CAPSTONE PROJECT

Task 1 – Academic Management System:

1. Database Creation:

```
create database student_database;
use student_database;
```

a) Creating table StudentInfo:

```
Create table StudentInfo (
STU_ID int,
STU_NAME varchar(100),
DOB DATE,
PHONE_NO VARCHAR(10),
EMAIL_ID varchar(50),
ADDRESS varchar(250),
primary key (STU_ID));
```

b) creating table CourseInfo:

```
Create table CourseInfo(
COURSE_ID INT ,
COURSE_NAME VARCHAR(100),
COURSE_INSTRUCTOR_NAME VARCHAR(100),
primary key (COURSE_ID));
```

c) creating table EnrollmentInfo:

```
Create table EnrollmentInfo(
ENROLLMENT_ID INT,
STU_ID int,
COURSE_ID INT,
ENROLL_STATUS varchar(20),
primary key (ENROLLMENT_ID),
FOREIGN KEY (STU_ID) REFERENCES StudentInfo(STU_ID),
FOREIGN KEY (COURSE_ID) REFERENCES CourseInfo(COURSE_ID));
```

2) Data Creation

a) Insert Into StudentInfo

```
(STU_ID, STU_NAME, DOB, PHONE_NO, EMAIL_ID ,ADDRESS) Values
('101', 'Virat', '1991-02-13',9898989898, 'Virat123@gamil.com', 'Banglore'),
('102', 'Dhoni', '1984-08-04',9999999995, 'Dhoni201@gamil.com', 'Banglore'),
('103', 'Bhuvi', '1994-08-15',9999999976, 'Bhuvi301@gamil.com', 'Chennai'),
('104', 'Robin', '1994-01-23',9999999669, 'Robin401@gamil.com', 'Mumbai'),
('105', 'John Austin', '1993-06-24',9999993456, 'John501@gamil.com', 'Hyderabad'),
('106', 'Daniel', '1992-07-27',9999997654, 'Dan601@gamil.com', 'Hyderabad');
```

b) Insert Into CourseInfo(COURSE_ID,COURSE_NAME, COURSE_INSTRUCTOR_NAME) values

```
(001, 'SQL','Sofy'),
(002, 'Python','Arvind'),
(003, 'AWS','Seema'),
(004, 'JAVA','Harry'),
(005, 'CSS','Jacky');
```

c) insert into EnrollmentInfo(ENROLLMENT_ID, STU_ID, COURSE_ID, ENROLL_STATUS) values

(10001, 101, 001,'ENROLLED'),

(10002, 103, 002,'ENROLLED'),

(10003, 104, 004,'ENROLLED'),

(10004, 102, 003,'ENROLLED'),

(10005, 105, 003,'NOT ENROLLED'),

(10006, 106, 005,'ENROLLED');

3) **Retrieve the Student Information**

a) Write a query to retrieve Student details, Such as Student Name, Contact Informations and Enrollment Status

Select s.STU_NAME, s.PHONE_NO, s.ADDRESS,

e.ENROLL_STATUS

from StudentInfo s join EnrollmentInfo e

on s.STU_ID=e.STU_ID

order by e.ENROLL_STATUS ;

STU_NAME	PHONE_NO	ADDRESS	ENROLL_STATUS
Virat	9898989898	Banglore	ENROLLED
Bhuvi	9999999976	Chennai	ENROLLED
Robin	9999999669	Mumbai	ENROLLED
Dhoni	9999999995	Banglore	ENROLLED
Daniel	9999997654	Hyderabad	ENROLLED
John Austin	9999993456	Hyderabad	NOT ENROLLED

b) Write a query to retrieve a list of courses in which a specific student enrolled

```
select c.COURSE_NAME, s.STU_NAME
from EnrollmentInfo e
join CourseInfo c on e.course_id = c.course_id
join StudentInfo s on s.STU_ID = e.STU_ID;
```

COURSE_NAME	STU_NAME
SQL	Virat
SQL	Bhuvi
Python	Robin
JAVA	Dhoni
AWS	John Austin
CSS	Daniel

C) Write a query to retrieve course information, including course name, instructor information

```
Select * From CourseInfo;
```

COURSE_ID	COURSE_NAME	COURSE_INSTRUCTOR_NAME
1	SQL	Sofy
2	Python	Arvind
3	AWS	Seema
4	JAVA	Harry
5	CSS	Jacky

d) Write a query to retrieve course information for a specific course

```
Select COURSE_ID, COURSE_NAME, COURSE_INSTRUCTOR_NAME
 from CourseInfo
 where COURSE_NAME='SQL';
```

COURSE_ID	COURSE_NAME	COURSE_INSTRUCTOR_NAME
1	SQL	Sofy

e) Write a query to retrieve course information for multiple courses

```
Select COURSE_ID,
        COURSE_NAME,
        COURSE_INSTRUCTOR_NAME
from CourseInfo
where COURSE_NAME in ('SQL','Python');
```

COURSE_ID	COURSE_NAME	COURSE_INSTRUCTOR_NAME
1	SQL	Sofy
2	Python	Arvind

f) Test the queries to ensure accurate retrieval of Student Information

(Execute queries and verify the results against the expected output)

```
Select * From StudentInfo;
```

STU_ID	STU_NAME	DOB	PHONE_NO	EMAIL_ID	ADDRESS
101	Virat	1991-02-13	9898989898	Virat123@gamil.com	Banglore
102	Dhoni	1984-08-04	9999999995	Dhoni201@gamil.com	Banglore
103	Bhuvi	1994-08-15	9999999976	Bhuvi301@gamil.com	Chennai
104	Robin	1994-01-23	9999999669	Robin401@gamil.com	Mumbai
105	John Austin	1993-06-24	9999993456	John501@gamil.com	Hyderabad
106	Daniel	1992-07-27	9999997654	Dan601@gamil.com	Hyderabad

4) Reporting and Analytics (Using joining queries)

a) Write a query to retrieve the number of students enrolled in each course

```
Select c.Course_Name ,
count(e.stu_id) as numberofStudents
from CourseInfo c join EnrollmentInfo e
on c.course_id=e.course_ID
where e.enroll_status = 'ENROLLED'
group by 1;
```

COURSE_NAME	numberofStudents
CSS	1
JAVA	1
Python	1
SQL	2

b) Write a query to retrieve the list of students enrolled in a specific course

```
select e.COURSE_ID,
c.COURSE_NAME,
s.STU_NAME
from CourseInfo c join EnrollmentInfo e on c.course_id=e.course_ID
join StudentInfo s on s.STU_ID = e.STU_ID
where e.enroll_status = 'ENROLLED' and c.COURSE_NAME = 'SQL';
```

COURSE_ID	COURSE_NAME	STU_NAME
1	SQL	Virat
1	SQL	Bhuvi

c) Write a query to retrieve the count of enrolled students for each instructor

```
Select c.COURSE_INSTRUCTOR_NAME , count(e.Stu_id) as numberofStudents
from CourseInfo c join EnrollmentInfo e
on c.course_id=e.course_ID
where e.enroll_status = 'ENROLLED'
group by 1;
```

COURSE_INSTRUCTOR_NAME	numberofStudents
Arvind	1
Harry	1
Jacky	1
Sofy	2

d) Write a query to retrieve the list of students who are enrolled in a multiple course

```
Select e.stu_id , count(c.course_id)
from CourseInfo c join EnrollmentInfo e
on c.course_id=e.course_ID
where e.enroll_status = 'ENROLLED'
group by 1
having count(c.course_id) >1
```

Query has no result

e) Write a query to retrieve the courses that have the highest number of enrolled students(arranging from highest to lowest)

```
Select c.course_id,c.COURSE_NAME,count(e.stu_id) as numberofStud
from CourseInfo c join EnrollmentInfo e
on c.course_id=e.course_ID
where e.enroll_status = 'ENROLLED'
group by 1,2
having count(e.stu_id) >=1
order by count(e.stu_id) desc;
```

COURSE_ID	COURSE_NAME	numberofStud
1	SQL	2
2	Python	1
4	JAVA	1
5	CSS	1

Task 2: Student Database Management System(Postgresql)

-- Database: Student_Database

-- DROP DATABASE IF EXISTS "Student_Database";

1) Database setup

```
CREATE DATABASE "Student_Database"
    WITH
```

```
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'English_United States.1252'
    LC_CTYPE = 'English_United States.1252'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;

create table Student_table
(Student_id int, Stu_name varchar(100),  Department varchar(50), email_id varchar(50),
Phone_no numeric, Address varchar(250),  Date_Of_Birth DATE, Gender Varchar(30),
Major Varchar(50), GPA numeric, Grade varchar(10));
```

2) Data entry

```
Insert Into Student_table
(Student_id, Stu_name, Department, email_id, Phone_no, Address,  Date_Of_Birth, Gender, Major,
GPA, Grade) Values
('1', 'Pravalika Gourishetty', 'Business', 'PravalikaG@gmail.com', '9999999991',  'Delhi', '1999-04-26',
'Female', 'MBA', '8.8', 'A'),
 ('2', 'Krishna Kumar', 'Arts and Sciences', 'krishnakumar@gmail.com',  '9999999992', 'Bangalore', '1992-
07-15', 'Male', 'Mathematics', '8.6', 'A'),
 ('3', 'Ravi Nema', 'Business', 'Ravinema@gmail.com', '9999999993',  'Delhi', '1995-06-28', 'Male', 'MBA',
'8.5', 'A'),
 ('4', 'Deepa Navya', 'Arts and Sciences', 'Deepanavya@gmail.com',  '9999999994', 'Dehradhun', '1997-
01-12', 'Male', 'Physics', '7.6', 'B'),
 ('5', 'Katrina', 'Arts and communication', 'Katrina@gmail.com', '9999999995', 'Mumbai', '1995-10-26',
'Female', 'Communication', '6.9', 'B'),
 ('6', 'Santosh', 'Arts and Sciences', 'Santosh@gmail.com',  '9999999996', 'Goa', '2000-01-03', 'Female',
'Computer Science', '5.5', 'C'),
 ('7', 'Hima', 'Human Development', 'Hima@gmail.com', '9999999997', 'Uttar Pradesh', '2000-10-05',
'Male', 'Counseling', '7.0', 'B'),
```

('8', 'Pranavi Yadav', 'Arts and communication', 'Pranavi@gmail.com', '9999999998', 'Gujarat', '1999-11-16', 'Male', 'Theatre', '9.2', 'A'),

('9', 'Sampath', 'Business', 'Sampath@gmail.com', '9999999999', 'Rajasthan', '2003-08-14', 'Male', 'Accounting', '7.9', 'B'),

('10', 'Sandhya', 'Arts and Sciences', 'Sandhya@gmail.com', '9999999911', 'Mumbai', '1997-03-12', 'Female', 'Physics', '4.9', 'C');

Select * from Student_table;

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
1	Pravalika Gourishetty	Business	PravalikaG@gmail.com	9999999991	Delhi	1999-04-26T00:00:00.000Z	Female	MBA	8.8	A
2	Krishna Kumar	Arts and Sciences	krishnakumar@gmail.com	9999999992	Bangalore	1992-07-15T00:00:00.000Z	Male	Mathematics	8.6	A
3	Ravi Nema	Business	Ravinema@gmail.com	9999999993	Delhi	1995-06-28T00:00:00.000Z	Male	MBA	8.5	A
4	Deepa Navya	Arts and Sciences	Deepanavya@gmail.com	9999999994	Dehradhun	1997-01-12T00:00:00.000Z	Male	Physics	7.6	B
5	Katrina	Arts and communication	Katrina@gmail.com	9999999995	Mumbai	1995-10-26T00:00:00.000Z	Female	Communication	6.9	B
6	Santosh	Arts and Sciences	Santosh@gmail.com	9999999996	Goa	2000-01-03T00:00:00.000Z	Female	Computer Science	5.5	C
7	Hima	Human Development	Hima@gmail.com	9999999997	Uttar Pradesh	2000-10-05T00:00:00.000Z	Male	Counseling	7.0	B
8	Pranavi Yadav	Arts and communication	Pranavi@gmail.com	9999999998	Gujarat	1999-11-16T00:00:00.000Z	Male	Theatre	9.2	A
9	Sampath	Business	Sampath@gmail.com	9999999999	Rajasthan	2003-08-14T00:00:00.000Z	Male	Accounting	7.9	B
10	Sandhya	Arts and Sciences	Sandhya@gmail.com	9999999911	Mumbai	1997-03-12T00:00:00.000Z	Female	Physics	4.9	C

3) Student information retrieval

Select * from Student_table

Order By GPA desc, Grade;

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
8	Pranavi Yadav	Arts and communication	Pranavi@gmail.com	9999999998	Gujarat	1999-11-16T00:00:00.000Z	Male	Theatre	9.2	A
1	Pravalika Gourishetty	Business	PravalikaG@gmail.com	9999999991	Delhi	1999-04-26T00:00:00.000Z	Female	MBA	8.8	A
2	Krishna Kumar	Arts and Sciences	krishnakumar@gmail.com	9999999992	Bangalore	1992-07-15T00:00:00.000Z	Male	Mathematics	8.6	A
3	Ravi Nema	Business	Ravinema@gmail.com	9999999993	Delhi	1995-06-28T00:00:00.000Z	Male	MBA	8.5	A
9	Sampath	Business	Sampath@gmail.com	9999999999	Rajasthan	2003-08-14T00:00:00.000Z	Male	Accounting	7.9	B
4	Deepa Navya	Arts and Sciences	Deepanavya@gmail.com	9999999994	Dehradhun	1997-01-12T00:00:00.000Z	Male	Physics	7.6	B
7	Hima	Human Development	Hima@gmail.com	9999999997	Uttar Pradesh	2000-10-05T00:00:00.000Z	Male	Counseling	7.0	B
5	Katrina	Arts and communication	Katrina@gmail.com	9999999995	Mumbai	1995-10-26T00:00:00.000Z	Female	Communication	6.9	B
6	Santosh	Arts and Sciences	Santosh@gmail.com	9999999996	Goa	2000-01-03T00:00:00.000Z	Female	Computer Science	5.5	C
10	Sandhya	Arts and Sciences	Sandhya@gmail.com	9999999911	Mumbai	1997-03-12T00:00:00.000Z	Female	Physics	4.9	C

4) Male students

Select * from Student_table

where Gender = 'Male';

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
2	Krishna Kumar	Arts and Sciences	krishnakumar@gmail.com	9999999992	Bangalore	1992-07-15T00:00:00.000Z	Male	Mathematics	8.6	A
3	Ravi Nema	Business	Ravinema@gmail.com	9999999993	Delhi	1995-06-28T00:00:00.000Z	Male	MBA	8.5	A
4	Deepa Navya	Arts and Sciences	Deepanavya@gmail.com	9999999994	Dehradhun	1997-01-12T00:00:00.000Z	Male	Physics	7.6	B
7	Hima	Human Development	Hima@gmail.com	9999999997	Uttar Pradesh	2000-10-05T00:00:00.000Z	Male	Counseling	7.0	B
8	Pranavi Yadav	Arts and communication	Pranavi@gmail.com	9999999998	Gujarat	1999-11-16T00:00:00.000Z	Male	Theatre	9.2	A
9	Sampath	Business	Sampath@gmail.com	9999999999	Rajasthan	2003-08-14T00:00:00.000Z	Male	Accounting	7.9	B

