**Титульный лист**

**Тема**

Разработка консольной системы учета автомобилей на парковке

Участники проекта:

Гузь Давид Иванович, группа [РПО 23/9/1]

Титов Данил Романович, группа [РПО 23/9/1]

Дата: 01 октября 2025 года

**Введение**

**Цель проекта**

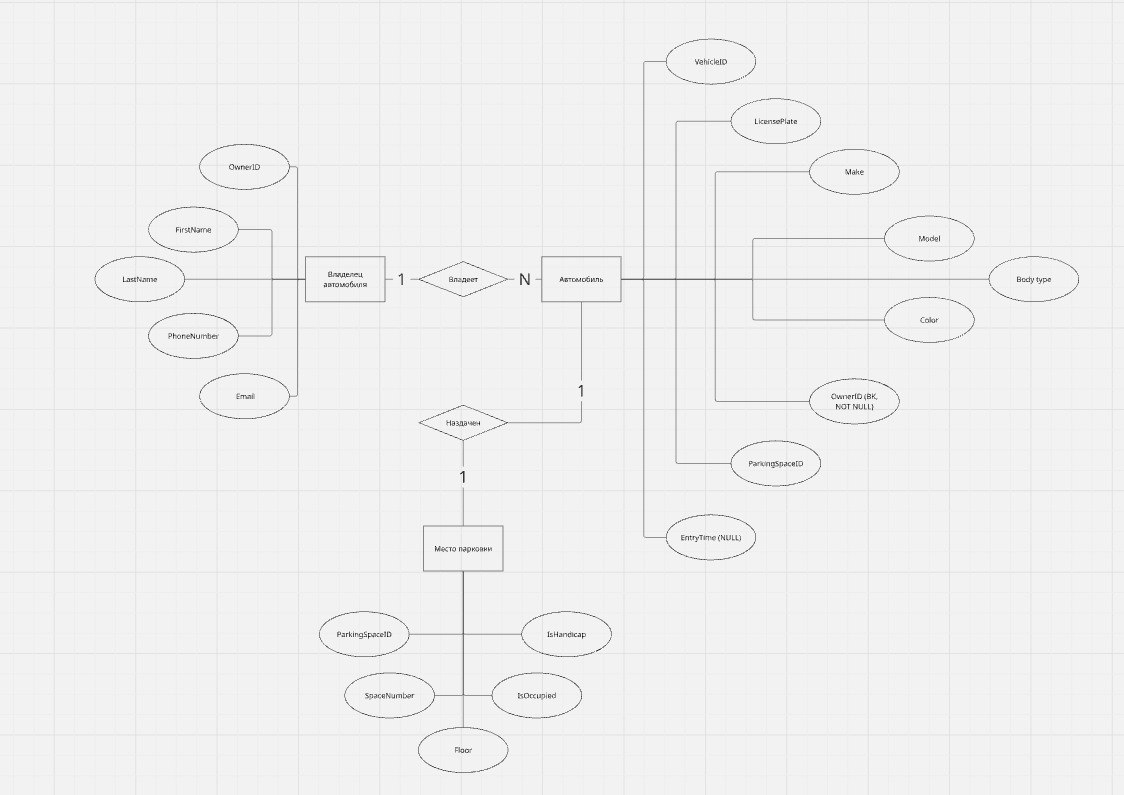
Разработать проект консольной системы учета автомобилей на парковке, позволяющий отображать и управлять данными о парковочных местах и автомобилях.

**Задачи проекта**

1. Разработать базу данных для хранения информации о парковке.
   1. Указать объекты:
      1. Место парковки
      2. Владелец автомобиля
      3. Автомобиль
   2. Описать таблицы в ER-диаграмме(написать атрибуты, типы данных, ключи, ограничения)
   3. Нормализовать БД до 3NF
   4. Создать таблицы, используя СУБД MS SQL
   5. Сделать диаграмму БД из SSMS
   6. Заполнить таблицы базы данных тестовыми данными.
2. Реализовать консольный интерфейс с функциями добавления, редактирования, удаления и просмотра данных, просмотр данных используя фильтрацию и сортировку
   1. Использовать ЯП C#
   2. Использовать БД технологию EF Core
   3. Использовать IDE Visual Studio
3. Провести тестирование программы для проверки корректности работы.
4. Подготовить отчет о результатах продельной работы.
5. Подготовить презентацию для защиты проекта.

**База данных**

**ER-диаграмма**



**Описание таблиц (атрибуты, типы данных, ключи, ограничения)**

**1. Таблица: Место парковки (ParkingSpace)**

**Атрибуты, типы данных и ограничения**

| **Атрибут** | **Тип данных** | **Описание** | **Ключи и ограничения** |
| --- | --- | --- | --- |
| ParkingSpaceID | INT | Уникальный идентификатор места парковки | Первичный ключ (PK), автоинкремент (IDENTITY(1,1)) |
| SpaceNumber | VARCHAR(10) | Номер парковочного места (например, "A1") | Уникальный (UNIQUE), NOT NULL |
| IsOccupied | BIT | Статус занятости места (0 = свободно, 1 = занято) | По умолчанию 0 (DEFAULT 0) |
| IsHandicap | BIT | Индикатор места для людей с ограниченными возможностями (0 = нет, 1 = да) | По умолчанию 0 (DEFAULT 0) |

#### Описание и обоснование

* **ParkingSpaceID**: Используется как уникальный идентификатор для каждого парковочного места. Автоинкремент упрощает добавление новых записей.
* **SpaceNumber**: Уникальный номер места, который должен быть задан при создании записи (NOT NULL) и не может повторяться (UNIQUE).
* **IsOccupied**: Бинарный атрибут для отслеживания занятости места, по умолчанию свободно (0).
* **IsHandicap**: Указывает, предназначено ли место для инвалидов, по умолчанию не предназначено (0).
* **Отсутствие внешних ключей**: Эта таблица является независимой и не ссылается на другие таблицы.

### 2. Таблица: Владелец автомобиля (VehicleOwner)

#### Атрибуты, типы данных и ограничения

| **Атрибут** | **Тип данных** | **Описание** | **Ключи и ограничения** |
| --- | --- | --- | --- |
| OwnerID | INT | Уникальный идентификатор владельца | Первичный ключ (PK), автоинкремент (IDENTITY(1,1)) |
| FirstName | VARCHAR(50) | Имя владельца | NOT NULL |
| LastName | VARCHAR(50) | Фамилия владельца | NOT NULL |
| PhoneNumber | VARCHAR(15) | Номер телефона владельца | Уникальный (UNIQUE), NOT NULL |
| Email | VARCHAR(100) | Электронная почта владельца | Уникальный (UNIQUE), NULL разрешен |

#### Описание и обоснование

* **OwnerID**: Уникальный идентификатор владельца, генерируется автоматически.
* **FirstName и LastName**: Обязательные поля (NOT NULL), так как имя и фамилия являются ключевыми данными владельца.
* **PhoneNumber**: Уникальный контактный номер, обязательный для связи с владельцем (NOT NULL, UNIQUE).
* **Email**: Необязательный атрибут (NULL разрешен), но если указан, должен быть уникальным для предотвращения дублирования.
* **Отсутствие внешних ключей**: Эта таблица независима, но используется как родительская для таблицы "Автомобиль".

