

Task2

Project: Student Database Management System(PostgreSQL)

Objective: Design and implement a student database management system using PostgreSQL that allows storing and retrieving student information efficiently. The project will include the following tasks:

1. Database Setup

Create a database named "student_database."

Create a table called " student_table " with the following columns: Student_id (integer), Stu_name (text), Department (text), email_id (text),Phone_no (numeric), Address (text), Date_of_birth (date), Gender (text), Major (text), GPA (numeric),Grade (text) should be A,B,C etc.

#	Time	Action	Message
1	22:47:01	CREATE DATABASE student_database	1 row(s) affected
2	22:47:23	use student_database	0 row(s) affected
3	22:47:38	CREATE TABLE student_table (Student_id INTEGER PRIMARY KEY, Stu_name TEXT, Department TE...	0 row(s) affected

2. Data Entry

Insert 10 sample records into the "student_table" using INSERT command.

Student_id	Stu_name	Department	email_id	Phone_no	Address	Date_of_birth	Gender	Major	GPA	Grade
1	Ayush Chauhan	Computer Science	ayush.chauhan@example.com	9876543210	Delhi	2002-05-14	Male	AI	8.50	A
2	Priya Sharma	Mechanical	priya.sharma@example.com	9876543211	Mumbai	2001-03-19	Female	Thermodynamics	6.70	B
3	Rohan Mehta	Electrical	rohan.mehta@example.com	9876543212	Pune	2003-12-30	Male	Power Systems	4.30	C
4	Ananya Iyer	Civil	ananya.iyer@example.com	9876543213	Chennai	2002-11-21	Female	Structural	7.90	A
5	Karan Singh	Computer Science	karan.singh@example.com	9876543214	Noida	2001-09-15	Male	ML	5.20	B
6	Sneha Nair	Electrical	sneha.nair@example.com	9876543215	Bangalore	2002-04-08	Female	Robotics	9.10	A
7	Manish Tiwari	Mechanical	manish.tiwari@example.com	9876543216	Lucknow	2000-10-10	Male	Design	3.90	D
8	Pooja Verma	Civil	pooja.verma@example.com	9876543217	Jaipur	2001-08-25	Female	GeoTech	6.80	B
9	Arjun Das	Computer Science	arjun.das@example.com	9876543218	Kolkata	2003-07-05	Male	Data Science	9.50	A
10	Neha Reddy	Mechanical	neha.reddy@example.com	9876543219	Hyderabad	2002-02-17	Female	Automation	4.80	C

3. Student Information Retrieval

Develop a query to retrieve all students' information from the "student_table" and sort them in descending order by their grade.

Student_id	Stu_name	Department	email_id	Phone_no	Address	Date_of_birth	Gender	Major	GPA	Grade
7	Manish Tiwari	Mechanical	manish.tiwari@example.com	9876543216	Lucknow	2000-10-10	Male	Design	3.90	D
3	Rohan Mehta	Electrical	rohan.mehta@example.com	9876543212	Pune	2003-12-30	Male	Power Systems	4.30	C
10	Neha Reddy	Mechanical	neha.reddy@example.com	9876543219	Hyderabad	2002-02-17	Female	Automation	4.80	C
2	Priya Sharma	Mechanical	priya.sharma@example.com	9876543211	Mumbai	2001-03-19	Female	Thermodynamics	6.70	B
5	Karan Singh	Computer Science	karan.singh@example.com	9876543214	Noida	2001-09-15	Male	ML	5.20	B
8	Pooja Verma	Civil	pooja.verma@example.com	9876543217	Jaipur	2001-08-25	Female	GeoTech	6.80	B
1	Ayush Chauhan	Computer Science	ayush.chauhan@example.com	9876543210	Delhi	2002-05-14	Male	AI	8.50	A
4	Ananya Iyer	Civil	ananya.iyer@example.com	9876543213	Chennai	2002-11-21	Female	Structural	7.90	A
6	Sneha Nair	Electrical	sneha.nair@example.com	9876543215	Bangalore	2002-04-08	Female	Robotics	9.10	A
9	Arjun Das	Computer Science	arjun.das@example.com	9876543218	Kolkata	2003-07-05	Male	Data Science	9.50	A

4. Query for Male Students:

.Implement a query to retrieve information about all male students from the "student_table."

[illegible]

5. Query for Students with GPA less than 5.0

Create a query to fetch the details of students who have a GPA less than 5.0 from the "student_table."

