****

Московский государственный технический университет имени Н. Э. Баумана

Факультет «Информатика и системы управления»

Кафедра ИУ5

Отчёт по

лабораторной работе № 2

**«Объектно-ориентированные возможности языка Python»**

Вариант 5

|  |  |  |
| --- | --- | --- |
|  |  | Выполнила:  Калашникова А.В.  Группа ИУ5-54Б |

**Москва**

**2020**

1. **Описание задания**
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». Для вычисления площади используется константа math.pi из модуля math.
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 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.
12. **Текст программы**

GeometricFigure.py:

from abc import ABC, abstractmethod  
  
class geometricFigure(ABC):  
  
 @abstractmethod  
 def area(self) -> object: #абстрактный метод для нахождения площади фигуры  
 pass  
  
 @abstractmethod  
 def printfigure(self):  
 pass

FigureColor.py:

class figureColor:  
  
 def \_\_init\_\_(self):  
 self.\_color = None  
  
 # геттер  
 @property  
 def colorf(self):  
 return self.\_color  
  
 # сеттер  
 @colorf.setter  
 def colorf(self, value):  
 self.\_color = value

rectangle.py:

from lab\_python\_oop.GeometricFigure import geometricFigure  
from lab\_python\_oop.FigureColor import figureColor  
  
  
class Rectangle(geometricFigure):  
 figure = **"Прямоугольник"** def printfigure(self):  
 return self.figure  
  
 def \_\_init\_\_(self, figcolor, width, height):  
 self.w = width  
 self.h = height  
 self.col = figureColor()  
 self.col.colorf=figcolor  
  
 def area(self):  
 return self.w \* self.h  
  
 def \_\_repr\_\_(self):  
 return (**"{}, {} цвет, ширина = {}; высота = {}; площадь = {}"**.format(self.figure, self.col.colorf, self.w, self.h, self.area()))

circle.py:

import math  
from lab\_python\_oop.GeometricFigure import geometricFigure  
from lab\_python\_oop.FigureColor import figureColor  
  
  
class Circle(geometricFigure):  
 figure = **"Круг"** def printfigure(self):  
 return self.figure  
  
 def \_\_init\_\_(self, figcolor, radius):  
 self.r = radius  
 self.col = figureColor()  
 self.col.colorf=figcolor  
  
 def area(self):  
 return math.pi \* pow(self.r, 2)  
  
 def \_\_repr\_\_(self):  
 return **"{}, {} цвет, радиус = {}; площадь = {}"**.format(self.figure, self.col.colorf, self.r, self.area())

square.py:

from lab\_python\_oop.rectangle import Rectangle  
from lab\_python\_oop.FigureColor import figureColor  
  
  
class Square(Rectangle):  
 figure: str = **"Квадрат"** def printfigure(self):  
 return self.figure  
  
 def \_\_init\_\_(self, figcolor, side):  
 self.s = side  
 self.col = figureColor()  
 self.col.colorf = figcolor  
  
 def area(self):  
 return pow(self.s, 2)  
  
 def \_\_repr\_\_(self):  
 return (**"{}, {} цвет, сторона = {}; площадь = {}"**.format(self.figure, self.col.colorf, self.s, self.area()))

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():  
 rect = Rectangle(**"синий"**, 5, 5)  
 circ = Circle(**"зеленый"**, 5)  
 sq = Square(**"красный"**, 5)  
  
 print(rect)  
 print(circ)  
 print(sq)  
  
if \_\_name\_\_ == **"\_\_main\_\_"**:  
 main()

1. **Экранные формы с примерами выполнения программы**