5) GPA less than 5

Select * from Student_table

where GPA <5.0;

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
10	Sandhya	Arts and Sciences	Sandhya@gmail.com	9999999911	Mumbai	1997-03-12T00:00:00.000Z	Female	Physics	4.9	C

6) Update email_id and Grade

```
UPDATE Student_table

SET email_id = 'Sarada@gmail.com', Grade = 'D'

Where Student_id = '9';
```

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
9	Sampath	Business	Sarada@gmail.com	9999999999	Rajasthan	2003-08-14T00:00:00.000Z	Male	Accounting	7.9	D

7) Grade B

```
SELECT Stu_name, date_part('year',age(Date_Of_Birth)) as Age

FROM Student_table

Where Grade = 'B';
```

stu_name	age
Deepa Navya	26
Katrina	28
Hima	23

8) Grouping and calculation

```
Select Department, Gender, Avg(GPA)

From Student_table

Group By 1,2;
```

department	gender	avg
Business	Male	8.2000000000000000
Arts and communication	Female	6.9000000000000000
Business	Female	8.8000000000000000
Arts and Sciences	Male	8.1000000000000000
Arts and communication	Male	9.2000000000000000
Human Development	Male	7.0000000000000000
Arts and Sciences	Female	5.2000000000000000

9) Renaming

Alter table Student_table

Rename to Student_info;

Select * from Student_info;

student_id	stu_name	department	email_id	phone_no	address	date_of_birth	gender	major	gpa	grade
1	Pravalika Gourishetty	Business	PravalikaG@gmail.com	9999999991	Delhi	1999-04-26T00:00:00.000Z	Female	MBA	8.8	A
2	Krishna Kumar	Arts and Sciences	krishnakumar@gmail.com	9999999992	Bangalore	1992-07-15T00:00:00.000Z	Male	Mathematics	8.6	A
3	Ravi Nema	Business	Ravinema@gmail.com	9999999993	Delhi	1995-06-28T00:00:00.000Z	Male	MBA	8.5	A
4	Deepa Navya	Arts and Sciences	Deepanavya@gmail.com	9999999994	Dehradhun	1997-01-12T00:00:00.000Z	Male	Physics	7.6	B
5	Katrina	Arts and communication	Katrina@gmail.com	9999999995	Mumbai	1995-10-26T00:00:00.000Z	Female	Communication	6.9	B
6	Santosh	Arts and Sciences	Santosh@gmail.com	9999999996	Goa	2000-01-03T00:00:00.000Z	Female	Computer Science	5.5	C
7	Hima	Human Development	Hima@gmail.com	9999999997	Uttar Pradesh	2000-10-05T00:00:00.000Z	Male	Counseling	7.0	B
8	Pranavi Yadav	Arts and communication	Pranavi@gmail.com	9999999998	Gujarat	1999-11-16T00:00:00.000Z	Male	Theatre	9.2	A
10	Sandhya	Arts and Sciences	Sandhya@gmail.com	9999999911	Mumbai	1997-03-12T00:00:00.000Z	Female	Physics	4.9	C
9	Sampath	Business	Sarada@gmail.com	9999999999	Rajasthan	2003-08-14T00:00:00.000Z	Male	Accounting	7.9	D

10) Students with highest GPA

Select Stu_name, GPA

From Student_info where GPA = (Select Max(GPA) From Student_info);

stu_name	gpa
Pranavi Yadav	9.2

Task 3: Event Management Sysytem (Postgresql)

```
-- Database: EventsManagement

-- DROP DATABASE IF EXISTS "EventsManagement";


CREATE DATABASE "EventsManagement"
    WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'English_United States.1252'
    LC_CTYPE = 'English_United States.1252'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;
```

