



МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ
Московский государственный технический университет
им. Н.Э. Баумана
(МГТУ им. Н.Э. Баумана)

Кафедра «Системы обработки информации и управления» (ИУ5)

Отчёт по лабораторной работе № 2

По курсу: «Базовые компоненты интернет-технологий»

Выполнил:

Строганов Георгий Константинович
студент группы ИУ5-31Б.

Проверил:

Дата: _____.____.2022г.

Подпись: _____.

г. 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 `"get"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format` - <https://pyformat.info/>
10. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
11. 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.

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 sys

def get_side_rad(prompt = "Введите значение стороны"):
    print(prompt)
    try:
        # Попробуем прочесть значение из командной строки
        side_str = sys.argv
    except:
        # Вводим с клавиатуры
        side_str = input()
    # Обрабатываем неправильный ввод
    while True:
        try:
            side = float(side_str)
        except:
            print("Введены неправильные данные.")
            side_str = input()
        else:
            return side
def get_color(termp):
    print("Введите цвет будущего", termp)
    return str(input())
def main():
    print("Выберете тип ввода:\n1 - с клавиатуры в консоли\n2 - вариант 16")
    while True:
        choice = int(input())
        if choice == 1 or choice == 2:
            break
        else:
            print("Введено неверное значение. Введите 1 или 2")

    if choice == 1:
        r = Rectangle(get_color("прямоугольника"), get_side_rad(), get_side_rad())
        c = Circle(get_color("круга"), get_side_rad("Введите радиус"))
        s = Square(get_color("квадрата"), get_side_rad())
        print(r, c, s, sep = "\n")
    else:
        print(Rectangle("синего", 16, 16))
        print(Circle("зеленого", 16))
        print(Square("красного", 16))

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

circle.py

```
from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor
```

```

import math

class Circle(Figure):
    """
    Класс «Круг» наследуется от класса «Геометрическая фигура».
    """
    type = "Круг"

    def __init__(self, color_param, r_param):
        self.r = r_param
        self.fc = FigureColor()
        self.fc.colorproperty = color_param

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

    def __repr__(self):
        return '{} {} цвета радиусом {} площадью {}'.format(Circle.type,
self.fc.colorproperty, self.r, self.square())

```

color.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

```

figure.py

```

from abc import ABC, abstractmethod

#Абстрактный класс «Геометрическая фигура»
class Figure(ABC):
    @abstractmethod
    def square(self):
        pass

```

rectangle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor

class Rectangle(Figure):
    """
    Класс «Прямоугольник» наследуется от класса «Геометрическая фигура».
    """
    type = "Прямоугольник"

```

```

def __init__(self, color_param, width_param, height_param):
    self.width = width_param
    self.height = height_param
    self.fc = FigureColor()
    self.fc.colorproperty = color_param

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

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

```

square.py

```

from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):
    """
    Класс «Квадрат» наследуется от класса «Прямоугольник».
    """
    type = "Квадрат"

    def __init__(self, color_param, side_param):
        self.side = side_param
        super().__init__(color_param, self.side, self.side)

    def __repr__(self):
        return '{} {} цвета со стороной {} площадью {}'.format(Square.type,
self.fc.colorproperty, self.side, self.square())

```

Результаты тестирования:

```

Введите коэффициент A:
1
Введите коэффициент B:
0
Введите коэффициент C:
-4
Два корня: 1.4142135623730951, -1.4142135623730951
Press any key to continue . . .

```

```

Введите коэффициент A:
1
Введите коэффициент B:
-24
Введите коэффициент C:
44
Четыре корня:
1.4142135623730951, 4.69041575982343, -4.69041575982343, -1.4142135623730951
Press any key to continue . . .

```

