

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

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

Курс «Парадигмы и конструкции языков программирования»  
Отчет по лабораторной работе №5  
«Объектно-ориентированные возможности языка Python»

Выполнил:  
студент группы ИУ5-33Б  
Смирнов Артём

Проверил:  
преподаватель каф. ИУ5  
Гапанюк Ю. Е.

Москва, 2023 г.

## 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.
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 "герг", 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`.
11. **Дополнительное задание.** Протестируйте корректность работы Вашей программы с помощью модульного теста.

## Текст Программы:

### Файл **abstract.py**

```
import abc

class figure(abc.ABC):
    @abc.abstractmethod
    def square(self):
        pass

    @abc.abstractmethod
    def repr(self):
        pass

    @abc.abstractmethod
    def get_na
```

### Файл **circle.py**

```
from lab_python_oop import abstract, colour
from math import pi

class circle(abstract.figure):
    def __init__(self, rad=10, colour_="Белый "):
        self.__name = "Круг"
        self.__colour = colour.colour(colour_)
        self.__rad = rad

    def square(self):
        res = pi * self.__rad**2
        return res

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

```

def get_colour(self):
    return self.__colour

def repr(self):
    print(
        f"Название фигуры: {self.get_name()}\n"
        f"Цвет фигуры: {self.get_colour()}\n"
        f"Площадь фигуры: {self.square()}\n" + "\n"
    )

```

### **Файл colour.py**

```

class colour:
    def __init__(self, colour_="Белый"):
        self.colour = colour_

    def __str__(self):
        return self.colour

```

### **Файл rectangle.py**

```

from lab_python_oop import abstract, colour

class rectangle(abstract.figure):
    def __init__(self, height=0, length=0, colour_="Белый"):
        self._length = length
        self.__height = height
        self._colour = colour.colour(colour_)
        self._name = "Прямоугольник"

    def square(self):
        res = self.__height * self._length
        return res

```

```

def get_name(self):
    return self._name

def get_colour(self):
    return self._colour

def repr(self):
    print(
        f"Название фигуры: {self.get_name()}\n"
        f"Цвет фигуры: {self.get_colour()}\n"
        f"Площадь фигуры: {self.square()}\n" + "\n"
    )

```

## Файл square.py

```

from lab_python_oop import abstract, colour, rectangle

class kvadrat(rectangle.rectangle):
    def __init__(self, length=10, colour_="Белый"):
        rectangle.rectangle.__init__(self, length=length, colour_=colour_)
        self.__name = "Квадрат"

    def square(self):
        res = self._length**2
        return res

    def get_name(self):
        return self.__name

    def get_colour(self):
        return self._colour

    def repr(self):
        print(
            f"Название фигуры: {self.get_name()}\n"
            f"Цвет фигуры: {self.get_colour()}\n"

```

```
        f"Площадь фигуры: {self.square()}\n" + "\n"  
    )
```

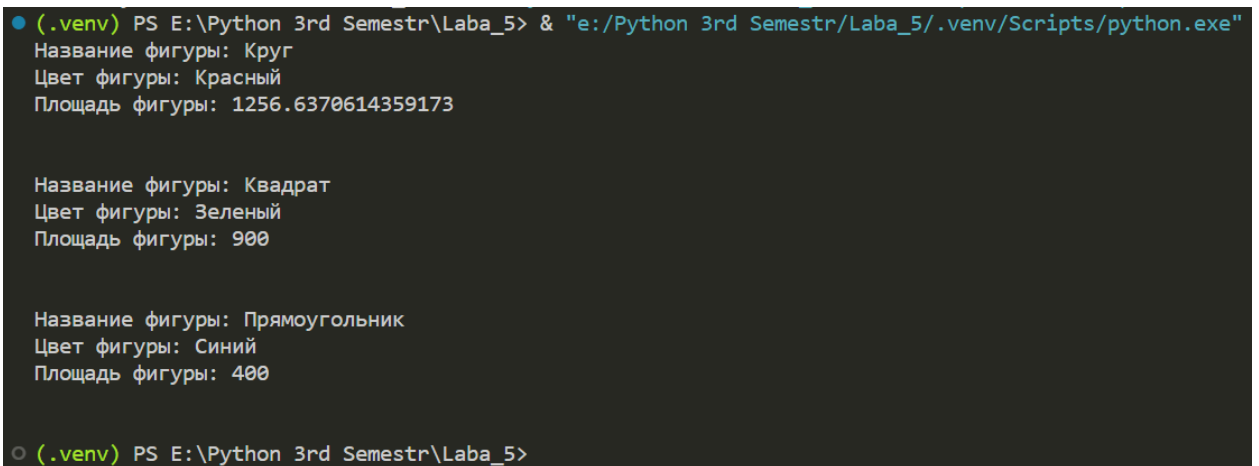
## Файл main.py

```
from lab_python_oop import abstract , circle , square , rectangle  
import black
```

```
def main():  
    circle_ = circle.circle(20 , 'Красный')  
    circle_.repr()  
  
    kvadrat_ = square.kvadrat(30 , 'Зеленый')  
    kvadrat_.repr()  
  
    rectangle_ = rectangle.rectangle(10, 40 , 'Синий')  
    rectangle_.repr()
```

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

## Результат выполнения:



```
(.venv) PS E:\Python 3rd Semestr\Laba_5> & "e:/Python 3rd Semestr/Laba_5/.venv/Scripts/python.exe"  
Название фигуры: Круг  
Цвет фигуры: Красный  
Площадь фигуры: 1256.6370614359173  
  
Название фигуры: Квадрат  
Цвет фигуры: Зеленый  
Площадь фигуры: 900  
  
Название фигуры: Прямоугольник  
Цвет фигуры: Синий  
Площадь фигуры: 400  
  
(.venv) PS E:\Python 3rd Semestr\Laba_5>
```