Assignment 1

Problem 1

Step 1: Websites and Information gathered

Websites	Information Gathered
https://www.tripadvisor.ca/Hotel_Review-g1182254-d469438-Reviews-Harbourview_Inn-Smith_s_Cove_Digby_Southwest_Nova_Scotia_Nova_Scotia.html And https://www.theharbourviewinn.com/	Here the website trip advisor is a site which contains information and reviews on all the Bed and breakfast hotels. The new thing I noticed in this website review and rating system which can be considered as an entity. The second link is the actual link to the hotel website which contains explore option which can be used as an entity which provides details of how many adventures the hotel offers to its customers
https://www.thecoopersinn.com/	As I searched through the site I saw the entities such as room, breakfast and explore. The website also had available check in and check out dates which can be attributes of Room entity.
https://www.tripadvisor.ca/Hotel_Review-g499213-d579039-Reviews-Auberge_Gisele_s_Country_Inn-Baddeck_Cape_Breton_Island_Nova_Scotia.html	This website has dining option which gave me an idea of Reservation item where customers can make reservations of food items they want for their breakfast
https://bbcapebretonisland.com/index.html	This website also contained different types of rooms where rooms can be considered as an entity. This also contains a reservation for the room.
https://www.booking.com/hotel/ca/star-of-the-sea-b-amp-b.en-gb.html?aid=356980&label=gog235jc-1BCAMYIAQoJ0IHaGFsaWZheEgzWANoJ4gBAZgBCbgBF8gBDNgBAegBAYgCAagCA7gC59vtowbAAgHSAiRjNzgzN2UwMy0xNzQ4LTQ0YTctOTRmZi0xNGZhM2FkZmFkOGTYAgXgAgE&sid=4a42512c59d528b0cdcee35acec81983	This hotel website contains an outdoor swimming pool. As the client is not offering this facility, the coupon offered can be considered as an entity which can contain the facility offered by the coupon along with its description.
http://castlerockcountryinn.com/	One unique thing about the website is the style in which reviews are recorded. First attribute can be a review and the next can be a description saying why the customer is giving the review.
https://www.antigonishvictorianinn.ca/	This website is not offering breakfast as it is facing some issues, but this will cause a reduction in price or

	as a result of coupon offering which can be used as an entity for the client's database.
https://www.comefromawayinn.com/	This website also has a “things to do” option which describes the extra activities offered to their customers which can be used as an entity. This is also similar to the explore option given in the other’s website.

Step 2: Assumption on entities consideration

Staff: The staff entity represents the "Hello12 Management team" and their bed and breakfast establishment. It is a strong entity as it is the central focus of the system and holds essential information about how the workflow of the hotel goes.

Customer: The Customer entity represents the individuals or groups who book and stay at the bed and breakfast. It is a strong entity as it plays a significant role in the system. Customers have their own attributes, such as customer ID, name, contact information, and reservation history, and they can exist independently.

Room: The Room entity represents the different types of rooms available at the bed and breakfast. It is a strong entity as it is a core component of the system. Rooms have their own attributes, such as room number, room type, capacity, and price per night, and they can exist independently.

Explore: The Explore entity represents the additional activities or amenities offered by the hotel to its customers. it has attributes such as activity name, description, and location. Customers can explore these activities during their stay.

Breakfast: The Breakfast entity represents the various types of breakfast options offered by the bed and breakfast. It is a strong entity as it is an essential aspect of the business. Breakfast options have their own attributes, such as breakfast ID, name, description, and price, and they can exist independently.

Coupon: The Coupon entity represents the coupons provided by the bed and breakfast for accessing facilities outside, such as swimming pools or gyms. It is a strong entity as it has its own attributes, such as coupon code, discount percentage, and validity period. Coupons can be issued and managed independently.

Review: The Review entity represents the reviews or feedback provided by customers about their experience at the bed and breakfast. Reviews have attributes such as review ID, rating, comments, and the associated customer and room.

Reservation: The Reservation entity represents the booking or reservation made by a customer for a specific room. It is a strong entity as it holds information about the duration and details of the stay. Reservations have their own attributes, such as reservation ID, check-in date, check-out date, and the associated customer and room, and they can exist independently.

Reason why I considered review and explore as a weak entity.

Review and Explore depends on the existence of strong entities. For instance, a Review is connected to a customer and a particular room, whereas an Explore activity is connected to weather and presence of staff. For these weak entities to be uniquely identified and associated, their matching strong entities must be present. They are dependent on a strong entity for their existence and identification because they are unable to exist on their own.

