**Объектно-ориентированные возможности языка Python.**

Выполнил: Шебеко Николай  
Группа: ИУ5-33Б  
Дата: 15.11.24г.

Описание задания:

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#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. 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.
11. **Дополнительное задание.** Протестируйте корректность работы Вашей программы с помощью модульного теста.

Программный код:

Main.py

from lab\_python\_oop.rectangle import Rectangle  
from lab\_python\_oop.circle import Circle  
from lab\_python\_oop.square import Square  
import matplotlib.pyplot as plt  
  
  
def main():  
 N = 10  
  
 rect = Rectangle(N, N, "синий")  
 circle = Circle(N, "зеленый")  
 square = Square(N, "красный")  
  
 print(rect)  
 print(circle)  
 print(square)  
  
 # Пример использования внешнего пакета matplotlib  
 plt.bar(["Прямоугольник", "Круг", "Квадрат"], [rect.area(), circle.area(), square.area()])  
 plt.title("Площади фигур")  
 plt.show()  
  
if \_\_name\_\_ == "\_\_main\_\_":  
 main()

Rectangle.py

from lab\_python\_oop.geometric\_figure import GeometricFigure  
from lab\_python\_oop.color import FigureColor  
  
class Rectangle(GeometricFigure):  
 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 "{} {} цвета шириной {} и высотой {} с площадью {:.2f}".format(  
 self.figure\_name,  
 self.color.color,  
 self.width,  
 self.height,  
 self.area()  
 )

Circle.py

from math import pi  
from lab\_python\_oop.geometric\_figure import GeometricFigure  
from lab\_python\_oop.color import FigureColor  
  
class Circle(GeometricFigure):  
 figure\_name = "Круг"  
  
 def \_\_init\_\_(self, radius, color):  
 self.radius = radius  
 self.color = FigureColor(color)  
  
 def area(self):  
 return pi \* (self.radius \*\* 2)  
  
 def \_\_repr\_\_(self):  
 return "{} {} цвета радиусом {} с площадью {:.2f}".format(  
 self.figure\_name,  
 self.color.color,  
 self.radius,  
 self.area()  
 )

Square.py

from lab\_python\_oop.rectangle import Rectangle  
  
class Square(Rectangle):  
 figure\_name = "Квадрат"  
  
 def \_\_init\_\_(self, side, color):  
 super().\_\_init\_\_(side, side, color)  
  
 def \_\_repr\_\_(self):  
 return "{} {} цвета со стороной {} с площадью {:.2f}".format(  
 self.figure\_name,  
 self.color.color,  
 self.width,  
 self.area()  
 )

Снимки экрана:  
Изображение выглядит как текст, снимок экрана, дисплей, Прямоугольник

Автоматически созданное описание