

Московский государственный технический
Университет им. Н.Э. Баумана

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»
Отчет по лабораторной работе №2
«Объектно-ориентированные возможности языка Python»

Выполнил:
студент группы ИУ5-36Б
Микулин М.А.

Проверил:
Гапанюк Е.Ю.

2024 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 («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.

11. Дополнительное задание. Протестируйте корректность работы Вашей программы с помощью модульного теста.

Код программы:

CIRCLE.py

```
from lab_python_oop.geometry import Geometry
from lab_python_oop.color import Color
```

```
import math
```

```
class Circle(Geometry):
```

```
    def __init__(self, rad, color):
        self.rad = rad
        super().__init__(color, "Круг")
```

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

```
    def __repr__(self):
        return "{} радиуса: {} и цвета: {}, площади: {}".format(
            self.TYPE,
            self.rad,
            self.color.get_color(),
            self.area())
```

COLOR.py

```
class Color:
```

```
    def __init__(self, color):
        self.color = color
```

```
    def get_color(self):
        return self.color
```

GEOMETRY.py

```
import abc
```

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

```
class Geometry(abc.ABC):
```

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

```
self.TYPE = type
```

```
def get_T(self):  
    return self.TYPE
```

```
@abc.abstractmethod  
def area(self): pass
```

```
from lab_python_oop.geometry import Geometry  
from lab_python_oop.color import Color
```

RECTANGLE.py

```
class Rectangle(Geometry):
```

```
    def __init__(self, h, w, color, type = "Прямоугольник"):  
        self.h = h  
        self.w = w  
        super().__init__(color, type)
```

```
    def area(self):  
        return self.h * self.w
```

```
    def __repr__(self):  
        return "{}: {}:{} и цвета: {}, площадью: {}".format(  
            self.get_T(),  
            self.w, self.h ,  
            self.color.get_color(),  
            self.area())
```

SQUARE.py

```
from lab_python_oop.rectangle import Rectangle
```

```
class Square(Rectangle):
```

```
    def __init__(self, side, color):  
        super().__init__(side, side, color, "Квадрат")
```

```
    def __repr__(self):  
        return "{} со стороной {} и цвета: {}, площадью: {}".format(  
            self.get_T(),  
            self.w,  
            self.color.get_color(),  
            self.area())
```