

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

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

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

**Выполнил:
студент группы ИУ5-36Б**

**Мироненков Арсений
Максимович**

Подпись и дата:

**Проверил:
преподаватель каф.
ИУ5**

Нардид А.Н.

Подпись и дата:

Москва, 2024

Цель лабораторной работы: изучение объектно-ориентированных возможностей языка Python.

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. (<https://docs.python.org/3/library/abc.html>)
6. 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>)
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`. (<https://docs.python.org/3/library/math.html>)
9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - Определите метод `getr`, 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). (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:

`Circle.py`

```
import math

from lab_python_oop.Geometry import shape
from lab_python_oop.Color import Color
```

```

class Circle(shape):
    SHAPE_TYPE = "Круг"
    @classmethod
    def get_shape_type(cls):
        return cls.SHAPE_TYPE
    def __init__(self, color: str, radius):
        self.color = Color()
        self.color.colorName = color
        self.radius = radius
    def area(self):
        if type(self.radius) not in [int, float]:
            raise TypeError
        if self.radius < 0:
            raise ValueError
        return math.pi * self.radius ** 2
    def __repr__(self):
        return f"{Circle.get_shape_type()}, цвет: {self.color.colorName}, радиус: {self.radius}, площадь: {self.area()}"

```

Color.py

```

class Color:

    def __init__(self):
        self.color = ""
    @property
    def colorName(self):
        return self.color
    @colorName.setter
    def colorName(self, color: str):
        self.color = color

```

Geometry.py

```

import abc

class shape(abc.ABC):
    def area():
        pass

```

Rectangle.py

```

from lab_python_oop.Geometry import shape
from lab_python_oop.Color import Color

class Rectangle(shape):
    SHAPE_TYPE = "Прямоугольник"
    @classmethod
    def get_shape_type(cls):
        return cls.SHAPE_TYPE
    def __init__(self, color: str, width: int, height: int):
        self.color = Color()
        self.color.colorName = color
        self.width = width
        self.height = height
    def area(self):
        return self.width * self.height
    def __repr__(self):
        return f"{Rectangle.get_shape_type()}, цвет: {self.color.colorName}, ширина: {self.width}, высота: {self.height}, площадь: {self.area()}"

```

Square.py

```
from lab_python_oop.Geometry import shape
from lab_python_oop.Rectangle import Rectangle
from lab_python_oop.Color import Color

class Square(Rectangle):
    SHAPE_TYPE = "Квадрат"
    @classmethod
    def get_shape_type(cls):
        return cls.SHAPE_TYPE
    def __init__(self, color: str, side: int):
        self.color = Color()
        self.color.colorName = color
        self.width = side
        self.height = side
    def __repr__(self):
        return f"[Square.get_shape_type()], цвет: {self.color.colorName}, сторона: {self.width}, площадь: {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
from colorama import init
init()
from colorama import Fore

def main():
    r = Rectangle("синий", 4, 5)
    c = Circle("зеленый", 6)
    s = Square("красный", 4)
    print(Fore.RED + str(s))
    print(Fore.GREEN + str(c))
    print(Fore.BLUE + str(r))
if __name__ == "__main__":
    main()
```

Вывод программы:

```
(venv) PS C:\Users\DNS\PycharmProjects\laba1.2> python main.py
Квадрат, цвет: красный, сторона: 4, площадь: 16
Круг, цвет: зеленый, радиус: 6, площадь: 113.09733552923255
Прямоугольник, цвет: синий, ширина: 4, высота: 5, площадь: 20
(venv) PS C:\Users\DNS\PycharmProjects\laba1.2>
```