### 3. Таблица: Автомобиль (Vehicle)

#### Атрибуты, типы данных и ограничения

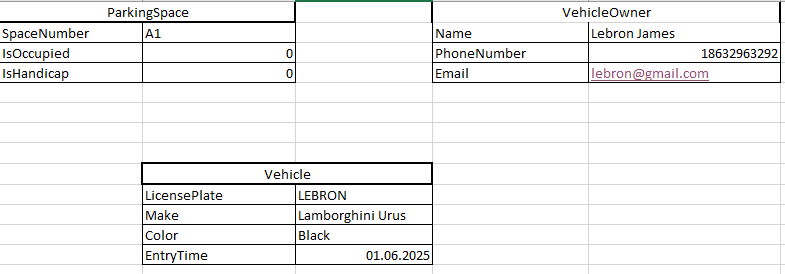
| **Атрибут** | **Тип данных** | **Описание** | **Ключи и ограничения** |
| --- | --- | --- | --- |
| VehicleID | INT | Уникальный идентификатор автомобиля | Первичный ключ (PK), автоинкремент (IDENTITY(1,1)) |
| LicensePlate | VARCHAR(20) | Номерной знак автомобиля | Уникальный (UNIQUE), NOT NULL |
| Make | VARCHAR(50) | Марка автомобиля (например, Toyota) | NOT NULL |
| Model | VARCHAR(50) | Модель автомобиля (например, Camry) | NOT NULL |
| Color | VARCHAR(30) | Цвет автомобиля | NOT NULL |
| OwnerID | INT | Идентификатор владельца | Внешний ключ (FK), ссылается на VehicleOwner(OwnerID), NOT NULL |
| ParkingSpaceID | INT | Идентификатор занятого парковочного места | Внешний ключ (FK), ссылается на ParkingSpace(ParkingSpaceID), NULL разрешен |
| EntryTime | DATETIME | Время въезда автомобиля на парковку | NULL разрешен |

#### Описание и обоснование

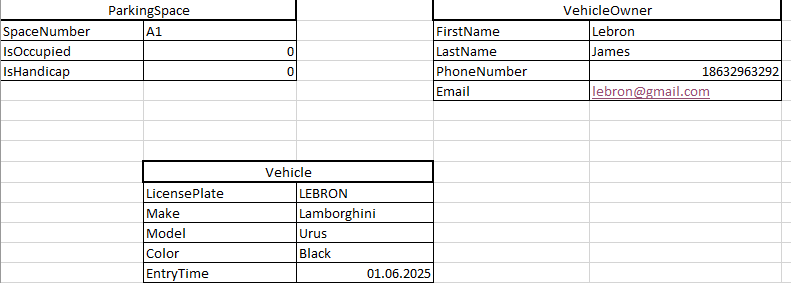
* **VehicleID**: Уникальный идентификатор автомобиля, генерируется автоматически.
* **LicensePlate**: Уникальный номерной знак, обязательный для идентификации автомобиля (NOT NULL, UNIQUE).
* **Make, Model, Color**: Обязательные атрибуты (NOT NULL), описывающие характеристики автомобиля.
* **OwnerID**: Внешний ключ, ссылающийся на OwnerID в таблице "Владелец автомобиля". Обязателен (NOT NULL), так как каждый автомобиль должен иметь владельца.
* **ParkingSpaceID**: Внешний ключ, ссылающийся на ParkingSpaceID в таблице "Место парковки". Может быть NULL, если автомобиль не припаркован.
* **EntryTime**: Время въезда, устанавливается при занятии места и может быть NULL, если автомобиль не на парковке.
* **Ограничения референциальной целостности**: При удалении владельца или места парковки необходимо определить поведение (например, CASCADE или RESTRICT), чтобы избежать "висячих" записей.

