

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

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

Курс «Базовые компоненты интернет-технологий»

Отчет по лабораторной работе №2
«Объектно-ориентированные возможности языка Python»

Выполнил:
студент группы ИУ5-34Б

Угрюмов Михаил

Проверил:
преподаватель каф.
ИУ5

Гапанюк Ю. Е.

2022 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`.

Текст программы

Color.py

```
class Color:
    def __init__(self):
        self.param = None

    @property
    def parametr(self):
        return self.param

    @parametr.setter
    def parametr(self, param):
        self.param = param
```

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):
    TYPE = 'прямоугольник'

    @classmethod
    def type(cls):
        return cls.TYPE

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

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

    def __repr__(self):
        return '{} {} шириной {} высотой {} площадью {}'.format(
            self.color.parametr,
            Rectangle.type(),
            self.width,
            self.height,
            self.square()
        )
```

Square.py

```
from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):
    TYPE = 'квадрат'

    @classmethod
    def type(cls):
        return cls.TYPE

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

    def __repr__(self):
        return '{} {} со стороной {} площадью {}'.format(
            self.color.parametr,
            Square.type(),
            self.side,
            self.square()
        )
```

Circle.py

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

class Circle(Figure):
    TYPE = 'круг'

    @classmethod
    def type(cls):
        return cls.TYPE

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

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

    def __repr__(self):
        return '{} {} радиусом {} площадью {}'.format(
            self.color.parametr,
            Circle.type(),
            self.radius,
            self.square()
        )
```

Cube.py (дополнительное задание при защите лабораторной)

```
from lab_python_oop.square import Square

class Cube(Square):
    TYPE = 'куб'

    @classmethod
    def type(cls):
        return cls.TYPE

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

    def volume(self):
        return self.side ** 3

    def __repr__(self):
        return '{} {} со стороной {} объёмом {}'.format(
            self.color.parametr,
            Cube.type(),
            self.side,
            self.volume()
        )
    )
```

main.py

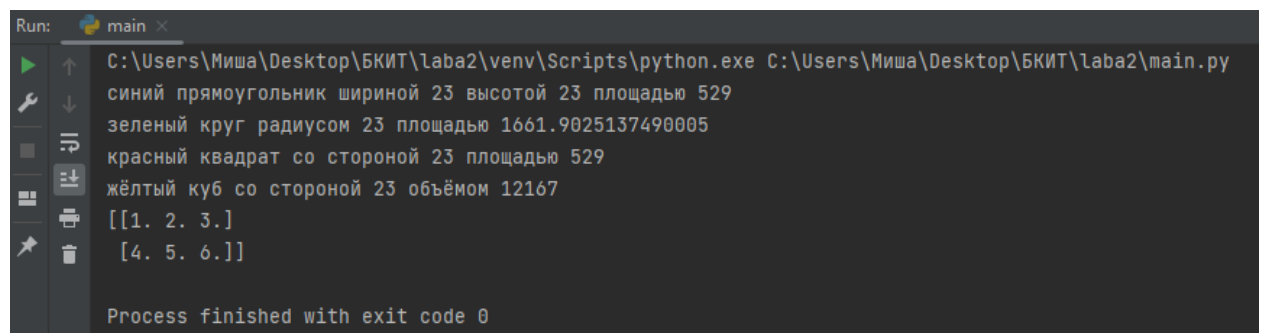
```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
from lab_python_oop.cube import Cube
import numpy as np

def main():
    r = Rectangle(23, 23, 'синий')
    c = Circle(23, 'зеленый')
    s = Square(23, 'красный')
    cube = Cube(23, 'жёлтый')
    print(r)
    print(c)
    print(s)
    print(cube)

    a = np.array([[1, 2, 3], [4, 5, 6]], float)
    print(a)

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

Результат выполнения программы:



```
Run: main x
C:\Users\Мisha\Desktop\БКИТ\laba2\venv\Scripts\python.exe C:\Users\Мisha\Desktop\БКИТ\laba2\main.py
синий прямоугольник шириной 23 высотой 23 площадью 529
зеленый круг радиусом 23 площадью 1661.9025137490005
красный квадрат со стороной 23 площадью 529
жёлтый куб со стороной 23 объёмом 12167
[[1. 2. 3.]
 [4. 5. 6.]]

Process finished with exit code 0
```