

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

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

Курс «Разработка интернет приложений»

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

Выполнил:

студент группы ИУ5-52Б
Васильченко Дарья

Проверил:

преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2021 г.

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. 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:

main.py:

```
from lab_python_oop.Rectangle import Rectangle
from lab_python_oop.Circle import Circle
from lab_python_oop.Square import Square

def main():
    rect = Rectangle(6, 6, "синего")
    circle = Circle(6, "зеленого")
    square = Square(6, "красного")
    print(rect)
    print(circle)
    print(square)

if __name__ == '__main__':
    main();
```

GeomFigure.py:

```
from abc import ABC, abstractmethod

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

FigureColor.py:

```
class FigureColor:

    def __init__(self):
        self._color = None

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

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

Rectangle.py:

```
from lab_python_oop.GeomFigure import GeomFigure
from lab_python_oop.FigureColor import FigureColor

class Rectangle(GeomFigure):
    FIGURE_TYPE = "Прямоугольник"

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

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

```

        self.color = FigureColor()
        self.color.colorproperty = color

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

    def __repr__(self):
        return '{} {} цвета шириной {} и высотой {} площадью {}'.format(
            Rectangle.get_figure_type(),
            self.color.colorproperty,
            self.width,
            self.height,
            self.square()
        )

```

Circle.py:

```

from lab_python_oop.GeomFigure import GeomFigure
from lab_python_oop.FigureColor import FigureColor
import math

class Circle(GeomFigure):
    FIGURE_TYPE = "Круг"

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

    def __init__(self, radius, color):
        self.radius = radius
        self.color = FigureColor()
        self.color.colorproperty = color

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

    def __repr__(self):
        return '{} {} цвета радиусом {} площадью {}'.format(
            Circle.get_figure_type(),
            self.color.colorproperty,
            self.radius,
            self.square()
        )

```

Square.py:

```

from lab_python_oop.FigureColor import FigureColor
from lab_python_oop.Rectangle import Rectangle

class Square(Rectangle):
    FIGURE_TYPE = "Квадрат"

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

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

```

```
def __repr__(self):  
    return '{} {} цвета со стороной {} площадью {}'.format(  
        Square.get_figure_type(),  
        self.color.colorproperty,  
        self.length,  
        self.square()  
    )
```

Экранные формы с примерами выполнения программы:

<i>Пример</i>	<i>Вывод программы</i>
1	Прямоугольник синего цвета шириной 6 и высотой 6 площадью 36 Круг зеленого цвета радиусом 6 площадью 113.09733552923255 Квадрат красного цвета со стороной 6 площадью 36