**TASK 1**

CREATE DATABASE academic\_management\_system;

USE academic\_management\_system;

CREATE TABLE StudentInfo (

STU\_ID INT AUTO\_INCREMENT PRIMARY KEY,

STU\_NAME VARCHAR(100),

DOB DATE,

PHONE\_NO VARCHAR(15),

EMAIL\_ID VARCHAR(100),

ADDRESS TEXT

);

CREATE TABLE CoursesInfo (

COURSE\_ID INT AUTO\_INCREMENT PRIMARY KEY,

COURSE\_NAME VARCHAR(100),

COURSE\_INSTRUCTOR\_NAME VARCHAR(100)

);

CREATE TABLE EnrollmentInfo (

ENROLLMENT\_ID INT AUTO\_INCREMENT PRIMARY KEY,

STU\_ID INT,

COURSE\_ID INT,

ENROLL\_STATUS ENUM('Enrolled', 'Not Enrolled'),

FOREIGN KEY (STU\_ID) REFERENCES StudentInfo(STU\_ID),

FOREIGN KEY (COURSE\_ID) REFERENCES CoursesInfo(COURSE\_ID)

);

INSERT INTO StudentInfo (STU\_NAME, DOB, PHONE\_NO, EMAIL\_ID, ADDRESS)

VALUES

('Atul Singh', '1990-01-15', '1234567890', 'atul@email.com', '123 Main St, City'),

('Bharat Chauhan', '1991-05-20', '9876543210', 'bharat@email.com', '456 Elm St, Town'),

('Deepak Tomar', '1992-07-12', '5555555555', 'deepak@email.com', '789 Oak St, Village'),

('Hitesh Pundir', '1993-03-03', '6666666666', 'hitesh@email.com', '101 Pine St, Hamlet'),

('Nitesh Sharma', '1994-09-09', '4444444444', 'nitesh@email.com', '222 Birch St, Town'),

('Akshay Dravid', '1995-02-18', '7777777777', 'akshay@email.com', '345 Cedar St, City'),

('Prabhat Tendulkar', '1996-04-30', '8888888888', 'prabhat@email.com', '789 Redwood St, Town'),

('Rahul Chauhan', '1997-08-27', '9999999999', 'rahul@email.com', '456 Sequoia St, Village'),

('Rohan Singh', '1998-12-10', '1111111111', 'rohan@email.com', '567 Maple St, City'),

('Puneet Tomar', '1999-06-06', '2222222222', 'puneet@email.com', '654 Fir St, Town');

INSERT INTO CoursesInfo (COURSE\_NAME, COURSE\_INSTRUCTOR\_NAME)

VALUES

('Math 101', 'Professor Math'),

('Physics 101', 'Professor Physics'),

('Chemistry 101', 'Professor Chemistry'),

('Biology 101', 'Professor Biology'),

('Computer Science 101', 'Professor CS'),

('History 101', 'Professor History'),

('Geography 101', 'Professor Geography'),

('Literature 101', 'Professor Literature'),

('Economics 101', 'Professor Economics'),

('Psychology 101', 'Professor Psychology');

INSERT INTO EnrollmentInfo (STU\_ID, COURSE\_ID, ENROLL\_STATUS)

VALUES

(1, 1, 'Enrolled'),

(1, 2, 'Enrolled'),

(2, 1, 'Enrolled'),

(2, 3, 'Enrolled'),

(3, 1, 'Enrolled'),

(4, 2, 'Enrolled'),

(5, 2, 'Enrolled'),

(6, 3, 'Enrolled'),

(7, 3, 'Enrolled'),

(8, 1, 'Enrolled'),

(8, 4, 'Enrolled'),

(9, 4, 'Enrolled'),

(9, 5, 'Enrolled'),

(10, 5, 'Enrolled'),

(10, 6, 'Enrolled');

SELECT SI.STU\_NAME, SI.PHONE\_NO, SI.EMAIL\_ID, EI.ENROLL\_STATUS

FROM StudentInfo SI

JOIN EnrollmentInfo EI ON SI.STU\_ID = EI.STU\_ID;

SELECT CI.COURSE\_NAME

FROM CoursesInfo CI

JOIN EnrollmentInfo EI ON CI.COURSE\_ID = EI.COURSE\_ID

WHERE EI.STU\_ID = 1;

SELECT COURSE\_NAME, COURSE\_INSTRUCTOR\_NAME

FROM CoursesInfo;

SELECT \*

FROM CoursesInfo

WHERE COURSE\_ID = 1;

SELECT COURSE\_NAME, COURSE\_INSTRUCTOR\_NAME

FROM CoursesInfo

WHERE COURSE\_ID IN (1, 2, 3);

SELECT CI.COURSE\_NAME, COUNT(EI.STU\_ID) AS NUM\_ENROLLED

FROM CoursesInfo CI

LEFT JOIN EnrollmentInfo EI ON CI.COURSE\_ID = EI.COURSE\_ID

GROUP BY CI.COURSE\_NAME;

SELECT SI.STU\_NAME

FROM StudentInfo SI

JOIN EnrollmentInfo EI ON SI.STU\_ID = EI.STU\_ID

WHERE EI.COURSE\_ID = 1;

SELECT CI.COURSE\_INSTRUCTOR\_NAME, COUNT(EI.STU\_ID) AS NUM\_ENROLLED

FROM CoursesInfo CI

LEFT JOIN EnrollmentInfo EI ON CI.COURSE\_ID = EI.COURSE\_ID

GROUP BY CI.COURSE\_INSTRUCTOR\_NAME;

SELECT SI.STU\_NAME

FROM StudentInfo SI

JOIN EnrollmentInfo EI ON SI.STU\_ID = EI.STU\_ID

WHERE EI.ENROLL\_STATUS = 'Enrolled'

GROUP BY SI.STU\_NAME

HAVING COUNT(DISTINCT EI.COURSE\_ID) > 1;

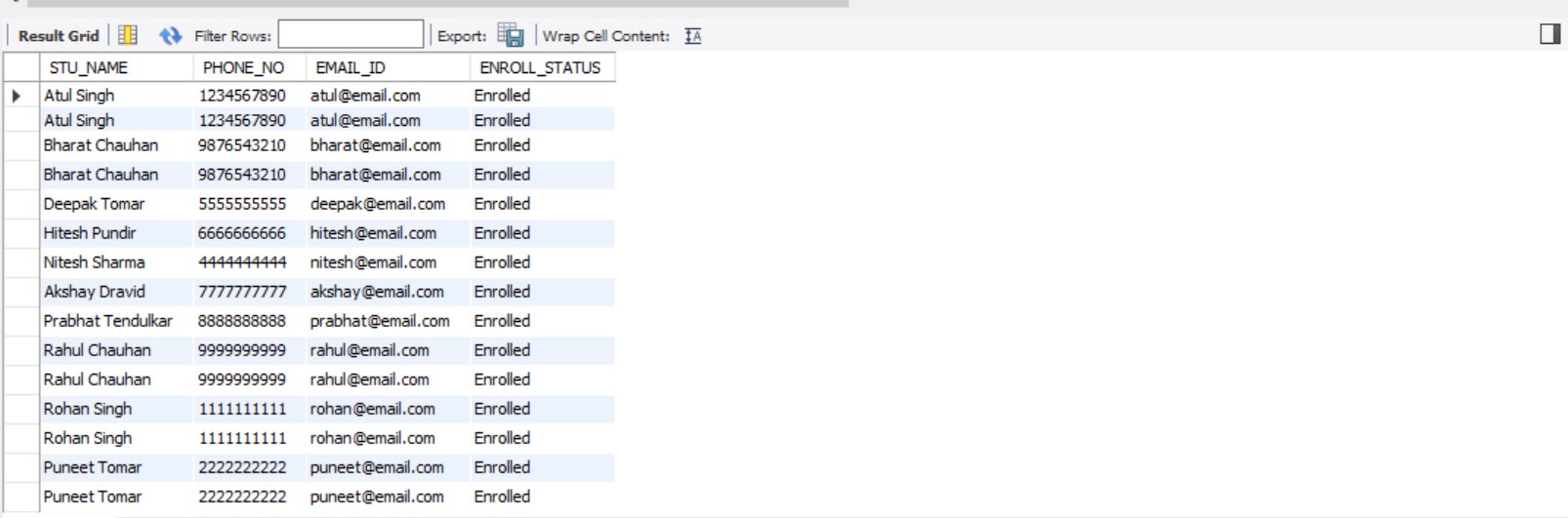
SELECT CI.COURSE\_NAME, COUNT(EI.STU\_ID) AS NUM\_ENROLLED

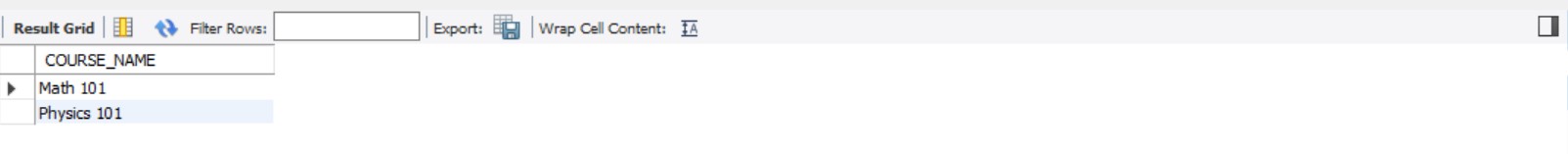
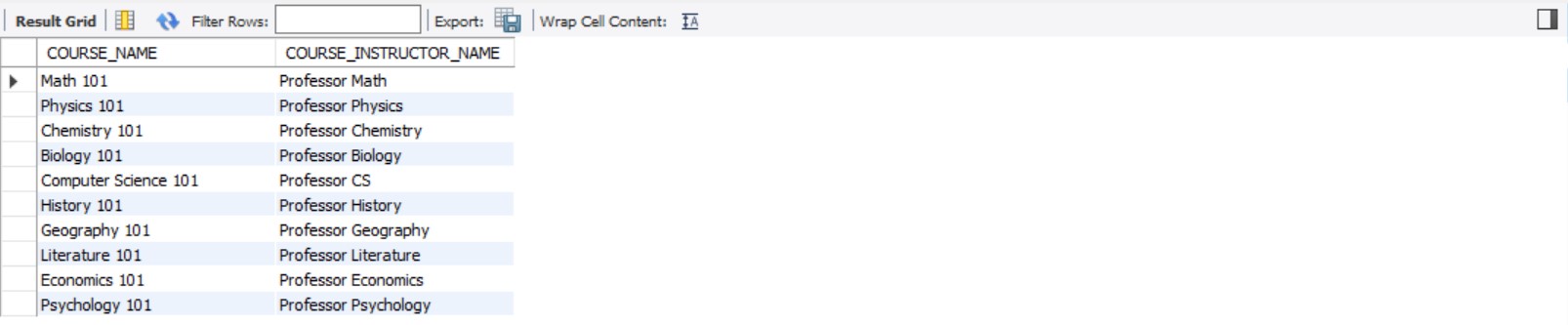
FROM CoursesInfo CI

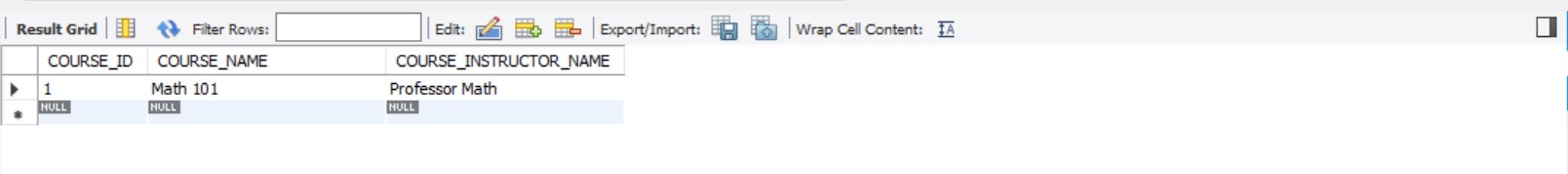
LEFT JOIN EnrollmentInfo EI ON CI.COURSE\_ID = EI.COURSE\_ID

