**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**

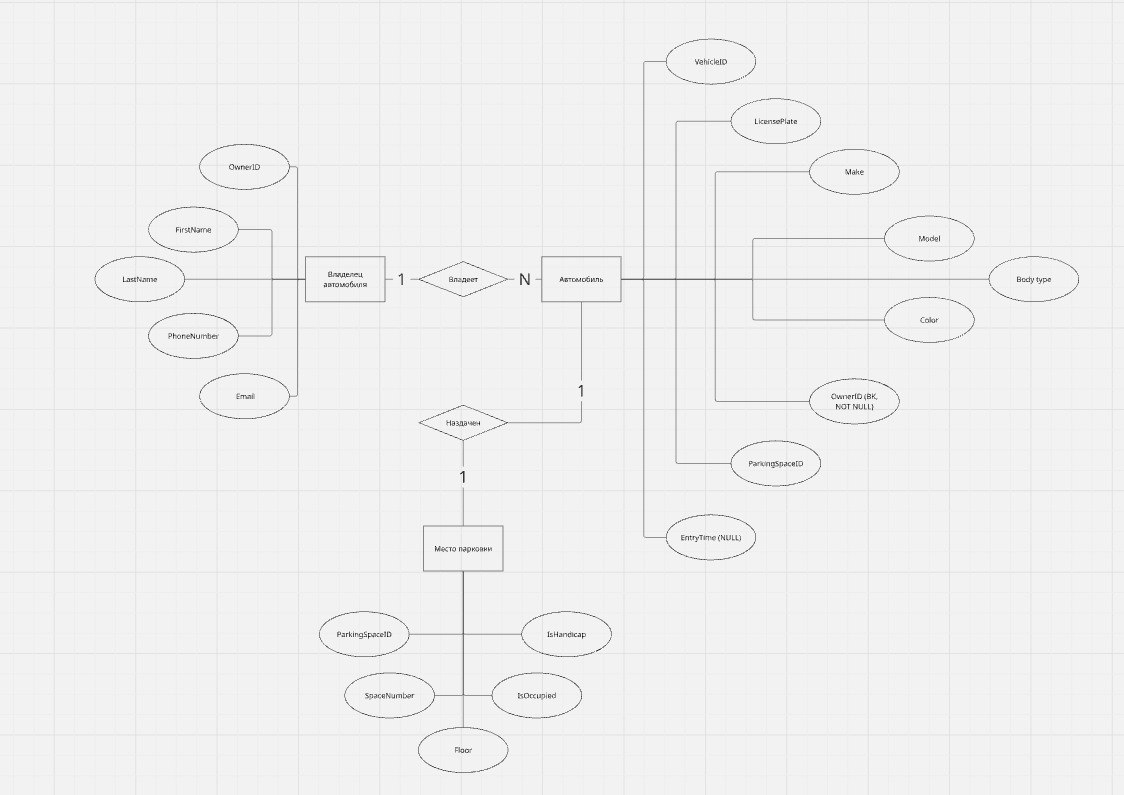
Разработать проект консольной системы учета автомобилей на парковке, позволяющий отображать и управлять данными о парковочных местах и автомобилях.

**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

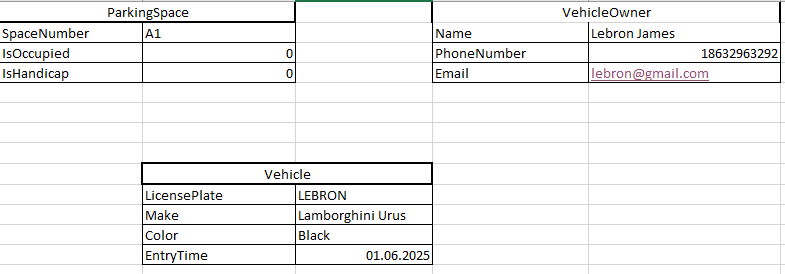
| **Атрибут** | **Тип данных** | **Описание** | **Ключи и ограничения** |
| --- | --- | --- | --- |
| VehicleID | INT | Уникальный идентификатор автомобиля | Первичный ключ (PK), автоинкремент (IDENTITY(1,1)) |
| LicensePlate | VARCHAR(20) | Номерной знак автомобиля | Уникальный (UNIQUE), NOT NULL |
| 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

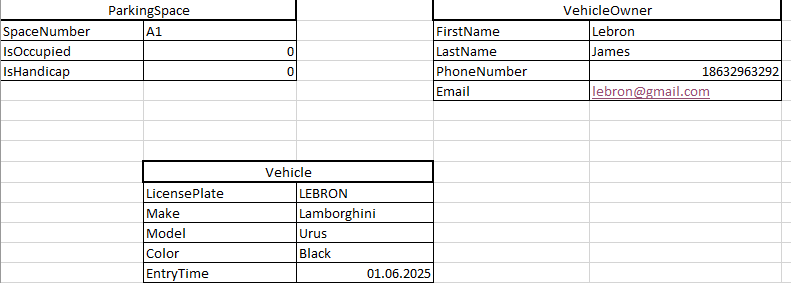
* **VehicleID**: Уникальный идентификатор автомобиля, генерируется автоматически.
* **LicensePlate**: Уникальный номерной знак, обязательный для идентификации автомобиля (NOT NULL, UNIQUE).
* **Make, Model, Color**: Обязательные атрибуты (NOT NULL), описывающие характеристики автомобиля.
* **OwnerID**: 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**



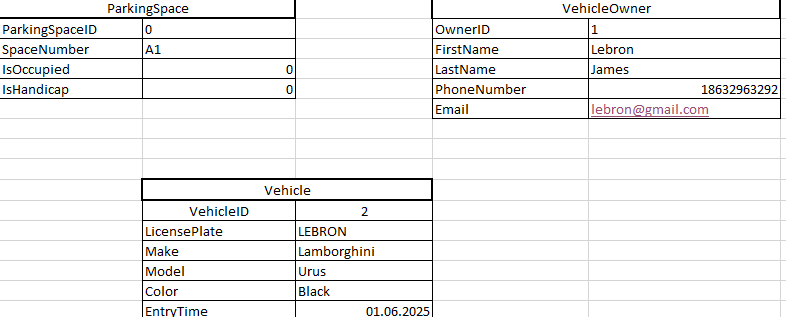
**Нормализация в 1NF**

****

**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

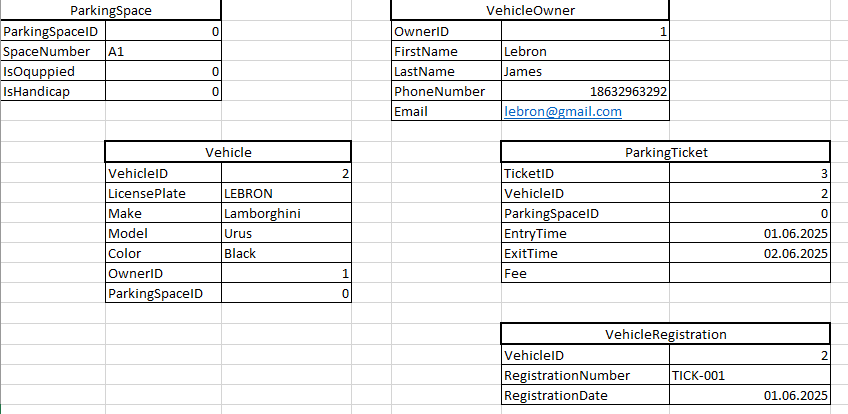
**Нормализация в 2NF**

****

**Что было изменено:**

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

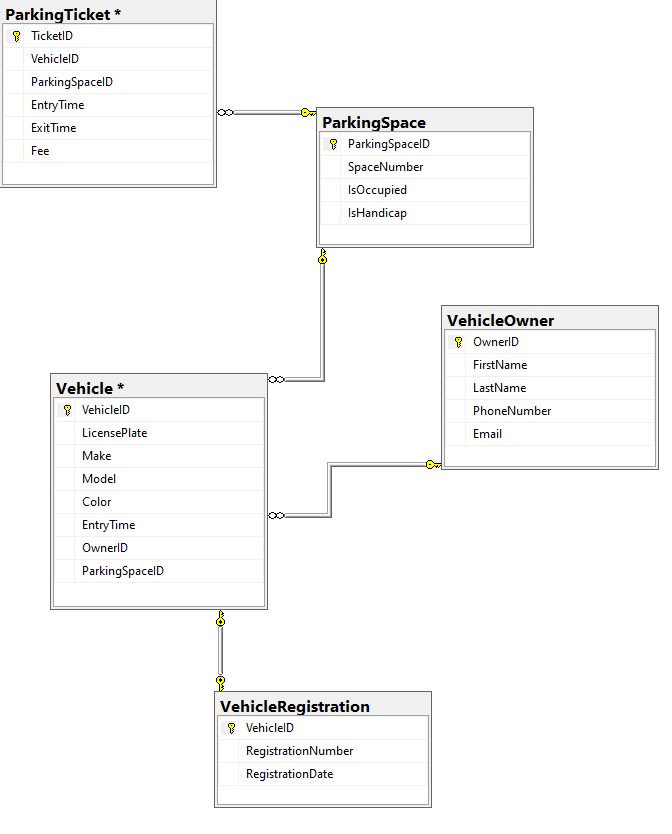
**Нормализация в 3NF**



**Что было изменено:**

* Была добавлена новая таблица ParkingTicket 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
* Добавлена таблица VehicleRegistration для добавления регистрации автомобиля

