

# TRAVEL BOOKING SYSTEM

---

*PROJECT BY ASHISH UBHALE*

---

## ❖ *TRAVEL BOOKING SYSTEM:*

### PROJECT FOR SQL MODULE

#### 1. DESCRIPTION:

The following database schema is designed to function as a backend storage database for a web application built to manage a travel booking system. By storing information in a relational database, all tasks related to travel booking and

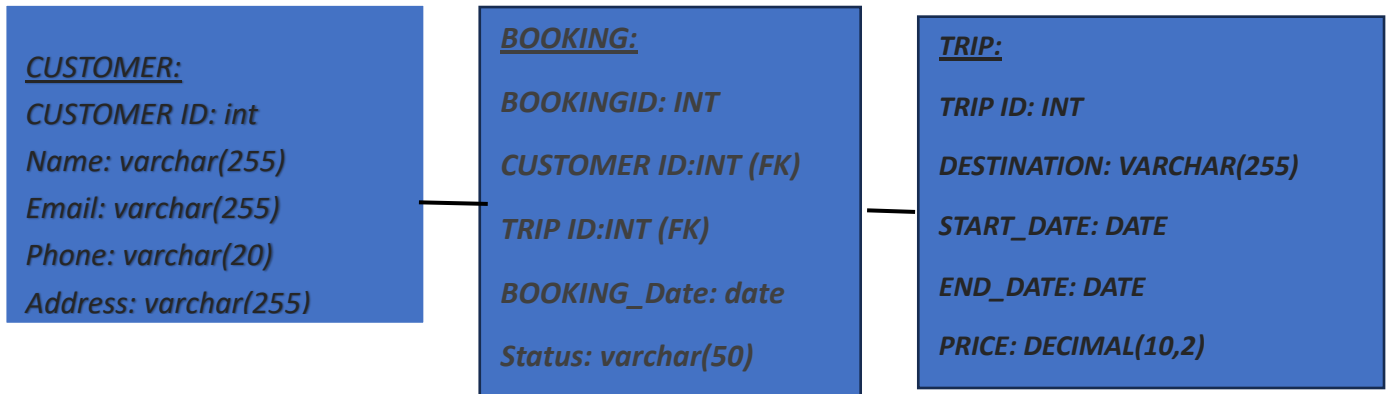
management can be performed efficiently. Some benefits of using this system to store data over traditional methods are outlined below:

1. **Efficient Booking Management:** Updating and modifying travel bookings, including flights, hotels, and rental cars, is streamlined and efficient.
2. **Customer Management:** Maintaining customer details, preferences, and booking history is straightforward and efficient.
3. **Automated Processes:** Calculations such as total booking costs, available inventory, and revenue can be automated by the Database Management System (DBMS), reducing human error.
4. **Data Analysis:** The Relational Database Management System (RDBMS) provides various tools for analyzing available data, enabling informed decisions about inventory management, pricing strategies, and customer preferences.

This database contains the following tables:

1. **Customer:** Stores information about customers, including customerID, name, email, phone, and address.
2. **Booking:** Records customer bookings, including bookingID, customerID (foreign key), tripID (foreign key), booking\_date, and status.
3. **Trip:** Stores details of trips, including tripID, destination, start\_date, end\_date, and price.
4. **Flight:** Contains information about flights, including flightID, airline, departure\_city, arrival\_city, departure\_time, and arrival\_time.
5. **Hotel:** Stores details of hotels, including hotelID, name, location, and price\_per\_night.

## 2.ER DIAGRAM (ENTITY RELATION- DIAGRAM) FOR TRAVEL BOOKING SYSTEM:



### **3.TABLE DESCRIPTIONS:**

#### **1.CUSTOMER:**

<u>FIELD</u>	<u>TYPE</u>	<u>NULL</u>	<u>KEY</u>	<u>DEFAULT</u>	<u>EXTRA</u>
<u>1)Customerid</u>	<b>Int(11)</b>	<b>No</b>	<b>PRI</b>	<b>NULL</b>	
<u>2)Name</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>3)Email</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>4)Phone</u>	<b>Varchar(20)</b>	<b>YES</b>		<b>NULL</b>	
<u>5)Address</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	

#### **2.BOOKING:**

<u>FIELD</u>	<u>TYPE</u>	<u>NULL</u>	<u>KEY</u>	<u>DEFAULT</u>	<u>EXTRA</u>
<u>1)BOOKING ID</u>	<b>Int(11)</b>	<b>NO</b>	<b>PRI</b>	<b>NULL</b>	

<u>2)CUSTOMER ID</u>	<b>Int(11)</b>	<b>YES</b>	<b>MUL</b>	<b>NULL</b>	<b>FK</b>
<u>3)TRIPID</u>	<b>Int(11)</b>	<b>YES</b>	<b>MUL</b>	<b>NULL</b>	<b>FK</b>
<u>4)BOOKING DATE</u>	<b>date</b>	<b>YES</b>		<b>NULL</b>	
<u>5)STATUS</u>	<b>Varchar (50)</b>	<b>YES</b>		<b>NULL</b>	

### **3.TRIP:**

<u>FIELD</u>	<u>TYPE</u>	<u>NULL</u>	<u>KEY</u>	<u>DEFAULT</u>	<u>EXTRA</u>
<u>1)TRIPID</u>	<b>Int(11)</b>	<b>NO</b>	<b>PRI</b>	<b>NULL</b>	
<u>2)DESTINATION</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>3)STARTDATE</u>	<b>date</b>	<b>YES</b>		<b>NULL</b>	
<u>4)ENDDATE</u>	<b>date</b>	<b>YES</b>		<b>NULL</b>	
<u>5)PRICE</u>	<b>Decimal(10,2)</b>	<b>YES</b>		<b>NULL</b>	

### **4.FLIGHT:**

<u>FIELD</u>	<u>TYPE</u>	<u>NULL</u>	<u>KEY</u>	<u>DEFAULT</u>	<u>EXTRA</u>
<u>1)FLIGHTID</u>	<b>Int(11)</b>	<b>NO</b>	<b>PRI</b>	<b>NULL</b>	
<u>2)AIRLINE</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>3)DEPARTURE CITY</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>4)ARRIVAL CITY</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>5)DEPARTURE TIME</u>	<b>time</b>	<b>YES</b>		<b>NULL</b>	
<u>6)ARRIVAL TIME</u>	<b>time</b>	<b>YES</b>		<b>NULL</b>	

### **5.HOTEL:**

<u>FIELD</u>	<u>TYPE</u>	<u>NULL</u>	<u>KEY</u>	<u>DEFAULT</u>	<u>EXTRA</u>
<u>1)HOTELID</u>	<b>Int(11)</b>	<b>NO</b>	<b>PRI</b>	<b>NULL</b>	
<u>2)NAME</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>3)LOCATION</u>	<b>Varchar(255)</b>	<b>YES</b>		<b>NULL</b>	
<u>4)PRICEPER NIGHT</u>	<b>Decimal(10,2)</b>	<b>YES</b>		<b>NULL</b>	

### **4.COMMANDS:**

**-- Create Customer table**

**CREATE TABLE Customer (**

**CustomerID INT AUTO INCREMENT PRIMARY KEY,**

**Name VARCHAR(255),**

**Email VARCHAR(255),**

**Phone VARCHAR(20),**

**Address VARCHAR(255)**

**);**

**-- Create Booking table**

CREATE TABLE Booking (  
BookingID INT AUTO INCREMENT PRIMARY KEY,  
CustomerID INT,  
TripID INT,  
BookingDate DATE,  
Status VARCHAR(50),  
FOREIGN KEY (CustomerID) REFERENCES Customer(CustomerID),  
FOREIGN KEY (TripID) REFERENCES Trip(TripID)  
);

-- Create Trip table

CREATE TABLE Trip (  
TripID INT AUTO INCREMENT PRIMARY KEY,  
Destination VARCHAR(255),  
StartDate DATE,  
EndDate DATE,  
Price DECIMAL(10,2)  
);

-- Create Flight table

CREATE TABLE Flight (  
FlightID INT AUTO INCREMENT PRIMARY KEY,  
Airline VARCHAR(255),  
DepartureCity VARCHAR(255),

ArrivalCity VARCHAR(255),

DepartureTime TIME,

ArrivalTime TIME

);

-- Create Hotel table

CREATE TABLE Hotel (

HotelID INT AUTO INCREMENT PRIMARY KEY,

Name VARCHAR(255),

Location VARCHAR(255),

PricePerNight DECIMAL(10,2)

);

-- Insert sample data into [1\)Customer table](#)

