

## 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,Cetc.

The screenshot shows a PostgreSQL query editor with a single query in the editor: `CREATE DATABASE student_database;`. The output window displays a list of executed queries and their results.

#	Time	Action	Message	Duration / Fetch
7	21:44:21	SELECT STU_NAME, PHONE_NO, EMAIL_ID, ENROLL_STATUS FROM Studen...	10 row(s) returned	0.000 sec / 0.000 sec
8	21:46:54	SELECT C.COURSE_NAME, C.COURSE_INSTRUCTOR_NAME FROM Enrollmen...	2 row(s) returned	0.000 sec / 0.000 sec
9	21:48:05	SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME FROM CoursesInfo L...	5 row(s) returned	0.000 sec / 0.000 sec
10	21:48:34	SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME FROM CoursesInfo ...	1 row(s) returned	0.000 sec / 0.000 sec
11	21:49:02	SELECT COURSE_NAME, COURSE_INSTRUCTOR_NAME FROM CoursesInfo ...	2 row(s) returned	0.000 sec / 0.000 sec
12	21:54:38	SELECT C.COURSE_NAME, COUNT(E.STU_ID) AS NumberOfStudents FROM C...	5 row(s) returned	0.000 sec / 0.000 sec
13	21:55:08	SELECT S.STU_NAME FROM StudentInfo S JOIN EnrollmentInfo E ON S.STU_ID...	3 row(s) returned	0.000 sec / 0.000 sec
14	21:55:37	SELECT C.COURSE_INSTRUCTOR_NAME, COUNT(E.STU_ID) AS NumberOfSt...	5 row(s) returned	0.000 sec / 0.000 sec
15	21:56:01	SELECT S.STU_NAME, COUNT(E.COURSE_ID) AS NumberOfCourses FROM Stu...	5 row(s) returned	0.015 sec / 0.000 sec
16	21:56:23	SELECT C.COURSE_NAME, COUNT(E.STU_ID) AS NumberOfStudents FROM C...	5 row(s) returned	0.016 sec / 0.000 sec
17	21:58:28	CREATE DATABASE student_database	1 row(s) affected	0.015 sec

The screenshot shows a SQL IDE window with a query editor and an output pane. The query editor contains the following SQL command:

```

1 CREATE TABLE student_table (
2     Student_id INT PRIMARY KEY,
3     Stu_name TEXT,
4     Department TEXT,
5     Email_id TEXT,
6     Phone_no NUMERIC,
7     Address TEXT,
8     Date_of_birth DATE,
9     Gender TEXT,
10    Major TEXT

```

The output pane shows the execution result:

#	Time	Action	Message
1	22:13:03	CREATE TABLE student_table ( Student_id INT PRIMARY KEY, Stu_name TEX...	0 row(s) affected

## 2.Data Entry

Insert 10 sample records into the "student\_table" using INSERT command.

The screenshot shows a SQL IDE window with a query editor and an output pane. The query editor contains the following SQL command:

```

1 INSERT INTO student_table (Student_id, Stu_name, Department, email_id, Phone_no, Address, Date_of_birth, Gender, Major, GPA, Grade)
2
3 ('bruce@wayne.com', 1234567890, 'Gotham', '1980-05-27', 'Male', 'Physics', 3.9, 'A'),
4 ('clark@dailyplanet.com', 9876543210, 'Metropolis', '1978-06-18', 'Male', 'Communication', 3.8, 'B'),
5 ('diana@themiscira.com', 5556667777, 'Themyscira', '1984-03-22', 'Female', 'Mythology', 4.0, 'A'),
6 ('barry@ccpd.com', 3334445555, 'Central City', '1992-09-20', 'Male', 'Chemistry', 3.7, 'B'),
7 ('hal@usaf.com', 9998887777, 'Coast City', '1980-02-20', 'Male', 'Engineering', 3.5, 'B'),
8 ('victor@cyborg.com', 8887776666, 'Detroit', '1995-11-29', 'Male', 'Robotics', 3.9, 'A'),
9 ('steve@superman.com', 6665554444, 'Star City', '1983-07-15', 'Male', 'Law', 3.3, 'C')
10

```

The output pane shows the execution results:

#	Time	Action	Message
1	22:13:03	CREATE TABLE student_table ( Student_id INT PRIMARY KEY, Stu_name TEX...	0 row(s) affected
2	22:17:14	INSERT INTO student_table (Student_id, Stu_name, Department, email_id, Phone_no...	10 row(s) affected, 9 warning(s): 1265 Data truncated for column

