Московский государственный технический

университет им. Н.Э. Баумана.

Факультет «Информатика и управление»

Кафедра «Системы обработки информации и управления»

Курс «Базовые компоненты интернет технологий»

Отчет по лабораторной работе №2

«Объектно-ориентированные возможности языка Python.»

Выполнил:   
студент группы ИУ5-31Б   
Калинников Даниил

Подпись и дата:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

24.10.2021

Проверил:  
преподаватель кафедры ИУ5  
Гапанюк Ю. Е.

Подпись и дата:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Москва, 2021 г.

**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.
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры.
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.
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.

**Текст программы main():**

from lab\_python\_oop import rectangle,circle,quadrate  
  
rect = rectangle.Rectangle()  
  
rect.setc("синий")  
rect.seth(8)  
rect.setw(8)  
rect.setname()  
print(rect.getname(), rect.repr())  
  
circle = circle.Circle()  
circle.setc("зеленый")  
circle.setr(8)  
circle.setname()  
print(circle.getname(), circle.repr())  
  
quadrate = quadrate.Quadrate()  
quadrate.setc("красный")  
quadrate.setw(8)  
quadrate.setname()  
print(quadrate.getname(), quadrate.repr())

class Colour():  
 def \_\_init\_\_(self):  
 self.colour = None  
 def getc(self):  
 return self.colour  
  
 def setc(self, value):  
 self.colour = value

from abc import ABC, abstractmethod  
  
class Geometry\_f(ABC):  
 @abstractmethod  
 def square(self):  
 return round(self.w \* self.h, 2)

from lab\_python\_oop import colour, geometry\_f  
  
class Rectangle(geometry\_f.Geometry\_f):  
 def \_\_init\_\_(self):  
 self.w = None  
 self.h = None  
 self.c = colour.Colour()  
 self.name = None  
  
 def setname(self):  
 Name = input("Введите название фигуры: ")  
 self.name = Name  
  
 def getname(self):  
 return self.name  
  
 def square(self):  
 return self.w \* self.h  
  
 def geth(self):  
 return self.h  
 def getw(self):  
 return self.w  
 def getc(self):  
 return self.c.getc()  
  
 def setw(self, value):  
 self.w = value  
  
 def seth(self, value):  
 self.h = value  
  
 def setc(self, string):  
 self.c.setc(string)  
  
 def repr(self):  
 a = Rectangle.square(self)  
 colour = self.getc()  
 return "Основыне параметры фигуры: ширина - {0}, высота - {1}, площадь - {2}, цвет - {3}".format(self.w,self.h,a,colour)

from lab\_python\_oop import colour, geometry\_f  
  
class Quadrate(geometry\_f.Geometry\_f):  
 def \_\_init\_\_(self):  
 self.w = None  
 self.c = colour.Colour()  
 self.name = None  
  
 def setname(self):  
 Name = input("Введите название фигуры: ")  
 self.name = Name  
  
 def getname(self):  
 return self.name  
  
 def square(self):  
 return self.w \* self.w  
  
 def getw(self):  
 return self.w  
 def getc(self):  
 return self.c.getc()  
  
 def setw(self, value):  
 self.w = value  
  
 def setc(self, string):  
 self.c.setc(string)  
  
 def repr(self):  
 a = Quadrate.square(self)  
 colour = self.getc()  
 return "Основыне параметры фигуры: длина строноны - {0},площадь - {2}, цвет - {3}".format(self.w,self.w,a,colour)

from lab\_python\_oop import colour, geometry\_f  
import math  
  
class Circle(geometry\_f.Geometry\_f):  
 def \_\_init\_\_(self):  
 self.r = None  
 self.c = colour.Colour()  
 self.name = None  
  
 def setname(self):  
 Name = input("Введите название фигуры: ")  
 self.name = Name  
  
 def getname(self):  
 return self.name  
  
 def square(self):  
 return math.pi\*self.r\*self.r  
  
 def getr(self):  
 return self.r  
 def getc(self):  
 return self.c.getc()  
  
 def setr(self, value):  
 self.r = value  
 def setc(self, string):  
 self.c.setc(string)  
  
 def repr(self):  
 a = Circle.square(self)  
 colour = self.getc()  
 return "Основыне параметры фигуры: радиус - {0}, площадь - {1}, цвет - {2}".format(self.r,a,colour)

**Экранная форма с примерами выполнения программы:**

