**Московский государственный технический**

**университет им. Н. Э. Баумана**

Факультет «Радиотехнический»

Кафедра РТ5 «Системы обработки информации и управления»

Курс «Парадигмы и конструкции языков программирования»

Отчет по лабораторной работе №1

|  |  |  |
| --- | --- | --- |
| Выполнил: |  | Проверил: |
| студент группы РТ5-31Б |  | преподаватель каф. ИУ5 |
| Суровец Е. А. |  | Гапанюк Ю.Е. |
|  |  |  |

Москва, 2024 г.

Задание:

Изображение выглядит как текст, снимок экрана, документ, Шрифт

Автоматически созданное описание

Код программы:

Circle.py

from Lab1.lab\_python\_oop.geom\_figure import GeomFigure  
from Lab1.lab\_python\_oop.figure\_color import FigureColor  
from math import pi  
  
class Circle(GeomFigure):  
 def \_\_init\_\_(self, radius, color):  
 self.radius = radius  
 self.color = FigureColor(color)  
 self.name = "Circle "  
  
 def area(self):  
 return 2 \* pi \* self.radius  
  
 def \_\_repr\_\_(self):  
 return "{} Radius: {} Color: {}".format(self.get\_name(), self.radius, self.color.get\_color())  
  
 def get\_name(self):  
 return self.name

figure\_color.py

class FigureColor():  
 def \_\_init\_\_(self, color):  
 self.color = color  
  
 def get\_color(self):  
 return self.color

geom\_figure.py

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

rectangle.py

from Lab1.lab\_python\_oop.geom\_figure import GeomFigure  
from Lab1.lab\_python\_oop.figure\_color import FigureColor  
  
class Rectangle(GeomFigure):  
 def \_\_init\_\_(self, wid, len, color):  
 self.wid = wid  
 self.len = len  
 self.color = FigureColor(color)  
 self.name = "Rectangle "  
  
 def area(self):  
 return self.wid \* self.len  
  
 def \_\_repr\_\_(self):  
 return "{} Length: {} Width: {} Color: {}".format(self.get\_name(), self.len, self.wid, self.color.get\_color())  
  
 def get\_name(self):  
 return self.name

square.py

from Lab1.lab\_python\_oop.rectangle import Rectangle  
  
class Square(Rectangle):  
 def \_\_int\_\_(self, len, color):  
 self.side = len  
 super().\_\_init\_\_(self.side, self.side, color)  
 self.name = "Square "  
  
 def area(self):  
 return self.len \* self.len  
  
 def \_\_repr\_\_(self):  
 return "{} Side: {} Color: {}".format(self.get\_name\_square(), self.len, self.color.get\_color())  
  
 def get\_name\_square(self):  
 return self.name

main.py

from Lab1.lab\_python\_oop.rectangle import Rectangle  
from Lab1.lab\_python\_oop.circle import Circle  
from Lab1.lab\_python\_oop.square import Square  
  
def main():  
 n = 10  
 rect = Rectangle(n, n, "blue")  
 circ = Circle(n, "green")  
 sqr = Square(n, n, "red")  
 print(rect.\_\_repr\_\_())  
 print(circ.\_\_repr\_\_())  
 print(sqr.\_\_repr\_\_())  
  
if \_\_name\_\_ == '\_\_main\_\_':  
 main()

Результат:

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание