



**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 «Радиотехнический»  
Кафедра «Системы обработки информации и управления»**

**Лабораторная работа № 2**  
**по дисциплине «Разработка интернет-приложений».**

Выполнил:  
студент(ка) группы № РТ5-51Б  
А. С. Пакало  
подпись, дата

Проверил:  
преподаватель  
Ю. Е. Гапанюк  
подпись, дата

# Оглавление

Цель работы .....	2
Задание .....	2
Выполнение .....	3
main.py .....	3
circle.py .....	3
color.py .....	4
figure.py .....	5
rectangle.py .....	5
square.py .....	7
Результаты выполнения.....	8
Вывод.....	8
На данной лабораторной работе я изучил возможности объектно-ориентированного программирования в языке Python. ....	8

## Цель работы

Изучение объектно-ориентированных возможностей языка Python.

## Задание

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.

## Выполнение

### main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square

def main() -> None:
    r = Rectangle("синего", 12, 12)
    c = Circle("зеленого", 12)
    s = Square("красного", 12)
    print(r)
    print(c)
    print(s)

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()
        )
```

## Результаты выполнения

```
dubuntus@DS13: ... /__t3.1-Course/code/lab2_code$ python main.py
Прямоугольник синего цвета шириной 12 и высотой 12 площадью 144.
Круг зеленого цвета радиусом 12 площадью 452.3893421169302.
Квадрат красного цвета со стороной 12 площадью 144.
```

*рис. 1 результат выполнения задания*

## Вывод

На данной лабораторной работе я изучил возможности объектно-ориентированного программирования в языке Python.