**Диаграмма БД в MS SQL**

****

**ДОКУМЕНТАЦИЯ**

**Скриншоты интерфейса**

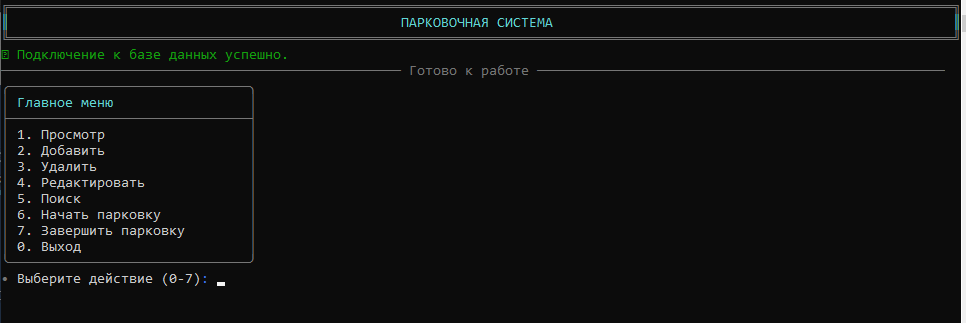


Рис. №1 Подключение к базе данных и выборка действий

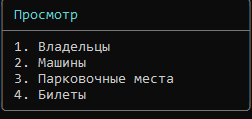


Рис №2 Просмотр данных

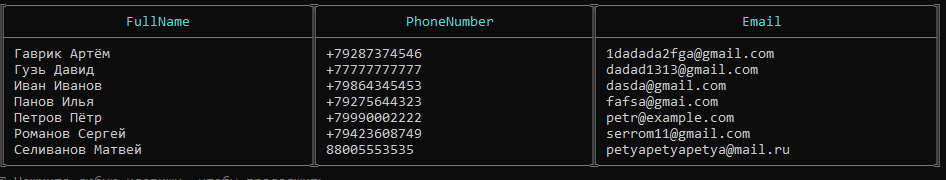


Рис №2.1 Просмотр владельццев

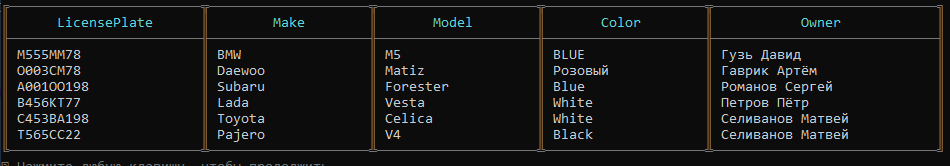


Рис №2.2 Просмотр автомобилей и их владельцев

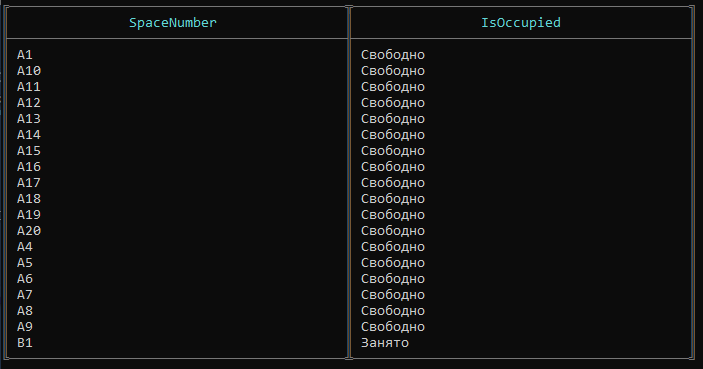


Рис №2.3 Просмотр статус занятости парковочных мест

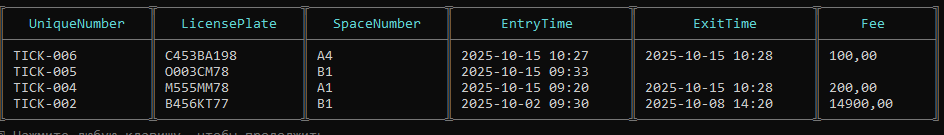


Рис №2.4 Просмотр билетов

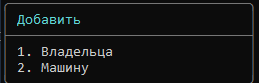
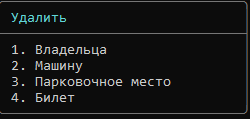
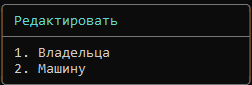


Рис. №3 Выборка добавления данных



Рис№4 Выборка удаления данных



Рис№5 Выборка редактирования данных

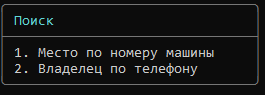


Рис №6 Поиск данных

**Test-cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **ID-теста** | **сценарий** | **предусловия** | **шаги** | **результат** |
| TC-001 | Проверка успешного добавления владельца | Отсутствует владелец с таким же ФИ | 1. Войти в БД 2. Выбрать опцию 2. 3. В появившемся окне выбрать опцию 1. 4.Ввести данные владельца(Имя, Фамилия, телефон, email) | Вывод: "Владелец добавлен!". Добавление владельца в таблицу VehicleOwner |
| ТС-002 | Проверка добавления владельца с фамилией/именем в 1 букву | База данных подключена | 1. Выбрать опцию 2. 2. В появившемся окне выбрать опцию 1. 3. Ввести фамилию/имя, состоящее из одной буквы. | Вывод: "Ошибка: фамилия должна иметь хотя бы 2 буквы" |
| ТС-003 | Проверка успешного добавления автомобиля | Существует настоящий владелец, у владельца нет машины с номером | 1. Выбрать опцию 2. 2. В появившемся окне выбрать опцию 2. 3. Ввести данные авто(госномер, марка, модель, цвет) и ФИ владельца | Вывод: "Машина добавлена", добавление записи в таблицу Vehicle. |
| ТС-004 | Проверка успешного добавления автомобиля с уже существующшим номером | Существует машина с номером Т242ОР16 | 1. Выбрать опцию 2. 2. В появившемся окне выбрать опцию 2. 3. Ввести данные авто(госномер, марка, модель, цвет) и ФИ владельца. Госномер должен совпадать с другим добавленным автомобилем | Вывод: "Ошибка: Машина с таким госномером уже существует", запись не добавляется |
| ТС-005 | Проверка редактирования владельца с неверным email. | Существует владелец | 1. Выбрать опцию 4. 2. В появившемся окне выбрать опцию 1. 3. Ввести новые данные владельца(ФИ, телефон, почта). Почта не должна содержать домена(например: @yandex.ru | Вывод: "Ошибка: неверный формат email." запись не редактируется |
| ТС-006 | Проверка редактирования данных автомобиля | Существует машина | 1. Выбрать опцию 4. 2. В появившемся окне выбрать опцию 2. 3. Ввести госномер авто. 4. Ввести новый цвет | Вывод: "данные машины обновлены", запись редактируется |
| ТС-007 | Проверка удаления владельца без авто | Существует владелец, не имеющий авто | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 1. 3. Ввести ФИ владельца без авто | Вывод: "Владелец удален.", запись удаляется из VehicleOwner |
| ТС-008 | Проверка удаления владельца, имеющий авто | Существует владелец, имеющий авто | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 1. 3. Ввести ФИ владельца, имеющий авто | Вывод: "Ошибка. Нельзя удалить владельца, у которого есть машины", запись не удаляется |
| ТС-010 | Проверка удаления машины без активных билетов | Существует авто без активного билета | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 2. 3. Ввести госномер авто без активного билета | Вывод: "Машина удалена", запись удаляется из таблицы Vehicle |
| ТС-011 | Проверка удаления машины с несуществующим номером | Не существует авто с госномером/не существует госномера | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 2. 3. Ввести несуществующий госномер | Вывод: "Ошибка: Машина с таким госномером не найдена", запись не редактируется |
| ТС-012 | Проверка удаления машины с активным билетом | Существует машина с активным билетом | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 2. 3. Ввести госномер с активным билетом | Вывод: "Ошибка: нельзя удалить машину с активным билетом". |
| ТС-013 | Проверка удаления свободного парковочного места | Существует свободное парковочное место | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 3. 3. Ввести номер свободного места. | Вывод: "Парковочное место удалено", запись удаляется из таблицы ParkingSpace |
