

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 ИУБ-35Б
Евдокимов М. С.
Преподаватель:
Гапанюк Ю. Е.

Москва 2025

Цель лабораторной работы

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

Задание лабораторной работы

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 неперечисленных классов должен располагаться в отдельном файле пакета lab_python_oop.
5. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
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.
 - 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. 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 программы

Файл figure.py

```
from abc import ABC, abstractmethod

class Figure(ABC):
    @abstractmethod
    def area(self):
        pass
```

Файл circle.py

```
import math
from .figure import Figure
from .color import Color

class Circle(Figure):
    name = "Круг"

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

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

    def __repr__(self):
        return f'{self.name} {self.color.color_name} цвета, радиус = {self.radius}, площадь = {self.area():.2f}'
```

Файл rectangle.py

```
from .figure import Figure
from .color import Color

class Rectangle(Figure):
    name = "Прямоугольник"

    def __init__(self, width, height, color_name):
        self.width = width
        self.height = height
        self.color = Color(color_name)

    def area(self):
        return self.width * self.height

    def __repr__(self):
        return f'{self.name} {self.color.color_name} цвета, ширина = {self.width}, высота = {self.height}, площадь = {self.area()}'
```

Файл color.py

```
class Color:
    def __init__(self, color_name):
        self.__color_name = color_name

    @property
    def color_name(self):
        return self.__color_name
```

```
@color_name.setter  
def color_name(self, value):  
    self.__color_name = value
```

Файл square.py

```
from .rectangle import Rectangle  
  
class Square(Rectangle):  
    name = "Квадрат"  
  
    def __init__(self, side, color_name):  
        super().__init__(side, side, color_name)  
        self.side = side  
  
    def __repr__(self):  
        return f"{self.name} {self.color.color_name} цвета, сторона = {self.side}, площадь = {self.area()}"
```

Файл main.py

```
from lab_python_oop.rectangle import Rectangle  
from lab_python_oop.circle import Circle  
from lab_python_oop.square import Square  
from termcolor import colored  
  
def main():  
    N = 7  
    rect = Rectangle(N, N, "синего")  
    circle = Circle(N, "зелёного")  
    square = Square(N, "красного")  
  
    print(colored(rect, "blue"))  
    print(colored(circle, "green"))  
    print(colored(square, "red"))  
  
if __name__ == "__main__":  
    main()
```

Скриншоты работы программы

```
~/Desktop/PyCharmMiscProject/ПикЯП/Лабы/lab_2 git:(main) 2 files changed, 0 insertions(+), 0 deletions(-) (0.031s)
source venv/bin/activate

venv ~/Desktop/PyCharmMiscProject/ПикЯП/Лабы/lab_2 git:(main) 2 files changed, 0 insertions(+), 0 deletions(-) (0.094s)
python3 main.py
Прямоугольник синего цвета, ширина = 7, высота = 49
Круг зелёного цвета, радиус = 7, площадь = 153.94
Квадрат красного цвета, сторона = 7, площадь = 49

(venv) ~/Desktop/PyCharmMiscProject/ПикЯП/Лабы/lab_2 git:(main) (0.025s)
deactivate
```