[illegible]

6. Update Student Email and Grade

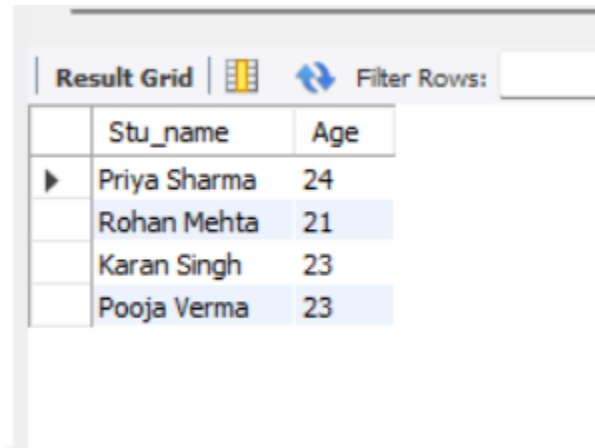
Write an update statement to modify the email and grade of a student with a specific ID in the "student table."

```
61
62 • Update student_table
63 Set email_id = 'updated.rohan@example.com',
64     Grade = 'B'
65 Where Student_id = 3;
66
67 • Select * from student_table
68 Where Student_id = 3;
69
```

[illegible]

7. Query for Students with Grade "B"

Develop a query to retrieve the names and ages of all students who have a grade of "B" from the "student_table."

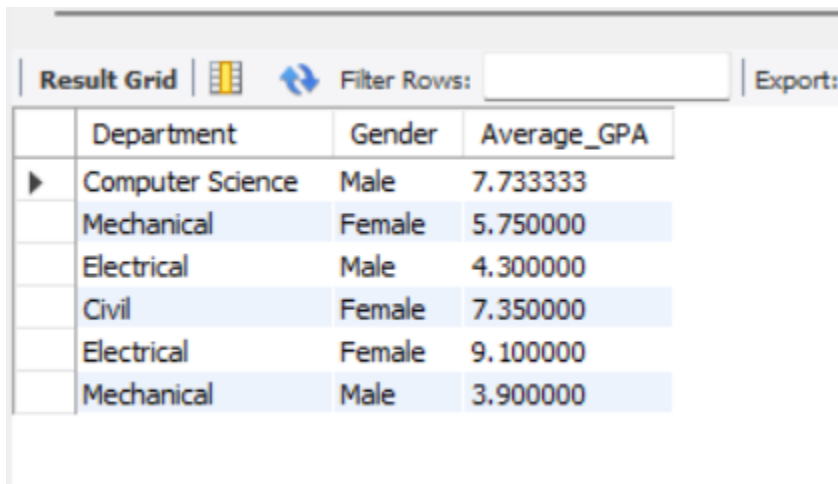


The screenshot shows a database interface with a 'Result Grid' tab. It contains a table with two columns: 'Stu_name' and 'Age'. There are four rows of data, each with a small triangle icon to its left. The rows represent students with a grade of 'B': Priya Sharma (24), Rohan Mehta (21), Karan Singh (23), and Pooja Verma (23).

	Stu_name	Age
▶	Priya Sharma	24
	Rohan Mehta	21
	Karan Singh	23
	Pooja Verma	23

8. Grouping and Calculation

Create a query to group the "student_table" by the "Department" and "Gender" columns and calculate the average GPA for each combination.



The screenshot shows a database interface with a 'Result Grid' tab. It contains a table with four columns: 'Department', 'Gender', and 'Average_GPA'. There are six rows of data, each with a small triangle icon to its left. The rows represent the average GPA for each combination of department and gender: Computer Science Male (7.733333), Mechanical Female (5.750000), Electrical Male (4.300000), Civil Female (7.350000), Electrical Female (9.100000), and Mechanical Male (3.900000).

	Department	Gender	Average_GPA
▶	Computer Science	Male	7.733333
	Mechanical	Female	5.750000
	Electrical	Male	4.300000
	Civil	Female	7.350000
	Electrical	Female	9.100000
	Mechanical	Male	3.900000

9. Table Renaming

Rename the "student_table" to "student_info" using the appropriate SQL statement.

The screenshot shows the SQL Server Enterprise Manager interface. On the left, the 'student_database' is expanded, showing 'Tables' with 'student_info' highlighted. The 'task1' task is also visible. The main pane displays a SQL script with the following content:

```
65 Where Student_id = 3;
66
67 • Select * from student_table
68 Where Student_id = 3;
69
70 -- Develop a query to retrieve the names and ages of all students who have a grade of "B" from the "
71
72 • SELECT
73 Stu_name,
74 TIMESTAMPDIFF(YEAR, Date_of_birth, CURDATE()) AS Age
75 FROM student_table
76 WHERE Grade = 'B';
77
78 -- Create a query to group the "student_table" by the "Department" and "Gender" columns and
79 -- calculate the average GPA for each combination.
80
81 • Select Department, Gender,
82 AVG(GPA) As Average_GPA
83 From student_table
84 Group By Department, Gender;
85
86 -- Rename the "student_table" to "student_info" using the appropriate SQL statement.
87 • Rename TABLE student_table TO student_info;
```

The 'Output' pane at the bottom shows the execution results:

#	Time	Action	Message
16	23:39:02	Select Department, Gender, AVG(GPA) As Average_GPA From student_table Group By Department, Gender	6 row(s) returned
17	23:40:02	Rename TABLE student_table TO student_info	0 row(s) affected

10. Retrieve Student with Highest GPA

Write a query to retrieve the name of the student with the highest GPA from the "student_info" table.

The screenshot shows the 'Result Grid' in SQL Server. It displays a single row with the student's name 'Arjun Das' under the column 'Stu_name'.

Stu_name
Arjun Das