

**Московский государственный технический
Университет 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:
Власов Александр,
ИУ5-33Б

Проверил:
Гапанюк Е.Ю.

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. 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 "get", 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). 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`.

Текст программы

Main.py

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

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()
```

Abstract.py

```
import abc

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

    @abc.abstractmethod
    def repr(self):
        pass

    @abc.abstractmethod
    def get_name(self):
        pass
```

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"
    )

```

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"
        )

```

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"
        )

```

Colour.py

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

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

Примеры выполнения программы

```
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
```