INSERT INTO Customer (Name, Email, Phone, Address) VALUES

('John Doe', 'john@example.com', '1234567890', '123 Main St'),

('Alice Smith', 'alice@example.com', '9876543210', '456 Elm St'),

('Bob Johnson', 'bob@example.com', '5551234567', '789 Oak St'),

('Emily Davis', 'emily@example.com', '3215559876', '321 Pine St'),

('Michael Lee', 'michael@example.com', '8883334444', '555 Cedar St'),

('Samantha Brown', 'samantha@example.com', '7778889999', '777 Maple St'),

('David Wilson', 'david@example.com', '9991112222', '999 Walnut St'),

('Jessica Taylor', 'jessica@example.com', '3332221111', '888 Birch St'),

('Matthew Clark', 'matthew@example.com', '4447778888', '444 Pine St'),

('Jennifer Martinez', 'jennifer@example.com', '6665554444', '666 Elm St'),

('Daniel Anderson', 'daniel@example.com', '2229998888', '222 Oak St'),

('Sarah Hernandez', 'sarah@example.com', '1113334444', '111 Walnut St'),

('Christopher Walker', 'christopher@example.com', '7771112222', '777 Cedar St'),

('Lisa King', 'lisa@example.com', '4446667777', '444 Maple St'),

('Mark Lopez', 'mark@example.com', '6668889999', '666 Birch St');

-- Insert sample data into [2\)Booking table](#)

INSERT INTO Booking (CustomerID, TripID, BookingDate, Status)  
VALUES

(1, 1, '2024-05-01', 'Confirmed'),

(2, 2, '2024-05-05', 'Pending'),

(3, 3, '2024-05-10', 'Confirmed'),

(4, 4, '2024-05-15', 'Confirmed'),

(5, 5, '2024-05-20', 'Pending'),

(6, 6, '2024-05-25', 'Confirmed'),

(7, 7, '2024-05-30', 'Pending'),



(8, 8, '2024-06-01', 'Confirmed'),  
(9, 9, '2024-06-05', 'Confirmed'),  
(10, 10, '2024-06-10', 'Pending'),  
(11, 11, '2024-06-15', 'Confirmed'),  
(12, 12, '2024-06-20', 'Pending'),  
(13, 13, '2024-06-25', 'Confirmed'),  
(14, 14, '2024-06-30', 'Confirmed'),  
(15, 15, '2024-07-01', 'Pending');

-- Insert sample data into [3\)Trip table](#)

INSERT INTO Trip (Destination, StartDate, EndDate, Price) VALUES

('New York', '2024-06-01', '2024-06-07', 1500.00),  
('Los Angeles', '2024-06-10', '2024-06-15', 1800.00),  
('London', '2024-06-20', '2024-06-27', 2000.00),  
('Paris', '2024-07-01', '2024-07-07', 2200.00),  
('Tokyo', '2024-07-10', '2024-07-18', 2500.00),  
('Sydney', '2024-07-20', '2024-07-28', 2700.00),  
('Dubai', '2024-08-01', '2024-08-07', 2300.00),  
('Rome', '2024-08-10', '2024-08-17', 2100.00),  
('Barcelona', '2024-08-20', '2024-08-28', 2400.00),  
('Rio de Janeiro', '2024-09-01', '2024-09-07', 2800.00),  
('Cancun', '2024-09-10', '2024-09-17', 2600.00),  
('Bangkok', '2024-09-20', '2024-09-27', 3000.00),

('Amsterdam', '2024-10-01', '2024-10-07', 1900.00),

('Berlin', '2024-10-10', '2024-10-15', 1700.00),

('Singapore', '2024-10-20', '2024-10-28', 2800.00);

-- Insert sample data into [4\)Flight table](#)

INSERT INTO Flight (Airline, DepartureCity, ArrivalCity,  
DepartureTime, ArrivalTime) VALUES

('Delta', 'New York', 'Los Angeles', '08:00:00', '11:30:00'),

('United', 'Los Angeles', 'London', '12:00:00', '20:30:00'),

('British Airways', 'London', 'Paris', '10:00:00', '12:00:00'),

('Air France', 'Paris', 'Tokyo', '14:00:00', '18:00:00'),

('Japan Airlines', 'Tokyo', 'Sydney', '09:00:00', '17:00:00'),

('Emirates', 'Sydney', 'Dubai', '13:00:00', '19:00:00'),

('Etihad', 'Dubai', 'Rome', '10:00:00', '14:00:00'),

('Alitalia', 'Rome', 'Barcelona', '08:30:00', '11:30:00'),

('Iberia', 'Barcelona', 'Rio de Janeiro', '14:30:00', '21:00:00'),

('LATAM Airlines', 'Rio de Janeiro', 'Cancun', '10:00:00', '16:30:00'),

('Aeromexico', 'Cancun', 'Bangkok', '12:00:00', '23:00:00'),

('Thai Airways', 'Bangkok', 'Amsterdam', '08:00:00', '16:00:00'),

('KLM', 'Amsterdam', 'Berlin', '09:30:00', '11:30:00'),

('Lufthansa', 'Berlin', 'Singapore', '13:00:00', '21:00:00'),

('Singapore Airlines', 'Singapore', 'New York', '22:00:00', '06:30:00');

-- Insert sample data into [5\)Hotel table](#)

INSERT INTO Hotel (Name, Location, PricePerNight) VALUES

('Hilton New York', 'New York', 200.00),

('Marriott Los Angeles', 'Los Angeles', 180.00),

('Intercontinental London', 'London', 220.00),

('Novotel Paris', 'Paris', 180.00),

('Tokyo Marriott Hotel', 'Tokyo', 250.00),

('Sydney Harbour Marriott', 'Sydney', 280.00),

('JW Marriott Marquis Dubai', 'Dubai', 230.00),

('Rome Cavalieri', 'Rome', 210.00),

('W Barcelona', 'Barcelona', 240.00),

('Grand Hyatt Rio de Janeiro', 'Rio de Janeiro', 280.00),

('Marriott Cancun Resort', 'Cancun', 260.00),

('Mandarin Oriental Bangkok', 'Bangkok', 300.00),

('Hotel Okura Amsterdam', 'Amsterdam', 190.00),

('Ritz-Carlton Berlin', 'Berlin', 170.00),

('Marina Bay Sands', 'Singapore', 320.00);

## 5.SUB-QUERIES:

A) WRITE Subquery to Retrieve Customers with No Bookings:

QUERY:

SELECT Name

FROM Customer

WHERE CustomerID NOT IN (SELECT DISTINCT CustomerID FROM Booking);

**RESULT:**

<b>Name:</b>
Michael Lee
Samantha Brown
David Wilson
Jessica Taylor
Matthew Clark
Jennifer Martinez
Daniel Anderson
Sarah Hernandez
Christopher Walker
Lisa King
Mark Lopez

**B) Subquery to Find the Cheapest Trip:**

**QUERY:**

SELECT \*

FROM Trip

WHERE Price = (SELECT MIN(Price) FROM Trip);

**RESULT:**

TripID	Destination	StartDate	EndDate	Price
1	New York	2024-06-01	2024-06-07	1500.00

**C) Subquery to Calculate Average Trip Price:**

**QUERY:**

***SELECT AVG(Price)***

***FROM Trip***

***WHERE StartDate >= CURDATE();***

***RESULT:***

***| AVG(Price) |***

***| 2286.666667 |***

**D) Subquery to Identify Fully Booked Hotels:**

**QUERY:**

***SELECT Name, Location***

***FROM Hotel***

***WHERE HotelID IN (SELECT HotelID FROM Booking GROUP BY  
HotelID HAVING COUNT(\*) >= 10);***

***RESULT:***

***NAME***

***LOCATION***

Hilton New York	New York
Marriott Los Angeles	Los Angeles
Intercontinental London	London
Novotel Paris	Paris
Tokyo Marriott Hotel	Tokyo
Sydney Harbour Marriott	Sydney
JW Marriott Marquis Dubai	Dubai
Rome Cavalieri	Rome
W Barcelona	Barcelona
Grand Hyatt Rio de Janeiro	Rio de Janeiro
Marriott Cancun Resort	Cancun
Mandarin Oriental Bangkok	Bangkok
Hotel Okura Amsterdam	Amsterdam
Ritz-Carlton Berlin	Berlin

Marina Bay Sands	Singapore
------------------	-----------

**E) Subquery to List Customers with Multiple Bookings:**

**QUERY:**

**SELECT Name**

**FROM Customer**

**WHERE CustomerID IN (SELECT CustomerID FROM Booking GROUP BY CustomerID HAVING COUNT(\*) >= 1);**

**RESULT:**

John Doe
Alice Smith
Bob Johnson
Emily Davis
Michael Lee
Samantha Brown
David Wilson
Jessica Taylor
Matthew Clark
Jennifer Martinez
Daniel Anderson
Sarah Hernandez
Christopher Walker
Lisa King
Mark Lopez

