

**Московский государственный технический
университет им. Н. Э. Баумана
Факультет «Информатика и системы управления»**

Кафедра «Системы обработки информации и управления»
Курс «Разработка интернет-приложений»

Отчет по лабораторной работе №3

Группа: ИУ5-52Б

Студент: Морозенков О.Н.

Преподаватель: Гапанюк Ю.Е.

Москва, 2019 г.

Задание

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.
 - 3.1. 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>
 - 3.2. 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>
 - 3.3. 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.
 - 3.4. 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>
 - 3.5. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
 - 3.6. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - 3.6.1. Определите метод "get", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод format - <https://pyformat.info/>
 - 3.6.2. 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).
Создайте следующие объекты и выведите о них информацию в консоль:
 - 4.1. Прямоугольник синего цвета шириной 3 и высотой 2.
 - 4.2. Круг зеленого цвета радиусом 5.
 - 4.3. Квадрат красного цвета со стороной 5.

Исходный код

librip/figure.py

```
from abc import ABC, abstractmethod

class Figure(ABC):
    @abstractmethod
    def area(self):
        raise NotImplementedError
```

librip/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
```

librip/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, color, width, height):
        self._width = width
        self._height = height
        self._fc = FigureColor()
        self._fc.color = color

    def area(self):
```

```

        return self._width * self._height

    def __repr__(self):
        return '{} {} цвета шириной {} и высотой {} площадью  

        {}'.format(
            Rectangle.get_figure_type(),
            self._fc.color,
            self._width,
            self._height,
            self.area()
        )

```

librip/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, 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.area()
        )

```

librip/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, color, radius):
    self._radius = radius
    self._fc = FigureColor()
    self._fc.color = color

def area(self):
    return math.pi * (self._radius ** 2)

def __repr__(self):
    return '{} {} цвета радиусом {} площадью {}'.format(
        Circle.get_figure_type(),
        self._fc.color,
        self._radius,
        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():
    r = Rectangle("синего", 3, 2)
    c = Circle("зеленого", 5)
    s = Square("красного", 5)
    print(r)
    print(c)
    print(s)

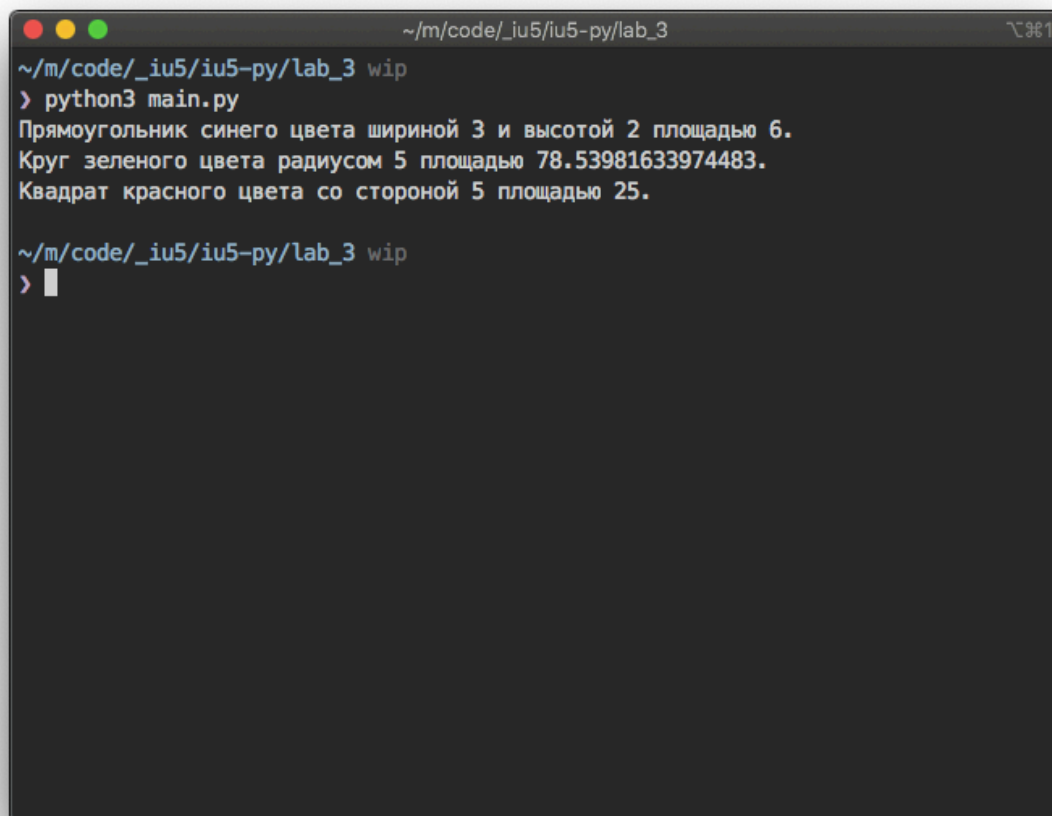
```

```

if __name__ == "__main__":
    main()

```

Вывод задания



```
~/m/code/_iu5/iu5-py/lab_3 wip
> python3 main.py
Прямоугольник синего цвета шириной 3 и высотой 2 площадью 6.
Круг зеленого цвета радиусом 5 площадью 78.53981633974483.
Квадрат красного цвета со стороной 5 площадью 25.

~/m/code/_iu5/iu5-py/lab_3 wip
> █
```

The image shows a terminal window with a dark background and light-colored text. The window title bar at the top indicates the current directory is `~/m/code/_iu5/iu5-py/lab_3`. The terminal content shows a user prompt `>` followed by the command `python3 main.py`. The script outputs three lines of text: "Прямоугольник синего цвета шириной 3 и высотой 2 площадью 6.", "Круг зеленого цвета радиусом 5 площадью 78.53981633974483.", and "Квадрат красного цвета со стороной 5 площадью 25.". After the output, the terminal shows another user prompt `>` followed by a cursor, indicating the session is still active.