Step 3: ERD using Chen Model

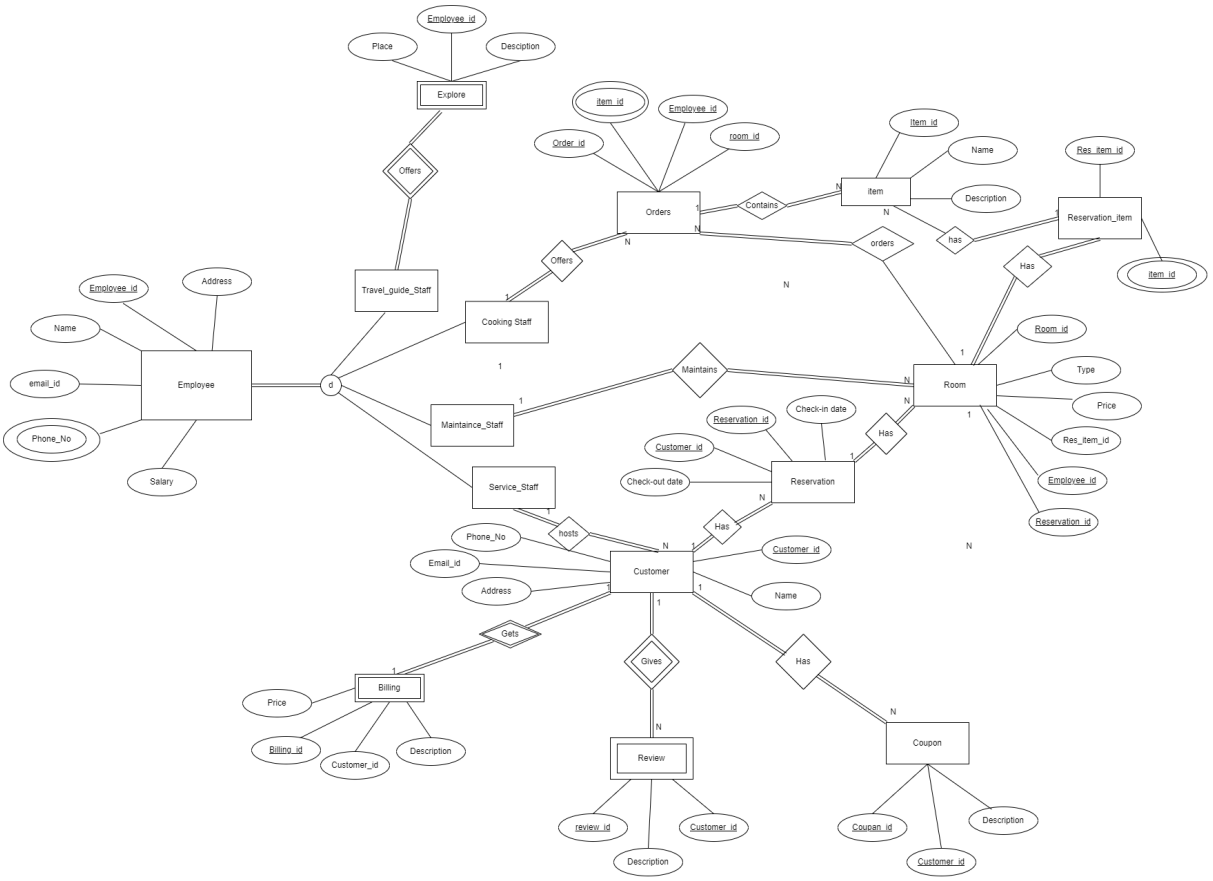


Figure 1 : Chen model for bed and breakfast.

Step 4: Design Issues

Created a separate entity called Salary to handle the time variant nature of the salary attribute. This approach allows for better management and tracking of salary-related information over time. By separating the Salary entity from the Employee entity, it becomes possible to capture changes in salaries for employees throughout their employment tenure.

This is also true for room and their prices as the room price will change as the days go. Therefore I created a separate entity to track room prices.

