

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

Факультет «Информатика и управление»  
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 ИУ5-32Б

Плюшко Дмитрий Андреевич

Подпись и дата:

Проверил:

преподаватель каф. ИУ5

Подпись и дата:

Москва, 2024 г.

## Постановка задачи

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 "гер", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь.
  - 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. 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.

### Код программы

#### **main.py**

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

def main():
    a = Rectangle(17, 17, 'blue')
    b = Circle(17, 'green')
    c = Square(17, 'red')

    print(f'{a}\n{b}\n{c}')

if __name__ == '__main__':
    sys.exit(main())
```

#### **figure.py**

```
from abc import ABC, abstractmethod
```

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

#### **color\_figure.py**

```
class ColorFigure:
    def __init__(self, COLOR):
        self.color = COLOR

    def __repr__(self):
        return f'{self.color}'
```

#### **circle.py**

```
from lab_python_oop.figure import Figure
from lab_python_oop.color_figure import ColorFigure
from math import pi
```

```

class Circle(Figure):
    def __init__(self, RADIUS, COLOR):
        self.name = "Circle"
        self.radius = RADIUS
        self.color = ColorFigure(COLOR)

    def __repr__(self):
        return f'Figure: {self.name}, Radius: {self.radius},
Color: {self.color}, Square: {self.square()}'

    def square(self):
        return pi * (self.radius ** 2)

```

### ***rectangle.py***

```

from lab_python_oop.figure import Figure
from lab_python_oop.color_figure import ColorFigure

```

```

class Rectangle(Figure):
    def __init__(self, HEIGHT, WIDTH, COLOR):
        self.name = "Rectangle"
        self.height = HEIGHT
        self.width = WIDTH
        self.color = ColorFigure(COLOR)

    def __repr__(self):
        return f'Figure: {self.name}, Height: {self.height},
Width: {self.width}, Color: {self.color}, Square:
{self.square()}'

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

```

### ***square.py***

```

from lab_python_oop.rectangle import Rectangle
from lab_python_oop.color_figure import ColorFigure

```

```

class Square(Rectangle):
    def __init__(self, HEIGHT, COLOR):
        super().__init__(HEIGHT, HEIGHT, COLOR)

```

```
self.name = "Square"
self.color = ColorFigure(COLOR)

def __repr__(self):
    return f'Figure: {self.name}, Side of square:
{self.height}, Color: {self.color}, Square: {self.square()}'
```

## Тестирование

Figure: Rectangle, Height: 17, Width: 17, Color: blue, Square: 289

Figure: Circle, Radius: 17, Color: green, Square: 907.9202768874502

Figure: Square, Side of square: 17, Color: red, Square: 289