|  |  |
| --- | --- |
|  | Министерство науки и высшего образования Российской Федерации  Федеральное государственное бюджетное образовательное учреждение  высшего образования  «Московский государственный технический университет  имени Н.Э. Баумана  (национальный исследовательский университет)»  (МГТУ им. Н.Э. Баумана) |

Факультет «Радиотехнический»

Кафедра «Системы обработки информации и управления»

Лабораторная работа № 2

по дисциплине «Разработка интернет-приложений».

Выполнил:

студент(ка) группы № РТ5-51Б

А. С. Пакало

подпись, дата

Проверил:

преподаватель

Ю. Е. Гапанюк   
подпись, дата

Оглавление

[Цель работы 2](#_Toc86615042)

[Задание 2](#_Toc86615043)

[Выполнение 3](#_Toc86615044)

[main.py 3](#_Toc86615045)

[circle.py 3](#_Toc86615046)

[color.py 4](#_Toc86615047)

[figure.py 5](#_Toc86615048)

[rectangle.py 5](#_Toc86615049)

[square.py 7](#_Toc86615050)

[Результаты выполнения 8](#_Toc86615051)

[Вывод 8](#_Toc86615052)

[На данной лабораторной работе я изучил возможности объектно-ориентированного программирования в языке Python. 8](#_Toc86615053)

# Цель работы

Изучение объектно-ориентированных возможностей языка Python.

# Задание

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.

# Выполнение

## 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() -> None:

r = Rectangle("синего", 12, 12)

c = Circle("зеленого", 12)

s = Square("красного", 12)

print(r)

print(c)

print(s)

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».

"""

FIGURE\_TYPE = "Круг"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, r\_param):

"""

Класс должен содержать конструктор по параметрам «радиус» и «цвет». В конструкторе создается объект класса «Цвет фигуры» для хранения цвета.

"""

self.r = r\_param

self.fc = FigureColor()

self.fc.colorproperty = color\_param

def square(self):

"""

Класс должен переопределять метод, вычисляющий площадь фигуры.

"""

return math.pi\*(self.r\*\*2)

def \_\_repr\_\_(self):

return '{} {} цвета радиусом {} площадью {}.'.format(

Circle.get\_figure\_type(),

self.fc.colorproperty,

self.r,

self.square()

)

## color.py

class FigureColor:

"""

Класс «Цвет фигуры»

"""

def \_\_init\_\_(self):

self.\_color = None

@property

def colorproperty(self):

"""

Get-аксессор

"""

return self.\_color

@colorproperty.setter

def colorproperty(self, value):

"""

Set-аксессор

"""

self.\_color = value

## figure.py

from abc import ABC, abstractmethod

class Figure(ABC):

"""

Абстрактный класс «Геометрическая фигура»

"""

@abstractmethod

def square(self):

"""

содержит виртуальный метод для вычисления площади фигуры.

"""

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».

"""

FIGURE\_TYPE = "Прямоугольник"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, width\_param, height\_param):

"""

Класс должен содержать конструктор по параметрам «ширина», «высота» и «цвет».

В конструкторе создается объект класса «Цвет фигуры» для хранения цвета.

"""

self.width = width\_param

self.height = height\_param

self.fc = FigureColor()

self.fc.colorproperty = color\_param

def square(self):

"""

Класс должен переопределять метод, вычисляющий площадь фигуры.

"""

return self.width\*self.height

def \_\_repr\_\_(self):

return '{} {} цвета шириной {} и высотой {} площадью {}.'.format(

Rectangle.get\_figure\_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».

"""

FIGURE\_TYPE = "Квадрат"

@classmethod

def get\_figure\_type(cls):

return cls.FIGURE\_TYPE

def \_\_init\_\_(self, color\_param, side\_param):

"""

Класс должен содержать конструктор по параметрам «сторона» и «цвет».

"""

self.side = side\_param

super().\_\_init\_\_(color\_param, self.side, self.side)

def \_\_repr\_\_(self):

return '{} {} цвета со стороной {} площадью {}.'.format(

Square.get\_figure\_type(),

self.fc.colorproperty,

self.side,

self.square()

)

## Результаты выполнения

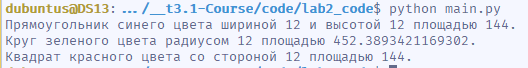


рис. 1 результат выполнения задания

# Вывод

# На данной лабораторной работе я изучил возможности объектно-ориентированного программирования в языке Python.