**Разработка программных модулей Windows Forms**

Отчёт по лабораторной работе

Студента группы ИСП9-21-2

Щербакова Тимофея Андреевича

Тема**: Разработка консольной программы Парковка машин**

**Цель работы**: разработка программы для учета автомобилей, парковок

**Задачи**:

1. Созданием функциональности и меню в консоли

2. Разработка классов Машины и Парковки

3. Написание функций

**Разработка приложения**

Класс Program – основной класс проекта, делающий инициализацию программы.

На рисунке 1 показан старт программы, а также главные переменные программы.

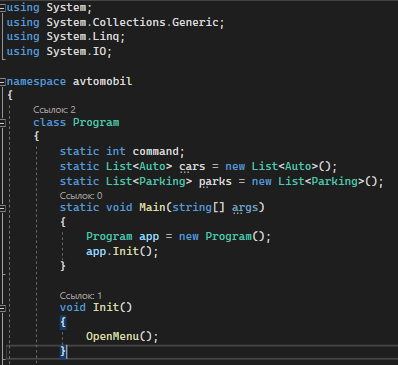


Рисунок 1 - Старт программы и переменные

Функция OpenMenu используется в старте, и выписывает список доступных действий пользователю. Она показана на рисунке 2.

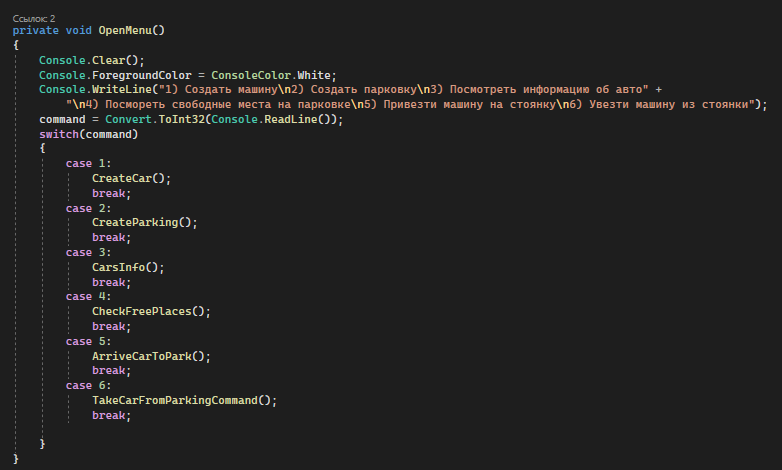


Рисунок 2 – Меню

В проекте также находятся другие классы, без которых проект не будет работать. Первый класс Datecounter. Он позволяет получить локальную дату пользователя. Класс показан на рисунке 3.

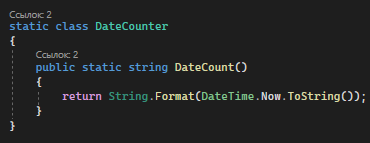


Рисунок 3 - Класс DateCounter

Следующий класс – класс Auto. Класс содержит все атрибуты автомобиля, а так же на рисунке 4 показана инициализация класса.

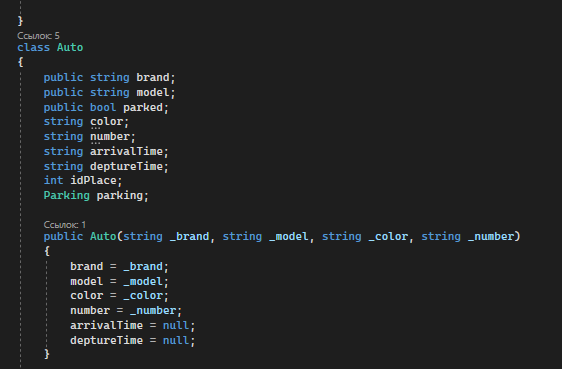


Рисунок 4 - Переменные и инициализация класса Auto

Данный класс также содержит четыре функции, позволяющие получить информацию об определенном авто, припарковать машину, забрать машину с парковки, и получить информацию о парковке, на которой находится определенная машина.

На рисунке 5 показаны данные функции.

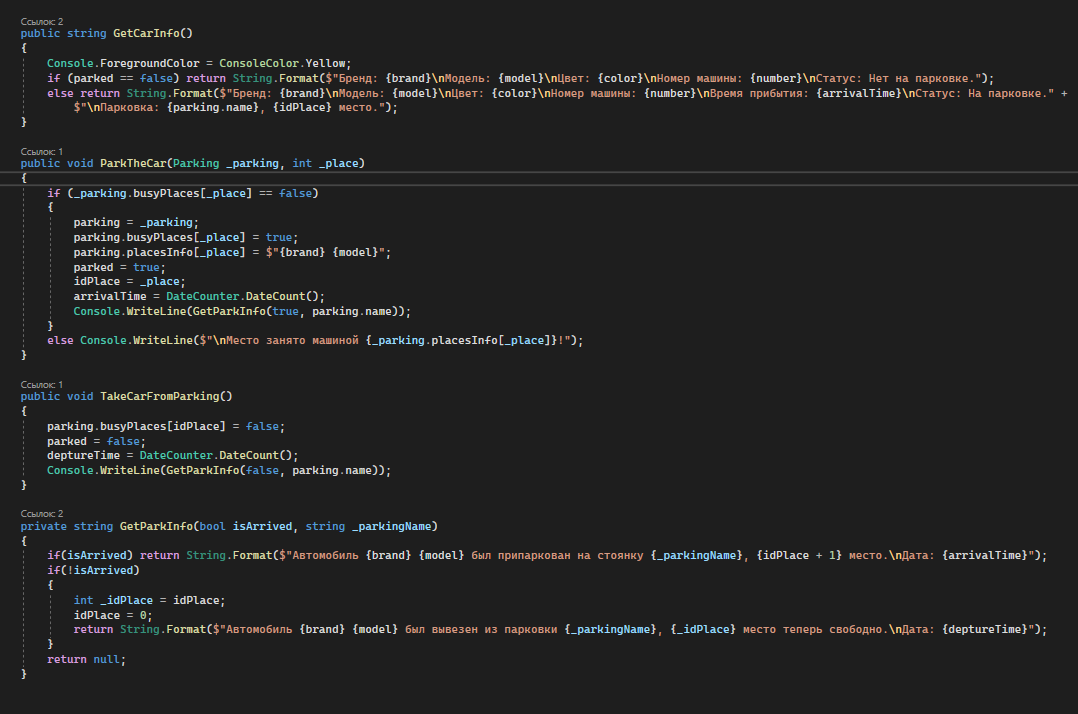


Рисунок 5 - Функции класса Auto

Класс имеет переменную другого класса, под названием Parking. Данная переменная используется в функциях класса Auto, для получения или изменения информации о парковке, на которой расположена определенная машина.

Класс Parking показан на рисунке 6.

Класс содержит одну функцию, для получения информации о парковке, такую как статус, и какая машина припаркована на парковке.

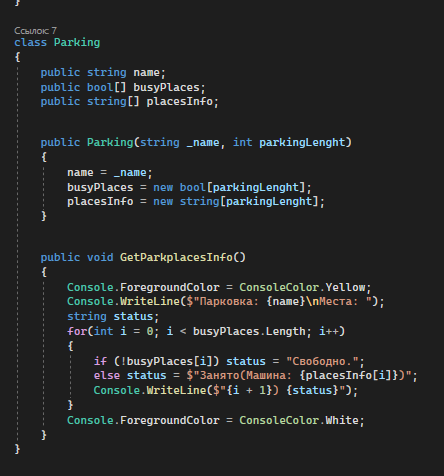


Рисунок 6 - Класс Parking

После создания данных классов, необходимо их взаимодействие. Функция класса Program CheckCorrectAnswer проверяет текст на корректность. Она используется в функциях для добавления какой либо информации. Функция показана на рисунке 7.