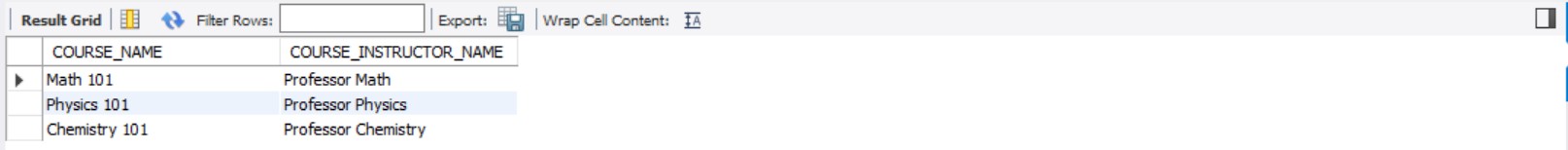
GROUP BY CI.COURSE\_NAME

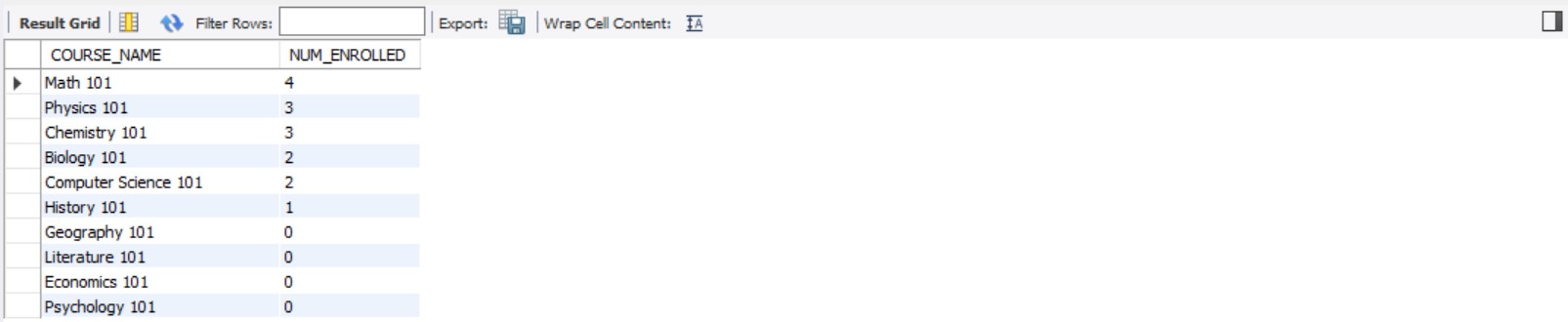
ORDER BY NUM\_ENROLLED DESC;





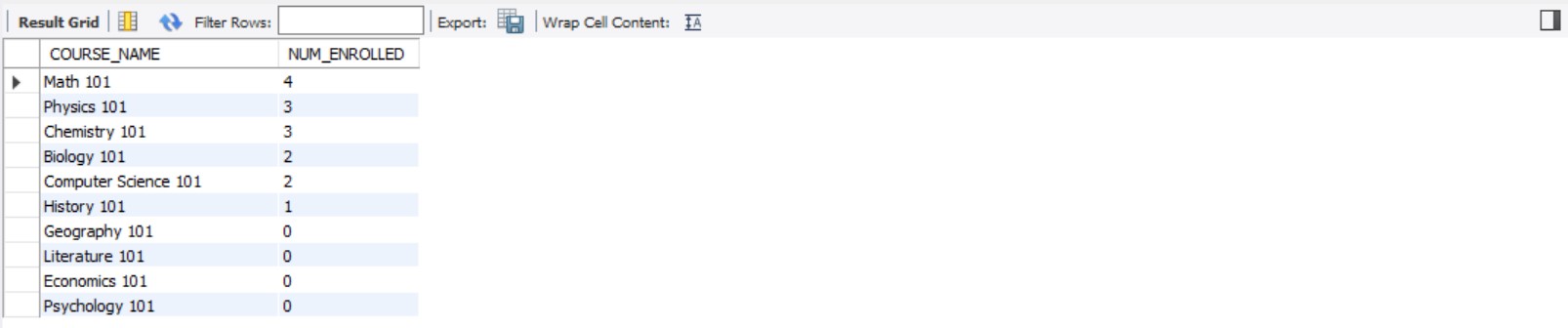












**TASK 2**  
  
CREATE DATABASE student\_database;

CREATE TABLE student\_info (

student\_id SERIAL PRIMARY KEY,

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

);

INSERT INTO student\_info (stu\_name, department, email\_id, phone\_no, address, date\_of\_birth, gender, major, gpa, grade)

VALUES

('Atul Singh', 'Computer Science', 'atul@yahoo.com', 1234567890, 'Mumbai', '1995-05-15', 'Male', 'Computer Engineering', 3.9, 'A'),

('Bharat Chauhan', 'Mathematics', 'bharat@yahoo.com', 9876543210, 'Delhi', '1996-03-20', 'Male', 'Applied Mathematics', 4.2, 'A'),

('Deepak Tomar', 'Electrical Engineering', 'deepak@yahoo.com', 5555555555, 'Chennai', '1997-07-12', 'Male', 'Power Systems', 3.7, 'B'),

('Hitesh Pundir', 'Physics', 'hitesh@yahoo.com', 6666666666, 'Kolkata', '1998-09-03', 'Male', 'Quantum Physics', 4.0, 'A'),

('Nitesh Sharma', 'Chemistry', 'nitesh@yahoo.com', 4444444444, 'Bangalore', '1999-01-09', 'Male', 'Inorganic Chemistry', 3.2, 'B'),

('Akshay Dravid', 'Economics', 'akshay@yahoo.com', 7777777777, 'Hyderabad', '1995-02-18', 'Male', 'Macroeconomics', 3.8, 'A'),

('Prabhat Tendulkar', 'Computer Science', 'prabhat@yahoo.com', 8888888888, 'Pune', '1996-04-30', 'Male', 'Machine Learning', 4.3, 'A'),

('Rahul Chauhan', 'Physics', 'rahul@yahoo.com', 9999999999, 'Jaipur', '1997-08-27', 'Male', 'Astrophysics', 3.5, 'B'),

('Rohan Singh', 'History', 'rohan@yahoo.com', 1111111111, 'Lucknow', '1998-12-10', 'Male', 'World History', 3.9, 'A'),

('Puneet Tomar', 'Mathematics', 'puneet@yahoo.com', 2222222222, 'Mumbai', '1999-06-06', 'Male', 'Number Theory', 3.6, 'B');

SELECT \* FROM student\_info ORDER BY grade DESC;

SELECT \* FROM student\_info WHERE gender = 'Male';

SELECT \* FROM student\_info WHERE gpa < 5.0;

UPDATE student\_info

SET email\_id = 'new\_email@gmail.com', grade = 'D'

WHERE student\_id = 3;

SELECT stu\_name, date\_part('year', age(current\_date, date\_of\_birth)) AS age

FROM student\_info

WHERE grade = 'B';

SELECT department, gender, AVG(gpa) AS avg\_gpa

FROM student\_info

GROUP BY department, gender;

