



Московский государственный технический университет имени Н. Э. Баумана

Факультет «Информатика и системы управления»

Кафедра ИУ5

Отчёт по

лабораторной работе №2

«Объектно-ориентированные возможности языка Python.»

Подготовил:

Алпеев Владислав Сергеевич

Группа ИУ5-54Б

**Москва
2020**

1. Описание задания.

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.
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»:
 - 9.1. Определить метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Использовать метод `format`.
 - 9.2. 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 классов. Создайте следующие объекты и выведите о них информацию в консоль (N - номер варианта по списку группы):
 - 10.1. Прямоугольник синего цвета шириной N и высотой N.
 - 10.2. Круг зеленого цвета радиусом N.
 - 10.3. Квадрат красного цвета со стороной N.
 - 10.4. Также вызовите один из методов внешнего пакета, установленного с использованием `pip`.

2. Текст программы.

Class geometricalFigure:

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

```
class GeometricalFigure(ABC):
```

```
    @abstractmethod
```

```
    def findSquare(self):
```

```
        pass
```

Class rectangle:

```
from lab_python_oop.geometricalFigure import GeometricalFigure
```

```
from lab_python_oop.figureColor import FigureColor
```

```
class Rectangle(GeometricalFigure):
```

```
    FIGURE_TYPE = "прямоугольник"
```

```
    @classmethod
```

```
    def getFigureType(cls):
```

```
        return cls.FIGURE_TYPE
```

```
    def __init__(self, width, height, _color):
```

```
        self.width = width
```

```
        self.height = height
```

```
        self._color = FigureColor()
```

```
        self._color.setColor(_color)
```

```
    def findSquare(self):
```

```
        return self.width * self.height
```

```
    def __repr__(self):
```

```
        return "{} {} с шириной равной {}, высотой равной {} и площадью равной {}  
{}".format(
```

```
            self._color.getColor(),
```

```
            self.getFigureType(),
```

```
            self.width,
```

```
            self.height,
```

```
            self.findSquare())
```

Class circle:

```
from lab_python_oop.geometricalFigure import GeometricalFigure
```

```
from lab_python_oop.figureColor import FigureColor
```

```
from math import pi
```

```
class Circle(GeometricalFigure):
```

```
    FIGURE_TYPE = "круг"
```

```
    @classmethod
```

```
    def getFigureType(cls):
```

```
        return cls.FIGURE_TYPE
```

```
    def __init__(self, radius, _color):
```

```
        self.radius = radius
```

```
        self._color = FigureColor()
```

```
        self._color.setColor(_color)
```

```
    def findSquare(self):
```

```
        return pi * (self.radius ** 2)
```

```

def __repr__(self):
    return "{} {} с радиусом {} и площадью равной {}".format(
        self._color.getColor(),
        self.getFigureType(),
        self.radius,
        self.findSquare())

```

Class square:

```

from lab_python_oop.rectangle import Rectangle
from lab_python_oop.figureColor import FigureColor

class Square(Rectangle):

    FIGURE_TYPE = "квадрат"

    @classmethod
    def getFigureType(cls):
        return cls.FIGURE_TYPE

    def __init__(self, side, _color):
        super().__init__(side, side, _color)

    def __repr__(self):
        return "{} {} со стороной {} и площадью равной {}".format(
            self._color.getColor(),
            self.getFigureType(),
            self.width,
            self.findSquare())

```

Class figureColor:

```

class FigureColor:

    def __init__(self):
        self._color = "чёрный"

    def setColor(self, color):
        self._color = color

    def getColor(self):
        return self._color

```

main:

```

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

from colorama import Fore

def main():
    rectangle = Rectangle(1, 1, "синий")
    print(Fore.BLUE + str(rectangle))
    circle = Circle(1, "зелёный")
    print(Fore.GREEN + str(circle))
    square = Square(1, "красный")
    print(Fore.RED + str(square))

if __name__ == "__main__":
    main()

```

3. Экранные формы с примерами выполнения программы.

```

vladislavapeev@MacDonalds-VL Lab2 % source /Users/vladislavapeev/Documents/labs/labsPython/Lab2/.venv/bin/activate
(.venv) vladislavapeev@MacDonalds-VL Lab2 % /Users/vladislavapeev/Documents/labs/labsPython/Lab2/.venv/bin/python3 /Users/vladislavapeev/Documents/
labs/labsPython/Lab2/main.py
синий прямоугольник с шириной равной 1, высотой равной 1 и площадью равной 1
зелёный круг с радиусом 1 и площадью равной 3.141592653589793
красный квадрат со стороной 1 и площадью равной 1
(.venv) vladislavapeev@MacDonalds-VL Lab2 %

```