| ТС-014 | Проверка удаления занятое место | Существует занятое место | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 3. 3. Ввести номер занятого места. | Вывод: "Ошибка: нельзя удалить занятое место", запись не удаляется |
| ТС-015 | Проверка удаления использованного билета | Существует билет, завершивший парковку | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 4. 3. Ввести номер билета, завершивший парковку | Вывод: "Билет удален", запись удаляется из таблицы ParkingTicket |
| ТС-016 | Проверка удаления активного билета | Существует активный билет | 1. Выбрать опцию 3. 2. В появившемся окне выбрать опцию 4. 3. Ввести номер активного билета | Вывод: "Ошибка: Нельзя удалить активный билет", запись не удаляется |
| ТС-017 | Проверка начала парковки с существующим госномером | Существует авто | 1. Выбрать опцию 6. 2. Ввести появившемся окне ввести существующий госномер авто | Вывод: "Парковка начата", выдается билет |
| ТС-018 | Проверка начала парковки с несуществующим госномером | Не существует госномера | 1. Выбрать опцию 6. 2. В появившемся окне ввести несуществующий госномер. | Вывод: "Ошибка: Машина с таким госномером не найдена", билет не выдается |
| ТС-019 | Проверка завершения парковки с активным билетом | Существует активный билет | 1. Выбрать опцию 7. 2. В появившемся окне ввести номера активного билета | Вывод: "Парковка завершена". Запись обновляется |
| ТС-020 | Проверка завершения парковки с неактивным билетом | Существует билет, завершивший парковку | 1. Выбрать опцию 7. 2. В появившемся окне ввести номер неактивного билета | Вывод: "Ошибка: билет уже закрыт". Запись не редактируется |
| ТС-021 | Проверка завершения парковки с несуществующим билетом | Билет не существует | 1. Выбрать опцию 7. 2. В появившемся окне ввести номер несуществующего билета | Вывод: "Ошибка: билет не найден". Запись не редактируется |
| ТС-022 | Проверка отображения всех владельцев | Существуют владельцы | 1. Выбрать опцию 1. 2. В появившемся окне выбрать опцию 1. | Вывод таблицы с владельцами(столбцы: FullName, PhoneNumber, Email) |
| ТС-023 | Проверка отображения всех автомобилей | Существуют автомобили | 1. Выбрать опцию 1. 2. В появившемся окне выбрать опцию 2. | Вывод таблицы с автомобилями и их владельцами(столбцы: LicensePlate, Make, Model, Color, Owner) |
| ТС-024 | Проверка отображения парковочных мест | Существуют парковочные места | 1. Выбрать опцию 1. 2. В появившемся окне выбрать опцию 3. | Вывод таблицы с парковочными местами(столбцы: SpaceNumber, IsOccupied) |
| ТС-025 | Проверка отображения билетов | Существуют билеты | 1. Выбрать опцию 1. 2. В появившемся окне выбрать опцию 4. | Вывод таблицы с билетами(столбцы: UniqueNumber, LicensePlate, SpaceNumber, EntryTime, ExitTime, Fee) |
| ТС-026 | Проверка поиска места по госномеру машины | Существует активная парковка | 1. Выбрать опцию 5. 2. В появившемся окне выбрать опцию 1. 3. Ввести госномер авто с активным билетом | Вывод таблицы с активным билетом(столбцы: SpaceNumber, IsOccupied, TicketNumber, EntryTime, ExitTime) |
| ТС-027 | Проверка поиска места по несуществующему госномеру | Госномера авто не существует | 1. Выбрать опцию 5. 2. В появившемся окне выбрать опцию 1. 3. Ввести несуществующий госномер авто | Вывод: "Не найдено активных парковок для этого номера". |
| ТС-028 | Проверка поиска владельца по телефону | Существует владелец | 1. Выбрать опцию 5. В появившемся окне выбрать опцию 2. 3. Ввести телефон владельца | Вывод таблицы с владельцем(столбцы: FullName, PhoneNumber, Email) |
| ТС-029 | Проверка поиска владельца по несуществующему телефону | Телефона не существует | 1. Выбрать опцию 5. В появившемся окне выбрать опцию 2. 3. Ввести несуществующий телефон владельца | Вывод: "Владелец с таким телефоном не найден" |