

**Московский государственный технический
университет им. Н.Э. Баумана.**

Факультет «Системы обработки информации и управления»

**Отчет по лабораторной работе №2 по курсу
«Разработка Интернет-Приложений»**

Тема: «Введение в Python.»

Выполнил:
студент группы ИУ5-53

Кондратьев М.В.

Подпись и дата:

Проверил:
преподаватель каф.
ИУ5

Гапанюк Ю.Е.

Подпись и дата:

Москва, 2018 г.

Задание

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 python_lab_oop import Rectangle, Circle, Square

rect = Rectangle(3, 2, "Синий")
circle = Circle(5, "Зеленый")
square = Square(5, "Красный")
print(rect, circle, square)
```

GeometricFigure.py:

```
from abc import *

class GeometricFigure (metaclass=ABCMeta):
    @abstractmethod
    def find_square(self):
        pass
```

FigureColor.py:

```
class FigureColour:
    def __init__(self, colour="White"):
        self.colour = colour

    def get_colour(self):
        return self.colour

    def set_colour(self, colour):
        self.colour = colour

    def del_colour(self):
        del self.colour

    def __repr__(self):
        return self.colour
```

Rectangle.py:

```
from python_lab_oop.FigureColour import FigureColour
from python_lab_oop.GeometricFigure import GeometricFigure

class Rectangle(GeometricFigure):
    figure_name = "Прямоугольник"

    def __init__(self, width, height, colour):
        self.width = width
        self.height = height
        self.colour = FigureColour(colour)

    def find_square(self):
        return self.width*self.height

    def __repr__(self):
        return ("Фигура - {}, Ширина - {}, Высота - {}, Цвет - {}, Площадь - {} \n"
                .format(self.figure_name, self.width, self.height,
                        self.colour, self.find_square()))

    def get_name(self):
        return self.figure_name
```

Square.py:

```
class Square(Rectangle):
    figure_name = "Квадрат"

    def __init__(self, side, colour):
        self.side = side
        self.colour = FigureColour(colour)

    def find_square(self):
        return self.side**2

    def __repr__(self):
        return ("Фигура - {}, Сторона - {}, Цвет - {}, Площадь - {} \n"
                .format(self.figure_name, self.side, self.colour,
                        self.find_square()))
```

Circle.py:

```
from math import pi

from python_lab_oop.FigureColour import FigureColour
from python_lab_oop.GeometricFigure import GeometricFigure

class Circle(GeometricFigure):
    figure_name = "Круг"

    def __init__(self, radius, colour):
        self.radius = radius
        self.colour = FigureColour(colour)

    def find_square(self):
        return pi*self.radius**2

    def __repr__(self):
        return ("Фигура - {}, Радиус - {}, Цвет - {}, Площадь - {}\n"
                .format(self.figure_name, self.radius, self.colour,
                self.find_square()))

    def get_name(self):
        return self.figure_name
```

Результат выполнения:

Фигура - Прямоугольник, Ширина - 3, Высота - 2, Цвет - Синий, Площадь - 6
Фигура - Круг, Радиус - 5, Цвет - Зеленый, Площадь - 78.53981633974483
Фигура - Квадрат, Сторона - 5, Цвет - Красный, Площадь - 25