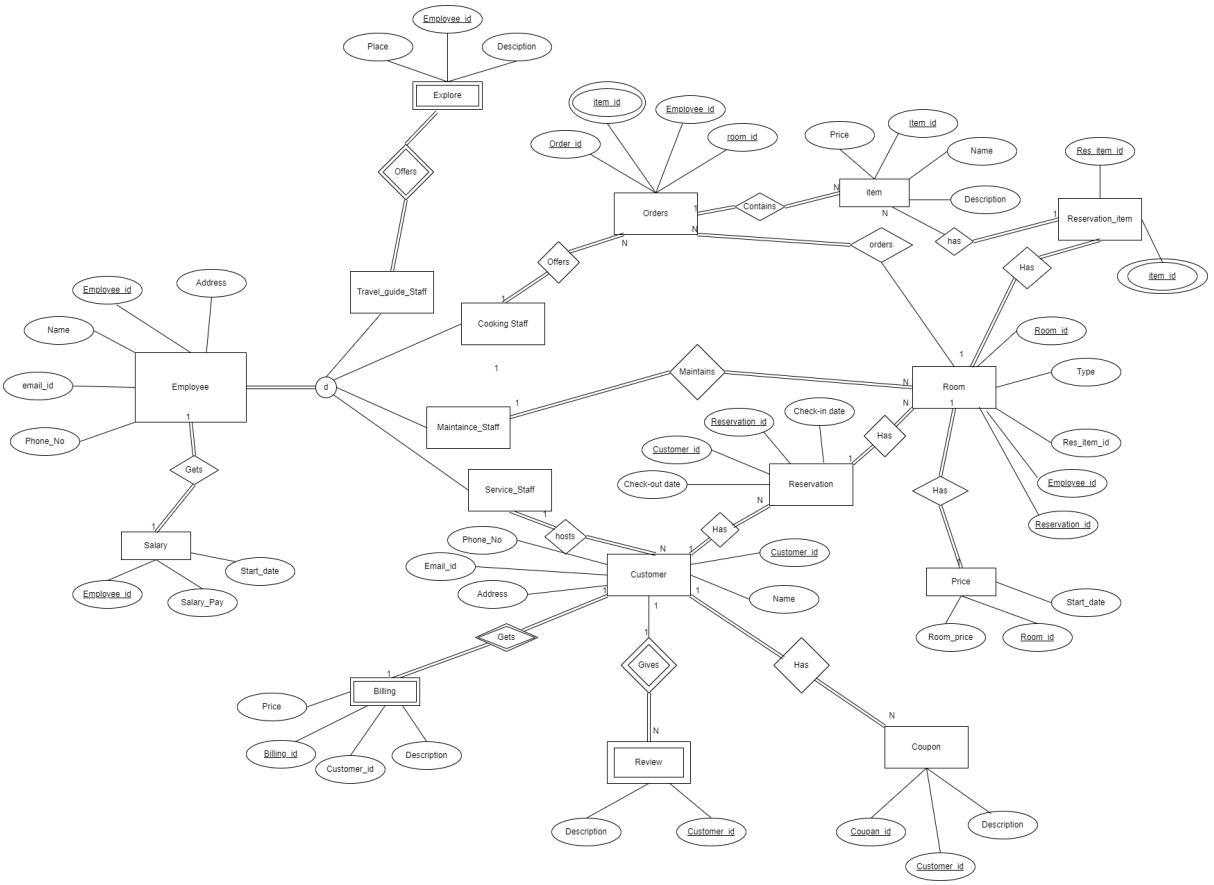


Figure 2: ERD of bed and breakfast(Design Issues)

Step 5: Tabular Structure

1. Employee

Employee_id(PK)	Name	Address	Email_id	Phone_No	Salary

2.Explore

Employee_id(FK)	Place(Partial Key)	Description

3. Orders

order_id(PK)	item_id(FK)	Employee_id(FK)	room_id(FK)

4. Item

item_id(PK)	Price	Name	Description

5. Reservation_item

Res_item_Id(PK)	item_id(FK)

6. Room

Room_id(PK)	Type	Res_item_id(FK)	Employee_id (FK)	Reservation_id (FK)	Price

7. Reservation

Reservation_id(PK)	Check-in date	Check-out date	Customer_id(FK)

8. Customer

Customer_id (PK)	Name	Phone_No	Email_id	Address

9. Review

Description(Partial Key)	Customer_id(FK)

10. Billing

Billing_id(PK)	Price	Customer_id(FK)	Description

11. Coupon

Coupon_id(PK)	Customer_id(FK)	Description

The given entities and their dependencies are:

- **Employee Dependencies**: None
- **Explore Dependencies**: Employee (Employee_id)
- **Orders Dependencies**: Employee (Employee_id), Item (item_id), Room (room_id)
- **Item Dependencies**: None
- **Reservation_item Dependencies**: Item (item_id)
- **Room Dependencies** :Reservation_item (Res_item_id), Employee (Employee_id), Reservation (Reservation_id)
- **Reservation Dependencies**: Customer (Customer_id)
- **Customer Dependencies**: None
- **Review Dependencies**: Customer (Customer_id)
- **Billing Dependencies**: Customer (Customer_id)
- **Coupon Dependencies**: Customer (Customer_id)

Employee entity is referenced by Explore and Orders entities.
Item entity is referenced by Orders and Reservation_item entities.
Reservation_item entity is referenced by Room entity.
Employee and Reservation entities are referenced by Room entity.
Customer entity is referenced by Reservation, Review, Billing, and Coupon entities.

Step 6 :Normalization

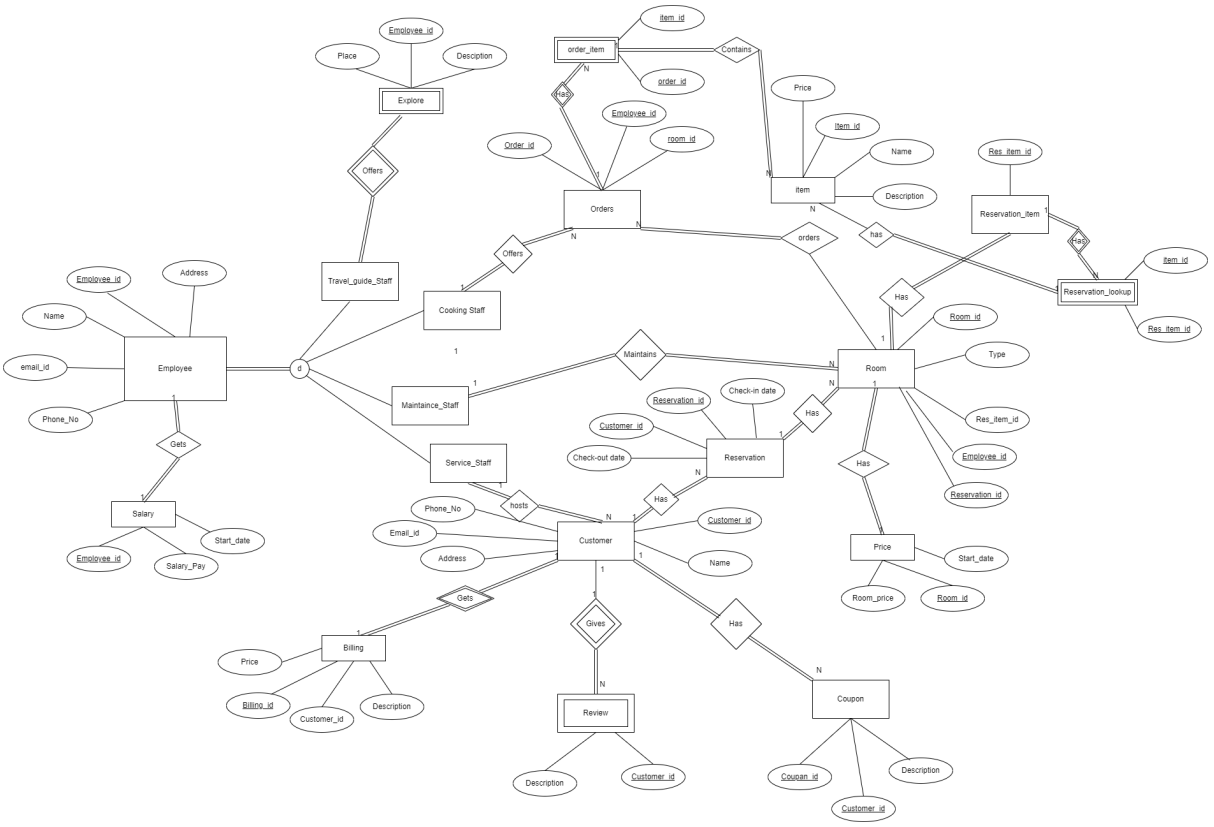


Figure 3: Normalised ERD of bed and breakfast

The entities orders and reservation_items had multivalued attributes namely “item_id” because one order can have multiple items in it. I added a lookup table for both the entities which brought the entire model to its first normal form(1NF).

Step 7: Query and Physical Model

```
create database BedBreakfast;
use BedBreakfast;
CREATE TABLE Employee (
  Employee_id INT PRIMARY KEY,
  `Name` VARCHAR(200),
  Address VARCHAR(200),
  Email_id VARCHAR(200),
  Phone_No VARCHAR(20)
);
CREATE TABLE SALARY(
          Employee_id INT,
    Salary_pay INT,
    Start_date date,
    FOREIGN KEY (Employee_id) references Employee(Employee_id)
);
CREATE TABLE Item (
  item_id INT PRIMARY KEY,
  Price DECIMAL(10, 2),
  `Name` VARCHAR(200),
  `Description` VARCHAR(200)
);
CREATE TABLE Customer (
  Customer_id INT PRIMARY KEY,
  `Name` VARCHAR(200),
  Phone_No VARCHAR(20),
  Email_id VARCHAR(200),
  Address VARCHAR(200),
  Host_id INT,
  FOREIGN KEY (Host_id) REFERENCES Employee(Employee_id)
);
CREATE TABLE Reservation_item (
  Res_item_Id INT PRIMARY KEY
);
CREATE TABLE Explore (
  Guide_id INT,
  Place VARCHAR(200),
  `Description` VARCHAR(200),
  FOREIGN KEY (Guide_id) REFERENCES Employee(Employee_id)
);
CREATE TABLE Reservation (
  Reservation_id INT PRIMARY KEY,
  `Check-in date` DATE,
  `Check-out date` DATE,
  Customer_id INT,
  FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id)
```

```
);
CREATE TABLE Room (
  Room_id INT PRIMARY KEY,
  `Type` VARCHAR(200),
  Res_item_id INT,
  Maintainer_id INT,
  Reservation_id INT,
  FOREIGN KEY (Res_item_id) REFERENCES Reservation_item(Res_item_Id),
  FOREIGN KEY (Maintainer_id) REFERENCES Employee(Employee_id),
  FOREIGN KEY (Reservation_id) REFERENCES Reservation(Reservation_id)
);
CREATE TABLE Room_Price(
        Room_id INT,
    Room_price DECIMAL(10,2),
    Start_date date,
    FOREIGN KEY (Room_id) REFERENCES Room(Room_id)
);
CREATE TABLE Orders (
  order_id INT PRIMARY KEY,
  Service_Provider_id INT,
  room_id INT,
  FOREIGN KEY (Service_Provider_id) REFERENCES Employee(Employee_id),
  FOREIGN KEY (room_id) REFERENCES Room(Room_id)
);
CREATE TABLE Order_Item (
        item_id int,
    order_id int,
    FOREIGN KEY (item_id) REFERENCES Item(item_id),
  FOREIGN KEY (order_id) REFERENCES Orders(order_id)
);
CREATE TABLE Reservation_Item_Lookup(
        Res_item_Id INT,
    item_id INT,
    FOREIGN KEY (Res_item_Id) REFERENCES Reservation_Item(Res_item_Id),
  FOREIGN KEY (item_id) REFERENCES Item(item_id)
);
CREATE TABLE Review (
  `Description` VARCHAR(200),
  Customer_id INT,
  FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id)
);
CREATE TABLE Billing (
  Billing_id INT PRIMARY KEY,
  Price DECIMAL(10, 2),
  Customer_id INT,
  `Description` VARCHAR(200),
  FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id)
);
```

```
CREATE TABLE Coupon (
  Coupon_id INT PRIMARY KEY,
  Customer_id INT,
  `Description` VARCHAR(200),
  FOREIGN KEY (Customer_id) REFERENCES Customer(Customer_id)
);
```

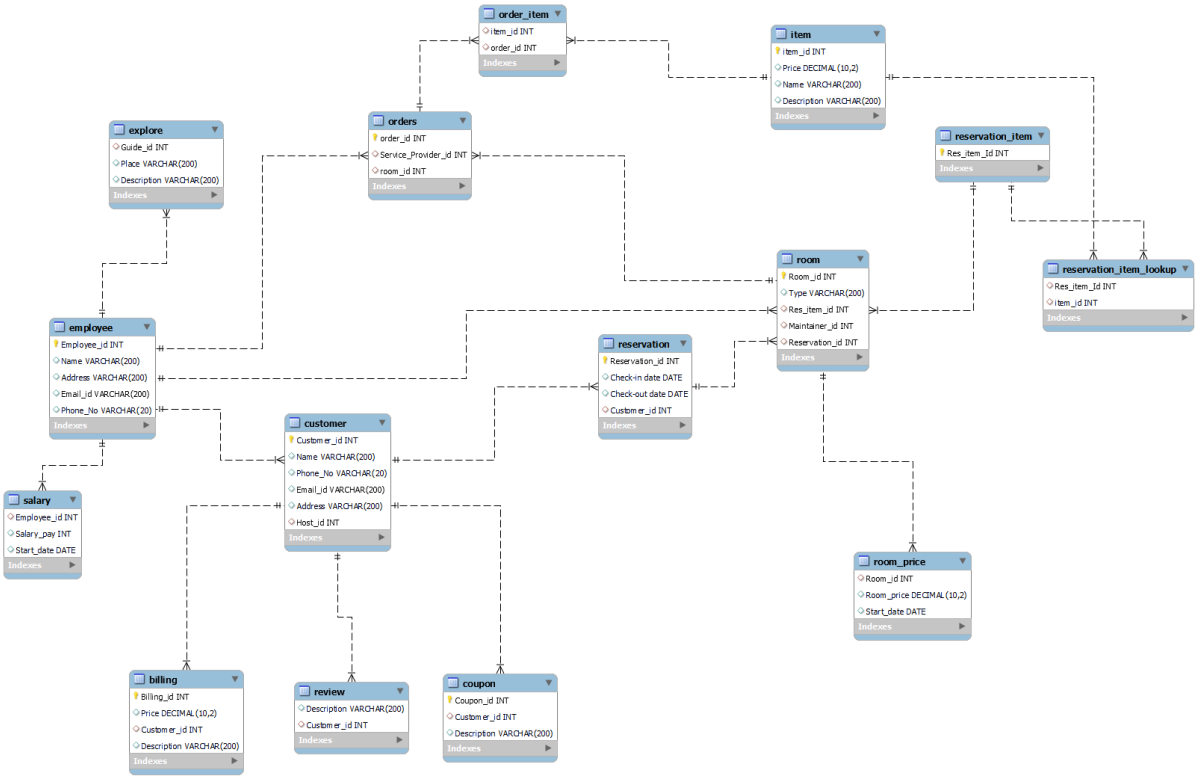


Figure 4: Physical Model of Bed and Breakfast

Problem 2

I have considered 8 entities namely Park, Features, Activities/Events, Reservation, Rules, News, Customers, Region.

The "Park" entity represents various parks in the system, including their unique identification, name, type (such as national, provincial, urban, or wildlife park), location, description, contact information, opening hours, facilities available, and a foreign key relationship with the "Region" entity to associate parks with their respective regions.

The "Features" entity represents distinct features found within parks, such as beaches, waterfalls, or forests. Each feature has a unique identification, name, description, and is associated with a specific park using a foreign key relationship.

The "Activities" entity represents recreational activities offered within parks, such as hiking, camping, or birdwatching. Each activity is identified by a unique ID, has a name, description, and is associated with a specific park using a foreign key relationship.

The "Reservation" entity represents reservations made by customers for specific parks and activities. Each reservation has a unique ID, is associated with a specific customer and park using foreign key relationships, includes the date of reservation, duration, and the number of participants.

