Amusement Park Management System

ID : 17.02.05.035
ID : 17.02.05.050
ID : 17.02.05.055

We have ensured up to 3rd normalization in the tables of our database. 13 tables of our database are mentioned below with necessary information.

1. **Visitor_Info:**

visitor_id	visitor_name	visitor_phone	visitor_gender	visitor_age

In this table we are storing the information of the visitors who are visiting the park. Here, visitor_id is our primary key.

```
CREATE TABLE Visitor_Info(
        visitor_id int  IDENTITY(1,1) PRIMARY KEY,
        visitor_name varchar(50) NOT NULL,
        visitor_phone int NOT NULL,
        visitor_gender varchar(10) NOT NULL CHECK
                (visitor_gender  IN  ('Male','Female','Other')),
        visitor_age int NOT NULL
)
```

2. **Payout_Amount:**

staff_designation	staff_salary

From this table, we can fetch the information about the salary amount of a specific designation of the staff. The primary key in this table is staff_designation.

```
CREATE TABLE Payout_Amount(
        staff_designation varchar(20) PRIMARY KEY,
        staff_salary int NOT NULL
)
```

3. **Staff_Info**

staff_id	staff_name	staff_phone	staff_gender	staff_age	staff_address	staff_designation	region_no

In this table we are storing the information about the staff working at the park. Here, staff_id is our primary key, staff_designation is a foreign key from the table Payout_Amount.

```
CREATE TABLE Staff_info(
        staff_id int IDENTITY(1,1) PRIMARY KEY,
        staff_name varchar(50) NOT NULL,
        staff_phone int NOT NULL,
        staff_gender varchar(10) NOT NULL
              CHECK(staff_gender IN('Male','Female','Other')),
        staff_age int NOT NULL,
        staff_address varchar(50) NOT NULL,
        staff_designation varchar(20) NOT NULL
              FOREIGN KEY REFERENCES  Payout_Amount  (staff_designation),
        staff_salary int NOT NULL,
        staff_hiredate varchar(20) NOT NULL
        region_no varchar(10) NOT NULL,
)
```

4. **Parking_Info**

license_no	vehicle_type	time_of_parking	garage_section_no

In this table we are storing the information about the vehicles that are parked in the parking area of the park. The primary key is license_no.

```
CREATE TABLE Parking_Info(
        license_no varchar(10) PRIMARY KEY ,
        vehicle_type varchar(20) NOT NULL,
        time_of_parking varchar(20) NOT NULL,
        garage_section_no varchar(10) NOT NULL
)
```

5. **Visitor_Vehicle:**

visitor_id	license_no

The primary is visitor_id and license_no is foreign from Parking_Info.

```
CREATE TABLE Visitor_Vehicle(
        visitor_id int PRIMARY KEY ,
        license_no varchar(10) NOT NULL
                FOREIGN KEY REFERENCES Parking_Info(license_no),
)
```

6. **Ride_Info**

ride_id	ride_name	region_no	ticket_price	age_limit

The information stored here is about the individual rides. Primary key is ride_id.

```
CREATE TABLE Ride_info(
        ride_id int IDENTITY(1,1) PRIMARY KEY,
        ride_name varchar(20) NOT NULL,
        region_no int NOT NULL,
        ticket_price int NOT NULL,
        age_limit int NOT NULL
)
```

7. **Food_Info**

food_id	food_name	food_price

The information about the foods that are served in the canteen area of the park can be found here. The primary key in this table is food_id.

```
CREATE TABLE Food_Info(
        food_id int IDENTITY(1,1) PRIMARY KEY,
        food_name varchar(30) NOT NULL,
        food_price int NOT NULL,
)
```

8. **Food_Counter**

order_sl_no	visitor_id	food_id	no_of_food	total_food_price	food_sold_date

The food order given by any visitor is stored in this table. The primary key is order_sl_no. The foreign keys here are visitor_id from Visitor_Info, food_id from Food_Info.

```
CREATE TABLE Food_Counter(
        order_sl_no INT IDENTITY(1,1) PRIMARY KEY,
        visitor_id int NOT NULL
                FOREIGN KEY REFERENCES Visitor_Info(visitor_id),
        food_id int NOT NULL
                FOREIGN KEY REFERENCES Food_Info(food_id),
        no_of_food int NOT NULL,
        total_food_price int NOT NULL,
        food_sold_date varchar(20) NOT NULL
)
```

9. **Entry_Ticket_Counter**

eticket_sl_no	visitor_id	no_of_etickets	total_eticket_price	eticket_sold_time	eticket_type

In this table we are storing the information about the tickets the visitor is buying for entering the park. The primary key in this table is eticket_sl_no. The foreign key visitor_id is from Visitor_Info. The foreign key eticket_type is from Entry_Ticket_type.

```
CREATE TABLE Entry_Ticket_counter (
        eticket_sl_no INT IDENTITY(1,1) PRIMARY KEY,
        visitor_id int NOT NULL
                FOREIGN KEY REFERENCES Visitor_Info(visitor_id),
        no_of_etickets int NOT NULL,
        total_eticket_price int NOT NULL,
        eticket_sold_time varchar(20) NOT NULL
        eticket_type varchar(20) NOT NULL
                FOREIGN KEY REFERENCES Entry_Ticket_type(eticket_type),

)
```

10. **Ride_Ticket_Counter**

rticket_sl_no	visitor_id	ride_id	no_of_rtickets	total_rticket_price	rticket_sold_time

In this table we are storing the information about the tickets the visitor is buying for riding any ride. The primary key in this table is eticket_sl_no. The foreign keys are visitor_id from Visitor_Info, ride_id from Ride_Info.

```
CREATE TABLE Ride_Ticket_counter (
        rticket_sl_no int IDENTITY(1,1) PRIMARY KEY,
        visitor_id int NOT NULL
              FOREIGN KEY REFERENCES Visitor_Info(visitor_id),
        ride_id int  NOT NULL
              FOREIGN KEY REFERENCES Ride_info(ride_id),
        no_of_etickets int NOT NULL,
        total_eticket_price int NOT NULL,
        eticket_sold_time varchar(20) NOT NULL
)
```

11. **Feedback**

feedback_sl_no	visitor_id	feedback_time	visitor_rating	comments

We are storing the feedback of the visitors in this table. The primary key is feedback_sl_no. Foreign key is visitor_id from Visitor_Info.

```
CREATE TABLE Feedback(
        feedback_sl_no int IDENTITY(1,1) PRIMARY KEY,
        visitor_id int NOT NULL
               FOREIGN KEY REFERENCES Visitor_Info(visitor_id),
        feedback_time varchar(20) NOT NULL,
        visitor_rating int NOT NULL,
        comments varchar(100)
)
```

12. **Service_region**

service_name	region_no

From this table we can know the service provided by a specific region. Primary key here is service_name.

```
CREATE TABLE Service_Region(
        service_name varchar(20) PRIMARY KEY ,
        region_no int NOT NULL
)
```

13. **Entry_Ticket_type:**

eticket_type	eticket_price

In this table we are storing the information about the ticket's price according to different offers and packages.

```
CREATE TABLE Entry_Ticket_type (
        eticket_type varchar(20) PRIMARY KEY NOT NULL,
        eticket_price int NOT NULL
)
```

The entity relationship diagram of our project is given below :