

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

Факультет «Радиотехнический»

**Кафедра РТ5 «Системы обработки информации и управления»
Отчёт по лабораторной работе №2**

Выполнил:
студент группы РТ5-31Б:
Кузнецов Марат Дмитриевич
Подпись и дата:

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю. Е.
Подпись и дата:

Москва, 2023 г.

Задание

1. Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием `pip`.
2. Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
3. Все файлы проекта (кроме основного файла `main.py`) должны располагаться в пакете `lab_python_oop`.
4. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета `lab_python_oop`.
5. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).
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.
8. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа `math.pi` из модуля `math`.
9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - Определите метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format` - <https://pyformat.info/>
 - Название фигуры должно задаваться в виде поля данных класса и возвращаться методом класса
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.

11. Дополнительное задание. Протестируйте корректность работы Вашей программы с помощью модульного теста. **Текст программы Main.py**

```
import numpy as np
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(18, 18,
"синего")      c = Circle(18,
"зеленого")
    s = Square(18, "красного")
print(r)  print(c)
    print(s)

a = np.array([[1, 2, 3], [4, 5, 6]], float)
print(a)

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

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 Color

class Rectangle(Figure):
    FIGURE_TYPE = "Прямоугольник"

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

    def __init__(self, width, length, color):
        self.width = width      self.length =
length      self.fc = Color()
        self.fc.colorproperty = color
```

```

    def square(self):          return
    self.width*self.length    def
    __repr__(self):
        return '{} {} цвета длиной {} и шириной {}. Площадь равна {}'.format(
            Rectangle.get_figure_type(),
            self.fc.colorproperty,
            self.length,        self.width,
            self.square()
        )

```

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, length, color):
        self.length = length
        super().__init__(color, self.length, self.length)

    def square(self):
        return (self.length**2)

    def __repr__(self):
        return '{} {} цвета с длиной сторон {}. Площадь равна {}'.format(
            Square.get_figure_type(),
            self.fc.colorproperty,    self.length,
            self.square()
        )

```

circle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.Color import Color
import math

```

```

class Circle(Figure):
    FIGURE_TYPE = "Круг"

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

    def __init__(self, radius, color):
        self.radius = radius    self.fc =
        Color()    self.fc.colorproperty
        = color

    def square(self):        return self.radius *
        self.radius * math.pi

    def __repr__(self):
        return '{} {} цвета радиусом {}. Площадь равна {}'.format(

```

```

        Circle.get_figure_type(),
        self.fc.colorproperty,          self.radius,
        self.square()
    )

```

color.py

```

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

    @property
    colorproperty(self):
        """
        Get-аксесор
        """
        return self._color

    @colorproperty.setter
    colorproperty(self, value):
        """
        Set-аксесор
        """
        self._color = value

```

test.py

```

import unittest
from main import main
from io import StringIO
from lab_python_oop.circle import Circle

```

```

class TestMain(unittest.TestCase):
    def test_main(self):
        captured_output = StringIO()
        expected_output = [
            'Прямоугольник синего цвета длиной 18 и шириной 18. Площадь равна 324\n',
            'Круг зеленого цвета с радиусом 18. Площадь равна 201.06192982974676\n',
            'Квадрат красного цвета с длиной сторон 18. Площадь равна 324\n',
            '[[1. 2. 3.]\n [4. 5. 6.]]\n'
        ]

    def test_square(self):
        c = Circle(18, "зеленого")
        self.assertAlmostEqual(c.square(), 1017.8760197630929, places=5)

if __name__ == '__main__':
    unittest.main()

```

Результаты работы

```
Прямоугольник синего цвета длиной 18 и шириной 18. Площадь равна 324
Круг зеленого цвета радиусом 18. Площадь равна 1017.8760197630929
Квадрат 18 цвета с длиной сторон 18. Площадь равна 324
[[1, 2, 3]
 [4, 5, 6]]

Press any key to continue . . .
```

```
..
-----
Ran 2 tests in 0.002s

OK
Press any key to continue . . .
```