DDL SCRIPT

```
create table Customer(
      CustId INT PRIMARY KEY,
      Password VARCHAR(20) NOT NULL,
      Fname VARCHAR(20) NOT NULL,
      Mname VARCHAR(20),
      Lname VARCHAR(20) NOT NULL,
      DOB DATE
);

create table Customer_Contact(
      Contact BIGINT,
      CustID INT,
      PRIMARY KEY (Contact, CustID),
      FOREIGN KEY (CustID) REFERENCES Customer(CustID)
      ON DELETE CASCADE
);

create table Customer_Email(
      Email VARCHAR(60),
      CustID INT,
      PRIMARY KEY (Email, CustID),
      FOREIGN KEY (CustID) REFERENCES Customer(CustID)
      ON DELETE CASCADE
);

create table Emergency_Contact(
      Contact_Name VARCHAR(40),
      CustID INT,
      Address VARCHAR(100),
      Relation VARCHAR(20),
      Contact_Number BIGINT NOT NULL,
      PRIMARY KEY (Contact_Name, CustID),
      FOREIGN KEY (CustID) REFERENCES Customer(CustID)
      ON DELETE CASCADE
);
```

```sql
create table Destination(
        DID INT PRIMARY KEY,
        Dname VARCHAR(30),
        Country VARCHAR(30),
        Description VARCHAR(100),
        Best_month_to_visit VARCHAR(10)
);

create table Popular_Attractions(
        Popular_Attractions VARCHAR(100),
        DID INT,
        PRIMARY KEY (Popular_Attractions, DID),
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE
);

create table Non_logged_in_user(
        User_Name VARCHAR(30),
        DID INT,
        PRIMARY KEY (User_Name, DID),
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE
);

create table Admin(
        AdminID INT PRIMARY KEY,
        Type VARCHAR(20) NOT NULL,
        Fname VARCHAR(20) NOT NULL,
        Mname VARCHAR(20),
        Lname VARCHAR(20) NOT NULL
);

create table Booking(
        BookingID BIGINT PRIMARY KEY,
        Booking_Date DATE NOT NULL,
```

```sql
        Booking_Status VARCHAR(20) NOT NULL,
        Booking_Type VARCHAR(30) NOT NULL,
        Total_Cost INT NOT NULL,
        No_of_Guests INT NOT NULL,
        Checkin_Date  DATE NOT NULL,
        Checkout_Date DATE NOT NULL,
        CustID INT,
        DID INT,
        AdminID INT,
        FOREIGN KEY (CustID) REFERENCES Customer(CustID)
        ON DELETE CASCADE,
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE,
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table Payment(
        TransactionID BIGINT PRIMARY KEY,
        Method VARCHAR(20) NOT NULL,
        Amount INT NOT NULL,
        Transaction_Date DATE NOT NULL,
        Transaction_Status VARCHAR(20) NOT NULL,
        CustID INT,
        DID INT,
        AdminID INT,
        BookingID INT,
        FOREIGN KEY (CustID) REFERENCES Customer(CustID)
        ON DELETE CASCADE,
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE,
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE,
        FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)
        ON DELETE CASCADE
);
```

```sql
create table Review(
        R_Type VARCHAR(20),
        Rating FLOAT NOT NULL,
        CustID INT,
        BookingID BIGINT,
        Text VARCHAR(100),
        R_Date DATE NOT NULL,
        PRIMARY KEY (R_Type, CustID, BookingID),
        FOREIGN KEY (CustID) REFERENCES Customer(CustID)
        ON DELETE CASCADE,
        FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)
        ON DELETE CASCADE
);

create table AdminEmail(
        Email VARCHAR(60),
        AdminID INT,
        PRIMARY KEY (AdminID,Email),
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table AdminContact(
        Contact BIGINT,
        AdminID INT,
        PRIMARY KEY (AdminID,Contact),
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table EventsAndFests(
        Event_Name VARCHAR(100),
        Event_Date DATE,
        Description VARCHAR(100),
        DID INT,
        PRIMARY KEY (Event_Name,Event_Date,DID),
        FOREIGN KEY (DID) REFERENCES Destination(DID)
```

```sql
            ON DELETE CASCADE
);

create table Transportation(
        Transportation_ID INT PRIMARY KEY,
        Price INT NOT NULL,
        Capacity INT NOT NULL,
        Trans_Type VARCHAR(20) NOT NULL,
        Trans_Name VARCHAR(40) NOT NULL,
        DID INT,
        AdminID INT,
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE,
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table Accommodation(
        AccommodationID INT PRIMARY KEY,
        Aname VARCHAR(40) NOT NULL,
        Atype VARCHAR(20) NOT NULL,
        Price_per_night INT NOT NULL,
        Availability INT NOT NULL,
        Docs_Required VARCHAR(100),
        DID INT,
        AdminID INT,
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE,
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table Activities(
        AcivityID INT PRIMARY KEY,
        Activity_type VARCHAR(20) NOT NULL,
        Activity_name VARCHAR(100) NOT NULL,
        Price INT NOT NULL,
```

```sql
        Availability INT,
        DID INT,
        AdminID INT,
        FOREIGN KEY (DID) REFERENCES Destination(DID)
        ON DELETE CASCADE,
        FOREIGN KEY (AdminID) REFERENCES Admin(AdminID)
        ON DELETE CASCADE
);

create table Refund(
        Refund_Status VARCHAR(20),
        BookingID BIGINT,
        TransactionID BIGINT,
        PRIMARY KEY (Refund_Status, BookingID, TransactionID),
        FOREIGN KEY (BookingID) REFERENCES Booking(BookingID)
        ON DELETE CASCADE,
        FOREIGN KEY (TransactionID) REFERENCES Payment(TransactionID)
        ON DELETE CASCADE
);
```