|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **Министерство образования и науки 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)** |

**Лабораторная работа № 2 по дисциплине «Разработка Интернет-Приложений»**

**Тема работы: " Классы, методы, поля. Наследование, полиморфизм, инкапсуляция. Модули. Работа с пакетным менеджером pip**.**"**

Студентка группы РТ5-51Б Бочарова М.А.

Вариант 2

Москва, 2020

1. **Цель лабораторной работы.**

Изучение объектно-ориентированных возможностей языка Python.

1. **Задание на лабораторную работу**
2. Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием pip.
3. Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.
4. Все файлы проекта (кроме основного файла main.py) должны располагаться в пакете lab\_python\_oop.
5. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab\_python\_oop.
6. Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры.
7. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
8. 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.
9. Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус».
10. 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.
11. В корневом каталоге проекта создайте файл main.py для тестирования Ваших классов. Создайте следующие объекты и выведите о них информацию в консоль
    * Прямоугольник синего цвета шириной 2 и высотой 2.
    * Круг зеленого цвета радиусом 2.
    * Квадрат красного цвета со стороной 2.
    * Также вызовите один из методов внешнего пакета, установленного с использованием pip.

# Ход выполнения лабораторной работы.

**main.py**

from lab\_python\_oop.rectangle import Rectangle

from lab\_python\_oop.krug import Krug

from lab\_python\_oop.kvadrat import Kvadrat

import numpy as np

if \_\_name\_\_ == "\_\_main\_\_":

print("Бочарова Маргарита Александровна, группа РТ5-51Б")

r=Rectangle(2,2,"Зелёный")

k=Krug(2,"Зелёный")

kv=Kvadrat("Зелёный", 2)

print(r)

print(k)

print(kv)

print('\n', np.ones((3, 20), float))

**Цвет**

class Color:

def \_init\_(self):

self.color=None

@property

def \_color(self):

return self.color

@\_color.setter

def \_color(self,value):

self.color=value

**figure.py**

from abc import ABC, abstractmethod

class Figure(ABC):

@abstractmethod

def Square(self):

pass

**Круг**

from lab\_python\_oop.figure import Figure

from lab\_python\_oop.Color import Color

import math

class Krug(Figure):

Type\_figure = "Круг"

@classmethod

def get\_type(cls):

return cls.Type\_figure

def \_\_init\_\_(self, R, color):

self.\_R=R

self.\_color=Color()

self.\_color.color\_figuri=color

def Square(self):

return math.pi \* self.\_R \* self.\_R

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

return "\nТип фигуры: {0} \nРадиус: {1} \nПлощадь: {2} \nЦвет: {3}".format(Krug.get\_type(),self.\_R,self.Square(), self.\_color.color\_figuri)

**Квадрат**

from lab\_python\_oop.rectangle import Rectangle

class Kvadrat(Rectangle):

Type\_figure = "Квадрат"

@classmethod

def get\_type(cls):

return cls.Type\_figure

def \_\_init\_\_(self, color, a):

self.\_a=a

super().\_\_init\_\_(color, a, a)

def Square(self):

return self.\_a \* self.\_a

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

return "\nТип фигуры: {0} \nСторона: {1} \nПлощадь: {2} \nЦвет: {3} ".format(Kvadrat.get\_type(),self.\_a,self.Square(),self.\_color.color\_figuri)

**Прямоугольник**

from lab\_python\_oop.figure import Figure

from lab\_python\_oop.Color import Color

class Rectangle(Figure):

Type\_figure = "Прямоугольник"

@classmethod

def get\_type(cls):

return cls.Type\_figure

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

self.\_width=width

self.\_height=height

self.\_color=Color()

self.\_color.color\_figuri=color

def Square(self):

return self.\_width \* self.\_height

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

return "\nТип фигуры: {0} Ширина: {1} Высота: {2} Площадь: {3} Цвет: {4}\n".format(Rectangle.get\_type(),self.\_width, self.\_height,self.Square(), self.\_color.color\_figuri)

# Результаты работы



