

МГТУ им Н.Э. Баумана, кафедра ИУ5

Лабораторная работа №3 по дисциплине "РИП"

Выполнил: Никольский Даниил, ИУ5-516
Проверил: Гапанюк Юрий Евгеньевич, ИУ5

Дата: 27 октября 2019 г.
Подпись: Никольский Д.Р.

Содержание

1	Задание	2
2	Исходный код	2
2.1	color.py	2
2.2	figure.py	3
2.3	main.py	4
3	Результат	5

1 Задание

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. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
4. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
5. 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.
6. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус».
7. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
8. Для классов «Прямоугольник», «Квадрат», «Круг»:
9. Определите метод `get`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь.
10. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
11. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов. Создайте следующие объекты и выведите о них информацию в консоль:
 - (a) Прямоугольник синего цвета шириной 3 и высотой 2.
 - (b) Круг зеленого цвета радиусом 5.
 - (c) Квадрат красного цвета со стороной 5.

2 Исходный код

2.1 `color.py`

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

    def get_color(self):
        return self._color

    def set_color(self, color):
        self._color = color

    def del_color(self):
        del self._color

color = property(get_color, set_color, del_color, "Color property")
```

2.2 figure.py

```
from abc import ABC, abstractmethod
from math import pi

from lab3.lab_python_oop.color import Color

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

class Rectangle(Figure):
    def __init__(self, height, width, color):
        self.height = height
        self.width = width

        self.color = Color()
        self.color.set_color(color)

        self.figure_name = "Rectangle"

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

    def __repr__(self):
        return f'<{self.figure_name}, \
            w: {self.width}, \
            h: {self.height}, \
            c: {self.color.get_color()}, \
            a: {self.area()}>'

class Circle(Figure):
    def __init__(self, rad, color):
        self.rad = rad

        self.color = Color()
        self.color.set_color(color)

        self.figure_name = "Circle"

    def area(self):
        return pi * self.rad ** 2

    def __repr__(self):
        return f'<{self.figure_name}, \
            r: {self.rad}, \
            c: {self.color.get_color()}, \
            a: {self.area()}>'
```

```

class Square(Rectangle):
    def __init__(self, a, color):
        super().__init__(a, a, color)

        self.figure_name = "Square"

    def __repr__(self):
        return f'<{self.figure_name}, \
            s: {self.width}, \
            c: {self.color.get_color()}, \
            a: {self.area()}>'

```

2.3 main.py

```

from lab3.lab_python_oop.figure import Rectangle, Circle, Square

```

```

def main():
    rect = Rectangle(2, 3, 'blue')
    circle = Circle(5, 'green')
    square = Square(5, 'red')
    print(rect)
    print(circle)
    print(square)

```

```

if __name__ == '__main__':
    main()

```

3 Результат

```
<Rectangle, w:3, h:2, c: blue, a: 6>  
<Circle, r: 5, c: green, a: 78.53981633974483>  
<Square, s: 5, c: red, a: 25>
```

Рис. 1: