

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

Факультет «Информатика и системы управления»

Кафедра ИУ5. Курс «Разработка Интернет-Приложений»

Отчет по лабораторной работе № 2

«Объектно-ориентированные возможности языка Python»

Выполнил:

студент группы ИУ5-53

Терентьев Владислав

Москва, 2020 г.

## 1. 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. Подробнее про абстрактные классы и методы Вы можете прочитать [здесь](#).
6. Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать [здесь](#).
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»:
10. Определите метод `"repr"`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Используйте метод `format` - <https://pyformat.info/>
11. Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
12. В корневом каталоге проекта создайте файл `main.py` для тестирования Ваших классов (используйте следующую конструкцию - [https://docs.python.org/3/library/\\_\\_main\\_\\_.html](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):
13. Прямоугольник синего цвета шириной N и высотой N.
14. Круг зеленого цвета радиусом N.
15. Квадрат красного цвета со стороной N.
16. Также вызовите один из методов внешнего пакета, установленного с использованием `pip`.

## 2. Текст программы

### 2.1. `lab_python_oop\GeomFigure.py`

```
from abc import ABC, abstractmethod

class GeomFigure(ABC):
    def __init__(self):
        self.shape
        self.__area
```

```

@property
def area(self):
    return self.__area

@area.setter
def area(self, area):
    if area < 0:
        raise Exception("Площадь не может быть меньше 0")
    else:
        self.__area = area

@abstractmethod
def CalcArea(self):
    pass

def name(self):
    return self.shape

```

## 2.2. lab\_python\_oop\FigureColor.py

```

import pymorphy2

morph = pymorphy2.MorphAnalyzer()

class FigureColor:
    def __init__(self, color):
        self.color = color

    @property
    def color(self):
        return self.__color

    @color.setter
    def color(self, color):
        color = morph.parse(color)[0]
        self.__color = color.inflect({'gent'}).word

```

## 2.3. lab\_python\_oop\Rectangle.py

```

from lab_python_oop.GeomFigure import GeomFigure
from lab_python_oop.FigureColor import FigureColor

class Rectangle(GeomFigure):
    def __init__(self, weight, height, color):
        self.shape = 'Прямоугольник'
        self.weight = weight
        self.height = height
        self.color = FigureColor(color)
        self.area = self.CalcArea()

```

```

def CalcArea(self):
    return self.height * self.weight

def __repr__(self):
    return "{0} площадью {1} {2} цвета шириной {3} и высотой {4}".format(self
.shape, self.area, self.color.color, self.weight, self.height)

```

## 2.4. lab\_python\_oop\Circle.py

```

from lab_python_oop.GeomFigure import GeomFigure
from lab_python_oop.FigureColor import FigureColor
from math import pi

class Circle(GeomFigure):
    def __init__(self, radius, color):
        self.shape = 'Окружность'
        self.radius = radius
        self.color = FigureColor(color)
        self.area = self.CalcArea()

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

    def __repr__(self):
        return "{0} площадью {1:.2f} {2} цвета радиусом {3}".format(self.shape, s
elf.area, self.color.color, self.radius)

```

## 2.5. lab\_python\_oop\Square.py

```

from lab_python_oop.Rectangle import Rectangle

class Square(Rectangle):
    def __init__(self, leng, color):
        super().__init__(leng, leng, color)
        self.shape = 'Квадрат'

    def __repr__(self):
        return "{0} площадью {1} {2} цвета со стороной {3}".format(self.shape, se
lf.area, self.color.color, self.weight)

```

## 2.6. 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():

```

```
r, c, s = Rectangle(21, 21, 'синий'), Circle(21, 'зеленый'), Square(21, 'красный')
print(r, c, s, sep='\n')
#print(s.name())

if __name__ == "__main__":
    main()
```

### 3. Экранные формы с примерами выполнения программы

OUTPUT   TERMINAL   DEBUG CONSOLE   PROBLEMS

Try the new cross-platform PowerShell <https://aka.ms/pscore6>

```
PS C:\Users\webma\iCloudDrive\MGTU\5\1 DIA\github\DIA\lab2> & C:,
Прямоугольник площадью 441 синего цвета шириной 21 и высотой 21
Окружность площадью 1385.44 зелёного цвета радиусом 21
Квадрат площадью 441 красного цвета со стороной 21
PS C:\Users\webma\iCloudDrive\MGTU\5\1 DIA\github\DIA\lab2>
```

### 4. Ссылка на репозиторий

<https://github.com/iYroglif/DIA>