

МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
им. Н.Э. Баумана

Факультет “Информатика и системы управления”  
Кафедра “Системы обработки информации и управления”



Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по лабораторной работе №2  
“Объектно-ориентированные возможности языка Python”

**Выполнил:**  
Студент группы ИУ5-35Б  
Рябов М.А.  
**Преподаватель:**  
Гапанюк Ю.Е.

Москва 2025

## 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](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.

### Листинг кода

#### circle.py

```
import math

from lab_python_oop.figure import Figure
from lab_python_oop.color import Color

class Circle(Figure):
    figure_type = "Круг"

    @classmethod
    def get_figure_type(obj):
        return obj.figure_type

    def __init__(self, radius, color):
        self._radius = radius
        self._color = Color()
        self._color.value = color

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

    def __repr__(self):
        return "{} {} с радиусом {} и площадью {}".format(
            self._color.value,
            Circle.get_figure_type(),
            self._radius,
            self.square()
        )
```

## color.py

```
class Color:  
    def __init__(self):  
        self._value = None  
  
    @property  
    def value(self):  
        return self._value  
  
    @value.setter  
    def value(self, value):  
        self._value = value
```

## 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(obj):  
        return obj.figure_type  
  
    def __init__(self, width, height, color):  
        self._width = width  
        self._height = height  
        self._color = Color()  
        self._color.value = color  
  
    def square(self):  
        return self._width * self._height  
  
    def __repr__(self):  
        return "{} {} со сторонами {} и {} и площадью {}".format(  
            self._color.value,  
            Rectangle.get_figure_type(),  
            self._width,  
            self._height,
```

```
        self.square()
    )
```

### square.py

```
from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):
    figure_type = "Квадрат"

    @classmethod
    def get_figure_type(obj):
        return obj.figure_type

    def __init__(self, side, color):
        self._side = side
        super().__init__(self._side, self._side, color)

    def square(self):
        return self._width * self._width

    def __repr__(self):
        return '{} {} со стороной {} и площадью {}'.format(
            self._color.value,
            Square.get_figure_type(),
            self._side,
            self.square()
        )
```

### main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square

from cowsay import cowsay

def main():
    rectangle = Rectangle(19, 19, "Синий")
    circle = Circle(19, "Зеленый")
    square = Square(19, "Красный")

    print(rectangle)
    print(circle)
    print(square)

    message = """
    There are always two roads in life: one is the first and the other is the
    second.
```

```
-- Jason Statham
""".strip()

print(cowsay(message))

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

## Выполнение программы

```
(.venv) mmmmm@LAPTOPC:~/projects/pcpl_2025/lab_2$ python3 main.py
Синий Прямоугольник со сторонами 19 и 19 и площадью 361
Зеленый Круг с радиусом 19 и площадью 1134.1149479459152
Красный Квадрат со стороной 19 и площадью 361

/ There are always two roads in life: one \
| is the first and the other is the      |
| second.                                |
\ -- Jason Statham                      /
```

---

```
\ ^__^
 \  (oo)\_____
   (__)\       )\/\
     ||----w |
     ||     ||
```