



**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)**

**Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»**

**Курс «Базовые компоненты интернет-технологий»
Отчет по лабораторной работе №2**

«Объектно-ориентированные возможности языка Python»

**Выполнил:
студент группы ИУ5-33Б
Климов Н.С.**

**Проверил:
Канев А.И.**

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`
 - 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. 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:

`figure.py`

```

from abc import ABC, abstractmethod

class Figure(ABC):

    @abstractmethod
    def square(self):

        pass

```

rectangle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor

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 «ширина», «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.py

```

from lab_python_oop.rectangle import Rectangle

class Square(Rectangle):

    FIGURE_TYPE = "Квадрат"

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

```

```

def __init__(self, color_param, side_param):

    self.side = side_param
    super().__init__(color_param, self.side, self.side)

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

```

color.py

```

class FigureColor:

    def __init__(self):
        self._color = None

    @property
    def colorproperty(self):

        return self._color

    @colorproperty.setter
    def colorproperty(self, value):

        self._color

```

circle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.color import FigureColor
import math

class Circle(Figure):

    FIGURE_TYPE = "Круг"

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

    def __init__(self, color_param, r_param):

        self.r = r_param
        self.fc = FigureColor()
        self.fc.colorproperty = color_param

    def square(self):

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

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

```

init .py

main.py

```
from lab_python_oop import *

def main():
    r = Rectangle("голубой", 21, 21)
    c = Circle("зеленый", 21)
    s = Square("красный", 21)
    print(r)
    print(c)
    print(s)
    print("loading...")
    for i in tqdm(range(100), colour = "blue"):
        sleep(random.uniform(0.01, 0.1))

    print("successfully")

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

Экранные формы с примерами выполнения программы

```
(base) 192:Lab2 evgeniayurova$ source venv/bin/activate
(venv) (base) 192:Lab2 evgeniayurova$ python main.py
Прямоугольник голубой цвета шириной 21 и высотой 21 площадью 441.
Круг зеленый цвета радиусом 21 площадью 1385.4423602339987.
Квадрат красный цвета со стороной 21 площадью 441.
loading...
100%|██████████████████████████████████████████████████████████████████████████████| 100/100 [00:05<00:00, 18.50it/s]
successFully
(venv) (base) 192:Lab2 evgeniayurova$
```

TODD Problems Terminal Python Packages Python Console Event Log