1) Database Creation

```
Create table Events (
Event_Id Int,
Event_Name Varchar(30),
 Event_Date Date,
Event_Location Varchar(100),
Event_Description Varchar(200),
primary key (Event_Id));


Create table Attendees (
Attendee_Id Int,
Attendee_Name Varchar(30),
Attendee_Phone numeric,
Attendee_Email Varchar(30),
```

```
Attendee_City Varchar(20),
primary key (Attendee_Id));

Create table Registrations (
Registration_Id Int,
Event_Id Int,
Attendee_Id Int,
Registration_Date Date,
Registration_Amount numeric,
primary key (Registration_Id),
FOREIGN KEY (Event_Id) REFERENCES Events(Event_Id),
FOREIGN KEY (Attendee_Id) REFERENCES Attendees(Attendee_Id));
```

2) Data Creation

```
Insert into Events
(Event_Id, Event_Name, Event_Date, Event_Location, Event_Description) values
('101', 'Taylor swift', '2024-01-08', 'Mumbai', 'Music Show'),
('102', 'Comicon', '2023-11-23', 'Bangalore', 'Exhibition'),
('103', 'IIMF', '2023-11-12', 'Hyderabad', 'Music Show'),
('104', 'Hitcki', '2024-03-06', 'Kochi', 'Comedy'),
('105', 'Ted Talk', '2024-02-23', 'Kolkata', 'Comedy')
;
Select * from Events;
```

event_id	event_name	event_date	event_location	event_description
101	Taylor swift	2024-01-08T00:00:00.000Z	Mumbai	Music Show
102	Comicon	2023-11-23T00:00:00.000Z	Bangalore	Exhibition
103	IIMF	2023-11-12T00:00:00.000Z	Hyderabad	Music Show
104	Hitcki	2024-03-06T00:00:00.000Z	Kochi	Comedy
105	Ted Talk	2024-02-23T00:00:00.000Z	Kolkata	Comedy

Insert into Attendees (Attendee_Id, Attendee_Name, Attendee_Phone, Attendee_Email, Attendee_City) values

('1001', 'Pravalika', '9999999991', 'Pravalika@gmail.com', 'Hyderabad'),

('1002', 'Raj kumar', '9999999992', 'Raj@gmail.com', 'Delhi'),

('1003', 'Abhiram', '9999999993', 'Abhiram@gmail.com', 'Mumbai'),

('1004', 'Kumar', '9999999994', 'Kumar@gmail.com', 'Uttar Pradesh'),

('1005', 'Katty', '9999999995', 'Katty@gmail.com', 'Mumbai'),

('1006', 'Shiv', '9999999996', 'Shiv@gmail.com', 'Goa'),

('1007', 'Dinesh', '9999999997', 'Dinesh@gmail.com', 'Uttrakhand'),

('1008', 'Sarthak', '9999999998', 'Sarthak@gmail.com', 'Delhi')

;

Select * from Attendees;

attendee_id	attendee_name	attendee_phone	attendee_email	attendee_city
1001	Pravalika	9999999991	Pravalika@gmail.com	Hyderabad
1002	Raj kumar	9999999992	Raj@gmail.com	Delhi
1003	Abhiram	9999999993	Abhiram@gmail.com	Mumbai
1004	Kumar	9999999994	Kumar@gmail.com	Uttar Pradesh
1005	Katty	9999999995	Katty@gmail.com	Mumbai
1006	Shiv	9999999996	Shiv@gmail.com	Goa
1007	Dinesh	9999999997	Dinesh@gmail.com	Uttrakhand
1008	Sarthak	9999999998	Sarthak@gmail.com	Delhi

Insert into Registrations

(Registration_Id, Event_Id, Attendee_Id, Registration_Date, Registration_Amount) values

