# Практические задания

Тема: Классы и методы

Класс **Person**

Создайте публичный класс Person – владелец ТС. Класс характеризуется именем и фамилией (строки). Конструкторы:

− принимающий два параметра

– имя и фамилию

Общедоступная константа класса:

− безымянная персона (характеризуется пустыми строками в качестве имени и фамилии).

Методы:

− возвращающий значение имени

− возвращающий значение фамилии

Создайте несколько экземпляров класса. Протестируйте работу конструкторов и методов класса.

Класс **Vehicle**

Создайте публичный класс Vehicle – транспортное средство Класс характеризуется гос. номером, производителем и моделью (строки)

Конструкторы:

− по-умолчанию, инициализирует все три параметра пустыми строками

− принимающий три параметра – гос. номер, производитель и модель

Методы:

− возвращающий значение гос. номера

− устанавливающий новое значение гос. номера

− возвращающий значение производителя

− устанавливающий новое значение производителя

− возвращающий значение марки

− устанавливающий новое значение марки

Создайте публичный класс Vehicle – транспортное средство Класс характеризуется гос. номером, производителем и моделью (строки)

Класс **Space**

Создайте публичный класс Space – парковочное место. Класс содержит ссылки на владельца (Person) и ТС (Vehicle).

Конструкторы:

− по-умолчанию, инициализирует атрибуты как безымянная персона, и ТС с параметрами – пустыми строками

− принимающий два параметра – ссылки на ТС и владельца

Методы:

− возвращающий ссылку на владельца

− устанавливающий новую ссылку на владельца

− возвращающий ссылку на ТС

− устанавливающий новую ссылку на ТС

− определяющий является ли это место пустым (ссылка на Vehicle – пустая, или гос. номер у ТС – пустая строка)

Создайте несколько экземпляров класса. Протестируйте работу конструкторов и методов класса.

Класс **OwnersSpacer (дополнительно, по желанию)**

Создайте публичный класс OwnersSpace – парковки, на которой все парковочные места находятся в собственности частных лиц. Этот класс характеризуется массивом парковочных мест.

**Логика работы класса**:

Емкость – это размер массива. Число элементов – это текущее число элементов, имеющих ссылки на экземпляр класса Space (их может быть меньше, чем размер массива).

При добавлении парковочного места, ссылка на его экземпляр записывается в первый свободный (имеющий значение null) элемент массива, затем следующий и т.д. Если при попытке добавить элемент выясняется, что размер массива этого не позволяет (в массиве все элементы имеют значение), размер массива удваивается. Т.е. создается новый массив, имеющий емкость в два раза большую, чем текущий, и, начиная с 0-го, в него копируются элементы текущего массива. После чего ссылка на текущий массив заменяется ссылкой на созданный.

При удалении парковочного места емкость массива не меняется. Все элементы с 0-го до удаляемого (не включительно) остаются без изменений. А элементы, начиная со следующего, после удаляемого, копируются в предыдущий элемент. Последний, имеющий значение не null после копирования заменяется значением null. При этом в класс добавляется приватный атрибут, который хранит текущее число элементов. При добавлении элемента его значение – увеличивается, при удалении – уменьшается.

Конструкторы:

− по-умолчанию, инициирующий массив из 16 элементов (сами элементы имеют значение null)

− принимающий целое число

– емкость массива, инициирующий массив указанным числом элементов (сами элементы имеют значение null)

− принимающий массив парковочных мест. В этом конструкторе происходит копирование элементов в новый массив, и ссылка на него записывается в атрибут.

Методы:

− добавляющий парковочное место в конец списка (принимает ссылку на экземпляр класса Space). Пока этот метод возвращает истину после выполнения операции добавления элемента.

− добавляющий парковочное место в заданное место (принимает номер и ссылку на экземпляр класса Space). Пока этот метод возвращает истину после выполнения операции добавления элемента

− возвращающий ссылку на экземпляр класса Space по его номеру в массиве

− возвращающий ссылку на экземпляр класса Space, с которым связанно транспортное средство с определенным гос. номером. В качестве параметра принимает строку – гос. номер.

− определяющий, есть ли на этаже парковочное место, связанное с транспортным средством с определенным гос. номером. В качестве параметра принимает строку – гос. номер. Возвращает логическое значение

− изменяющий ссылку на экземпляр класса Space по его номеру в массиве. Принимает в качестве параметров номер и ссылку на экземпляр класса Space. Возвращает ссылку, которую заменили

− удаляющий парковочное место из массива по его номеру (принимает номер в качестве параметра). Возвращает удаленную из массива ссылку на экземпляр класса Space

− удаляющий парковочное место из массива по гос. номеру автомобиля (принимает гос. номер в качестве параметра). Возвращает удаленную из массива ссылку на экземпляр класса Space

− возвращающий общее число парковочных мест на этаже

− возвращающий массив парковочных мест на этаже (значений null в массиве быть не должно, его размер должен быть равен числу элементов в исходном массиве).

− возвращающий массив транспортных средств на этаже (значений null в массиве быть не должно, его размер должен быть равен числу элементов в массиве).

Создайте несколько экземпляров класса. Протестируйте работу конструкторов и методов класса.