# Assignment One:
# Database Design and Implementation

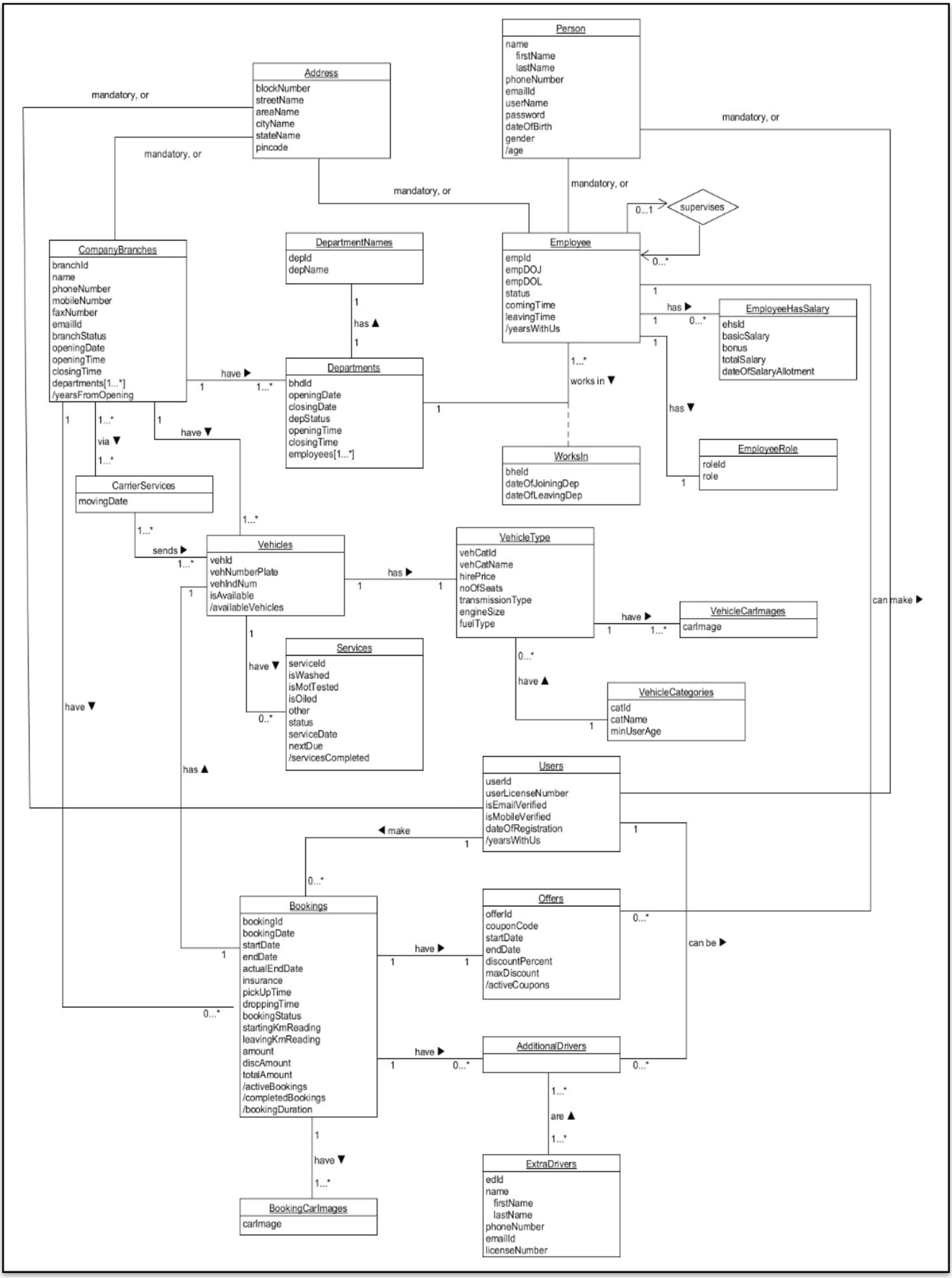
**Group No. :**
1

**Group Members :**
Akshay Arunkumar Garg,
Daniella Anwar,
Jaden Vadivelu,
Rithin Thomas,
Sana Sajesh,
Vidisha Jain

**Course Name :**
F28DM

**Department :**
School Of Mathematics and Computer Science(MACS)

# Task 1: - Conceptual model



Recursive Relationship: In ER Diagram, we have used recursive relationships in the Employees table where an employee is supervised by any other employee. In addition, one employee can supervise many other employees.

Derived Attributes: Ten derived attributes are observed in this ER Diagram, which is presented in many tables making it easy to calculate the fields which are required at the time of searching or executing.

Composite Attributes: The composite attributes which we added is 'name' in the Person table and Extra Drivers entity.

Composite Keys: We have presented composite keys in four tables: Additional Drivers, CarrrierServices, VehicleCarImages, and BookingCarImages.

Repeating Attributes: These are the attributes that are declared in CompanyBranches table and Departments table. They are declared in these two tables only because one company branch can have many departments and also in Departments table one department can have many employees this shows that this is a repeating attribute.

Use of generalization and specialisation: We have used this in Address and Person entities. The entity specialisation we used in these entities is Mandatory Or, where we created a relation for each subtype, added attributes for common attributes, added attributes specific to that entity subtype. Also, all relations have the same primary key.

# Task 2: - Translation into Relational Schema

CompanyBranches(**branchId**, name, phoneNumber, mobileNumber, faxNumber, emailId, blockNumber, streetName, areaName, cityName, stateName, pincode, openingDate, openingTime, closingTime, branchStatus)

DepartmentNames(**depId**, depName)

Departments(**bhdId**, *depId, branchId*, openingDate, closingDate, openingTime, closingTime, depStatus)

EmployeeRole(**roleId**, role)

Employee(**empId**, firstName, lastName, phoneNumber, emailId, userName, password, dateOfBirth, gender, *roleId,* empDOJ, empDOL, comingTime, leavingTime, *supervisor* blockNumber, streetName, areaName, cityName, stateName, pincode, status)

WorksIn(**bheId**, *bhdId, empId,* dateOfJoiningDep, dateOfLeavingDep)

EmployeeHasSalary(**ehsId**, *empId*, basicSalary, bonus, totalSalary, dateOfSalaryAllotment)

VehicleCategories(**catId**, catName, minUserAge)

VehicleType(**vehCatId**, vehCatName, hirePrice, noOfSeats, *catId*, transmissionType, engineSize, fuelType)

VehicleCarImages(***vehCatId***, **carImage**)

Vehicles(**vehId**, *vehCatId*, vehNumberPlate, vehIndNum, *homeBranchId*, isAvailable)

CarrierServices(***vehIndNum, movingFrom, movingTo***, **movingDate**)

Services(**serviceId**, *vehIndNum*, isWashed, isMotTested, isOiled, other, serviceDate, nextDue, status)

Users( **userId**, firstName, lastName, phoneNumber, emailId, userName, password, dateOfBirth, gender, blockNumber, streetName, areaName, cityName, stateName, pincode, licenseNumber, isEmailVerified, isMobileVerified, dateOfRegistration)

ExtraDrivers(**edId**, licenseNumber, firstName, lastName, phoneNumber, emailId)

Offers(**offerId**, couponCode, startDate, endDate, discountPercent, maxDiscount, *empId*)

Bookings(**bookingId**, *vehIndNum, userPhoneNum*, bookingDate, startDate, endDate, actualEndDate, haveInsurance, *pickUpBranchId, leavingBranchId*, pickUpTime, droppingTime, bookingStatus, startingKmReading, leavingKmReading, amount, *couponCode*, discAmount, totalAmount)

AdditionalDrivers(***edId***, ***userId***, ***bookingId***)

BookingCarImages(***bookingId***, **carImage**)

# Task 3: - Implementation of the Schema in MariaDB

1. **Company Branches Table:**

```
#Creating table for Company branches to store the details of the branches in UK.
CREATE TABLE CompanyBranches
(
  branchId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  name VARCHAR(20) NOT NULL UNIQUE,
  phoneNumber VARCHAR(13) NOT NULL UNIQUE,
  mobileNumber VARCHAR(13),
  faxNumber VARCHAR(13),
  emailId VARCHAR(30) NOT NULL DEFAULT "contactus@hwmotors.uk",
  blockNumber VARCHAR(6),
  streetName VARCHAR(30),
  areaName VARCHAR(30),
  cityName VARCHAR(30),
  stateName VARCHAR(30),
  pincode VARCHAR(6),
  openingDate DATE DEFAULT CURDATE(),
  openingTime TIME DEFAULT CURTIME(),
  closingTime TIME,
  branchStatus ENUM('active', 'inActive') NOT NULL DEFAULT "active"
)ENGINE=INNODB;
```

2. **Department Names Table:**

```
#Creating table for storing the unique name of the departments so that we dont have to
duplicate the names
CREATE TABLE DepartmentNames
(
  depId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  depName VARCHAR(20) NOT NULL UNIQUE
)ENGINE=INNODB;
```

3. **Departments Table:**

```
#Creating table for departments of each branch in UK.
CREATE TABLE Departments
(
  bhdId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  depId INT(11),
  branchId INT(11) NOT NULL,
  openingDate DATE DEFAULT CURDATE(),
  closingDate DATE,
  openingTime TIME DEFAULT CURTIME(),
  closingTime TIME,
  depStatus ENUM('active', 'inActive') DEFAULT "active",
  FOREIGN KEY (depId) REFERENCES DepartmentNames(depId) ON DELETE SET NULL,
  FOREIGN KEY (branchId) REFERENCES CompanyBranches(branchId) ON DELETE CASCADE
)ENGINE=INNODB;
```

4. **EmployeeRole Table:**

```
#Creating a table to define all the unique roles of an employee.
CREATE TABLE EmployeeRole
(
  roleId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  role VARCHAR(40) NOT NULL UNIQUE
)ENGINE=INNODB;
```

5. **Employee Table:**

```
#Creating Employee table for storing details for each employee in all the branches and departments.
CREATE TABLE Employee
(
  empId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  firstName VARCHAR(20) NOT NULL,
  lastName VARCHAR(20),
  phoneNumber VARCHAR(13) NOT NULL UNIQUE,
  emailId VARCHAR(30) UNIQUE,
  userName VARCHAR(20) NOT NULL UNIQUE,
  password VARCHAR(30) NOT NULL DEFAULT "emp1234",
  dateOfBirth DATE,
  gender ENUM('M','F','O'),
  roleId INT(11),
  empDOJ DATE DEFAULT CURDATE(),
  empDOL DATE,
  comingTime TIME DEFAULT CURTIME(),
  leavingTime TIME,
  supervisor INT(11),
  blockNumber VARCHAR(6),
  streetName VARCHAR(30),
  areaName VARCHAR(30),
  cityName VARCHAR(30),
  stateName VARCHAR(30),
  pincode VARCHAR(30),
  status ENUM('active', 'inActive') NOT NULL DEFAULT "active",
  FOREIGN KEY (roleId) REFERENCES EmployeeRole(roleId) ON DELETE SET NULL,
   FOREIGN KEY (supervisor) REFERENCES Employee(empId) ON DELETE SET NULL
)ENGINE=INNODB;
```

6. **Work In Table:**

```
#Creating table for storing data of employees working in each department. This table keep
records of and employee in which he/she worked
CREATE TABLE WorksIn
(
  bheId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  bhdId INT(11),
  empId INT(11),
  dateOfJoiningDep DATE DEFAULT CURDATE(),
  dateOfLeavingDep DATE,
  FOREIGN KEY (bhdId) REFERENCES Departments(bhdId) ON DELETE SET NULL,
  FOREIGN KEY (empId) REFERENCES Employee(empId) ON DELETE SET NULL
)ENGINE=INNODB;
```

7. **Employee Has Salary Table:**

```
#Creating a table to manage salaries of employees working in company.
CREATE TABLE EmployeeHasSalary
(
  ehsId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  empId INT(11) NOT NULL,
  basicSalary FLOAT DEFAULT 0,
  bonus FLOAT DEFAULT 0,
  totalSalary FLOAT DEFAULT 0,
  dateOfSalaryAllotment DATE,
  FOREIGN KEY (empId) REFERENCES Employee(empId) ON DELETE CASCADE
)ENGINE=INNODB;
```

8. **Vehicle Categories Table:**

```
#Creating table to store categories of vehicles like sports, luxury and all.
CREATE TABLE VehicleCategories
(
  catId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  catName VARCHAR(40) NOT NULL UNIQUE,
  minUserAge INT(3) NOT NULL DEFAULT 21
)ENGINE=INNODB;
```

9. **Vehicle Type Table:**

```
#Creating table for storing the types of vehicles in company.
CREATE TABLE VehicleType
(
  vehCatId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
  vehCatName VARCHAR(40) NOT NULL UNIQUE,
  hirePrice FLOAT DEFAULT 0,
  noOfSeats INT(2),
  catId INT(11),
  transmissionType ENUM('manual', 'automatic', 'autopilot') DEFAULT "automatic",
  engineSize VARCHAR(20),
  fuelType ENUM('diesel', 'petrol', 'electric'),
  FOREIGN KEY (catId) REFERENCES VehicleCategories(catId) ON DELETE SET NULL
)ENGINE=INNODB;
```

**10. Vehicle Car Images table:**

```
Creating table for storing vehicle car images for displaying on website
CREATE TABLE VehicleCarImages
(
    vehCatId INT(11) NOT NULL,
    carImage VARCHAR(50) NOT NULL UNIQUE,
    PRIMARY KEY(vehCatId,carImage),
    FOREIGN KEY (vehCatId) REFERENCES VehicleType(vehCatId) ON DELETE CASCADE
)ENGINE=INNODB;
```

**11. Vehicles Table**

```
#Creating table for vehicles showing information of the vehicle.
CREATE TABLE Vehicles
(
    vehId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    vehCatId INT(11),
    vehNumberPlate VARCHAR(20) UNIQUE,
    vehIndNum VARCHAR(16) NOT NULL UNIQUE,
    homeBranchId INT(11),
    isAvailable BOOLEAN DEFAULT 1,
    FOREIGN KEY (vehCatId) REFERENCES VehicleType(vehCatId) ON DELETE SET NULL,
    FOREIGN KEY (homeBranchId) REFERENCES CompanyBranches(branchId) ON DELETE SET
NULL
)ENGINE=INNODB;
```

**12. Carrier Services Table**

```
#Creating table for carrier service which sends vehicles from one branch to other
CREATE TABLE CarrierServices
(
    vehIndNum VARCHAR(16) NOT NULL,
    movingFrom INT(11),
    movingTo INT(11),
    movingDate DATE NOT NULL DEFAULT CURDATE(),
    PRIMARY KEY(vehIndNum,movingFrom,movingTo,movingDate),
    FOREIGN KEY (vehIndNum) REFERENCES Vehicles(vehIndNum) ON DELETE CASCADE,
    FOREIGN KEY (movingFrom) REFERENCES CompanyBranches(branchId) ON DELETE
CASCADE,
    FOREIGN KEY (movingTo) REFERENCES CompanyBranches(branchId) ON DELETE CASCADE
)ENGINE=INNODB;
```

**13. Services Table:**

```
#Creating table for services as the vehicles have services
CREATE TABLE Services
(
    serviceId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    vehIndNum VARCHAR(16) NOT NULL,
    isWashed BOOLEAN DEFAULT 0,
    isMotTested BOOLEAN DEFAULT 0,
    isOiled BOOLEAN DEFAULT 0,
    other BOOLEAN DEFAULT 0,
    serviceDate DATE DEFAULT CURDATE(),
    status ENUM('atService', 'completed') DEFAULT "atService",
    nextDue DATE,
    FOREIGN KEY (vehIndNum) REFERENCES Vehicles(vehIndNum) ON DELETE CASCADE
)ENGINE=INNODB;
```

**14. Users Table:**

```
#Creating table for users showing all their details and information.
CREATE TABLE Users
(
    userId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    firstName VARCHAR(20) NOT NULL,
    lastName VARCHAR(20),
    phoneNumber VARCHAR(13) NOT NULL UNIQUE,
    emailId VARCHAR(30) UNIQUE,
    userName VARCHAR(20) NOT NULL UNIQUE,
    password VARCHAR(30) NOT NULL DEFAULT "user@123",
    dateOfBirth DATE,
    gender ENUM('M', 'F', 'O'),
    blockNumber VARCHAR(6),
    streetName VARCHAR(30),
    areaName VARCHAR(30),
    cityName VARCHAR(30),
    stateName VARCHAR(30),
    pincode VARCHAR(30),
    licenseNumber VARCHAR(40) UNIQUE,
    isEmailVerified BOOLEAN NOT NULL DEFAULT 0,
    isMobileVerified BOOLEAN NOT NULL DEFAULT 0,
    dateOfRegistration DATE NOT NULL DEFAULT CURDATE()
)ENGINE=INNODB;
```

**15. Extra Drivers Table:**

```
#Creating table for extra drivers showing their names and details.
CREATE TABLE ExtraDrivers
(
    edId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    licenseNumber VARCHAR(40) NOT NULL UNIQUE,
    firstName VARCHAR(20) NOT NULL,
    lastName VARCHAR(20),
    phoneNumber VARCHAR(13) NOT NULL UNIQUE,
    emailId VARCHAR(30) UNIQUE
)ENGINE=INNODB;
```

**16. Offers Table:**

```
#Creating table for creating offers that can be used in booking to avail discount.
CREATE TABLE Offers
(
    offerId INT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    couponCode VARCHAR(10) NOT NULL UNIQUE,
    startDate DATE DEFAULT CURDATE(),
    endDate DATE,
    discountPercent FLOAT DEFAULT 0,
    maxDiscount FLOAT DEFAULT 0,
    empId INT(11),
    FOREIGN KEY (empId) REFERENCES Employee(empId) ON DELETE SET NULL
)ENGINE=INNODB;
```

**17. Booking Table:**

```
#Creating table for bookings taking all the deatils and information of all the bookings.
CREATE TABLE Bookings
(
    bookingId BIGINT(11) PRIMARY KEY AUTO_INCREMENT NOT NULL,
    vehIndNum VARCHAR(16),
    userPhoneNum VARCHAR(13),
    bookingDate DATE DEFAULT CURDATE(),
    startDate DATE DEFAULT CURDATE(),
    endDate DATE,
    actualEndDate DATE,
    haveInsurance BOOLEAN DEFAULT 1,
    pickUpBranchId INT(11),
    leavingBranchId INT(11),
    pickUpTime TIME,
    droppingTime TIME,
    bookingStatus ENUM('booked', 'active', 'completed') DEFAULT "booked",
    startingKmReading FLOAT DEFAULT 0,
    leavingKmReading FLOAT DEFAULT 0,
    amount FLOAT DEFAULT 0,
    couponCode VARCHAR(10),
    discAmount FLOAT DEFAULT 0,
    totalAmount FLOAT DEFAULT 0,
    FOREIGN KEY (vehIndNum) REFERENCES Vehicles(vehIndNum) ON DELETE SET NULL,
    FOREIGN KEY (userPhoneNum) REFERENCES Users(phoneNumber) ON DELETE SET NULL,
    FOREIGN KEY (pickUpBranchId) REFERENCES CompanyBranches(branchId) ON DELETE SET NULL,
    FOREIGN KEY (leavingBranchId) REFERENCES CompanyBranches(branchId) ON DELETE SET NULL,
    FOREIGN KEY (couponCode) REFERENCES Offers(couponCode) ON DELETE SET NULL
)ENGINE=INNODB;
```

**18. Additional Drivers Table:**

```
#Creating table for additional drivers, as some bookings have additional drivers.
CREATE TABLE AdditionalDrivers
(
    edId INT(11) NULL,
    bookingId BIGINT(11) NOT NULL,
    userId INT(11) NULL,
    PRIMARY KEY(edId,bookingId,userId),
    FOREIGN KEY (edId) REFERENCES ExtraDrivers(edId) ON DELETE CASCADE,
    FOREIGN KEY (bookingId) REFERENCES Bookings(bookingId) ON DELETE CASCADE,
    FOREIGN KEY (userId) REFERENCES Users(userId) ON DELETE CASCADE
)ENGINE=INNODB;
```

**19. Booking Car Images:**

```
#Creating table for storing car images that show car images while user makes booking.
CREATE TABLE BookingCarImages
(
    bookingId BIGINT(11) NOT NULL,
    carImage VARCHAR(50) NOT NULL UNIQUE,
    PRIMARY KEY(bookingId,carImage),
   FOREIGN KEY (bookingId) REFERENCES Bookings(bookingId) ON DELETE CASCADE
)ENGINE=INNODB;
```

# Task 4: - Indexes

```
#Index for Offers Table
CREATE INDEX OfferName ON Offers(couponCode);

#Index for Vehicle Type
CREATE INDEX ModelName ON VehicleType(vehCatName);

#Indexes for Bookings table
CREATE INDEX UserDetails ON Bookings(userPhoneNum);
CREATE INDEX VehicleDetails ON Bookings(vehIndNum);

#Index for Vehicles
CREATE INDEX VIN ON Vehicles(vehIndNum);

#Index for Number plate
CREATE INDEX VehicleNumberPlate ON Vehicles(vehNumberPlate);

#Index for company branches
CREATE INDEX BranchName ON CompanyBranches(name);

#Index for Extra Drivers
CREATE INDEX ExtraDriverLicense ON ExtraDrivers(licenseNumber);

#Index for Works in table to find the employee
CREATE INDEX EmployeeDetails ON WorksIn(empId);

#Indexes for users table
CREATE INDEX UserPhoneNumber ON Users(phoneNumber);
CREATE INDEX UserLoginName ON Users(userName);

#Indexes for Employee table
CREATE INDEX EmpPhoneNumber ON Employee(phoneNumber);
CREATE INDEX EmpUserName ON Employee(userName);
```

# Names and their Contributions

| Names | Contribution |
|---|---|
| Akshay Arunkumar Garg : | Database Designing and added constraints to the SQL create commands |
| Daniella Anwar : | Relationships in ER Diagram and Indexes for all the tables |
| Jaden Vadivelu : | SQL Create Commands for the tables |
| Rithin Thomas : | SQL Create Commands for the tables |
| Sana Sajesh : | Created ER Diagram and Entities in ER Diagram |
| Vidisha Jain : | Relational schema and SQL Create Commands |