

**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 ИУ5 «Системы обработки информации и управления»**

**Отчет по лабораторной работе №2  
по курсу «Разработка интернет-приложений»**

**Выполнила:  
студент группы ИУ5-54Б  
Подопригорова Н. С.**

**Проверил:  
Доцент кафедры ИУ5  
Гапанюк Ю. Э.**

**Москва, 2020 г.**

## 1. Описание задания

- 1.1. Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.
- 1.2. Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
- 1.3. Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.
- 1.4. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.
- 1.5. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
- 1.6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
- 1.7. 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.
- 1.8. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля `math`.
- 1.9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
  - Определите метод `getr`, 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.
- 1.10. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов. 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`.

## 2. 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", 7, 4)
    c = Circle("зелёного", 9)
    s = Square("красного", 2)
    print(r)
    print(c)
    print(s)

    a = np.array([r.square(), c.square(), s.square()], float)
    for f in a:
        print(f)

if __name__ == "__main__":
    main()

```

### **color.py**

```

class FigureColor:

    def __init__(self):
        self.__color = None

    @property
    def color(self):
        return self.__color

    @color.setter
    def color(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"

```

```

def __init__(self, color, width, height):
    self.width = width
    self.height = height
    self.fc = FigureColor()
    self.fc.color = color

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

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

def square(self):
    return self.width*self.height

```

## **square.py**

```

from lab_python_oop.rectangle import Rectangle

```

```

class Square(Rectangle):
    FIGURE_TYPE = "Квадрат"

    def __init__(self, color, side):
        self.side = side
        super().__init__(color, self.side, self.side)

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

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

```

## circle.py

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

```
class Circle(Figure):
```

```
    FIGURE_TYPE = "Круг"
```

```
    def __init__(self, color, rad):
```

```
        self.rad = rad
```

```
        self.fc = FigureColor()
```

```
        self.fc.color = color
```

```
    def __repr__(self):
```

```
        return '{} {} цвета радиусом {} площадью {}'.format(
```

```
            Circle.get_figure_type(),
```

```
            self.fc.color,
```

```
            self.rad,
```

```
            self.square()
```

```
        )
```

```
    def square(self):
```

```
        return math.pi*(self.rad**2)
```

```
    @classmethod
```

```
    def get_figure_type(cls):
```

```
        return cls.FIGURE_TYPE
```

```
~/my/pythonchic/rip/my_labs/lab2 master ➤ tree -L 3
.
├── env
│   ├── bin
│   │   ├── Activate.ps1
│   │   ├── activate
│   │   ├── activate.csh
│   │   ├── activate.fish
│   │   ├── easy_install
│   │   ├── easy_install-3.8
│   │   ├── f2py
│   │   ├── f2py3
│   │   ├── f2py3.8
│   │   ├── pip
│   │   ├── pip3
│   │   ├── pip3.8
│   │   ├── python -> python3
│   │   └── python3 -> /usr/local/bin/python3
│   ├── include
│   ├── lib
│   │   └── python3.8
│   └── pyvenv.cfg
├── lab_python_oop
│   ├── __init__.py
│   ├── __pycache__
│   │   ├── __init__.cpython-38.pyc
│   │   ├── circle.cpython-38.pyc
│   │   ├── color.cpython-38.pyc
│   │   ├── figure.cpython-38.pyc
│   │   ├── rectangle.cpython-38.pyc
│   │   └── square.cpython-38.pyc
│   ├── circle.py
│   ├── color.py
│   ├── figure.py
│   ├── rectangle.py
│   └── square.py
├── main.py
└── review.pages

7 directories, 29 files
```

### 3. Экранные формы с примерами выполнения программы

```
~/my/pythonchic/rip/my_labs/lab2 master ➤ python3 main.py
importing lab_python_oop
Прямоугольник синего цвета шириной 7 и высотой 4 площадью 28.
Круг зелёного цвета радиусом 9 площадью 254.46900494077323.
Квадрат красного цвета со стороной 2 площадью 4.
28.0
254.46900494077323
4.0
```