

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

**Факультет «Информатика и управление»**

Кафедра ИУ5. Курс «Разработка Интернет-приложений»

Отчет по лабораторной работе №3  
«Python. Объектно-ориентированные возможности»

Выполнил:

студент группы ИУ5-54Б

Тарасов Владислав

Проверил:

преподаватель каф. ИУ5

Гапанюк Ю. Е.

Подпись и дата:

Подпись и дата:

Москва, 2019 г.

## Описание задания лабораторной работы

Необходимо разработать программу, реализующую работу с классами.

1. Программа должна быть разработана в виде консольного приложения на языке Python 3.
2. Все файлы проекта (кроме основного файла main.py) должны располагаться в пакете lab\_python\_oop.
3. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab\_python\_oop.
  1. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать здесь - <https://docs.python.org/3/library/abc.html>
  2. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать здесь - <https://docs.python.org/3/library/functions.html#property>
  3. 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.
  4. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа math.pi из модуля math - <https://docs.python.org/3/library/math.html>
  5. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
  6. Для классов «Прямоугольник», «Квадрат», «Круг»:
    1. Определите метод "repr", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод format - <https://pyformat.info/>
    2. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
4. 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:
  1. Прямоугольник синего цвета шириной 3 и высотой 2.
  2. Круг зеленого цвета радиусом 5.
  3. Квадрат красного цвета со стороной 5.

## Исходный код

- **figure.py**

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

```
class Figure():  
    @abstractmethod  
    def area(self):  
        pass
```

- **color.py**

```
class Color:  
    def __init__(self):  
        self.__color = None
```

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

```
    @color.setter  
    def color(self, value):  
        self.__color = value
```

- **rectangle.py**

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

```
class Rectangle(Figure):  
    FIGURE_TYPE = "Rectangle"
```

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

```
    def __init__(self, width, height, color):  
        self.width = width  
        self.height = height  
        self.color = Color()
```

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

```
def area(self):
```

```
    return self.width * self.height
```

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

```
    return "{} color: {}, width: {}, height: {}, area: {}".format(
```

```
        Rectangle.get_figure_type(),
```

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

```
        self.width,
```

```
        self.height,
```

```
        self.area()
```

```
)
```

- **circle.py**

```
from lab_python_oop.figure import Figure
```

```
from lab_python_oop.color import Color
```

```
import math
```

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

```
    FIGURE_TYPE = "Circle"
```

```
    @classmethod
```

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

```
        return cls.FIGURE_TYPE
```

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

```
        self.radius = radius
```

```
        self.color = Color()
```

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

```
    def area(self):
```

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

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

```
        return "{} color: {}, radius: {}, area: {}".format(
```

```

        Circle.get_figure_type(),
        self.color.color,
        self.radius,
        self.area()
    )

```

- **square.py**

```

from lab_python_oop.rectangle import Rectangle

```

```

class Square(Rectangle):
    FIGURE_TYPE = "Square"

```

```

    @classmethod

```

```

    def get_figure_type(cls):
        return cls.FIGURE_TYPE

```

```

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

```

```

    def __repr__(self):
        return "{} color: {}, side: {}, area: {}".format(
            Square.get_figure_type(),
            self.color.color,
            self.side,
            self.area()
        )

```

- **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():
    rectangle = Rectangle(3, 2, "black")
    square = Square(5, "red")

```

```
circle = Circle(5, "green")
print(rectangle)
print(square)
print(circle)

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

## Скриншоты с результатами выполнения

```
v.tarasov@sinimawath: ~/univer/rip/labs/lab3 (master)$ python3 main.py
Rectangle color: black, width: 3, height: 2, area: 6
Square color: red, side: 5, area: 25
Circle color: green, radius: 5, area: 78.53981633974483
```