

"Heaven's Light is Our Guide"

Rajshahi University of Engineering & Technology, Rajshahi



Department of Electrical And Computer
Engineering

Course Code : ECE 2216

Course Title : Database Management System
Sessional

Experiment No. : 02

Experiment Date : 24-09-2024

Submission Date : 1-10-2024

Submitted To-
Oishi Jyoti
Assistant Professor,
ECE, RUET

Submitted By-
Nafisha Nawar
Roll:2110026

Experiment No.: 02

Experiment Name:

Write appropriate SQL query for the given task and show output

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

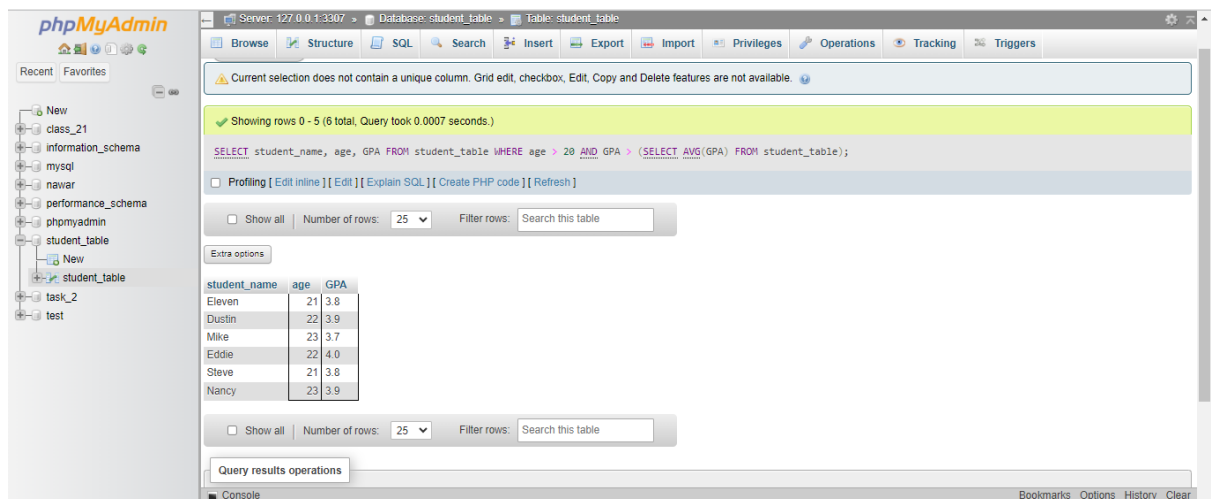
Task:

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: Learning different operations using database .

1.Query: Finding students who are older than 20 and have a GPA above the average GPA of all students

Output:

