



**Министерство образования и науки Российской Федерации Федеральное
государственное бюджетное образовательное учреждение высшего
образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)**

ФАКУЛЬТЕТ

ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА

СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ

**Отчет по лабораторной работе № 2
«Python. Объектно-ориентированные возможности»
по курсу “Разработка интернет-приложений”**

Исполнитель:

Студент группы ИУ5-52

Губайдуллина Карина

_____ 24.09.2018

Москва, 2018

Задание лабораторной работы

Необходимо разработать программу, реализующую работу с классами.

1. Программа должна быть разработана в виде консольного приложения на языке Python 3.
2. Все файлы проекта (кроме основного файла main.py) должны располагаться в пакете lab_python_oop.
3. Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab_python_oop.
 1. 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>
 2. 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>
 3. 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.
 4. 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>
 5. Класс «Квадрат» наследуется от класса «Прямоугольник». Класс должен содержать конструктор по длине стороны.
 6. Для классов «Прямоугольник», «Квадрат», «Круг»:
 1. Определите метод "repr", который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод format - <https://pyformat.info/>
 2. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
4. 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:
 1. Прямоугольник синего цвета шириной 3 и высотой 2.
 2. Круг зеленого цвета радиусом 5.
 3. Квадрат красного цвета со стороной 5.

Исходный код

main.py

```
from lab_python_oop.Rectangle import Rectangles
from lab_python_oop.Circle import Circles
from lab_python_oop.Square import Squares
if __name__ == "__main__":
    Rect = Rectangles('blue', 2, 3)
    print(Rect.repr())
    Circ = Circles('green', 5)
    print(Circ.repr())
    Squa = Squares('red', 5)
    print(Squa.repr())
```

Circle.py

```
from lab_python_oop.GeometricFigure import GeometricFigures
from lab_python_oop.ShapeColor import Colors from math
import pi
class Circles(GeometricFigures,
Colors):
    def __init__(self, color, radius, kind =
"Circle"):
        self.color = color
self.radius = radius #радиус
self.kind = kind
    def CircSpace(space, radius):
return pi * radius * radius
    def
Kind(self):
    return Circles.kind
    def
repr(self):
    return '{} : color {}, radius {}, space {}'.format(self.kind,
self.color, self.radius, self.CircSpace(self.radius))
```

GeometricFigure.py

```
from abc import ABCMeta, abstractmethod
class
GeometricFigures():

    __GeometricFigures__=ABCMeta
    def __init__(self):
        """Constructor"""
pass

    @abstractmethod
def space(self):
    """Вычисление площади"""
pass
```

Rectangle.py

```
from lab_python_oop.GeometricFigure import GeometricFigures from
lab_python_oop.ShapeColor import Colors
class Rectangles(GeometricFigures,
Colors):
    def __init__(self, color, height, width, kind =
"Rectangle"):
        self.color = color
self.height = height #высота
self.width = width #ширина
self.kind = kind
    def
RectSpace(self):
    return self.height * self.width
    def
Kind(self):
    return Rectangles.kind
    def
repr(self):
    return '{} : color {}, height {}, width {}, space
```

```
{}}'.format(self.kind, self.color, self.height, self.width, self.RectSpace())
```

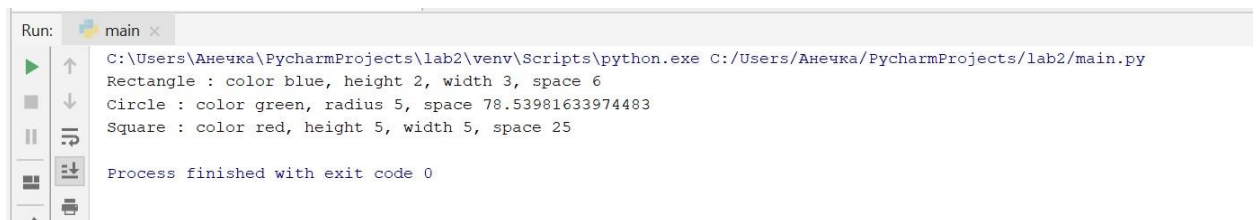
ShapeColor.py

```
class Colors(object):  
    def __init__(self,  
color):  
        """Constructor"""  
self.color = color  
    def  
color(self):  
        return self.color
```

Square.py

```
from lab_python_oop.Rectangle import Rectangles  
class  
Squares(Rectangles):  
    def __init__(self, color, size, kind =  
"Square"):  
        self.color = color  
self.height = size  
self.width = size          self.kind  
= kind  
    def  
repr(self):  
        return '{} : color {}, height {}, width {}, space  
{}}'.format(self.kind, self.color, self.height, self.width, self.RectSpace())
```

Результаты выполнения



```
Run: main x  
C:\Users\Анечка\PycharmProjects\lab2\venv\Scripts\python.exe C:\Users\Анечка\PycharmProjects\lab2/main.py  
Rectangle : color blue, height 2, width 3, space 6  
Circle : color green, radius 5, space 78.53981633974483  
Square : color red, height 5, width 5, space 25  
Process finished with exit code 0
```