

**Московский государственный технический  
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»  
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе № 2  
«Объектно-ориентированные возможности языка Python»

Выполнил:  
студент группы ИУ5-34Б  
Бромберг Е.А.

Проверил:  
преподаватель каф. ИУ5  
Нардид А.Н.

Москва, 2024 г.

## Постановка задачи

### Задание:

- Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.
- Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
- Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.
- Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.
- Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
- Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
- 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.
- Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля `math`.
- Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
  - Определите метод "герг", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format`.
  - Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
- В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов (используйте следующую конструкцию - [https://docs.python.org/3/library/\\_main\\_.html](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

main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
import numpy as np

def main():
    r = Rectangle("синего", 3, 3)
    c = Circle("зеленого", 3)
    s = Square("красного", 3)
    arr = np.array([r, c, s])
    print(*arr, sep='\n')
    print()
    print("Вот и numpy:")
    arr2 = np.array([Rectangle("фиолетового", 3, 4), Rectangle("красного", 5,
7)])
    arr3 = np.array([Square("зеленого", 9), Circle("черного", 2)])
    big_arr = np.concatenate((arr, arr2, arr3))
    print(*big_arr, sep='\n')

if __name__ == "__main__":
    main()
```

circle.py

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

class Circle(Figure):
    """
    Класс «Круг» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Круг"

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

    def __init__(self, color_param, r_param):
        """
        Класс должен содержать конструктор по параметрам «радиус» и «цвет». В
        конструкторе создается объект класса «Цвет фигуры» для хранения цвета.
        """
        self.r = r_param
        self.fc = FigureColor()
        self.fc.colorproperty = color_param

    def square(self):
        """
        Класс должен переопределять метод, вычисляющий площадь фигуры.
        """
        return math.pi*(self.r**2)

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

### color.py

```
class FigureColor:
    """
    Класс «Цвет фигуры»
    """

    def __init__(self):
        self._color = None

    @property
    def colorproperty(self):
        """
        Get-аксесор
        """
        return self._color

    @colorproperty.setter
    def colorproperty(self, value):
        """
        Set-аксесор
        """
        self._color = value
```

### figure.py

```
from abc import ABC, abstractmethod

class Figure(ABC):
    """
    Абстрактный класс «Геометрическая фигура»
    """

    @abstractmethod
    def square(self):
        """
        содержит виртуальный метод для вычисления площади фигуры.
        """
        pass
```

### rectangle.py

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

class Rectangle(Figure):
    """
    Класс «Прямоугольник» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Прямоугольник"

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

    def __init__(self, color_param, width_param, height_param):
        """
        Класс должен содержать конструктор по параметрам «ширина», «высота» и
        «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения
        цвета.
        """
        self.width = width_param
        self.height = height_param
        self.fc = FigureColor()
```

```

        self.fc.colorproperty = color_param

    def square(self):
        """
        Класс должен переопределять метод, вычисляющий площадь фигуры.
        """
        return self.width*self.height

    def __repr__(self):
        return '{} {} цвета шириной {} и высотой {} площадью {}.'.format(
            Rectangle.get_figure_type(),
            self.fc.colorproperty,
            self.width,
            self.height,
            self.square()
        )

```

#### square.py

```

from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):
    """
    Класс «Квадрат» наследуется от класса «Прямоугольник».
    """
    FIGURE_TYPE = "Квадрат"


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

    def __init__(self, color_param, side_param):
        """
        Класс должен содержать конструктор по параметрам «сторона» и «цвет».
        """
        self.side = side_param
        super().__init__(color_param, self.side, self.side)


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

```

## Экранные формы с примерами выполнения программы

 Командная строка


```
C:\Users\user\PycharmProjects>.\lab_python_oop\Scripts\activate
(lab_python_oop) C:\Users\user\PycharmProjects>cd lab_python_oop
```

 Командная строка

```
(lab_python_oop) C:\Users\user\PycharmProjects\lab_python_oop>python main.py
Прямоугольник синего цвета шириной 3 и высотой 3 площадью 9.
Круг зеленого цвета радиусом 3 площадью 28.274333882308138.
Квадрат красного цвета со стороной 3 площадью 9.

Вот и numpу:
Прямоугольник синего цвета шириной 3 и высотой 3 площадью 9.
Круг зеленого цвета радиусом 3 площадью 28.274333882308138.
Квадрат красного цвета со стороной 3 площадью 9.
Прямоугольник фиолетового цвета шириной 3 и высотой 4 площадью 12.
Прямоугольник красного цвета шириной 5 и высотой 7 площадью 35.
Квадрат зеленого цвета со стороной 9 площадью 81.
Круг черного цвета радиусом 2 площадью 12.566370614359172.

(lab_python_oop) C:\Users\user\PycharmProjects\lab_python_oop>
```

 Командная строка

```
(lab_python_oop) C:\Users\user\PycharmProjects\lab_python_oop>deactivate
C:\Users\user\PycharmProjects\lab_python_oop>
```