

Лабораторная работа №5

Объектно-Оrientированное Программирование

Lab 5.1 Создайте класс Person, у которого есть:

1. Метод `__init__`, принимающий 3 аргумента: name, surname, gender. Атрибут gender может принимать только 2 значения: "male" и "female", по умолчанию "male". Если в атрибут gender передается любое другое значение, печатать сообщение: "Значение не передано." и проставить атрибут gender значением "male"
2. Магический метод `__str__` определить следующим образом:
 - о если объект - мужчина (атрибут gender = "male"), возвращать строку "Гражданин: <Фамилия> <Имя>"
 - о если объект - женщина (атрибут gender = "female"), возвращать строку "Гражданка: <Фамилия> <Имя>"

```
p1 = Person('Maddison', 'Cynthia', 'female')
print(p1) # печатает "Гражданка Cynthia Maddison"
p2 = Person('Zoe', 'Amber', 'female')
print(p2) # печатает "Гражданка Amber Zoe"
p3 = Person('Oscar', 'Smith', True) # печатает "Значение не передано."
print(p3) # печатает "Гражданин Smith Oscar"
```

Lab 5.2 Создайте класс Vector. У класса Vector есть:

1. конструктор `__init__`, принимающий произвольное количество аргументов. Среди всех переданных аргументов необходимо оставить только целые числа и сохранить их в атрибут values в виде списка;
2. переопределить метод `__str__` так, чтобы экземпляр класса Vector выводился следующим образом:
 - о "Вектор(<value1>, <value2>, <value3>, ...)", если вектор не пустой. При этом значения должны быть упорядочены по возрастанию;
 - о "Пустой вектор", если наш вектор не хранит в себе значения

```
v1 = Vector(1, 12, 32, 'abc')
print(v1) # печатает "Вектор(1, 12, 32)"
v2 = Vector()
print(v2) # печатает "Пустой вектор"
```

Lab 5.3 Создайте класс City, у которого есть:

1. конструктор `__init__`, принимающий единственный аргумент - название города. Вам необходимо сохранить его в качестве атрибута экземпляра name, причем вам нужно преобразовать переданное имя города таким образом, чтобы первая буква каждого слова была заглавной, а остальные оказались строчными (пример "new york" -> "New York")
2. переопределить метод `__str__` таким образом, чтобы он возвращал имя города
3. переопределить метод `__bool__` так, чтобы он возвращал False, если название города заканчивается на любую гласную букву латинского алфавита (a, e, i, o, u), в противном случае True

```
p1 = City('new york')
print(p1) # печатает "New York"
print(bool(p1)) # печатает "True"
p2 = City('SaN frANCISco')
print(p2) # печатает "San Francisco"
print(p2 == True) # печатает "False"
```

Lab 5.4 Создать класс Quadrilateral(четыреугольник), в котором есть:

1. конструктор `__init__`. Он должен сохранять в экземпляр класса два атрибута: `width` и `height`. При этом в сам метод `__init__` может передаваться один аргумент(тогда в `width` и `height` присваивать это одно одинаковое значение, тем самым делать квадрат), либо два аргумента(первый идет в атрибут `width`, второй - в `height`)
2. переопределить метод `__str__` следующим образом:
 - о если `width` и `height` одинаковые, возвращать строку «Квадрат размером `<width>*<height>`»
 - о в противном случае, возвращать строку «Прямоугольник размером `<width>*<height>`»
3. переопределить метод `__bool__` так, чтобы он возвращал `True`, если объект является квадратом, и `False` в противном случае

```
q1 = Quadrilateral(10)
print(q1) # печатает "Квадрат размером 10*10"
print(bool(q1)) # печатает "True"
q2 = Quadrilateral(3, 5)
print(q2) # печатает "Прямоугольник размером 3*5"
print(q2 == True) # печатает "False"
```