**6.JOINS:**

**A) Retrieve the names of customers along with the details of their bookings and the corresponding trip destinations.**

**QUERY:**

**SELECT Customer.Name, Booking.BookingID, Trip.Destination**

**FROM Customer**

**INNER JOIN Booking ON Customer.CustomerID =  
Booking.CustomerID**

**INNER JOIN Trip ON Booking.TripID = Trip.TripID;**

**RESULT:**

NAME	BOOKING ID	DESTINATION
John Doe	31	New York
Alice Smith	32	Los Angeles
Bob Johnson	33	London
Emily Davis	34	Paris
Michael Lee	35	Tokyo
Samantha Brown	36	Sydney
David Wilson	37	Dubai
Jessica Taylor	38	Rome
Matthew Clark	39	Barcelona
Jennifer Martinez	40	Rio de Janeiro
Daniel Anderson	41	Cancun
Sarah Hernandez	42	Bangkok
Christopher Walker	43	Amsterdam
Lisa King	44	Berlin
Mark Lopez	45	Singapore

**B) Retrieve the names of all customers, along with the details of their bookings and the corresponding trip destinations, even if some customers have not made any bookings.**

**QUERY:**

**SELECT Customer.Name, Booking.BookingID, Trip.Destination**

**FROM Customer**

**LEFT JOIN Booking ON Customer.CustomerID =  
Booking.CustomerID**

**LEFT JOIN Trip ON Booking.TripID = Trip.TripID;**

**RESULT:**

NAME	BOOKING ID	DESTINATION
John Doe	31	New York
Alice Smith	32	Los Angeles
Bob Johnson	33	London
Emily Davis	34	Paris
Michael Lee	35	Tokyo
Samantha Brown	36	Sydney
David Wilson	37	Dubai
Jessica Taylor	38	Rome
Matthew Clark	39	Barcelona
Jennifer Martinez	40	Rio de Janeiro
Daniel Anderson	41	Cancun
Sarah Hernandez	42	Bangkok
Christopher Walker	43	Amsterdam
Lisa King	44	Berlin
Mark Lopez	45	Singapore

**C) Retrieve all possible combinations of customers and trips.**

**QUERY:**

**SELECT Customer.Name, Trip.Destination**

**FROM Customer**

**CROSS JOIN Trip;**

**RESULT:**

NAME	DESTINATION
John Doe	New York
Alice Smith	New York
Bob Johnson	New York
Emily Davis	New York
Michael Lee	New York
Samantha Brown	New York
David Wilson	New York



Jessica Taylor	New York
Matthew Clark	New York
Jennifer Martinez	New York
Daniel Anderson	New York
Sarah Hernandez	New York
Christopher Walker	New York
Lisa King	New York
Mark Lopez	New York
John Doe	Los Angeles
Alice Smith	Los Angeles
Bob Johnson	Los Angeles
Emily Davis	Los Angeles
Michael Lee	Los Angeles
Samantha Brown	Los Angeles
David Wilson	Los Angeles
Jessica Taylor	Los Angeles
Matthew Clark	Los Angeles
Jennifer Martinez	Los Angeles
Daniel Anderson	Los Angeles
Sarah Hernandez	Los Angeles
Christopher Walker	Los Angeles
Lisa King	Los Angeles
Mark Lopez	Los Angeles
John Doe	London
Alice Smith	London
Bob Johnson	London
Emily Davis	London
Michael Lee	London
Samantha Brown	London
David Wilson	London
Jessica Taylor	London
Matthew Clark	London
Jennifer Martinez	London
Daniel Anderson	London
Sarah Hernandez	London
Christopher Walker	London
Lisa King	London
Mark Lopez	London
John Doe	Paris
Alice Smith	Paris
Bob Johnson	Paris
Emily Davis	Paris
Michael Lee	Paris
Samantha Brown	Paris
David Wilson	Paris
Jessica Taylor	Paris
Matthew Clark	Paris
Jennifer Martinez	Paris

<b>Daniel Anderson</b>	Paris
<b>Sarah Hernandez</b>	Paris
<b>Christopher Walker</b>	Paris
<b>Lisa King</b>	Paris
<b>Mark Lopez</b>	Paris
<b>John Doe</b>	Tokyo
<b>Alice Smith</b>	Tokyo
<b>Bob Johnson</b>	Tokyo
<b>Emily Davis</b>	Tokyo
<b>Michael Lee</b>	Tokyo
<b>Samantha Brown</b>	Tokyo
<b>David Wilson</b>	Tokyo
<b>Jessica Taylor</b>	Tokyo
<b>Matthew Clark</b>	Tokyo
<b>Jennifer Martinez</b>	Tokyo
<b>Daniel Anderson</b>	Tokyo
<b>Sarah Hernandez</b>	Tokyo
<b>Christopher Walker</b>	Tokyo
<b>Lisa King</b>	Tokyo
<b>Mark Lopez</b>	Tokyo
<b>John Doe</b>	Sydney
<b>Alice Smith</b>	Sydney
<b>Bob Johnson</b>	Sydney
<b>Emily Davis</b>	Sydney
<b>Michael Lee</b>	Sydney
<b>Samantha Brown</b>	Sydney
<b>David Wilson</b>	Sydney
<b>Jessica Taylor</b>	Sydney
<b>Matthew Clark</b>	Sydney
<b>Jennifer Martinez</b>	Sydney
<b>Daniel Anderson</b>	Sydney
<b>Sarah Hernandez</b>	Sydney
<b>Christopher Walker</b>	Sydney
<b>Lisa King</b>	Sydney
<b>Mark Lopez</b>	Sydney
<b>John Doe</b>	Dubai
<b>Alice Smith</b>	Dubai
<b>Bob Johnson</b>	Dubai
<b>Emily Davis</b>	Dubai
<b>Michael Lee</b>	Dubai
<b>Samantha Brown</b>	Dubai
<b>David Wilson</b>	Dubai
<b>Jessica Taylor</b>	Dubai
<b>Matthew Clark</b>	Dubai
<b>Jennifer Martinez</b>	Dubai
<b>Daniel Anderson</b>	Dubai
<b>Sarah Hernandez</b>	Dubai
<b>Christopher Walker</b>	Dubai

<b>Lisa King</b>	Dubai
<b>Mark Lopez</b>	Dubai
<b>John Doe</b>	Rome
<b>Alice Smith</b>	Rome
<b>Bob Johnson</b>	Rome
<b>Emily Davis</b>	Rome
<b>Michael Lee</b>	Rome
<b>Samantha Brown</b>	Rome
<b>David Wilson</b>	Rome
<b>Jessica Taylor</b>	Rome
<b>Matthew Clark</b>	Rome
<b>Jennifer Martinez</b>	Rome
<b>Daniel Anderson</b>	Rome
<b>Sarah Hernandez</b>	Rome
<b>Christopher Walker</b>	Rome
<b>Lisa King</b>	Rome
<b>Mark Lopez</b>	Rome
<b>John Doe</b>	Barcelona
<b>Alice Smith</b>	Barcelona
<b>Bob Johnson</b>	Barcelona
<b>Emily Davis</b>	Barcelona
<b>Michael Lee</b>	Barcelona
<b>Samantha Brown</b>	Barcelona
<b>David Wilson</b>	Barcelona
<b>Jessica Taylor</b>	Barcelona
<b>Matthew Clark</b>	Barcelona
<b>Jennifer Martinez</b>	Barcelona
<b>Daniel Anderson</b>	Barcelona
<b>Sarah Hernandez</b>	Barcelona
<b>Christopher Walker</b>	Barcelona
<b>Lisa King</b>	Barcelona
<b>Mark Lopez</b>	Barcelona
<b>John Doe</b>	Rio de Janeiro
<b>Alice Smith</b>	Rio de Janeiro
<b>Bob Johnson</b>	Rio de Janeiro
<b>Emily Davis</b>	Rio de Janeiro
<b>Michael Lee</b>	Rio de Janeiro
<b>Samantha Brown</b>	Rio de Janeiro
<b>David Wilson</b>	Rio de Janeiro
<b>Jessica Taylor</b>	Rio de Janeiro
<b>Matthew Clark</b>	Rio de Janeiro
<b>Jennifer Martinez</b>	Rio de Janeiro
<b>Daniel Anderson</b>	Rio de Janeiro
<b>Sarah Hernandez</b>	Rio de Janeiro
<b>Christopher Walker</b>	Rio de Janeiro
<b>Lisa King</b>	Rio de Janeiro
<b>Mark Lopez</b>	Rio de Janeiro
<b>John Doe</b>	Cancun

