

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

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Разработка интернет-приложений»

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

Выполнил:  
студент группы ИУ5-51Б  
Андреев А.В.  
Подпись и дата:

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю.Е.  
Подпись и дата:

г. Москва, 2020 г.

## 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 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».
9. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны. Для классов «Прямоугольник», «Квадрат», «Круг»:
  1. Определите метод "get", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь.
  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 Ваших классов.

Создайте следующие объекты и выведите о них информацию в консоль (N - номер Вашего варианта по списку группы):

1. Прямоугольник синего цвета шириной N и высотой N.
2. Круг зеленого цвета радиусом N.
3. Квадрат красного цвета со стороной N.
4. Также вызовите один из методов внешнего пакета, установленного с использованием `pip`.

## Текст программы Main.py

```
from lab_python_oop.Rectangle import Rectangle
from lab_python_oop.Circle import Circle from
lab_python_oop.Square import Square import
arrow
def
main():
    print("ИУ5-51Б Андреев Александр Владимирович Лаб №2")
print(arrow.now(), "\n")

    rectangle = Rectangle("синего", 2, 2)
circle = Circle("зеленого", 2)    square
= Square("красного", 2)

    print(rectangle)
print(circle)    print(square)
    if __name__ ==
"__main__":
    main()
```

## GeometricFigure.py

```
from abc import ABC, abstractmethod
class
GeometricFigure(ABC):
    FIGURE_TYPE = None

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

    @abstractmethod
def _square(self):
    pass
```

## FigureColor.py

```
class FigureColor:    def
    __init__(self):
self._color = None

    @property    def
color_property(self):
return self._color

    @color_property.setter    def
color_property(self, value):
    self._color = value
```

## Rectangle.py

```
from lab_python_oop.GeometricFigure import GeometricFigure from
lab_python_oop.FigureColor import FigureColor
class
Rectangle(GeometricFigure):
    FIGURE_TYPE = "Прямоугольник"
    def __init__(self, color, width,
height):
        self._width = width
self._height = height
        self.figure_color = FigureColor()
self.figure_color.color_property = color
    def
_square(self):
        return self._width * self._height
    def
__repr__(self):
        return '{} {} цвета со длиной {} and высотой {} площадью
{}'.format(
            Rectangle.get_figure_type(),
self.figure_color.color_property,
self._width, self._height,
self._square()
        )
```

## Circle.py

```
class
Circle(GeometricFigure):
    FIGURE_TYPE = "Круг"
    def __init__(self, color,
radius):
        self.radius = radius
        self.figure_color = FigureColor()
self.figure_color.color_property = color
    def
_square(self):
        return math.pi * (self.radius ** 2)
    def
__repr__(self):
        return '{} {} цвета радиусом {} площадью {}'.format(
Circle.get_figure_type(),
self.figure_color.color_property, self.radius,
self._square()
        )
```

## Square.py

```

from lab_python_oop.Rectangle import Rectangle
class
Square(Rectangle):
    FIGURE_TYPE = "Квадрат"
    def __init__(self, color,
side):
        self._side = side
        super().__init__(color, self._side, self._side)
    def
__repr__(self):
    return '{} {} цвета со стороной {} площадью {}'.format(
Square.get_figure_type(),
self._figure_color.color_property, self._side,
self._square()
)

```

## Примеры работы программы

**ИУ5-51Б Андреев Александр Владимирович Лаб №2**

**Прямоугольник синего цвета со длиной 16 и высотой 16 площадью 256.**

**Круг зеленого цвета радиусом 16 площадью 804.247719318987.**

**Квадрат красного цвета со стороной 16 площадью 256.**