**Нормализация таблиц**

**Ненормализованная форма**



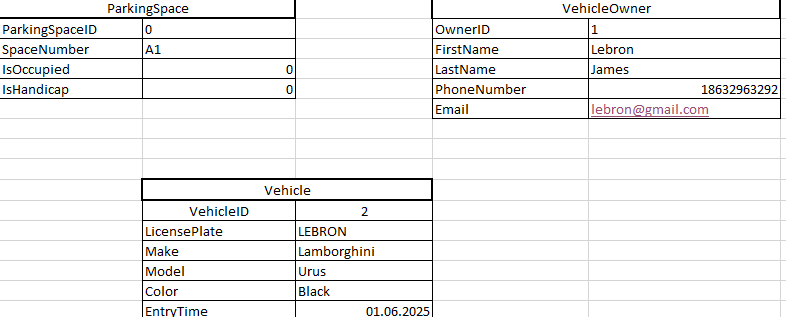
**Нормализация в 1NF**

****

**Что было изменено:**

* Были добавлены строки:
  + Таблица Vehicle:
    - Добавлена строка Model для удаления избыточности данных
  + Таблица VehicleOwner:
    - Удаление строки Name и добавление, заместо нее, строк FirstName и LastName для атомарности значений

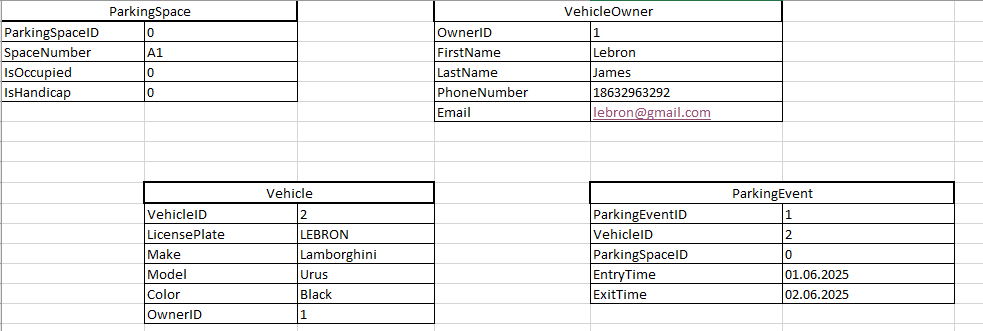
**Нормализация в 2NF**

****

**Что было изменено:**

* Были добавлены первичные ключи на каждую таблицу, чтобы неключевые поля зависели от всего ключа

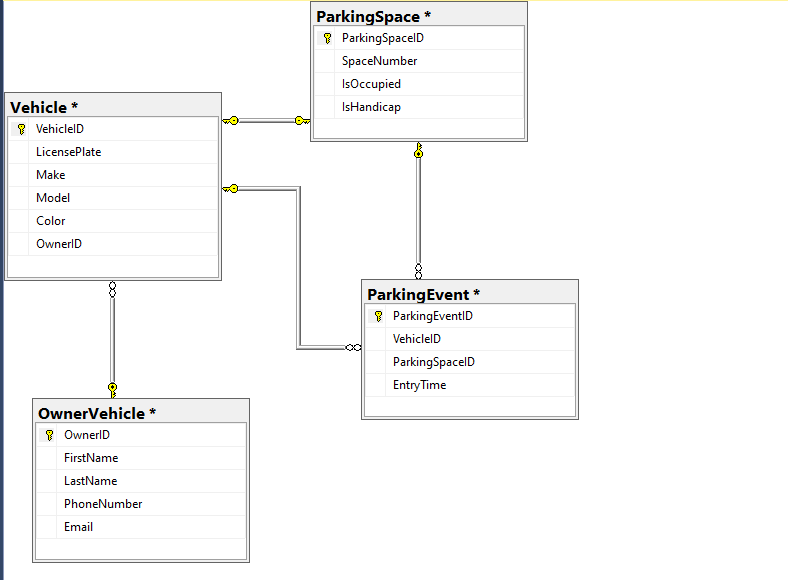
**Нормализация в 3NF**



**Что было изменено:**

* Была добавлена новая таблица из атрибутов других таблиц для:
  + Разделения постоянных и временных данных
    - В таблице Vehicle смешивались Make, Model(постоянные данные) и EntryTime(временные данные)
  + Удаления транзитивых зависимостей с помощью добавления внешних ключей

**Диаграмма БД в MS SQL**

****