Customer(<u>CustomerID</u>:Integer, Name:varchar(30), PhoneNumber:Integer, Email: varchar(50), **<u>TableID</u>**:Integer)

Candidate Keys: (CustomerID, TableID)

Functional Dependencies:

CustomerID, TableID ==> Name, PhoneNumber, Email

SQL DDL: CREATE TABLE Customer (
            CustomerID INTEGER PRIMARY KEY,
            Name VARCHAR(30) NOT NULL,

            PhoneNumber INTEGER NOT NULL,

            Email varchar(50),

            FOREIGN KEY (TableID) REFERENCES RestaurantTable

        );

RestaurantTable(<u>TableID</u>:Integer, Row:Integer, Spot:varchar(30))

Candidate Keys: (TableID)

Functional Dependencies:

TableID ==> Row, Spot, Reserves_Date, Reserves_Time, Reserves_Period

SQL DDL: CREATE TABLE RestaurantTable (
            TableID INTEGER PRIMARY KEY,
            Row INTEGER,

            Spot varchar(30)

            );

RoomLocation(<u>LocationID</u>:Integer, Name:varchar(30), Description:varchar(200))

Candidate Keys: (LocationID)

Functional Dependencies:

LocationID ==> Name, Description

SQL DDL: CREATE TABLE RoomLocation (
    LocationID INTEGER PRIMARY KEY,
    Name varchar(30),

    Description varchar(200)

);

EventOcation(OcationID:Integer, Description:varchar(200))

Candidate Keys: (OcationID)

Functional Dependencies:

OcationID ==> OcationID, Description

SQL DDL: CREATE TABLE EventOcation (
    OcationID INTEGER PRIMARY KEY,
    Description varchar(200)

);

Reserves(Date:Date, Time: Time, Period: Integer, **TableID**: Integer)

Candidate Keys: (TableID)

Functional Dependencies:

TableID ==> Date, Time, Period

SQL DDL: CREATE TABLE Reserves (
    TableID INTEGER PRIMARY KEY,
    Date Date,
    Time Time,
    Period Integer,
    PRIMARY KEY (TableID),
    FOREIGN KEY (TableID) REFERENCES RestaurantTable

    );

GiveInstruction(SpecialInstruction:varchar(200), **OcationID**:Integer)

Candidate Keys: (OcationID)

Functional Dependencies:

OcationID ==> SpecialInstruction

SQL DDL: CREATE TABLE Reserves (
      TableID INTEGER PRIMARY KEY,
      SpecialInstruction varchar(200),
      PRIMARY KEY (OcationID),
      FOREIGN KEY (OcationID) REFERENCES EventOcation

);