

МГТУ им. Н.Э. Баумана

Отчёт по лабораторной работе №2
по курсу «Парадигмы и конструкции языков программирования»
Тема: Объектно-ориентированное программирование.

Проверил:
Нардид А.Н.

Студент группы ИУ5-36Б
Мохаммед М. Н.

2024 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`.

Листинг:

Circle.py

```
import math
from lab_python_oop.shape_of_figure import Shape_of_figure
from lab_python_oop.color import Color_of_figure

class Circle(Shape_of_figure):
    shape = "Круг"
    @staticmethod
    def get_shape():
        return Circle.shape
    def __init__(self, color, radius):
        self.color = Color_of_figure()
        self.color.color_ = color
        self.radius = radius
    def area(self):
        return pow(self.radius, 2)*math.pi
    def __repr__(self):
        return f'Название фигуры - {Circle.get_shape()}, цвет - {self.color.color_}, радиус - {self.radius}, площадь - {self.area()}'
```

color.py

```
class Color_of_figure:
    def __init__(self):
        self.color = None
    @property
    def color_(self):
        return self.color
    @color_.setter
    def color_(self, new_color):
        self.color = new_color
```

main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
import numpy as np
from termcolor import colored

def main():
    rec = Rectangle("синий", 13, 13)
    cir = Circle("зеленый", 13)
    sq = Square("красный", 13)
    print(colored(rec, 'blue'))
    print(colored(cir, 'green'))
    print(colored(sq, 'red'))
    a = np.array([rec, cir, sq])
    print(a)

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

rectangle.py

```
from lab_python_oop.shape_of_figure import Shape_of_figure
from lab_python_oop.color import Color_of_figure

class Rectangle(Shape_of_figure):#дописать про цвет
    shape = "Прямоугольник"
    @classmethod
    def get_shape(cls):
```

```

        return cls.shape
    def __init__(self, color, width, height):
        self.color = Color_of_figure()
        self.color.color_ = color
        self.width = width
        self.height = height
    def area(self):
        return self.width*self.height
    def __repr__(self):
        return f'Название фигуры - {Rectangle.get_shape()}, цвет -
{self.color.color_}, ширина - {self.width}, высота - {self.height}, площадь -
{self.area()}'

```

shape_of_figure.py

```

import abc

class Shape_of_figure(abc.ABC):
    @abc.abstractmethod
    def area(self): pass      #площадь фигуры

```

square.py

```

from lab_python_oop.color import Color_of_figure
from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):
    shape = "Квадрат"
    Rectangle.get_shape()
    def __init__(self, color, side):
        self.color = Color_of_figure()
        self.color.color_ = color
        self.side = side
    def area(self):
        return pow(self.side, 2)
    def __repr__(self):
        return f'Название фигуры - {Square.get_shape()}, цвет -
{self.color.color_}, длина одной стороны - {self.side}, площадь -
{self.area()}'

```

Анализ результатов

```

C:\Users\mokha\PycharmProjects\lab_2\venv\Scripts\python.exe C:\Users\mokha\PycharmProject
Название фигуры - Прямоугольник, цвет - синий, ширина - 13, высота - 13, площадь - 169
Название фигуры - Круг, цвет - зеленый, радиус - 13, площадь - 530.929158456675
Название фигуры - Квадрат, цвет - красный, длина одной стороны - 13, площадь - 169
[Название фигуры - Прямоугольник, цвет - синий, ширина - 13, высота - 13, площадь - 169
Название фигуры - Круг, цвет - зеленый, радиус - 13, площадь - 530.929158456675
Название фигуры - Квадрат, цвет - красный, длина одной стороны - 13, площадь - 169]

Process finished with exit code 0

```

C:\Users\mokha\PycharmProjects\pythonProject1\venv\Scripts\python

Введите коэффициент: -2

Введите коэффициент: 2

Введите коэффициент: 1

Два корня: 1.1687708944803676, -1.1687708944803676

Введите коэффициент: 4

Введите коэффициент: 3

Введите коэффициент: 2

Нет корней

Введите коэффициент: 1

Введите коэффициент: 4

Введите коэффициент: 1

Нет корней

Введите коэффициент: -3

Введите коэффициент: 3

Введите коэффициент: 1

Два корня: 1.1241719689735967, -1.1241719689735967

Введите коэффициент: