Day 3 – Subqueries, Set Operators & CTEs

```sql
CREATE DATABASE AirlinesDB;
USE AirlinesDB;

CREATE TABLE air_passenger_profile (
    profile_id      INT IDENTITY(1,1) PRIMARY KEY,
    password        VARCHAR(20) NOT NULL,
    first_name      VARCHAR(20) NOT NULL,
    last_name       VARCHAR(20) NOT NULL,
    address         VARCHAR(100),
    mobile_number   BIGINT,
    email_id        VARCHAR(30)
);

CREATE TABLE air_flight (
    flight_id       INT IDENTITY(1,1) PRIMARY KEY,
    airline_id      VARCHAR(10) NOT NULL,
    airline_name    VARCHAR(30),
    from_location   VARCHAR(20),
    to_location     VARCHAR(20),
    departure_time TIME,
    arrival_time    TIME,
    duration_time   TIME,
    total_seats     INT
);

CREATE TABLE air_flight_details (
    flight_detail_id    INT IDENTITY(1,1) PRIMARY KEY,
    flight_id           INT NOT NULL,
    flight_departure_date DATE,
    price               DECIMAL(8,2),
    available_seats     INT,

    CONSTRAINT FK_flightdetails_flight
        FOREIGN KEY (flight_id)
        REFERENCES air_flight(flight_id)
);

CREATE TABLE air_credit_card_details (
    card_id             INT IDENTITY(1,1) PRIMARY KEY,
    profile_id          INT NOT NULL,
    card_number         BIGINT UNIQUE,
    card_type           VARCHAR(10),
    expiration_month INT,
    expiration_year     INT,

    CONSTRAINT FK_card_passenger
        FOREIGN KEY (profile_id)
        REFERENCES air_passenger_profile(profile_id)
);
```

```sql
CREATE TABLE air_ticket_info (
    ticket_id              INT IDENTITY(1,1) PRIMARY KEY,
    profile_id             INT NOT NULL,
    flight_id              INT NOT NULL,
    flight_departure_date  DATE,
    status                 VARCHAR(15),

    CONSTRAINT FK_ticket_passenger
        FOREIGN KEY (profile_id)
        REFERENCES air_passenger_profile(profile_id),

    CONSTRAINT FK_ticket_flight
        FOREIGN KEY (flight_id)
        REFERENCES air_flight(flight_id)
);

INSERT INTO air_passenger_profile
(password, first_name, last_name, address, mobile_number, email_id)
VALUES
('p1','Arun','Kumar','Chennai',9876543210,'arun@mail.com'),
('p2','Bhavya','Reddy','Hyderabad',9123456780,'bhavya@mail.com'),
('p3','Charan','Singh','Bangalore',9988776655,'charan@mail.com'),
('p4','Divya','Sharma','Delhi',9012345678,'divya@mail.com');

INSERT INTO air_flight
(airline_id, airline_name, from_location, to_location, departure_time,
arrival_time, duration_time, total_seats)
VALUES
('ABC01','ABC
Airlines','Chennai','Hyderabad','08:00','09:30','01:30',180),
('ABC02','ABC
Airlines','Hyderabad','Delhi','10:00','12:30','02:30',160),
('ABC03','ABC
Airlines','Chennai','Bangalore','14:00','15:30','01:30',150),
('IN01','Indigo','Delhi','Mumbai','16:00','18:00','02:00',180);

INSERT INTO air_flight_details
(flight_id, flight_departure_date, price, available_seats)
VALUES
(1,'2025-01-05',3500,120),
(1,'2025-02-10',3600,115),
(1,'2025-04-15',3800,110),
(2,'2025-01-20',5200,100),
(2,'2025-04-12',5400,95),
(3,'2025-02-08',3000,130),
(3,'2025-04-22',3200,125);
```

```
INSERT INTO air_ticket_info
(profile_id, flight_id, flight_departure_date, status)
VALUES
(1,1,'2025-01-05','BOOKED'),
(1,1,'2025-02-10','BOOKED'),
(2,1,'2025-04-15','BOOKED'),
(2,2,'2025-01-20','BOOKED'),
(3,2,'2025-04-12','BOOKED'),
(1,3,'2025-02-08','BOOKED'),
(4,3,'2025-04-22','BOOKED');
```

1. Write a query to display the average monthly ticket cost for each flight in ABC Airlines. The query should display the Flight_Id, From_location, To_Location, Month Name as "Month_Name" and average price as "Average_Price". Display the records sorted in ascending order based on flight id and then by Month Name.

```
select
    f.flight_id,
    f.from_location,
    f.to_location,
    datename(month, fd.flight_departure_date) as month_name,
    avg(fd.price) as average_price
from air_flight f
join air_flight_details fd
    on f.flight_id = fd.flight_id
where f.airline_name = 'ABC Airlines'
group by
    f.flight_id,
    f.from_location,
    f.to_location,
    datename(month, fd.flight_departure_date),
    month(fd.flight_departure_date)
order by
    f.flight_id,
    month(fd.flight_departure_date);
```

Results | Messages

	flight_id	from_location	to_location	month_name	average_price
1	1	Chennai	Hyderabad	January	3500.000000
2	1	Chennai	Hyderabad	February	3600.000000
3	1	Chennai	Hyderabad	April	3800.000000
4	2	Hyderabad	Delhi	January	5200.000000
5	2	Hyderabad	Delhi	April	5400.000000
6	3	Chennai	Bangalore	February	3000.000000
7	3	Chennai	Bangalore	April	3200.000000

2. Write a query to display the customer(s) who has/have booked least number of tickets in ABC Airlines. The Query should display profile_id, customer's first_name, Address and Number of tickets booked as "No_of_Tickets". Display the records sorted in ascending order based on customer's first name.

Query:

```
with ticket_count as (
    select
        p.profile_id,
        p.first_name,
        p.address,
        count(t.ticket_id) as no_of_tickets
    from air_passenger_profile p
    join air_ticket_info t
        on p.profile_id = t.profile_id
    group by
        p.profile_id,
        p.first_name,
        p.address
)
select *
from ticket_count
where no_of_tickets = (select min(no_of_tickets) from ticket_count)
order by first_name;
```

output:

	profile_id	first_name	address	no_of_tickets
1	3	Charan	Bangalore	1
2	4	Divya	Delhi	1

3. Write a query to display the number of flight services between locations in a month. The Query should display From_Location, To_Location, Month as "Month_Name" and number of flight services as "No_of_Services". Hint: The Number of Services can be calculated from the number of scheduled departure dates of a flight. The records should be displayed in ascending order based on From_Location and then by To_Location and then by month name.

Query:

```sql
select
    f.from_location,
    f.to_location,
    datename(month, fd.flight_departure_date) as month_name,
    count(fd.flight_departure_date) as no_of_services
from air_flight f
join air_flight_details fd
    on f.flight_id = fd.flight_id
group by
    f.from_location,
    f.to_location,
    datename(month, fd.flight_departure_date),
    month(fd.flight_departure_date)
order by
    f.from_location,
    f.to_location,
    month(fd.flight_departure_date);
```

output:

	from_location	to_location	month_name	no_of_services
1	Chennai	Bangalore	February	1
2	Chennai	Bangalore	April	1
3	Chennai	Hyderabad	January	1
4	Chennai	Hyderabad	February	1
5	Chennai	Hyderabad	April	1
6	Hyderabad	Delhi	January	1
7	Hyderabad	Delhi	April	1

4. Write a query to display the customer(s) who has/have booked maximum number of tickets in ABC Airlines. The Query should display profile_id, customer's first_name, Address and Number of tickets booked as "No_of_Tickets". Display the records in ascending order based on customer's first name.

Query:

```sql
with ticket_count as (
    select
        p.profile_id,
        p.first_name,
        p.address,
        count(t.ticket_id) as no_of_tickets
    from air_passenger_profile p
    join air_ticket_info t
        on p.profile_id = t.profile_id
    group by
        p.profile_id,
        p.first_name,
        p.address
)
select *
from ticket_count
where no_of_tickets = (select max(no_of_tickets) from ticket_count)
order by first_name;
```

output:

	profile_id	first_name	address	no_of_tickets
1	1	Arun	Chennai	3

5. Write a query to display the number of tickets booked from Chennai to Hyderabad. The Query should display passenger profile_id,first_name,last_name, Flight_Id , Departure_Date and number of tickets booked as "No_of_Tickets". Display the records sorted in ascending order based on profile id and then by flight id and then by departure date.

Query:

```sql
select
    p.profile_id,
    p.first_name,
    p.last_name,
    f.flight_id,
    t.flight_departure_date as departure_date,
    count(t.ticket_id) as no_of_tickets
from air_ticket_info t
join air_passenger_profile p
    on t.profile_id = p.profile_id
join air_flight f
    on t.flight_id = f.flight_id
where
    f.from_location = 'Chennai'
    and f.to_location = 'Hyderabad'
group by
    p.profile_id,
    p.first_name,
    p.last_name,
    f.flight_id,
    t.flight_departure_date
order by
    p.profile_id,
    f.flight_id,
    t.flight_departure_date;
```

Output:

	profile_id	first_name	last_name	flight_id	departure_date	no_of_tickets
1	1	Arun	Kumar	1	2025-01-05	1
2	1	Arun	Kumar	1	2025-02-10	1
3	2	Bhavya	Reddy	1	2025-04-15	1