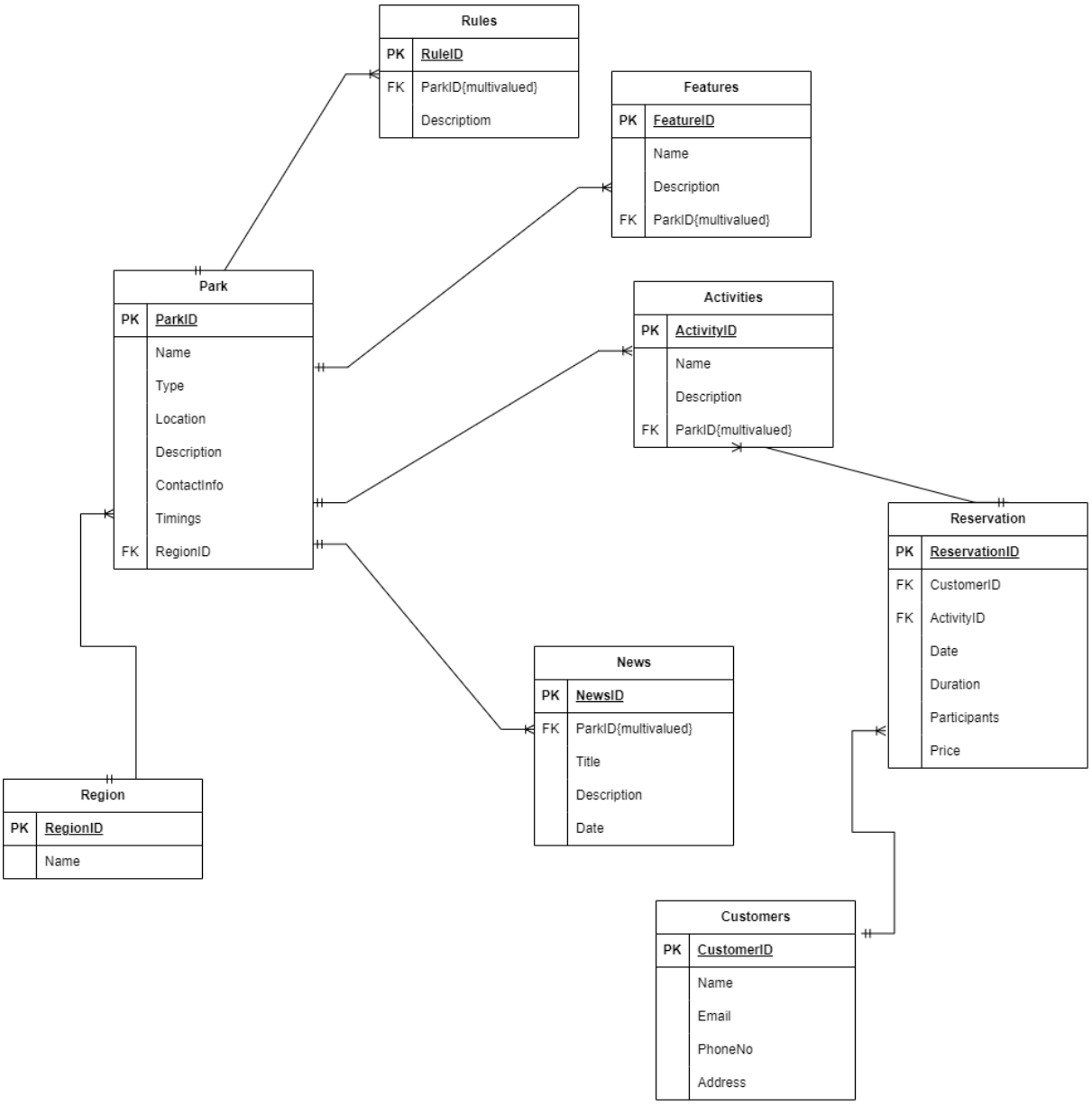
The "Rules" entity represents specific rules or regulations associated with individual parks. Each rule has a unique ID, description, and is associated with a specific park using a foreign key relationship.

The "News" entity represents news articles or updates related to parks. Each news article has a unique ID, title, description, date, and is associated with a specific park using a foreign key relationship.

The "Customers" entity represents customers who make reservations and engage with the park system. Each customer has a unique ID, name, email, phone number, and address.