('10001', '101', '1001', '2024-01-08', '7500'),

('10002', '102', '1003', '2023-09-25', '1000'),

('10003', '103', '1002', '2023-10-29', '2000'),

('10004', '104', '1004', '2023-10-20', '500'),

('10005', '101', '1005', '2023-09-10', '7500'),

('10006', '103', '1008', '2023-09-15', '2000'),

('10007', '101', '1007', '2023-11-23', '7500'),

('10008', '102', '1006', '2023-11-12', '1000')

;

Select *from Registrations;

registration_id	event_id	attendee_id	registration_date	registration_amount
10001	101	1001	2024-01-08T00:00:00.000Z	7500
10002	102	1003	2023-09-25T00:00:00.000Z	1000
10003	103	1002	2023-10-29T00:00:00.000Z	2000
10004	104	1004	2023-10-20T00:00:00.000Z	500
10005	101	1005	2023-09-10T00:00:00.000Z	7500
10006	103	1008	2023-09-15T00:00:00.000Z	2000
10007	101	1007	2023-11-23T00:00:00.000Z	7500
10008	102	1006	2023-11-12T00:00:00.000Z	1000

3) Manage Event Details

a) Insert New event

Insert INTO Events (Event_Id, Event_Name, Event_Date, Event_Location, Event_Description) VALUES

('106', 'Untold Stories', '2023-11-19', 'Delhi', 'Storytelling');

b) Update event's information

```
UPDATE Events
SET Event_Location = 'Bangalore' Where Event_Id = '104'
;
```

c) Deleting an event

```
Delete from Events where Event_Id = '105'
;
```

4) Manage Track attendees and handle events

a) Insert new attendee

```
Insert into Attendees (Attendee_Id, Attendee_Name, Attendee_Phone, Attendee_Email, Attendee_City)
values ('1009', 'Kavitha', '9999999999', 'Kavitha@gmail.com', 'Hyderabad');
```

b) Register attendee

```
Insert into Registrations (Registration_Id, Event_Id, Attendee_Id, Registration_Date, Registration_Amount)
values ('10009', '101', '1009', '2023-11-11', '7500');
```

5) Retrieve event information, Generate attendee list, Calculate event attendee statistics

```
Select * from Events;
Select * from Attendees;
select * from Registrations;
with Event1 as(
select E.Event_id, E.event_name, E.event_date, E.event_location,
sum(R.registration_amount) over(partition by E.event_id) as Amountgenperevent
from Events E join Registrations R on E.event_id = R.event_id
join Attendees A on A.attendee_id = R.attendee_id)
select Event_id, event_name, event_location, Amountgenperevent
```

from Event1

group by 1,2,3,4

event_id	event_name	event_location	amountgenperevent
103	IIMF	Hyderabad	4000
104	Hitcki	Kochi	500
101	Taylor swift	Mumbai	22500
102	Comicon	Bangalore	2000

Task 4: OLAP Operations (Postgresql)

```
-- Database: Sales Data


-- DROP DATABASE IF EXISTS "Sales Data ";


CREATE DATABASE "Sales Data "
    WITH
    OWNER = postgres
    ENCODING = 'UTF8'
    LC_COLLATE = 'English_United States.1252'
    LC_CTYPE = 'English_United States.1252'
    TABLESPACE = pg_default
    CONNECTION LIMIT = -1
    IS_TEMPLATE = False;
```

1) Database creation

```
Create table Sales_sample (Product_Id Int, Region Varchar(50), On_date Date,
Sales_Amount Numeric);
```

2) Data Creation

```
Insert into Sales_sample (Product_Id, Region, On_date, Sales_Amount) values
('1', 'East', '2023-11-10', '20000'),
('2', 'West', '2023-08-19', '50000'),
('2', 'East', '2023-11-21', '40000'),
('3', 'North', '2023-07-20', '15000'),
('4', 'North', '2023-08-07', '45000'),
('2', 'South', '2023-08-24', '45000'),
```

('5', 'North', '2023-11-22', '20000'),

('5', 'West', '2023-11-11', '60000'),

('3', 'East', '2023-09-19', '50000'),

('1', 'West', '2023-09-29', '70000')

;

Select * from Sales_Sample;

product_id	region	on_date	sales_amount
1	East	2023-11-10T00:00:00.000Z	20000
2	West	2023-08-19T00:00:00.000Z	50000
2	East	2023-11-21T00:00:00.000Z	40000
3	North	2023-07-20T00:00:00.000Z	15000
4	North	2023-08-07T00:00:00.000Z	45000
2	South	2023-08-24T00:00:00.000Z	45000
5	North	2023-11-22T00:00:00.000Z	20000
5	West	2023-11-11T00:00:00.000Z	60000
3	East	2023-09-19T00:00:00.000Z	50000
1	West	2023-09-29T00:00:00.000Z	70000

3) OLAP operations

a) Drill down

Select Region, Product_Id, Sum(Sales_Amount) as Sales_Amount

From Sales_Sample

Group By 1,2

Order By Region, Product_Id, Sales_Amount;

region	product_id	sales_amount
East	1	20000
East	2	40000
East	3	50000
North	3	15000
North	4	45000
North	5	20000
South	2	45000
West	1	70000
West	2	50000
West	5	60000

b) Roll Up

Select Region, Product_Id, Sum(Sales_Amount) as Sales_Amount

From Sales_Sample

Group By Rollup (1,2)

Order By Region;

region	product_id	sales_amount
East	1	20000
East	2	40000
East	3	50000
East		110000
North	3	15000
North	4	45000
North	5	20000
North		80000
South	2	45000
South		45000
West	1	70000
West	2	50000
West	5	60000
West		180000
		415000

c) Cube

Select Region, Product_Id, On_Date, Sum(Sales_Amount) as Sales_Amount

From Sales_Sample

Group By Cube (1,2,3)

Order By Region, Product_Id, On_Date, Sales_Amount;

region	product_id	on_date	sales_amount
East	1	2023-11-10T00:00:00.000Z	20000
East	1		20000
East	2	2023-11-21T00:00:00.000Z	40000
East	2		40000
East	3	2023-09-19T00:00:00.000Z	50000
East	3		50000
East		2023-09-19T00:00:00.000Z	50000
East		2023-11-10T00:00:00.000Z	20000
East		2023-11-21T00:00:00.000Z	40000
East			110000
North	3	2023-07-20T00:00:00.000Z	15000
North	3		15000
North	4	2023-08-07T00:00:00.000Z	45000
North	4		45000
North	5	2023-11-22T00:00:00.000Z	20000
North	5		20000
North		2023-07-20T00:00:00.000Z	15000

d) Slice

Select Region, Product_Id, On_Date, Sum(Sales_Amount) as Sales_Amount

From Sales_Sample

Where Region in('North', 'South') OR On_Date between To_date('2023-08-20','YYYY-MM-DD') And To_Date('2023-10-20','YYYY-MM-DD')

Group By 1,2,3

Order By Region, Product_Id, On_Date, Sales_Amount;

region	product_id	on_date	sales_amount
East	3	2023-09-19T00:00:00.000Z	50000
North	3	2023-07-20T00:00:00.000Z	15000
North	4	2023-08-07T00:00:00.000Z	45000
North	5	2023-11-22T00:00:00.000Z	20000
South	2	2023-08-24T00:00:00.000Z	45000
West	1	2023-09-29T00:00:00.000Z	70000

e) Dice

Select Region, Product_Id, On_Date, Sum(Sales_Amount) as Sales_Amount

From Sales_Sample

Where Region in('North', 'South') AND Product_Id IN (1,2) AND On_Date between To_date('2023-08-20','YYYY-MM-DD') And To_Date('2023-10-20','YYYY-MM-DD')

Group By 1,2,3

Order By Region, Product_Id, On_Date, Sales_Amount;

region	product_id	on_date	sales_amount
South	2	2023-08-24T00:00:00.000Z	45000