

**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 ИУ5-32Б:
Бекетов Роман Александрович
Подпись и дата:

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

Москва, 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.
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»:
 - Определите метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format`.
 - 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):
 - 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`.

Листинг

main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.quadrate import Quadrate

import numpy as np

def main():
    rec = Rectangle(3, 3, "синий")
    cir = Circle(3, "зелёный")
    squ = Quadrate(3, "красный")
    print(rec)
    print(cir)
    print(squ)

    print(np.ones((4,4)))

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

circle.py

```
from lab_python_oop.geometrical_figure import GeometricalFigure
from lab_python_oop.f_color import FigurColor
from math import pi

class Circle(GeometricalFigure):

    FIGURE_TYPE = "Круг"

    @classmethod
    def __init__(self, radius, color):
        self.radius = radius
        self.color = FigurColor()
        self.color.colorproperty = color

    def get_type(self):
        return self.FIGURE_TYPE

    def get_square(self):
        return pi*self.radius*self.radius

    def __repr__(self):
        return '{}, цвет - {}, радиус = {}, площадь = {}'.format(
            self.get_type(),
            self.color.colorproperty,
            self.radius,
            self.get_square()
        )
```

f_color.py

```
class FigurColor:

    def __init__(self):
        self._color = None

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

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

geometrical_figure.py

```
from abc import ABC, abstractmethod

class GeometricalFigure(ABC):

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

quadrate.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.f_color import FigurColor

class Quadrate(Rectangle):

    FIGURE_TYPE = "Квадрат"

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

    def get_type(self):
        return self.FIGURE_TYPE

    def __repr__(self):
        return '{}, цвет - {}, сторона = {}, площадь = {}'.format(
```

```

        self.get_type(),
        self.color.colorproperty,
        self.side,
        self.get_square()
    )

```

rectangle.py

```

from lab_python_oop.geometrical_figure import GeometricalFigure
from lab_python_oop.f_color import FigurColor

```

```

class Rectangle(GeometricalFigure):

```

```

    FIGURE_TYPE = "Прямоугольник"

```

```

    @classmethod

```

```

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

```

```

    def get_type(self):
        return self.FIGURE_TYPE

```

```

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

```

```

    def __repr__(self):
        return '{}, цвет - {}, ширина = {}, высота = {}, площадь = {}'.format(
            self.get_type(),
            self.color.colorproperty,
            self.width,
            self.height,
            self.get_square()
        )

```

Тесты

```

(.venv) romanbeketov@MacBook-Air-Roman src % python3 main.py
Прямоугольник, цвет - синий, ширина = 3, высота = 3, площадь = 9.
Круг, цвет - зелёный, радиус = 3, площадь = 28.274333882308138.
Квадрат, цвет - красный, сторона = 3, площадь = 9.
[[1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]
 [1. 1. 1. 1.]]

```