Alice Smith	Cancun
Bob Johnson	Cancun
Emily Davis	Cancun
Michael Lee	Cancun
Samantha Brown	Cancun
David Wilson	Cancun
Jessica Taylor	Cancun
Matthew Clark	Cancun
Jennifer Martinez	Cancun
Daniel Anderson	Cancun
Sarah Hernandez	Cancun
Christopher Walker	Cancun
Lisa King	Cancun
Mark Lopez	Cancun
John Doe	Bangkok
Alice Smith	Bangkok
Bob Johnson	Bangkok
Emily Davis	Bangkok
Michael Lee	Bangkok
Samantha Brown	Bangkok
David Wilson	Bangkok
Jessica Taylor	Bangkok
Matthew Clark	Bangkok
Jennifer Martinez	Bangkok
Daniel Anderson	Bangkok
Sarah Hernandez	Bangkok
Christopher Walker	Bangkok
Lisa King	Bangkok
Mark Lopez	Bangkok
John Doe	Amsterdam
Alice Smith	Amsterdam
Bob Johnson	Amsterdam
Emily Davis	Amsterdam
Michael Lee	Amsterdam
Samantha Brown	Amsterdam
David Wilson	Amsterdam
Jessica Taylor	Amsterdam
Matthew Clark	Amsterdam
Jennifer Martinez	Amsterdam
Daniel Anderson	Amsterdam
Sarah Hernandez	Amsterdam
Christopher Walker	Amsterdam
Lisa King	Amsterdam
Mark Lopez	Amsterdam
John Doe	Berlin
Alice Smith	Berlin
Bob Johnson	Berlin
Emily Davis	Berlin

Michael Lee	Berlin
Samantha Brown	Berlin
David Wilson	Berlin
Jessica Taylor	Berlin
Matthew Clark	Berlin
Jennifer Martinez	Berlin
Daniel Anderson	Berlin
Sarah Hernandez	Berlin
Christopher Walker	Berlin
Lisa King	Berlin
Mark Lopez	Berlin
John Doe	Singapore
Alice Smith	Singapore
Bob Johnson	Singapore
Emily Davis	Singapore
Michael Lee	Singapore
Samantha Brown	Singapore
David Wilson	Singapore
Jessica Taylor	Singapore
Matthew Clark	Singapore
Jennifer Martinez	Singapore
Daniel Anderson	Singapore
Sarah Hernandez	Singapore
Christopher Walker	Singapore
Lisa King	Singapore
Mark Lopez	Singapore

**D) Calculate Total Revenue Generated by Each Customer:**

**QUERY:**

**SELECT Customer.Name, SUM(Trip.Price) AS TotalRevenue**

**FROM Customer**

**JOIN Booking ON Customer.CustomerID = Booking.CustomerID**

**JOIN Trip ON Booking.TripID = Trip.TripID**

**GROUP BY Customer.CustomerID;**

**RESULT:**

NAME	TOTAL REVENUE
John Doe	1500.00

Alice Smith	1800.00
Bob Johnson	2000.00
Emily Davis	2200.00
Michael Lee	2500.00
Samantha Brown	2700.00
David Wilson	2300.00
Jessica Taylor	2100.00
Matthew Clark	2400.00
Jennifer Martinez	2800.00
Daniel Anderson	2600.00
Sarah Hernandez	3000.00
Christopher Walker	1900.00
Lisa King	1700.00
Mark Lopez	2800.00

