven's Light is Our Guide"

Rajshahi University of Engineering & Technology

Rajshahi, Bangladesh

Department of Electrical & Computer Engineering

Course Code: ECE 2216

Course Title: Database System Sessional

Experiment No: 02

Date of Experiment: 23 September 2024

Date of Submission: 30 September 2024

Submitted To

Oishi Jyoti
Assistant Professor
Electrical & Computer Engineering
RUET

Submitted By

Nur-E-Sharin Trisha
Roll:2110034
Reg.:1088/2021-2022

Lab No: 02

Lab Task: Create the following table:

St_id	St_name	Age	Department	GPA	Admission_ year	Fees_paid	Earned_ credit	Enroll_ Status
1	Eleven	21	Engineering	3.8	2021	10000	120	Active
2	Dustin	22	Science	3.9	2020	9000	110	Active
3	Will	19	Business	3.4	2022	8500	95	Active
4	Mike	23	Science	3.7	2021	9500	115	Inactive
5	Max	20	Engineering	3.5	2020	12000	130	Active
6	Eddie	22	Arts	4.0	2019	8000	140	Active
7	Billy	24	Engineering	2.9	2022	5000	60	Active
8	Alexei	25	Business	3.2	2018	7500	100	Inactive
9	Steve	21	Science	3.8	2021	10500	120	Active
10	Robin	20	Engineering	3.6	2022	11000	125	Active
11	Lucas	18	Engineering	2.7	2023	4000	50	Active
12	Nancy	23	Business	3.9	2019	9500	135	Active

Table 1: Students information table.

Problems:

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., enroll_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:

The objective of these tasks is to analyze a dataset containing student records to extract meaningful insights and solve specific queries. [1]The tasks involve identifying students based on various criteria such as age, GPA, enrollment status, and fees paid. This focuses on filtering data to find students with high academic performance, particularly those belonging to the Engineering department, and identifying patterns in student enrollment. [2]Additionally, it includes the computation of departmental statistics, such as total fees paid and average GPA, to provide a clearer understanding of student distribution across different disciplines. The aim is to apply data querying techniques to draw conclusions that could support academic and administrative decision-making.

Problem Statement 1: Find students who are older than 20 and have a GPA above the average GPA of all students

Query:

Run SQL query/queries on table lab_3.student_information:

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

Output :

student_id	student_name	age	GPA
1	Eleven	21	3.80
2	Dustin	22	3.90
4	Mike	23	3.70
6	Eddie	22	4.00
9	Steve	21	3.80
12	Nancy	23	3.90

Problem Statement 2: Find the top 5 students with the highest fees paid, ordered by GPA (in descending order) as a tiebreaker.

Query:

Run SQL query/queries on table lab_3.student_information:

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

Output:

student_id	student_name	fees_paid 1	GPA 2
5	Max	12000.00	3.50
10	Robin	11000.00	3.60
9	Steve	10500.00	3.80
1	Eleven	10000.00	3.80
12	Nancy	9500.00	3.90

Problem Statement 3: List students who belong to the "Engineering" department, have a GPA greater than 3.5, and are enrolled after 2020

Query:

Run SQL query/queries on table lab_3.student_information:

```
SELECT student_id,student_name, GPA, year_of_admission
FROM student_information
WHERE department = 'Engineering' AND GPA > 3.5 AND year_of_admission > 2020;

```

Output:

student_id	student_name	GPA	year_of_admission
1	Eleven	3.80	2021
10	Robin	3.60	2022

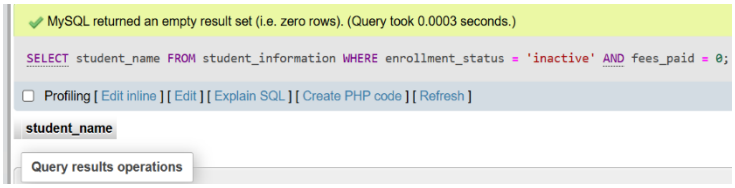
Problem Statement 4: Find students who are not active (i.e., enroll_status = 'inactive') and have not paid any fees (Fees_paid = 0)

Query:

Run SQL query/queries on table lab_3.student_information:

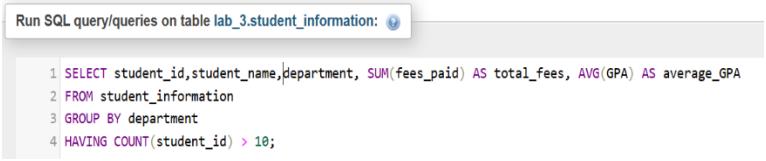
```
SELECT student_name
FROM student_information
WHERE enrollment_status = 'inactive' AND fees_paid = 0;

```

Output:

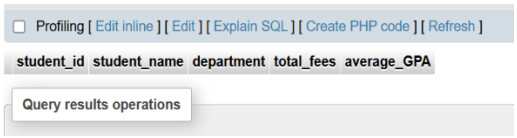


Problem Statement 5: Calculate the total fees paid and average GPA for each department, but only for departments with more than 10 students.

Query:



Output:



Reference:

[1] "Introduction of DBMS (Database Management System)." Accessed: Sep. 30, 2024. [Online]. Available: https://www.geeksforgeeks.org/introduction-of-dbms-database-management-system-set-1/

[2] "DBMS Tutorial | What is a Database Management System? - javatpoint." Accessed: Sep. 30, 2024. [Online]. Available: https://www.javatpoint.com/dbms-tutorial