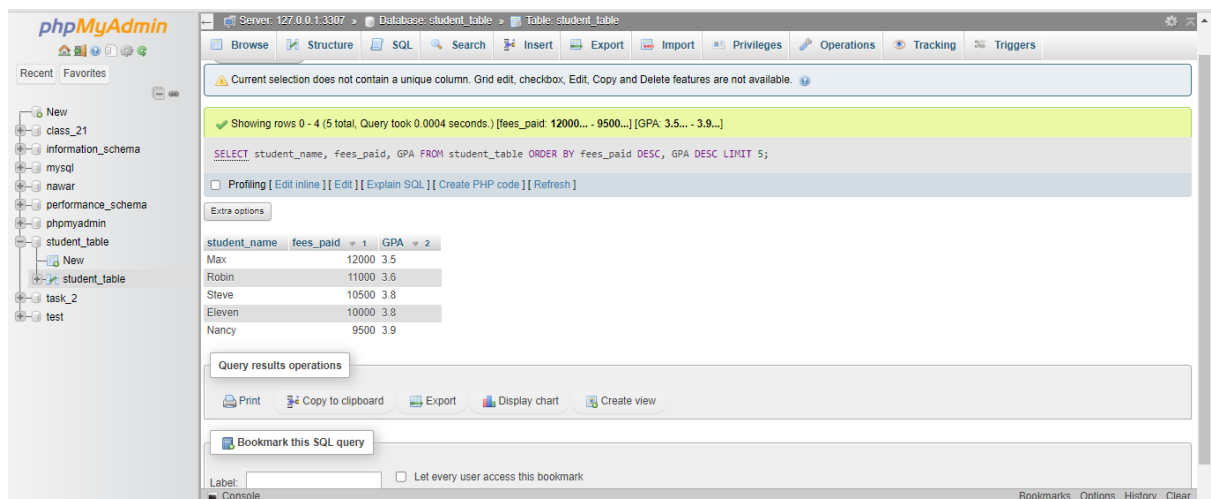


The screenshot shows the phpMyAdmin interface with a SQL query executed. The query is: `SELECT student_name, age, GPA FROM student_table WHERE age > 20 AND GPA > (SELECT AVG(GPA) FROM student_table);` The result shows 5 rows of data.

student_name	age	GPA
Eleven	21	3.8
Dustin	22	3.9
Mike	23	3.7
Eddie	22	4.0
Steve	21	3.8
Nancy	23	3.9

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

Output:

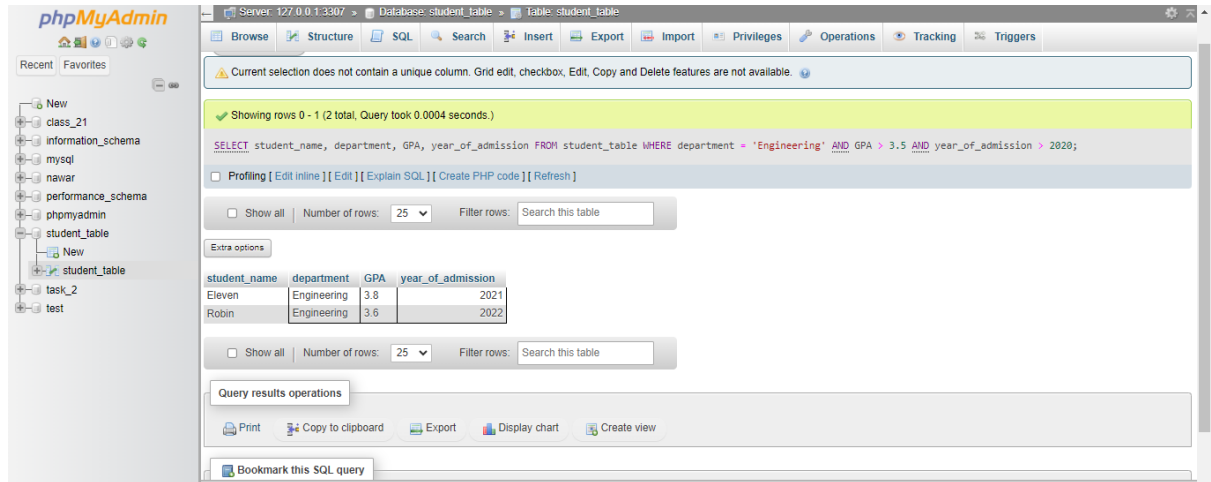


The screenshot shows the phpMyAdmin interface with a SQL query executed. The query is: `SELECT student_name, fees_paid, GPA FROM student_table ORDER BY fees_paid DESC, GPA DESC LIMIT 5;` The result shows 5 rows of data.

student_name	fees_paid	GPA
Max	12000	3.5
Robin	11000	3.6
Steve	10500	3.8
Eleven	10000	3.8
Nancy	9500	3.9

3.Query: Listing students who belong to the "Engineering" department, have a GPA greater than 3.5, and are enrolled after 2020

Output:

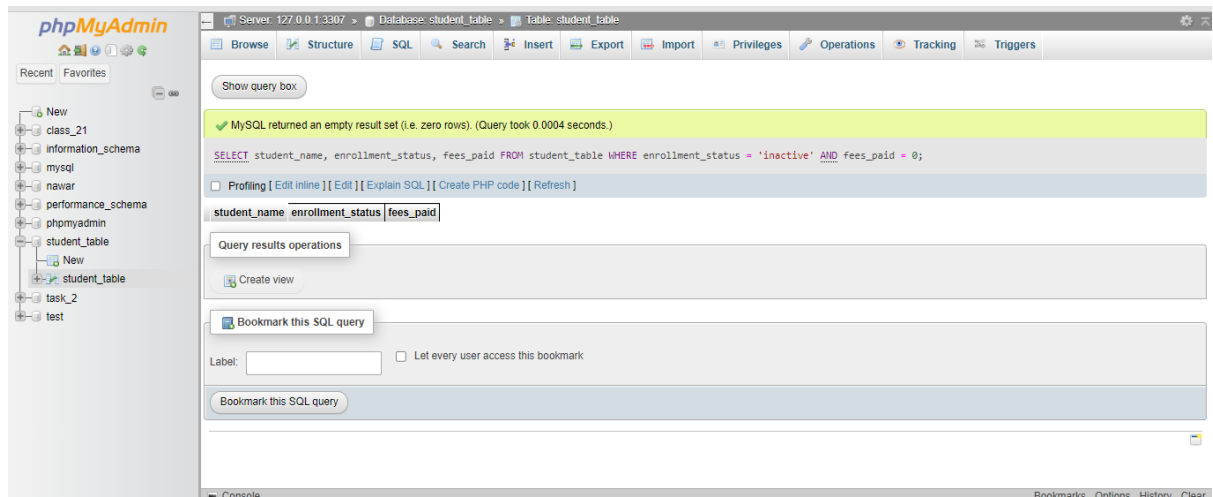


The screenshot shows the phpMyAdmin interface with the 'student_table' selected. A SQL query is executed, and the results are displayed in a table. The query filters for students in the 'Engineering' department with a GPA greater than 3.5 and an admission year after 2020. The results table shows two rows: 'Eleven' with a GPA of 3.8 and admission year 2021, and 'Robin' with a GPA of 3.6 and admission year 2022.

student_name	department	GPA	year_of_admission
Eleven	Engineering	3.8	2021
Robin	Engineering	3.6	2022

4.Query: Finding students who are not active (i.e., enrollment_status = 'inactive') and have not paid any fees (fees_paid = 0)

Output:



The screenshot shows the phpMyAdmin interface with the 'student_table' selected. A SQL query is executed, and the results are displayed. The query filters for students with an enrollment status of 'inactive' and fees paid equal to 0. The results table is empty, indicating that no students meet these criteria.

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

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

Output:

The screenshot shows the phpMyAdmin interface with the following components:

- Left Sidebar:** A tree view of the database structure. The 'student_table' is selected under the 'student' database.
- Top Bar:** Navigation tabs for 'Browse', 'Structure', 'SQL', 'Search', 'Insert', 'Export', 'Import', 'Privileges', 'Operations', 'Tracking', and 'Triggers'. The 'SQL' tab is active.
- Query Area:** A text box containing the SQL query: `SELECT department, SUM(fees_paid) AS total_fees, AVG(GPA) AS average_GPA FROM student_table GROUP BY department HAVING COUNT(student_id) > 10;`. Below the query box are links for 'Profiling', 'Edit inline', 'Edit', 'Explain SQL', 'Create PHP code', and 'Refresh'.
- Results Area:** A message stating 'MySQL returned an empty result set (i.e. zero rows). (Query took 0.0089 seconds.)'. Below this, a table header is shown:

department	total_fees	average_GPA
------------	------------	-------------

. The table body is empty.
- Operations Area:** A section with a 'Query results operations' dropdown menu, a 'Create view' button, and a 'Bookmark this SQL query' button.
- Bookmark Section:** A form with a 'Label:' text input, a checkbox labeled 'Let every user access this bookmark', and a 'Bookmark this SQL query' button.
- Footer:** A 'Console' tab and links for 'Bookmarks', 'Options', 'History', and 'Clear'.