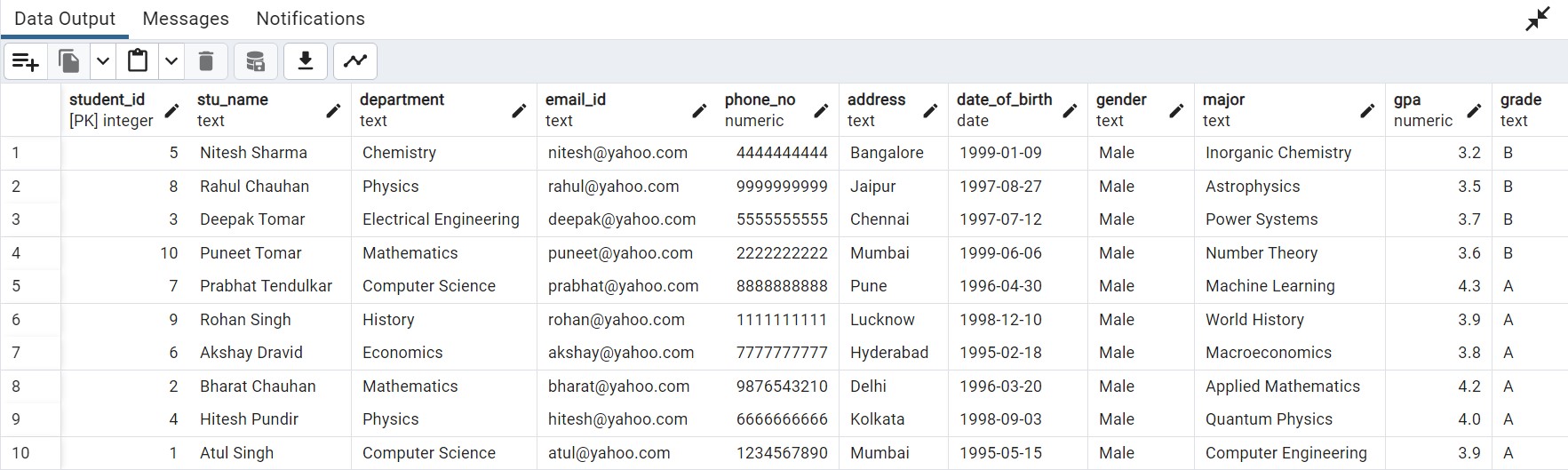
ALTER TABLE student\_info RENAME TO student\_table;

