

Министерство науки и высшего образования 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)

**Отчет по лабораторной работе № 2 по курсу
Базовые компоненты интернет-технологий**

“Объектно-ориентированные возможности языка Python”

ПРЕПОДАВАТЕЛЬ

Гапанюк Ю. Е.

(подпись)

ИСПОЛНИТЕЛЬ:

студентка группы ИУ5-
35Б

Гурова М.Д.

(подпись)

" " _____ 2021 г.

Москва - 2021

Задание:

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»:
 - 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 rectangle import Rectangle
from circle import Circle
from square import Square
import emoji

def main():
    r = Rectangle("синего", 3, 3)
    c = Circle("зеленого", 3)
    s = Square("красного", 3)
```

```

print(r)
print(c)
print(s)
print()
result =
emoji.emojize(':red_circle::red_circle::red_circle::red_circle::red_circle:')
print(result)

if __name__ == "__main__":
    main()

```

Circle

```

from figure import Figure
from color import FigureColor
import math

class Circle(Figure):
    """
    Класс «Круг» наследуется от класса «Геометрическая фигура».
    """
    FIGURE_TYPE = "Круг"

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __init__(self, color_param, r_param):
        """
        Класс должен содержать конструктор по параметрам «ради-
        ус» и «цвет». В конструкторе создается объект класса «Цвет фигу-
        ры» для хранения цвета.
        """
        self.r = r_param
        self.fc = FigureColor()
        self.fc.colorproperty = color_param

    def square(self):
        """

```

Класс должен переопределять метод, вычисляющий площадь фигуры.

```
    """
    return math.pi*(self.r**2)

def __repr__(self):
    return '{} {} цвета радиусом {} площадью {}'.format(
        Circle.get_figure_type(),
        self.fc.colorproperty,
        self.r,
        self.square()
    )
```

Color

```
class FigureColor:
    """
    Класс «Цвет фигуры»
    """

    def __init__(self):
        self._color = None

    @property
    def colorproperty(self):
        """
        Get-аксесор
        """
        return self._color

    @colorproperty.setter
    def colorproperty(self, value):
        """
        Set-аксесор
        """
        self._color = value
```

Figure

```
from abc import ABC, abstractmethod
```

```

class Figure(ABC):
    """
    Абстрактный класс «Геометрическая фигура»
    """

    @abstractmethod
    def square(self):
        """
        содержит виртуальный метод для вычисления площади фигу-
ры.
        """
        pass
from figure import Figure
from color import FigureColor

```

Rectangle

```

class Rectangle(Figure):
    """
    Класс «Прямоугольник» наследуется от класса «Геометрическая
фигура».
    """

    FIGURE_TYPE = "Прямоугольник"

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __init__(self, color_param, width_param, height_param):
        """
        Класс должен содержать конструктор по параметрам «шири-
на», «высота» и «цвет». В конструкторе создается объект класса
«Цвет фигуры» для хранения цвета.
        """
        self.width = width_param
        self.height = height_param
        self.fc = FigureColor()
        self.fc.colorproperty = color_param

```

```

def square(self):
    """
    Класс должен переопределять метод, вычисляющий площадь
    фигуры.
    """
    return self.width*self.height

def __repr__(self):
    return '{} {} цвета шириной {} и высотой {} площадью
    {}.format(
        Rectangle.get_figure_type(),
        self.fc.colorproperty,
        self.width,
        self.height,
        self.square()
    )

```

Square

```

from rectangle import Rectangle

```

```

class Square(Rectangle):
    """
    Класс «Квадрат» наследуется от класса «Прямоугольник».
    """
    FIGURE_TYPE = "Квадрат"

    @classmethod
    def get_figure_type(cls):
        return cls.FIGURE_TYPE

    def __init__(self, color_param, side_param):
        """
        Класс должен содержать конструктор по параметрам «сторо-
        на» и «цвет».
        """
        self.side = side_param
        super().__init__(color_param, self.side, self.side)

```

```

def __repr__(self):
    return '{0} {1} цвета со стороной {2} площадью {3}'.format(
        Square.get_figure_type(),
        self.fc.colorproperty,
        self.side,
        self.square()
    )

```

Пример работы программы:

```

mac@MacBook-Pro-Mac lab02py % mkvirtualenv lab_python_oop
zsh: command not found: mkvirtualenv
mac@MacBook-Pro-Mac lab02py % python3 -m venv lab_python_oop
mac@MacBook-Pro-Mac lab02py % source venv/bin/activate
source: no such file or directory: venv/bin/activate
mac@MacBook-Pro-Mac lab02py % source lab_python_oop/bin/activate
(lab_python_oop) mac@MacBook-Pro-Mac lab02py % pip3 list
Package      Version
-----
pip           19.2.3
setuptools    41.2.0

(lab_python_oop) mac@MacBook-Pro-Mac lab02py % pip3 install emoji
Collecting emoji
  Using cached https://files.pythonhosted.org/packages/04/f1/efe9360dbbdcf045620d8fb2fb79df36468a0a616578e907ccf0c71b52d7/emoji-1.5.0.tar.gz
Installing collected packages: emoji
  Running setup.py install for emoji ... done
Successfully installed emoji-1.5.0
WARNING: You are using pip version 19.2.3, however version 21.2.4 is available.
You should consider upgrading via the 'pip install --upgrade pip' command.
(lab_python_oop) mac@MacBook-Pro-Mac lab02py % cd /Users/mac/Desktop/lab02py ; /usr/bin/env /Users/mac/Desktop/lab02py/lab_python_oop/bin/python /Users/mac/.vscode/extensions/ms-python.python-2021.9.1246542782/pythonFiles/lib/python/debugpy/launcher 56251 -- /Users/mac/Desktop/lab02py/lab_python_oop/main.py
Прямоугольник синего цвета шириной 3 и высотой 3 площадью 9.
Круг зеленого цвета радиусом 3 площадью 28.274333882308138.
Квадрат красного цвета со стороной 3 площадью 9.

●●●●●
(lab_python_oop) mac@MacBook-Pro-Mac lab02py % pip3 list
Package      Version
-----
emoji         1.5.0
pip           19.2.3
setuptools    41.2.0
WARNING: You are using pip version 19.2.3, however version 21.2.4 is available.

```