## Information Retrieval

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

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • SELECT *
2 FROM student_table
3 ORDER BY Grade;
4

```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

Student_id	Stu_name	Department	Email_id	Phone_no	Address	Date_of_birth	Gender	Major	GPA	Grade
1	Bruce Wayne	Science	bruce@waynecorp.com	1234567890	Gotham	1980-05-27	Male	Physics	4	A
3	Diana Prince	History	diana@themiscira.com	5556667777	Themyscira	1984-03-22	Female	Mythology	4	A
7	Victor Stone	Computer Science	victor@cyborg.com	8887776666	Detroit	1995-11-29	Male	Robotics	4	A
2	Clark Kent	Journalism	clark@dailyplanet.com	9876543210	Metropolis	1978-06-18	Male	Communication	4	B
4	Barry Allen	Forensics	barry@ccpd.com	3334445555	Central City	1992-09-20	Male	Chemistry	4	B
6	Hal Jordan	Aerospace	hal@usaf.com	9998887777	Coast City	1980-02-20	Male	Engineering	4	B
9	Kara Danvers	Biology	kara@deo.com	4443332222	National City	1990-12-10	Female	Genetics	4	B
5	Arthur Curry	Marine Biology	arthur@atlantis.com	2223334444	Atlantis	1985-01-29	Male	Biology	4	C
8	Oliver Queen	Political Science	oliver@queenind.com	6665554444	Star City	1983-07-15	Male	Law	3	C
10	John Constantine	Occult Studies	john@hellblazer.com	7776665555	Liverpool	1975-05-10	Male	Theology	3	C

student\_table 1 x Apply

#### 4. Query for Male Students:

.Implement a query to retrieve information about all male students from the "student\_table."

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • SELECT *
2 FROM student_table
3 WHERE Gender = 'Male';

```

Result Grid Filter Rows: Edit: Export/Import: Wrap Cell Content:

Student_id	Stu_name	Department	Email_id	Phone_no	Address	Date_of_birth	Gender	Major	GPA	Grade
1	Bruce Wayne	Science	bruce@waynecorp.com	1234567890	Gotham	1980-05-27	Male	Physics	4	A
2	Clark Kent	Journalism	clark@dailyplanet.com	9876543210	Metropolis	1978-06-18	Male	Communication	4	B
4	Barry Allen	Forensics	barry@ccpd.com	3334445555	Central City	1992-09-20	Male	Chemistry	4	B
5	Arthur Curry	Marine Biology	arthur@atlantis.com	2223334444	Atlantis	1985-01-29	Male	Biology	4	C
6	Hal Jordan	Aerospace	hal@usaf.com	9998887777	Coast City	1980-02-20	Male	Engineering	4	B
7	Victor Stone	Computer Science	victor@cyborg.com	8887776666	Detroit	1995-11-29	Male	Robotics	4	A
8	Oliver Queen	Political Science	oliver@queenind.com	6665554444	Star City	1983-07-15	Male	Law	3	C
10	John Constantine	Occult Studies	john@hellblazer.com	7776665555	Liverpool	1975-05-10	Male	Theology	3	C

#### 5. Query for Students with GPA less than 5.0

Query 1SQL File 1\*SQL File 2\*Administration - Server Status

## 6. Update Student Email and Grade

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • UPDATE student_table
2   SET Grade = 'B',email_id = 'john@heavenblazer.com'
3   WHERE Stu_name = 'John Constantine';

```

SQLAdditions

My Snippets

Output

Action Output

#	Time	Action	Message
1	22:24:33	UPDATE student_table SET Grade = 'B',email_id = 'john@heavenblazer.com' WHERE Stu_name = 'Jo...	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0

## 7. Query for Students with Grade "B"

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • SELECT Stu_name, TIMESTAMPDIFF(YEAR, Date_of_birth, CURDATE()) AS Age
2   FROM student_table
3   WHERE Grade = 'B';

```

Result Grid

Filter Rows:

Export: Wrap Cell Content:

Stu_name	Age
Clark Kent	46
Barry Allen	31
Hal Jordan	44
Kara Danvers	33
John Constantine	49

## 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.

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • SELECT Department,Gender, AVG(GPA) AS AvgGPA
2 FROM student_table
3 GROUP BY Department,Gender;

```

Result Grid Filter Rows: Export: Wrap Cell Content:

	Department	Gender	AvgGPA
▶	Science	Male	4.0000
▶	Journalism	Male	4.0000
	History	Female	4.0000
	Forensics	Male	4.0000
	Marine Biology	Male	4.0000
	Aerospace	Male	4.0000
	Computer Science	Male	4.0000
	Political Science	Male	3.0000
	Biology	Female	4.0000
	Occult Studies	Male	3.0000

## 9. Table Renaming

Rename the "student\_table" to "student\_info" using the appropriate SQL statement.

Query 1 SQL File 1\* x SQL File 2\* Administration - Server Status

Limit to 1000 rows

```

1 • ALTER TABLE student_table RENAME TO student_info;

```

SQLAdditions My Snippets

Context Help Snippets

Output

Action Output

#	Time	Action	Message	Duration / Fetch
4	22:26:54	SELECT Department,Gender, AVG(GPA) AS AvgGPA FROM student_table GROUP BY Department...	10 row(s) returned	0.000 sec / 0.000 s
5	22:27:24	ALTER TABLE student_table RENAME TO student_info	0 row(s) affected	0.016 sec











## 10. Retrieve Student with Highest GPA

Query 1






SQL File 1\* x

SQL File 2\*

Administration - Server Status



Limit to 1000 rows



1

SELECT Stu\_name FROM student\_info

2


ORDER BY GPA DESC


3


LIMIT 1;


<


Result Grid



 Filter Rows:

Export: 

Wrap Cell Content: 

Fetch rows: 

	Stu_name
▶	Bruce Wayne