

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

**Факультет «Радиотехнический»
Кафедра «Системы обработки информации и управления»**

Курс «Парадигмы и конструкции языков программирования»

**Отчет по лабораторной работе №2
«Объектно-ориентированные возможности языка Python.»**

Выполнил:
студент группы РТ5-31Б:
Фёдоров И. А.

Подпись и дата:

Проверил:
преподаватель кафедры ИУ5
Гапанюк Ю. Е.

Подпись и дата:

Москва, 2024 г.

Постановка задачи

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 («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.

10. 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 (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`.

Текст программы

Файл «Shape.py»

```
from abc import ABCMeta, abstractmethod

class Shape:
    __metaclass__ = ABCMeta

    @abstractmethod
    def square(self):
        """Площадь"""

    @abstractmethod
    def __repr__(self):
        """Параметры"""
```

Файл «Rectangle.py»

```
from Color import ShapeColor
from Shape import Shape

class Rectangle(Shape):
    def __init__(self, a, b, color):
        self.a = a
        self.b = b
        self.color = ShapeColor(color.r, color.g, color.b)
        self.title = 'Rectangle'

    def square(self):
        return self.a * self.b

    def __repr__(self):
        return f"title: {self.title}; (a,b): ({self.a}, {self.b}); color(r, g, b): ({self.color.r}, {self.color.g}, {self.color.b}); square:{self.square()}"""
```

Файл «Color.py»

```
class ShapeColor:
    def __init__(self, r, g, b):
        self.r = r
        self.g = g
        self.b = b
```

Файл «Square.py»

```
from Rectangle import Rectangle

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

    def __repr__(self):
        return f"title: {self.title}; a = {self.a}; color(r, g, b):\n" \
            f"({self.color.r}, {self.color.g}, {self.color.b});\n" \
            f"square:{self.square()}"
```

Файл «Circle.py»

```
import math

from Color import ShapeColor
from Shape import Shape

class Circle(Shape):
    def __init__(self, r, color):
        self.r = r
        self.color = ShapeColor(color.r, color.g, color.b)
        self.title = 'Circle'

    def square(self):
        return math.pi * self.r ** 2

    def __repr__(self):
        return f"title: {self.title}; r = {self.r}; color(r, g, b):\n" \
            f"({self.color.r}, {self.color.g}, {self.color.b});\n" \
            f"square:{self.square()}"
```

Файл «main.py»

```
from Circle import Circle
from Color import ShapeColor
from Rectangle import Rectangle
from Square import Square

def main():
    color = ShapeColor(1, 2, 3)
    rectangle = Rectangle(1, 2, color)
    circle = Circle(1, color)
    square = Square(1, color)

    print(rectangle.__repr__())
    print(circle.__repr__())
    print(square.__repr__())
```

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

Экранные формы с примерами выполнения программы

```
title: Rectangle; (a,b): (1, 2); color(r, g, b):  
    (1, 2, 3); square:2  
title: Circle; r = 1; color(r, g, b):  
    (1, 2, 3); square:3.141592653589793  
title: Square; a = 1; color(r, g, b):  
    (1, 2, 3); square:1
```

```
Process finished with exit code 0
```