Assignment -2 DAY-3

Name : Nanda kishor Gudala

```
/* 1) Flights master */
CREATE TABLE Flights (
   Flight_Id      INT IDENTITY(1,1) PRIMARY KEY,
   From_Location  VARCHAR(50) NOT NULL,
   To_Location    VARCHAR(50) NOT NULL
);
GO

/* 2) Scheduled services (each departure date = one service) */
CREATE TABLE FlightSchedules (
   Schedule_Id     INT IDENTITY(1,1) PRIMARY KEY,
   Flight_Id       INT NOT NULL,
   Departure_Date  DATE NOT NULL,
   Ticket_Price    DECIMAL(10,2) NOT NULL,
   CONSTRAINT FK_Schedule_Flight
      FOREIGN KEY (Flight_Id) REFERENCES Flights(Flight_Id)
);
GO

/* 3) Customers / passengers */
CREATE TABLE CustomerProfiles (
   Profile_Id  INT IDENTITY(1,1) PRIMARY KEY,
   First_Name  VARCHAR(50) NOT NULL,
   Last_Name   VARCHAR(50) NOT NULL,
   Address     VARCHAR(100) NULL
);
```

```
GO

/* 4) Bookings (store number of tickets booked) */
CREATE TABLE TicketBookings (
   Booking_Id   INT IDENTITY(1,1) PRIMARY KEY,
   Profile_Id   INT NOT NULL,
   Schedule_Id  INT NOT NULL,
   Tickets      INT NOT NULL CHECK (Tickets > 0),

   CONSTRAINT FK_Booking_Profile
      FOREIGN KEY (Profile_Id) REFERENCES CustomerProfiles(Profile_Id),
   CONSTRAINT FK_Booking_Schedule
      FOREIGN KEY (Schedule_Id) REFERENCES FlightSchedules(Schedule_Id)
);
GO


/* Flights */
INSERT INTO Flights (From_Location, To_Location) VALUES
('Chennai', 'Hyderabad'),
('Chennai', 'Bengaluru'),
('Hyderabad', 'Delhi'),
('Mumbai', 'Chennai'),
('Chennai', 'Kolkata');

/* Flight Schedules (include April dates for Q6) */
-- Flight 1: Chennai -> Hyderabad
INSERT INTO FlightSchedules (Flight_Id, Departure_Date, Ticket_Price) VALUES
(1, '2025-01-10', 4200),
(1, '2025-01-20', 4000),
(1, '2025-02-05', 4300),
```

```
(1, '2025-04-12', 4600),
(1, '2025-04-20', 4550);

-- Flight 2: Chennai -> Bengaluru
INSERT INTO FlightSchedules (Flight_Id, Departure_Date, Ticket_Price) VALUES
(2, '2025-03-01', 3500),
(2, '2025-04-02', 3700);

-- Flight 3: Hyderabad -> Delhi
INSERT INTO FlightSchedules (Flight_Id, Departure_Date, Ticket_Price) VALUES
(3, '2025-04-15', 8000);

-- Flight 4: Mumbai -> Chennai
INSERT INTO FlightSchedules (Flight_Id, Departure_Date, Ticket_Price) VALUES
(4, '2025-02-10', 5000);

-- Flight 5: Chennai -> Kolkata
INSERT INTO FlightSchedules (Flight_Id, Departure_Date, Ticket_Price) VALUES
(5, '2025-01-25', 6000);

/* Customers */
INSERT INTO CustomerProfiles (First_Name, Last_Name, Address) VALUES
('Nanda',   'Kishor',  'KPHB, Hyderabad'),
('Riya',    'Sharma',  'Ameerpet, Hyderabad'),
('Ganesh',  'Kumar',   'Madhapur, Hyderabad'),
('Ruthwik', 'Reddy',   'Gachibowli, Hyderabad'),
('Mani',    'Rao',     'Kukatpally, Hyderabad'),
('Ramesh',  'Harsha',  'Begumpet, Hyderabad');

/* TicketBookings (Schedule_Id values depend on insertion order above)
   Based on our inserts, Schedule_Ids will be 1..10 in order. */
```

```
INSERT INTO TicketBookings (Profile_Id, Schedule_Id, Tickets) VALUES
(1, 1, 2),   -- Nanda, CHN-HYD (Jan 10)
(1, 4, 1),   -- Nanda, CHN-HYD (Apr 12)
(1, 6, 1),   -- Nanda, CHN-BLR (Mar 01)

(2, 2, 1),   -- Riya, CHN-HYD (Jan 20)
(2, 3, 2),   -- Riya, CHN-HYD (Feb 05)

(3, 4, 3),   -- Ganesh, CHN-HYD (Apr 12)

(4, 7, 2),   -- Ruthwik, CHN-BLR (Apr 02)
(4, 10, 1),  -- Ruthwik, CHN-KOL (Jan 25)

(5, 8, 1),   -- Mani, HYD-DEL (Apr 15)
(6, 9, 1);   -- Ramesh, MUM-CHN (Feb 10)
```

Q1) Average monthly ticket cost for each flight

Day3-SQL ServerCommands

```
SELECT
   f.Flight_Id,
   f.From_Location,
   f.To_Location,
   DATENAME(MONTH, s.Departure_Date) AS Month_Name,
   AVG(s.Ticket_Price) AS Average_Price
FROM Flights f
JOIN FlightSchedules s ON s.Flight_Id = f.Flight_Id
GROUP BY
   f.Flight_Id, f.From_Location, f.To_Location,
   DATENAME(MONTH, s.Departure_Date),
   MONTH(s.Departure_Date)
```

```
ORDER BY f.Flight_Id, MONTH(s.Departure_Date);
```

Q2) Customer(s) who booked the least number of tickets

Day3-SQL ServerCommands

```
WITH TicketTotals AS (
   SELECT p.Profile_Id, p.First_Name, p.Address, SUM(b.Tickets) AS No_of_Tickets
   FROM CustomerProfiles p
   JOIN TicketBookings b ON b.Profile_Id = p.Profile_Id
   GROUP BY p.Profile_Id, p.First_Name, p.Address
)
SELECT *
FROM TicketTotals
WHERE No_of_Tickets = (SELECT MIN(No_of_Tickets) FROM TicketTotals)
ORDER BY First_Name;
```

Q3) Number of flight services between locations in a month

Day3-SQL ServerCommands

```
SELECT
   f.From_Location,
   f.To_Location,
   DATENAME(MONTH, s.Departure_Date) AS Month_Name,
   COUNT(*) AS No_of_Services
FROM Flights f
JOIN FlightSchedules s ON s.Flight_Id = f.Flight_Id
GROUP BY
   f.From_Location, f.To_Location,
   DATENAME(MONTH, s.Departure_Date),
   MONTH(s.Departure_Date)
ORDER BY f.From_Location, f.To_Location, MONTH(s.Departure_Date);
```

Q4) Customer(s) who booked the maximum number of tickets

Day3-SQL ServerCommands

```
WITH TicketTotals AS (
   SELECT p.Profile_Id, p.First_Name, p.Address, SUM(b.Tickets) AS No_of_Tickets
```

```
    FROM CustomerProfiles p
    JOIN TicketBookings b ON b.Profile_Id = p.Profile_Id
    GROUP BY p.Profile_Id, p.First_Name, p.Address
)
SELECT *
FROM TicketTotals
WHERE No_of_Tickets = (SELECT MAX(No_of_Tickets) FROM TicketTotals)
ORDER BY First_Name;
```

Q5) Tickets booked from Chennai to Hyderabad (by passenger, flight, date)

Day3-SQL ServerCommands

```
SELECT
    p.Profile_Id,
    p.First_Name,
    p.Last_Name,
    f.Flight_Id,
    s.Departure_Date,
    SUM(b.Tickets) AS No_of_Tickets
FROM TicketBookings b
JOIN CustomerProfiles p ON p.Profile_Id = b.Profile_Id
JOIN FlightSchedules s   ON s.Schedule_Id = b.Schedule_Id
JOIN Flights f           ON f.Flight_Id = s.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, s.Departure_Date
ORDER BY p.Profile_Id, f.Flight_Id, s.Departure_Date;
```

Q6) Flight id, from, to, ticket price where departure month is April

Day3-SQL ServerCommands

```
SELECT
    f.Flight_Id,
    f.From_Location,
    f.To_Location,
    s.Ticket_Price
```

```
FROM Flights f
JOIN FlightSchedules s ON s.Flight_Id = f.Flight_Id
WHERE MONTH(s.Departure_Date) = 4;
```

Q7) Average ticket cost per flight across all scheduled dates

Day3-SQL ServerCommands

```
SELECT
   f.Flight_Id,
   f.From_Location,
   f.To_Location,
   AVG(s.Ticket_Price) AS Price
FROM Flights f
JOIN FlightSchedules s ON s.Flight_Id = f.Flight_Id
GROUP BY f.Flight_Id, f.From_Location, f.To_Location
ORDER BY f.Flight_Id, f.From_Location, f.To_Location;
```

Q8) Unique customers who booked Chennai to Hyderabad (unique irrespective of multiple tickets)

Day3-SQL ServerCommands

```
SELECT DISTINCT
   p.Profile_Id,
   CONCAT(p.First_Name, ',', p.Last_Name) AS Customer_Name,
   p.Address
FROM TicketBookings b
JOIN CustomerProfiles p ON p.Profile_Id = b.Profile_Id
JOIN FlightSchedules s   ON s.Schedule_Id = b.Schedule_Id
JOIN Flights f          ON f.Flight_Id = s.Flight_Id
WHERE f.From_Location = 'Chennai' AND f.To_Location = 'Hyderabad'
ORDER BY p.Profile_Id;
```

Q9) Profile id(s) of passenger(s) who booked maximum number of tickets

Day3-SQL ServerCommands

```
WITH TicketTotals AS (
   SELECT Profile_Id, SUM(Tickets) AS TotalTickets
   FROM TicketBookings
```

```
    GROUP BY Profile_Id
)
SELECT Profile_Id
FROM TicketTotals
WHERE TotalTickets = (SELECT MAX(TotalTickets) FROM TicketTotals)
ORDER BY Profile_Id;
```

Q10) Total tickets booked in each flight (only flights with at least 1 ticket)

Day3-SQL ServerCommands

```
SELECT
    f.Flight_Id,
    f.From_Location,
    f.To_Location,
    SUM(b.Tickets) AS No_of_Tickets
FROM TicketBookings b
JOIN FlightSchedules s ON s.Schedule_Id = b.Schedule_Id
JOIN Flights f         ON f.Flight_Id = s.Flight_Id
GROUP BY f.Flight_Id, f.From_Location, f.To_Location
HAVING SUM(b.Tickets) >= 1
ORDER BY f.Flight_Id;
```