



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ
КАФЕДРА СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

Отчет по лабораторной работе №2

Студент Вардумян Арсен Тигранович
фамилия, имя, отчество
Группа ИУ5-51Б

Студент 18.12.2021 Вардумян А.Т.
подпись, дата фамилия, и.о.
Преподаватель 18.12.2021 Гапанюк Ю.Е.
подпись, дата фамилия, и.о.

Описание задания:

1. Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.
2. Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
3. Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.
4. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.
5. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).
7. Класс «Прямоугольник» наследуется от класса «Геометрическая фигура». Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета. Класс должен переопределять метод, вычисляющий площадь фигуры.
8. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля [math](#).
9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - Определите метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format` - <https://pyformat.info/>
 - Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
10. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов (используйте следующую конструкцию

- https://docs.python.org/3/library/_main_.html). Создайте следующие объекты и выведите о них информацию в консоль (N - номер Вашего варианта по списку группы):

- Прямоугольник синего цвета шириной N и высотой N.
- Круг зеленого цвета радиусом N.
- Квадрат красного цвета со стороной N.
- Также вызовите один из методов внешнего пакета, установленного с использованием pip.

Текст программы:

“circle.py”:

```
from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor
import math

class Circle(Figure):
    FIGURE_TYPE = "Круг"

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __init__(self, colorParam, rParam):
        self.r = rParam
        self.figureColor = FigureColor()
        self.figureColor.colorproperty = colorParam

    def square(self):
        return math.pi * (self.r ** 2)

    def __repr__(self):
        return '{} {} цвета радиусом {} площадью {}'.format(
            Circle.get_figure_type(),
            self.figureColor.colorproperty,
            self.r,
            self.square()
        )
```

“color.py”:

```
class FigureColor:
    def __init__(self):
        self._color = None

    @property
    def colorproperty(self):
```

```
return self._color
```

```
@colorproperty.setter  
def colorproperty(self, value):  
    self._color = value
```

“figure.py”:

```
from abc import ABC, abstractmethod
```

```
class Figure(ABC):  
    @abstractmethod  
    def square(self):  
        pass
```

“rectangle.py”:

```
from lab_python_oop.figure import Figure  
from lab_python_oop.color import FigureColor
```

```
class Rectangle(Figure):  
    FIGURE_TYPE = "Прямоугольник"  
  
    @classmethod  
    def get_figure_type(cls):  
        return cls.FIGURE_TYPE  
  
    def __init__(self, colorParam, widthParam, heightParam):  
        self.width = widthParam  
        self.height = heightParam  
        self.figureColor = FigureColor()  
        self.figureColor.colorproperty = colorParam  
  
    def square(self):  
        return self.width * self.height  
  
    def __repr__(self):  
        return '{} {} цвета шириной {} и высотой {} площадью {}.'.format(  
            Rectangle.get_figure_type(),  
            self.figureColor.colorproperty,  
            self.width,  
            self.height,  
            self.square()  
        )
```

“square.py”:

```
from lab_python_oop.rectangle import Rectangle
```

```

class Square(Rectangle):
    FIGURE_TYPE = "Квадрат"

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __init__(self, colorParam, sideParam):
        self.side = sideParam
        super().__init__(colorParam, self.side, self.side)

    def __repr__(self):
        return '{} {} цвета со стороной {} площадью {}'.format(
            Square.get_figure_type(),
            self.figureColor.colorproperty,
            self.side,
            self.square()
        )

```

“main.py”:

```

from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
import requests as req

```

```

def main():
    n = 7
    r = Rectangle("синего", n, n)
    c = Circle("зеленого", n)
    s = Square("красного", n)
    print(r)
    print(c)
    print(s)

    r = req.get('https://api.github.com/events')
    print(r.status_code)

```

```

if __name__ == "__main__":
    main()

```

“requirements.txt”:

```

certifi==2021.5.30
charset-normalizer==2.0.6
idna==3.2
requests==2.26.0
urllib3==1.26.6

```

Экранные формы с примерами выполнения программы:

Пример 1:

```
def main():  
    n = 7  
    r = Rectangle("синего", n, n)  
    c = Circle("зеленого", n)  
    s = Square("красного", n)  
    print(r)  
    print(c)  
    print(s)  
  
    r = req.get('https://api.github.com/events')  
    print(r.status_code)
```

```
(venv_hm2) arsenvardumyan@MacBook-Pro-Arsen hm2 % python main.py  
Прямоугольник синего цвета шириной 7 и высотой 7 площадью 49.  
Круг зеленого цвета радиусом 7 площадью 153.93804002589985.  
Квадрат красного цвета со стороной 7 площадью 49.  
200
```

Пример 2:

```
def main():  
    n = 3  
    r = Rectangle("синего", n, n)  
    c = Circle("черного", n)  
    s = Square("красного", n)  
    print(r)  
    print(c)  
    print(s)  
  
    r = req.get('https://api.github.com/events')  
    print(r.status_code)
```

```
(venv_hm2) arsenvardumyan@MacBook-Pro-Arsen hm2 % python main.py  
Прямоугольник синего цвета шириной 3 и высотой 3 площадью 9.  
Круг черного цвета радиусом 3 площадью 28.274333882308138.  
Квадрат красного цвета со стороной 3 площадью 9.  
200
```