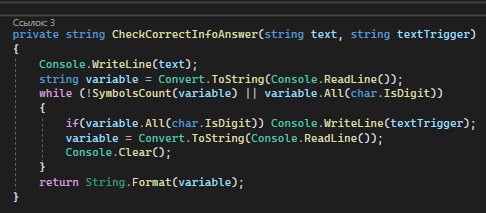


Рисунок 7 - Функция для проверки текста на корректность

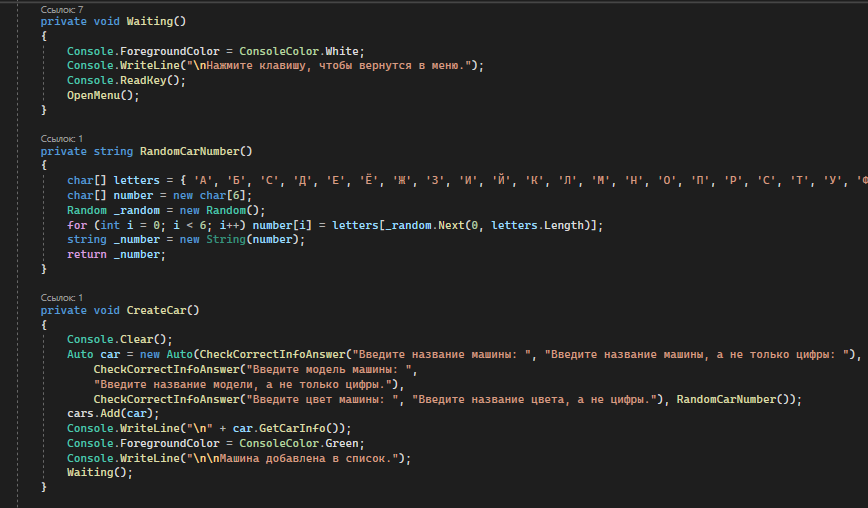


Рисунок 8 - Функции класса Program

На рисунке 8 показаны 3 функции. Функция Waiting используется как ожидание, после выбор какой-либо функции или окончания действия.

Функция RandomCarNumber создает строку из случайных символов, для присваивания данной строки номеру машины.

Функция CreateCar создает объект класса Auto,т.е. машину. При создании необходимо ввести название машины, модели, цвет, введенный текст будет проверятся с помощью функции CheckCorrectAnswer (см. рисунок 7). Номер создается с помощью функции выше. Машина создается, и добавляется в список.

Для создания объекта класса Parking, т.е. парковки, создана функция CreateParking. Данная функция проверяет текст на количество символов, и при верных условиях ввода создается парковка, и добавляется в список парковок. Функция показана на рисунке 9.

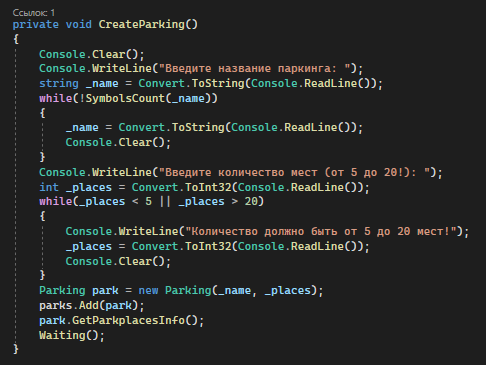


Рисунок 9 - Функция для создания парковки

Функция ObjectList выводит список машин, или парковок для чтения информации по ним.

Функция показана на рисунке 10.



Рисунок 10 - Функция для чтения списков машин или парковок

На рисунке 11 показаны последние четыре функции класса Program.



Рисунок 11 - Функции класса Program

Первая функция, CarsInfo, выводит список машин, а также предлагает сделать выбор номера из списка, для просмотра подробной информации.

Функция checkFreePlaces, выводит список парковок, и выводит информацию по ним. Занято какое-либо место на парковке, или нет.

Функция ArriveCarToPark позволяет припарковать машину на любое парковочное место, где есть свободное место.

Функция TakeCarFromParking позволяет забрать машину со стоянки. Она выводит информацию о машинах, и позволяет выбрать машину, для просмотра подробной информации.

Для всех данных функции необходимо вводить число индекса элемента списка, для выбора.