## **7.GENERAL QUERIES:**

**A) Find the Most Expensive Trip:**

**QUERY:**

**SELECT \***

**FROM Trip**

**WHERE Price = (SELECT MAX(Price) FROM Trip);**

**RESULT:**

TRIPID	DESTINATION	STARTDATE	ENDDATE	PRICE
12	Bangkok	2024-09-20	2024-09-27	3000.00

**B) List Customers Who Have Booked a Trip to Paris:**

**QUERY:**

**SELECT DISTINCT Customer.Name**

**FROM Customer**

**JOIN Booking ON Customer.CustomerID = Booking.CustomerID**

**JOIN Trip ON Booking.TripID = Trip.TripID**

**WHERE Trip.Destination = 'Paris';**

**RESULT:**

NAME
Emily Davis

**C) Retrieve the Earliest Booking Date for Each Customer:**

**QUERY:**

**SELECT c.Name, MIN(b.BookingDate) AS EarliestBookingDate**

**FROM Customer c**

**JOIN Booking b ON c.CustomerID = b.CustomerID**

**GROUP BY c.CustomerID;**

**RESULT:**

NAME	EARLIESTBOOKINGDATE
John Doe	2024-05-01
Alice Smith	2024-05-05
Bob Johnson	2024-05-10
Emily Davis	2024-05-15
Michael Lee	2024-05-20
Samantha Brown	2024-05-25
David Wilson	2024-05-30
Jessica Taylor	2024-06-01
Matthew Clark	2024-06-05

Jennifer Martinez	2024-06-10
Daniel Anderson	2024-06-15
Sarah Hernandez	2024-06-20
Christopher Walker	2024-06-25
Lisa King	2024-06-30
Mark Lopez	2024-07-01

**D) Find Customers Who Have Booked Trips Within a Specific Date Range: Suppose from between June 2024 and September 2024**

**QUERY:**

**SELECT DISTINCT c.Name**

**FROM Customer c**

**JOIN Booking b ON c.CustomerID = b.CustomerID**

**JOIN Trip t ON b.TripID = t.TripID**

**WHERE t.StartDate BETWEEN '2024-06-01' AND '2024-09-01';**

**RESULT:**

NAME
John Doe
Alice Smith
Bob Johnson
Emily Davis
Michael Lee
Samantha Brown
David Wilson
Jessica Taylor
Matthew Clark
Jennifer Martinez

***E)Get the top 5 customers who have spent the most on bookings.***

**QUERY:**



***SELECT c.Name, SUM(t.Price) AS TotalAmountSpent***  
***FROM Customer c***  
***JOIN Booking b ON c.CustomerID = b.CustomerID***  
***JOIN Trip t ON b.TripID = t.TripID***  
***GROUP BY c.CustomerID***  
***ORDER BY TotalAmountSpent DESC***  
***LIMIT 5;***

***RESULT:***

NAME	TOTALAMOUNTSPENT
Sarah Hernandez	3000.00
Jennifer Martinez	2800.00
Mark Lopez	2800.00
Samantha Brown	2700.00
Daniel Anderson	2600.00

***F)Identify customers who have booked trips with a price higher than the average price of all trips.***

***QUERY:***

***SELECT DISTINCT c.Name***  
***FROM Customer c***  
***JOIN Booking b ON c.CustomerID = b.CustomerID***  
***JOIN Trip t ON b.TripID = t.TripID***  
***WHERE t.Price > (SELECT AVG(Price) FROM Trip);***

***RESULT:***

NAME
Michael Lee
Samantha Brown
David Wilson

Matthew Clark
Jennifer Martinez
Daniel Anderson
Sarah Hernandez
Mark Lopez

**F) Find Hotels with Price Per Night Less Than a Certain Amount:(LESS THAN 200)**

**QUERY:**

```
SELECT *
FROM Hotel
WHERE PricePerNight < 200;
```

**RESULT:**

HOTELID	NAME	LOCATION	PRICEPERNIGHT
2	Marriott Los Angeles	Los Angeles	180.00
4	Novotel Paris	Paris	180.00
13	Hotel Okura Amsterdam	Amsterdam	190.00
14	Ritz-Carlton Berlin	Berlin	170.00

**G) Get the details of flights departing after a certain time (e.g., after 12:00 PM).**

**QUERY:**

```
SELECT *
FROM Flight
WHERE DepartureTime > '12:00';
```

**RESULT:**

FlightID	Airline	DepartureCity	ArrivalCity	DepartureTime	ArrivalTime
4	Air France	Paris	Tokyo	14:00:00	18:00:00

6	Emirates	Sydney	Dubai	13:00:00	19:00:00
9	Iberia	Barcelona	Rio de Janeiro	14:30:00	21:00:00
14	Lufthansa	Berlin	Singapore	13:00:00	21:00:00
15	Singapore Airlines	Singapore	New York	22:00:00	06:30:00

**H) Calculate the total number of flights:**

**QUERY:**

**SELECT COUNT(\*) AS TotalFlights**

**FROM Flight;**

**RESULT:**

<b>TotalFlights</b>
<b>15</b>

***As we conclude, I want to express my gratitude for the opportunity to share findings with you. I hope that you found the presentation insightful and valuable.***