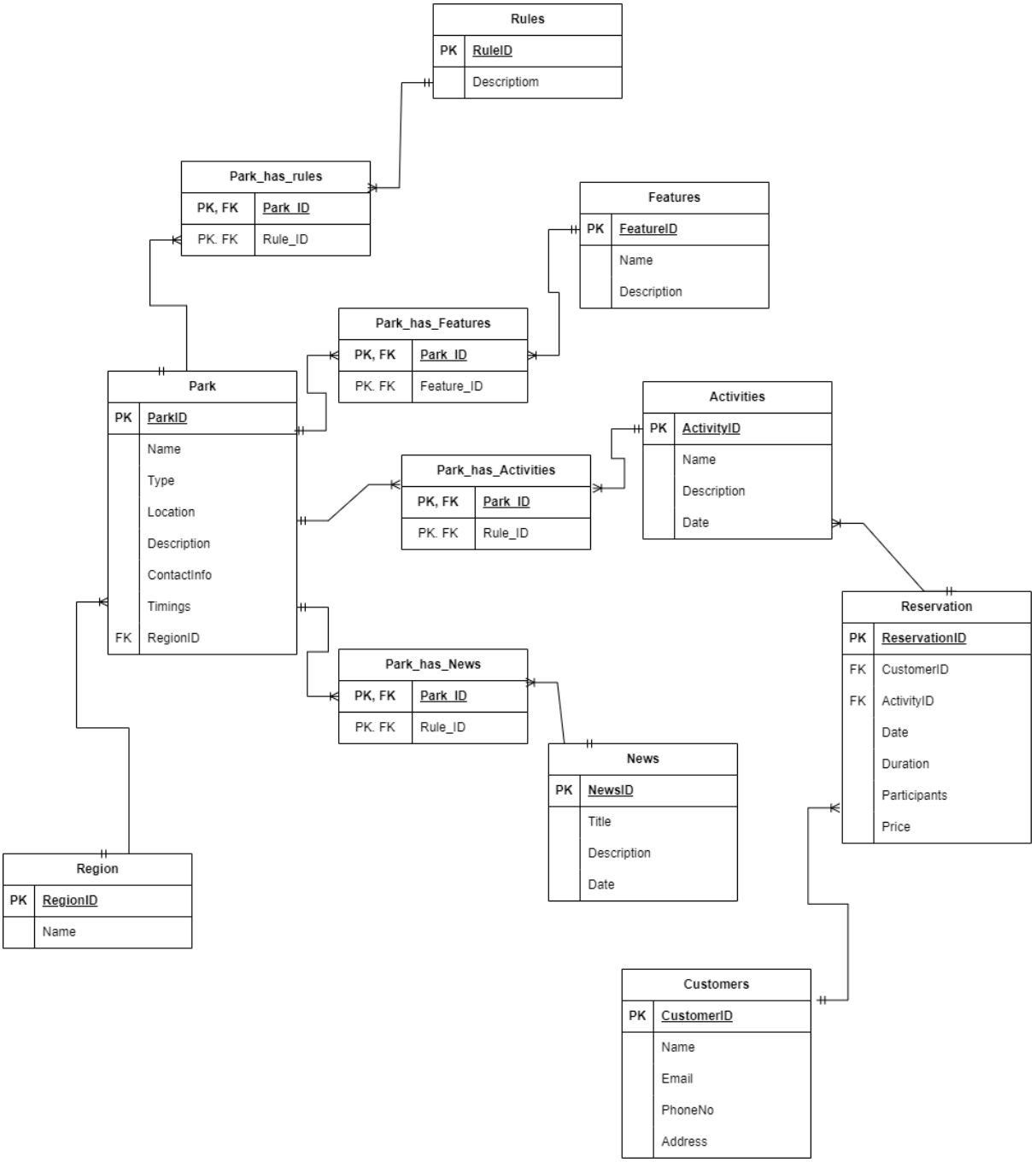
The "Region" entity represents different regions within Nova Scotia. Each region is identified by a unique ID and has a name. This entity allows parks to be associated with their respective regions using a foreign key relationship.

ER Diagram


Rules
PK RuleID
FK ParkID{multivalued}
Descriptiom
Features
PK FeatureID
Name
Description
FK ParkID{multivalued}
Park
PK ParkID
Name
Type
Location
Description
ContactInfo
Timings
FK RegionID
Activities
PK ActivityID
Name
Description
FK ParkID{multivalued}
Reservation
PK ReservationID
FK CustomerID
FK ActivityID
Date
Duration
Participants
Price
News
PK NewsID
FK ParkID{multivalued}
Title
Description
Date
Region
PK RegionID
Name
Customers
PK CustomerID
Name
Email
PhoneNo
Address

Normalization

Since ParkID is a multivalued attribute in rules, features, news and activities entity. I Have made a look up table which normalises the table.

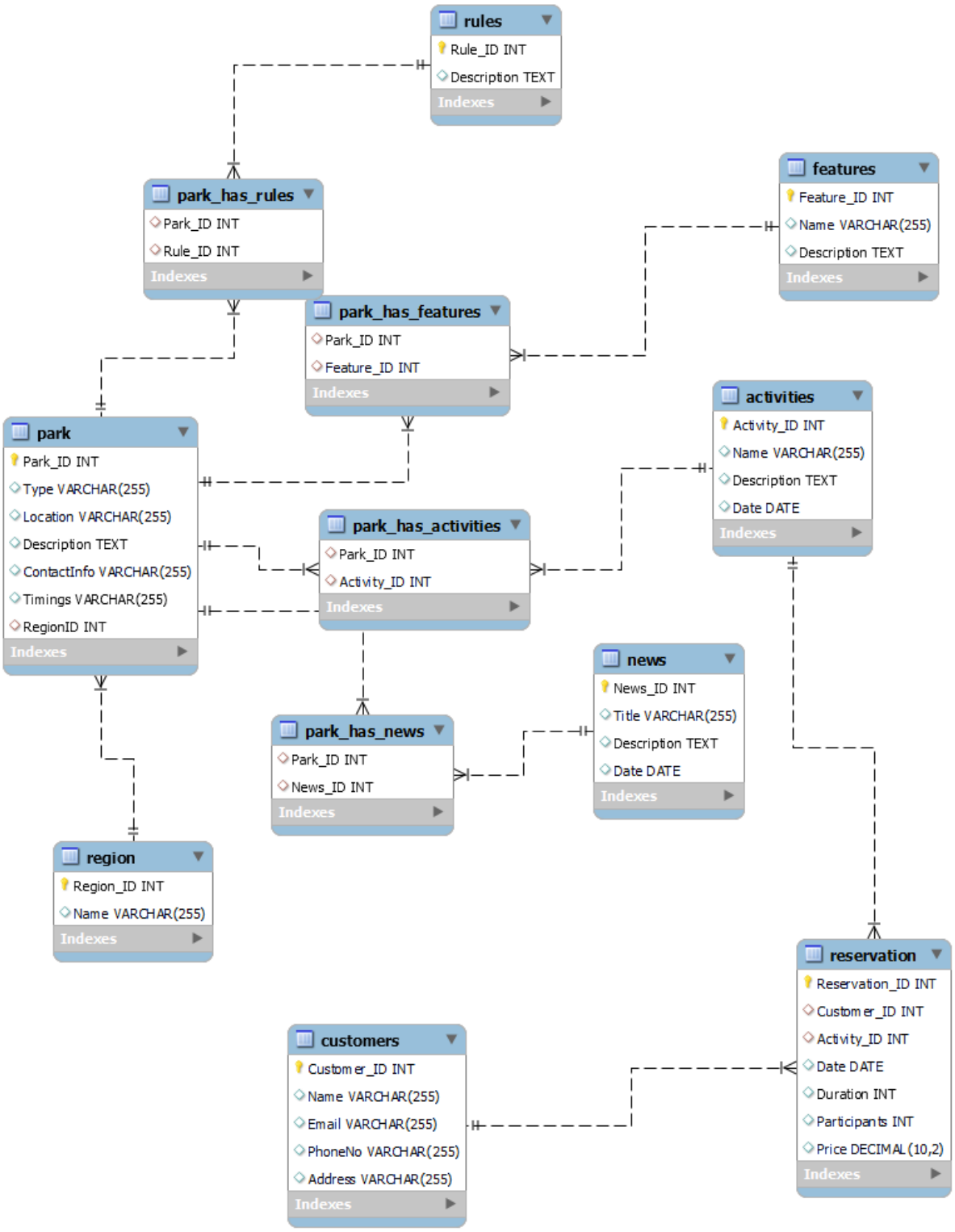


Query

```
create DATABASE ParkDatabase;
use ParkDatabase;
-- Customers Table
CREATE TABLE Customers (
  Customer_ID INT PRIMARY KEY,
  `Name` VARCHAR(255),
  Email VARCHAR(255),
  PhoneNo VARCHAR(255),
  Address VARCHAR(255)
);

CREATE TABLE Region(
        Region_ID INT PRIMARY KEY,
        `Name` VARCHAR(255)
);

CREATE TABLE Rules (
  Rule_ID INT PRIMARY KEY,
  `Description` TEXT
);

CREATE TABLE News (
  News_ID INT PRIMARY KEY,
  Title VARCHAR(255),
  `Description` TEXT,
  `Date` DATE
);

CREATE TABLE Park (
  Park_ID INT PRIMARY KEY,
  `Type` VARCHAR(255),
  Location VARCHAR(255),
  `Description` TEXT,
  ContactInfo VARCHAR(255),
  Timings VARCHAR(255),
  RegionID INT,
  FOREIGN KEY (RegionID) REFERENCES Region(Region_ID)
);
```

```
CREATE TABLE Park_has_rules (
  Park_ID INT,
  Rule_ID INT,
  FOREIGN KEY (Park_ID) REFERENCES Park(Park_ID),
  FOREIGN KEY (Rule_ID) REFERENCES Rules(Rule_ID)
);

CREATE TABLE Features (
  Feature_ID INT PRIMARY KEY,
  `Name` VARCHAR(255),
  `Description` TEXT
);

CREATE TABLE Park_has_Features (
  Park_ID INT,
  Feature_ID INT,
  FOREIGN KEY (Park_ID) REFERENCES Park(Park_ID),
  FOREIGN KEY (Feature_ID) REFERENCES Features(Feature_ID)
);

CREATE TABLE Activities (
  Activity_ID INT PRIMARY KEY,
  `Name` VARCHAR(255),
  `Description` TEXT,
  `Date` DATE
);

CREATE TABLE Park_has_Activities (
  Park_ID INT,
  Activity_ID INT,
  FOREIGN KEY (Park_ID) REFERENCES Park(Park_ID),
  FOREIGN KEY (Activity_ID) REFERENCES Activities(Activity_ID)
);

CREATE TABLE Park_has_News (
  Park_ID INT,
  News_ID INT,
  FOREIGN KEY (Park_ID) REFERENCES Park(Park_ID),
  FOREIGN KEY (News_ID) REFERENCES News(News_ID)
);
```

```
CREATE TABLE Reservation (
  Reservation_ID INT PRIMARY KEY,
  Customer_ID INT,
  Activity_ID INT,
  `Date` DATE,
  Duration INT,
  Participants INT,
  Price DECIMAL(10,2),
  FOREIGN KEY (Customer_ID) REFERENCES Customers(Customer_ID),
  FOREIGN KEY (Activity_ID) REFERENCES Activities(Activity_ID)
);
```



Similarities and Differences

The reverse engineered ERD and Manual ERD are almost similar. Both have the same cardinalities between most of the entities between them. The only difference between them is the cardinality difference between “reservation” and “Activities” entity. Apart from that all the cardinality information with entity and attribute match the same.