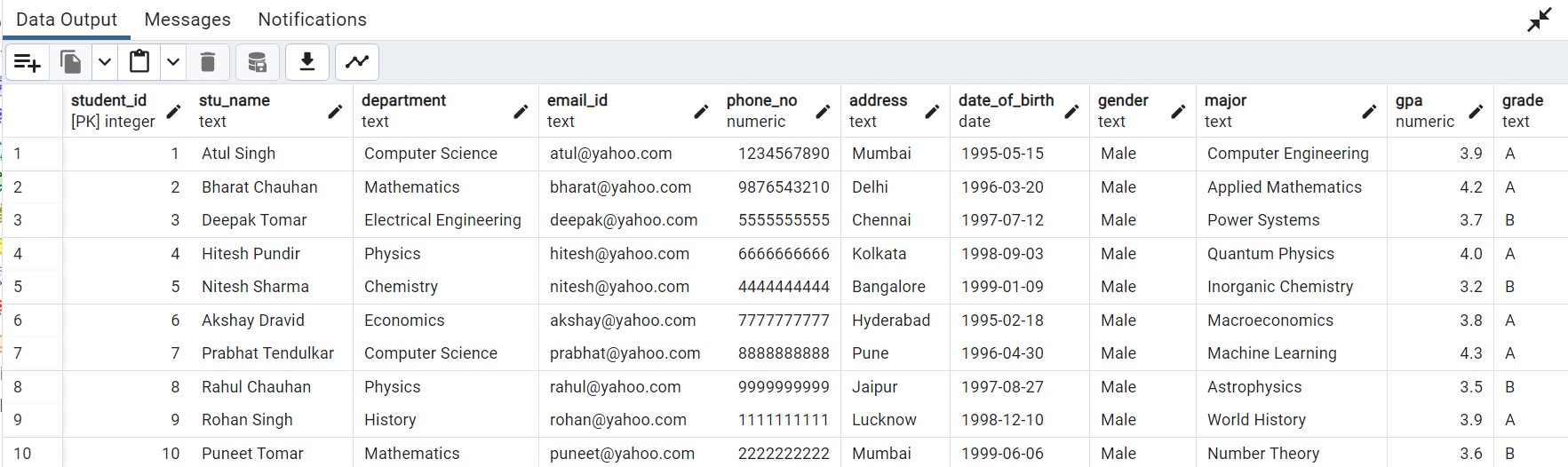
SELECT stu\_name

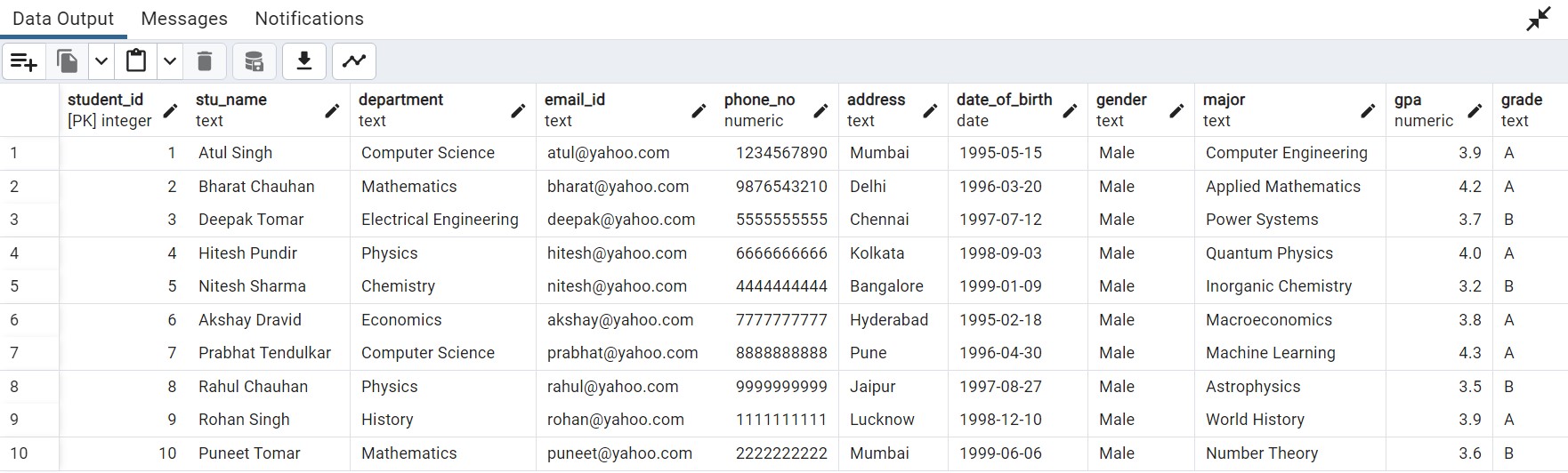
FROM student\_table

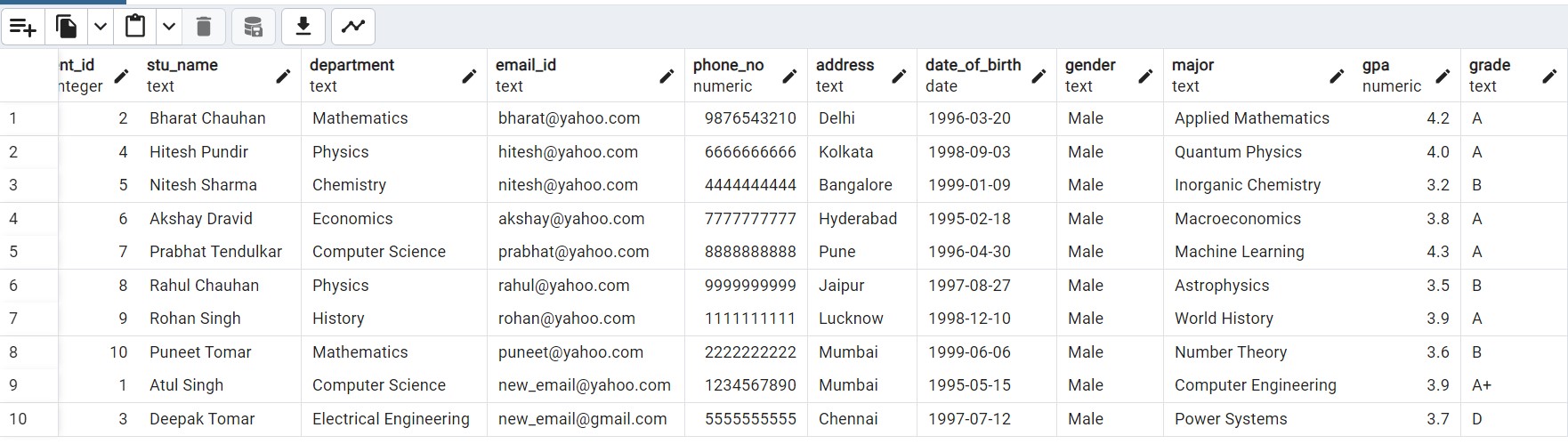
ORDER BY gpa DESC

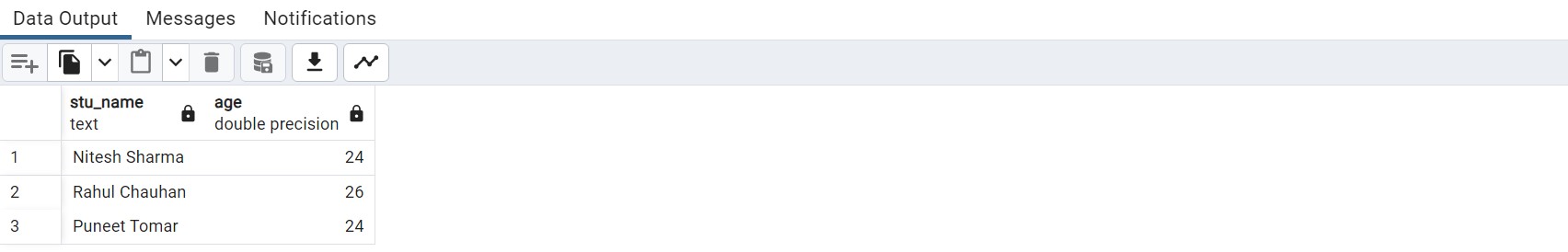
LIMIT 1;

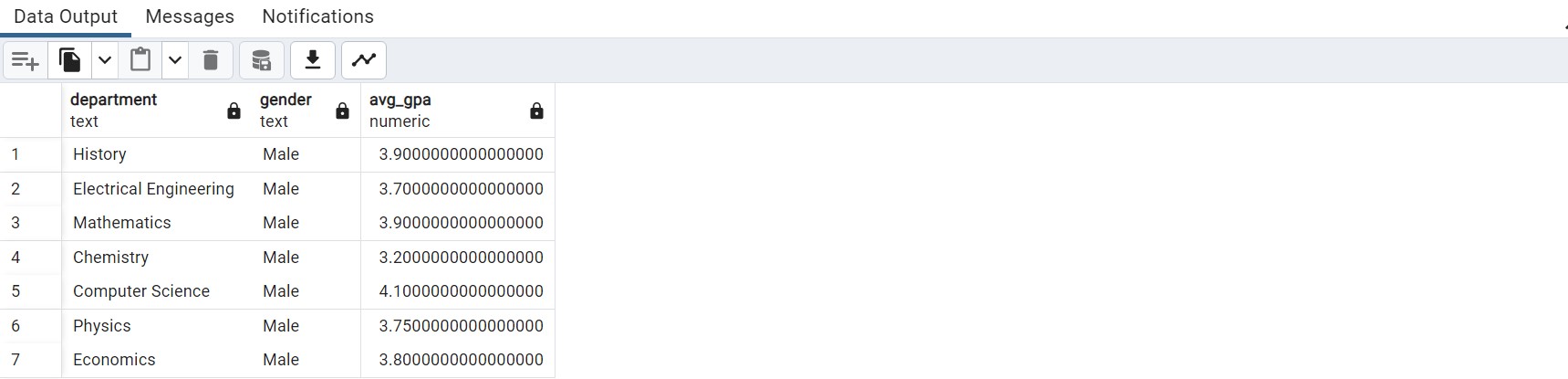


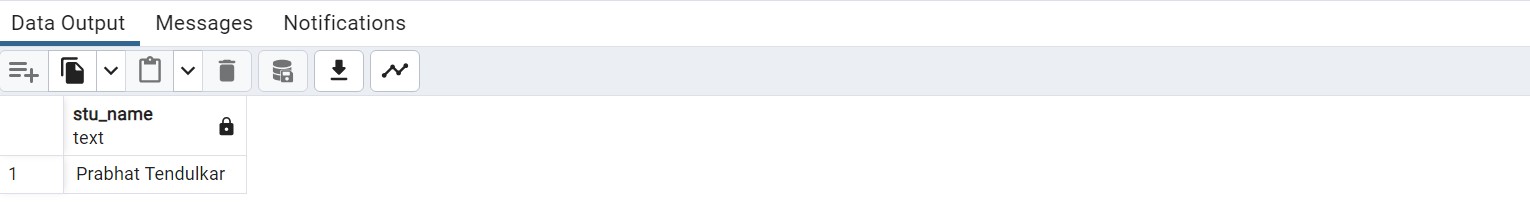












**TASK 3**  
  
  
CREATE DATABASE EventsManagement;

CREATE TABLE Events (

Event\_Id SERIAL PRIMARY KEY,

Event\_Name TEXT,

Event\_Date DATE,

Event\_Location TEXT,

Event\_Description TEXT

);

CREATE TABLE Attendees (

Attendee\_Id SERIAL PRIMARY KEY,

Attendee\_Name TEXT,

Attendee\_Phone NUMERIC,

Attendee\_Email TEXT,

Attendee\_City TEXT

);

CREATE TABLE Registrations (

Registration\_Id SERIAL PRIMARY KEY,

Event\_Id INT,

Attendee\_Id INT,

Registration\_Date DATE,

Registration\_Amount NUMERIC,

FOREIGN KEY (Event\_Id) REFERENCES Events(Event\_Id),

FOREIGN KEY (Attendee\_Id) REFERENCES Attendees(Attendee\_Id)

);

INSERT INTO Events (Event\_Name, Event\_Date, Event\_Location, Event\_Description)

VALUES

('Tech Conference', '2023-05-15', 'Bangalore', 'Annual technology conference'),

('Music Festival', '2023-06-25', 'Mumbai', 'Weekend music festival'),

('Sports Tournament', '2023-07-10', 'Delhi', 'Inter-school sports competition'),

('Food Expo', '2023-08-20', 'Chennai', 'Culinary delights from around the world'),

('Art Exhibition', '2023-09-05', 'Kolkata', 'Display of contemporary art');

INSERT INTO Attendees (Attendee\_Name, Attendee\_Phone, Attendee\_Email, Attendee\_City)

VALUES

('Akshay Sharma', 9876543210, 'akshay@gmail.com', 'Bangalore'),

('Deepika Chauhan', 9988776655, 'deepika@gmail.com', 'Mumbai'),

('Rahul Tomar', 7777777777, 'rahul@gmail.com', 'Delhi'),

('Neha Singh', 8888888888, 'neha@gmail.com', 'Kolkata'),

('Anjali Reddy', 9999999999, 'anjali@gmail.com', 'Chennai');

INSERT INTO Registrations (Event\_Id, Attendee\_Id, Registration\_Date, Registration\_Amount)

VALUES

(1, 1, '2023-04-20', 500),

(1, 2, '2023-04-21', 500),

(2, 2, '2023-05-05', 750),

(3, 3, '2023-06-10', 300),

(3, 4, '2023-06-11', 300);

INSERT INTO Events (Event\_Name, Event\_Date, Event\_Location, Event\_Description)

VALUES ('Tech Workshop', '2023-10-15', 'Pune', 'Hands-on technology workshop');

UPDATE Events

SET Event\_Description = 'Annual technology and innovation conference'

WHERE Event\_Id = 1;

DELETE FROM Events WHERE Event\_Id = 5;

INSERT INTO Attendees (Attendee\_Name, Attendee\_Phone, Attendee\_Email, Attendee\_City)

VALUES ('Sarika Verma', 7771112223, 'sarika@gmail.com', 'Hyderabad');

INSERT INTO Registrations (Event\_Id, Attendee\_Id, Registration\_Date, Registration\_Amount)

VALUES (2, 3, '2023-05-10', 750);

SELECT \* FROM Events;

SELECT A.Attendee\_Name, A.Attendee\_Email

FROM Attendees A

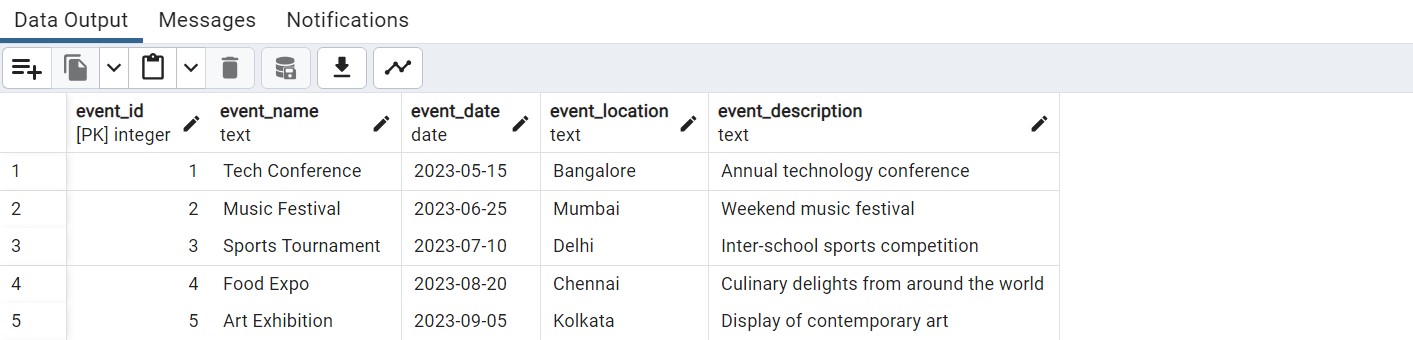
JOIN Registrations R ON A.Attendee\_Id = R.Attendee\_Id

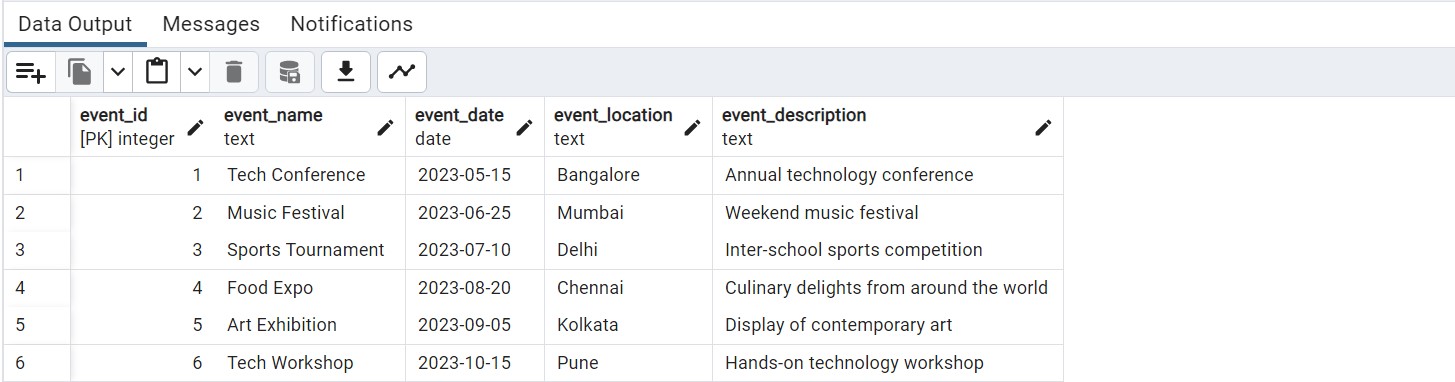
WHERE R.Event\_Id = 2;

