МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
им. Н.Э. Баумана

Факультет “Информатика и системы управления”  
Кафедра “Системы обработки информации и управления”

A logo with a gold star and blue ribbon

Description automatically generated

Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по лабораторной работе №2  
“Объектно-ориентированные возможности языка Python”

**Выполнил:**Студент группы ИУ5-34Б  
Изотов Е.А  
**Преподаватель:**Нардид А.Н.

Москва 2025

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.](https://docs.python.org/3/library/abc.html)
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь.](https://docs.python.org/3/library/functions.html" \l "property)
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.](https://docs.python.org/3/library/math.html)
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.

Код(main.py)

from lab\_python\_oop.rectangle import Rectangle  
from lab\_python\_oop.square import Square  
from lab\_python\_oop.circle import Circle  
import numpy as np  
  
def main():  
 N = 5  
 r = Rectangle(N, N, "синий")  
 c = Circle(N, "зеленый")  
 s = Square(N, "красный")  
  
 print(r)  
 print(c)  
 print(s)  
  
 arr = np.array([1, 2, 3, 4, 5])  
 print("Сумма элементов массива через numpy:", np.sum(arr))  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

Код(figure.py)

from abc import ABC, abstractmethod  
  
class Figure(ABC):  
 @abstractmethod  
 def area(self):  
 pass

Код(ShapeColour.py)

class FigureColor:  
 def \_\_init\_\_(self, color):  
 self.\_color = color  
  
 @property  
 def color(self):  
 return self.\_color  
  
 @color.setter  
 def color(self, value):  
 self.\_color = value

Код(rectangle.py)

from figure import Figure  
from color import FigureColor  
  
class Rectangle(Figure):  
 name = "Прямоугольник"  
  
 def \_\_init\_\_(self, width, height, color):  
 self.width = width  
 self.height = height  
 self.color = FigureColor(color)  
  
 def area(self):  
 return self.width \* self.height  
  
 def \_\_repr\_\_(self):  
 return "{} {}, ширина {}, высота {}, площадь {:.2f}".format(  
 self.name, self.color.color, self.width, self.height, self.area()  
 )

Код(square.py)

from rectangle import Rectangle  
  
class Square(Rectangle):  
 name = "Квадрат"  
  
 def \_\_init\_\_(self, side, color):  
 super().\_\_init\_\_(side, side, color)  
  
 def \_\_repr\_\_(self):  
 return "{} {}, сторона {}, площадь {:.2f}".format(  
 self.name, self.color.color, self.width, self.area()  
 )

Код(circle.py)

import math  
from figure import Figure  
from color import FigureColor  
  
class Circle(Figure):  
 name = "Круг"  
  
 def \_\_init\_\_(self, radius, color):  
 self.radius = radius  
 self.color = FigureColor(color)  
  
 def area(self):  
 return math.pi \* self.radius \* self.radius  
  
 def \_\_repr\_\_(self):  
 return "{} {}, радиус {}, площадь {:.2f}".format(  
 self.name, self.color.color, self.radius, self.area()  
 )

Работа приложения

