

Database Schema Mapping for the Car Servicing System

The schema contains tables that capture information about **customers**, **cars**, **services**, and **mechanics**. Below is the database schema, followed by the ER diagram.

Customers Table

customer_id	name	email_address	phone_number
-------------	------	---------------	--------------

Cars Table

car_id	reg_no	make	model	manufacture_date	customer_id
--------	--------	------	-------	------------------	-------------

Service Table

service_id	drop_off_date	description	mileage	next_service	car_id
------------	---------------	-------------	---------	--------------	--------

Mechanic Table

mechanic_id	name	phone_number	grade
-------------	------	--------------	-------

Service Mechanic Table

service_id	mechanic_id	time_spent
------------	-------------	------------

Description of tables

1. Customers Table

The Customers table holds information about customers.

Column Name	Data Type	Description
customer_id	INT (PK)	Unique identifier for each customer.
name	VARCHAR(255)	Customer's full name.
email_address	VARCHAR(255)	Customer's email address.
phone_number	VARCHAR(15)	Customer's phone number.

2. Cars Table

The Cars table holds information about the cars.

Column Name	Data Type	Description
car_id	INT (PK)	Unique identifier for each car.
reg_no	VARCHAR(15)	Car's registration number.
make	VARCHAR(100)	Make of the car (e.g., Toyota, Ford).
model	VARCHAR(100)	Model of the car.
manufacture_date	DATE	Date the car was manufactured.
customer_id	INT (FK)	The customer who owns the car (foreign key).

3. Services Table

The Services table holds information about each service.

Column Name	Data Type	Description
service_id	INT (PK)	Unique identifier for each service.
drop_off_datetime	DATETIME	Date and time the car was dropped off for service.
description	TEXT	Description of the work to be done.
mileage	INT	Mileage of the car at the time of service.
next_service_date	DATE	Date for the next scheduled service.
car_id	INT (FK)	The car associated with the service (foreign key).

4. Mechanics Table

The Mechanics table holds information about each mechanic.

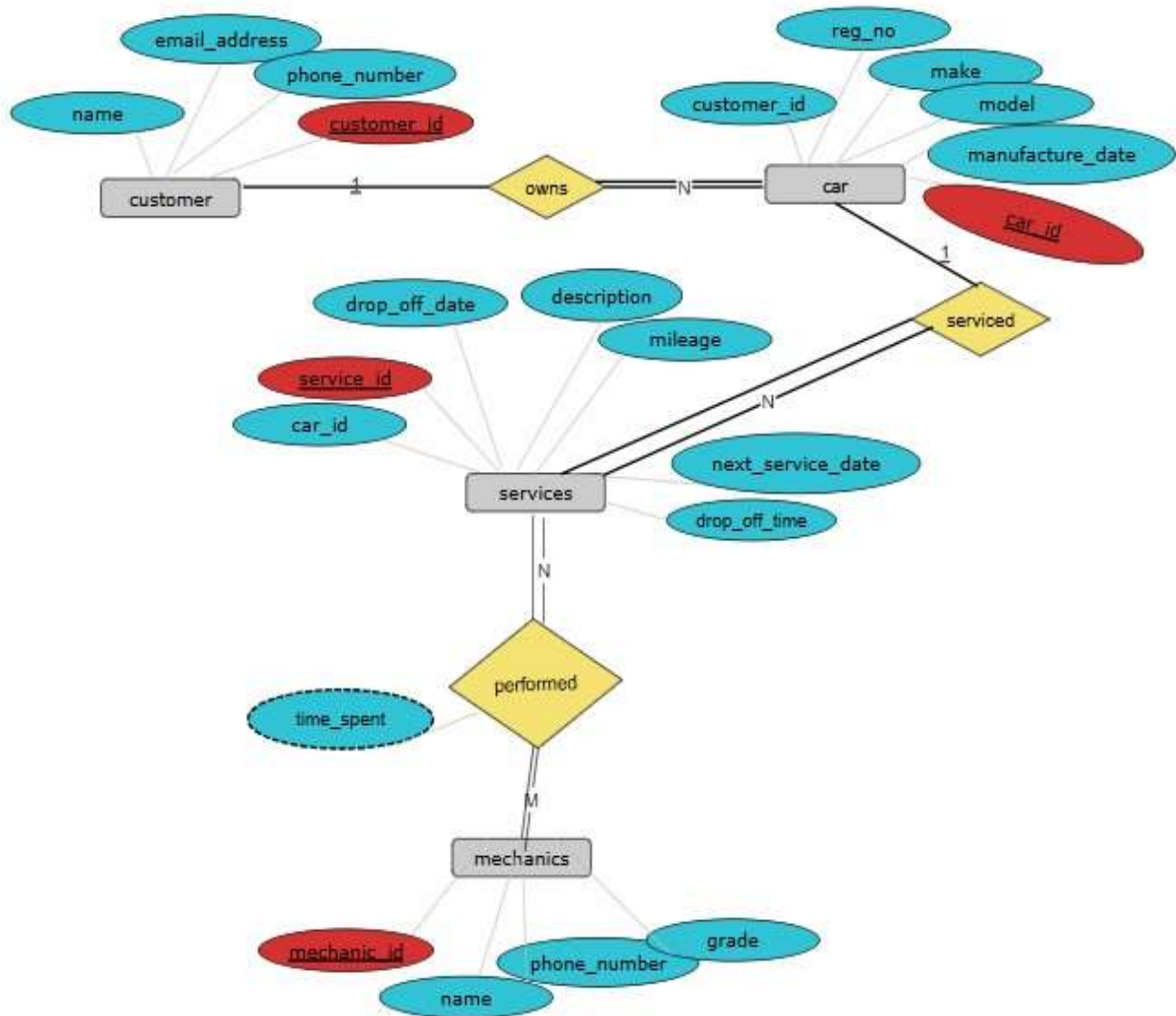
Column Name	Data Type	Description
mechanic_id	INT (PK)	Unique identifier for each mechanic.
name	VARCHAR(255)	Mechanic's full name.
phone_number	VARCHAR(15)	Mechanic's phone number.
grade	VARCHAR(50)	Grade of the mechanic (e.g., Junior, Senior).

5. Service Mechanics Table

The Service_Mechanics table holds information about which mechanics worked on which services, as multiple mechanics may work on one service.

Column Name	Data Type	Description
service_id	INT (FK)	The service ID (foreign key).
mechanic_id	INT (FK)	The mechanic ID (foreign key).
time_spent	INT	Time spent by the mechanic on the service (in hours).

ER Diagram



Here is how the entities relate to each other:

1. **Customers** and **Cars**: A customer can own many cars, but each car belongs to only one customer. The cardinality is **1 to many (1:N)** relationship between Customers and Cars.
2. **Cars** and **Services**: A car can have many services over time, but each service is associated with exactly one car. The cardinality is **1 to many (1:N)** relationship between Cars and Services.
3. **Services** and **Mechanics**: A service can be performed by one or more mechanics, and each mechanic can work on many services. The cardinality is **Many to many (N:M)** relationship between Services and Mechanics, which is implemented via the Service_Mechanics table.
4. **Mechanics** and **Services**: Each mechanic works on many services. The cardinality is **Many to many (N:M)** relationship between Services and Mechanics.

ER Diagram Description

- **Customer** (1) → (many) **Car** (one-to-many)
- **Car** (1) → (many) **Service** (one-to-many)
- **Service** (many) → (many) **Mechanic** (many-to-many through Service_Mechanics)

SQL Schema Code

-- Customers table

```
CREATE TABLE Customers (  
    customer_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(255),  
    email_address VARCHAR(255),  
    phone_number VARCHAR(15)  
);
```

-- Cars table

```
CREATE TABLE Cars (  
    car_id INT PRIMARY KEY AUTO_INCREMENT,  
    registration_no VARCHAR(15),  
    make VARCHAR(100),  
    model VARCHAR(100),  
    manufacture_date DATE,  
    customer_id INT,  
    FOREIGN KEY (customer_id) REFERENCES Customers(customer_id)  
);
```

-- Services table

```
CREATE TABLE Services (  
    service_id INT PRIMARY KEY AUTO_INCREMENT,  
    drop_off_datetime DATETIME,  
    description TEXT,  
    mileage INT,  
    next_service_date DATE,  
    car_id INT,  
    FOREIGN KEY (car_id) REFERENCES Cars(car_id)  
);
```

-- Mechanics table

```
CREATE TABLE Mechanics (  
    mechanic_id INT PRIMARY KEY AUTO_INCREMENT,  
    name VARCHAR(255),  
    phone_number VARCHAR(15),  
    grade VARCHAR(50)  
);
```

-- Service_Mechanics table (to handle the many-to-many relationship)

```
CREATE TABLE Service_Mechanics (  
    service_id INT,  
    mechanic_id INT,  
    time_spent INT,  
    PRIMARY KEY (service_id, mechanic_id),  
    FOREIGN KEY (service_id) REFERENCES Services(service_id),  
    FOREIGN KEY (mechanic_id) REFERENCES Mechanics(mechanic_id)  
);
```