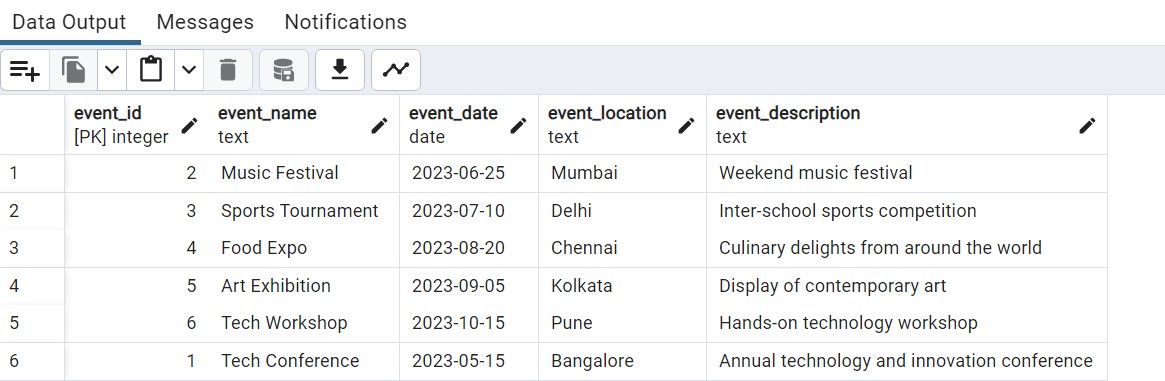
SELECT COUNT(\*) AS total\_attendees, AVG(Registration\_Amount) AS avg\_registration\_amount

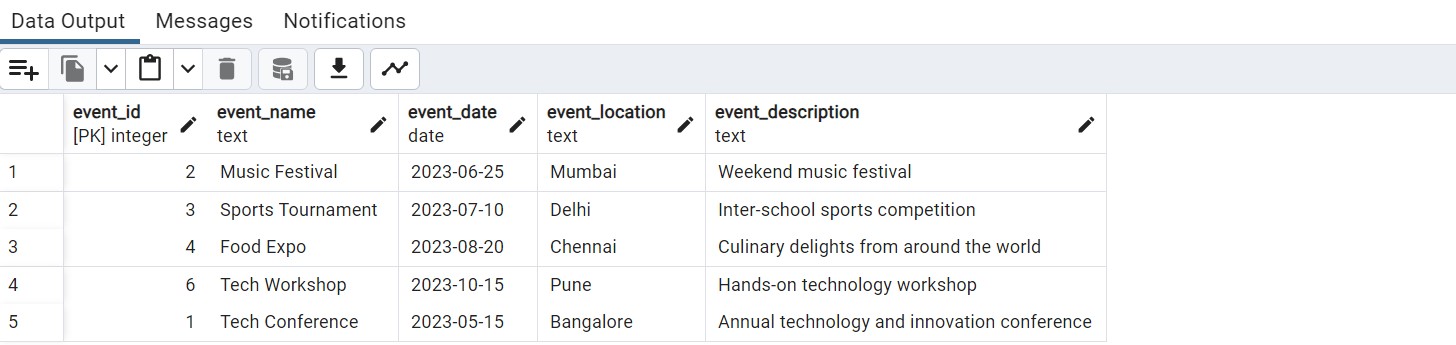
FROM Registrations

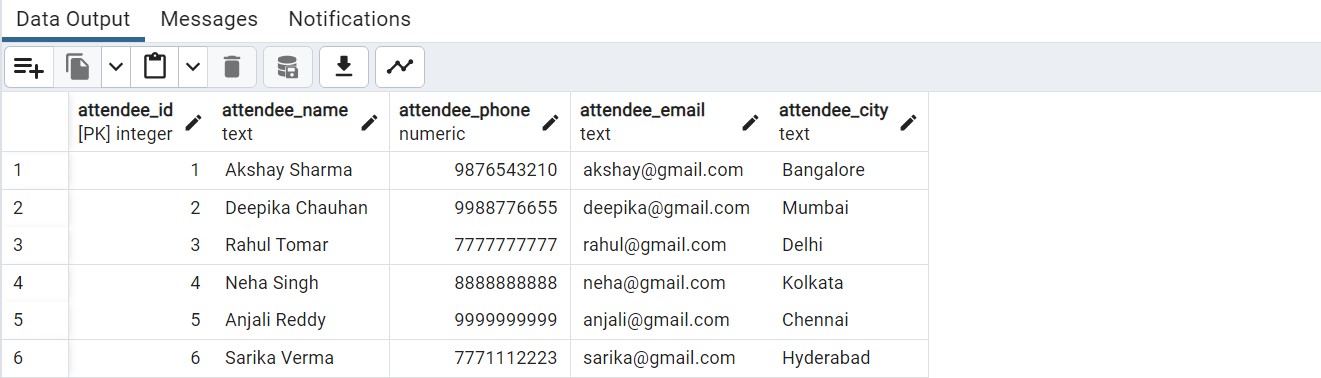
WHERE Event\_Id = 1;

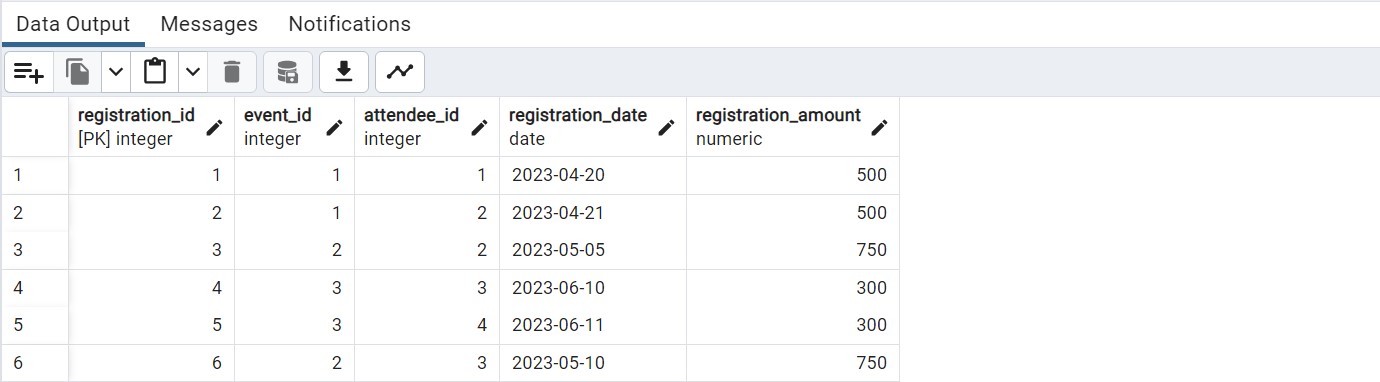


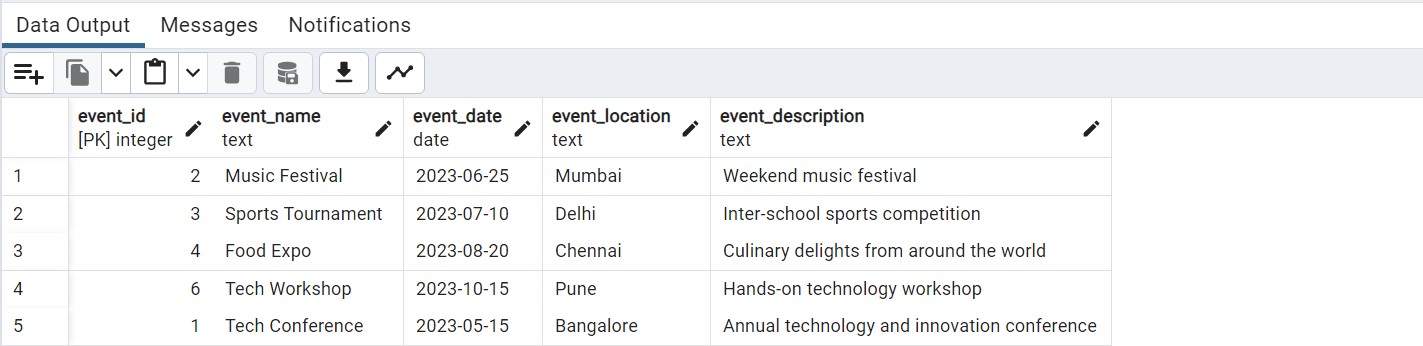


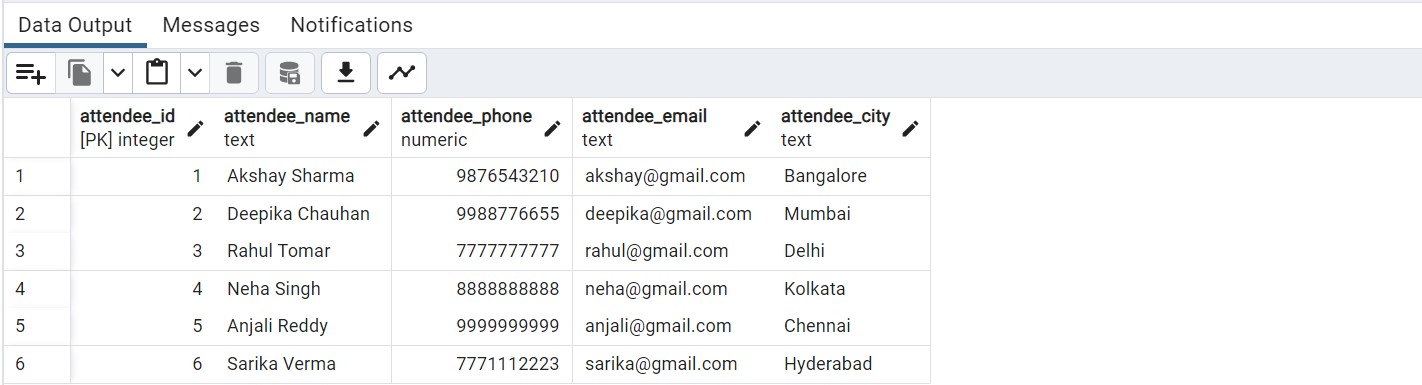


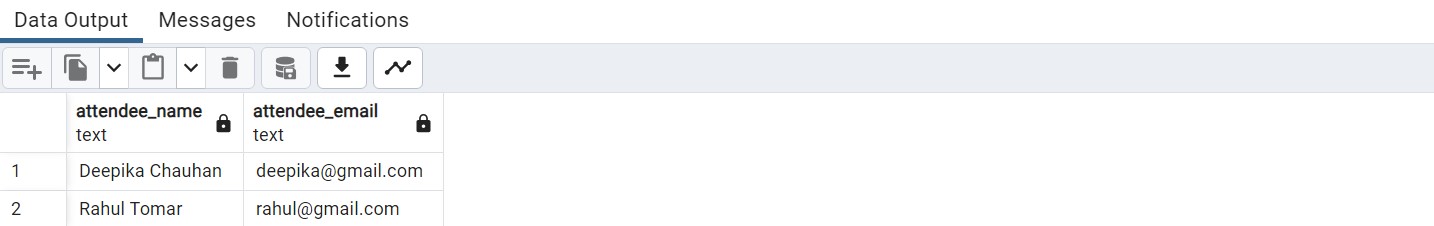


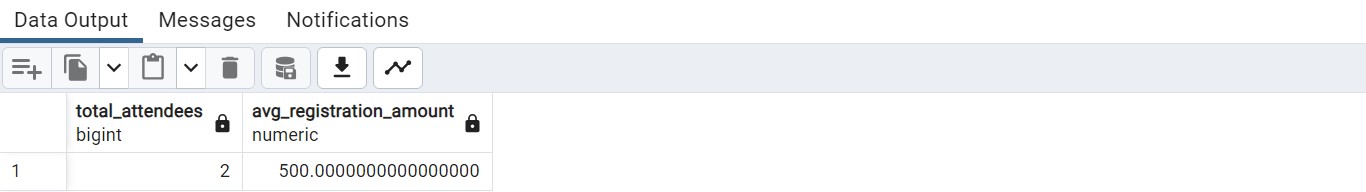












**TASK 4**  
  
CREATE DATABASE SalesData;

CREATE TABLE sales\_sample (

Product\_id SERIAL PRIMARY KEY,

Region VARCHAR(50),

Date DATE,

Sales\_Amount NUMERIC

);

INSERT INTO sales\_sample (Region, Date, Sales\_Amount)

VALUES

('East', '2023-01-05', 1000.00),

('West', '2023-01-10', 750.00),

('North', '2023-02-15', 1200.00),

('South', '2023-03-20', 900.00),

('East', '2023-04-25', 1100.00),

('West', '2023-05-30', 850.00),

('North', '2023-06-05', 1300.00),

('South', '2023-07-10', 950.00),

('East', '2023-08-15', 1050.00),

('West', '2023-09-20', 800.00);

SELECT Region, Product\_id, SUM(Sales\_Amount) AS Total\_Sales

FROM sales\_sample

GROUP BY Region, Product\_id

ORDER BY Region, Product\_id;

SELECT Region, SUM(Sales\_Amount) AS Total\_Sales

FROM sales\_sample

GROUP BY ROLLUP (Region, Product\_id)

ORDER BY Region, Product\_id;

SELECT Region, Product\_id, Date, SUM(Sales\_Amount) AS Total\_Sales

FROM sales\_sample

GROUP BY CUBE (Region, Product\_id, Date)

ORDER BY Region, Product\_id, Date;

SELECT Region, Date, SUM(Sales\_Amount) AS Total\_Sales

FROM sales\_sample

WHERE Region = 'East' AND Date BETWEEN '2023-04-01' AND '2023-06-30'

GROUP BY Region, Date

ORDER BY Date;

SELECT Region, Product\_id, Date, SUM(Sales\_Amount) AS Total\_Sales

FROM sales\_sample

WHERE Region = 'West' AND Product\_id = 2 AND Date BETWEEN '2023-01-01' AND '2023-03-31'

GROUP BY Region, Product\_id, Date

ORDER BY Date;

