

Факультет РТ Радиотехнический

Кафедра ИУ5 Системы обработки информации и управления

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

Исполнитель

Студент группы РТ5-31Б

Корсаков Н.А.

“___” _____ 2021 г.

Проверил

Доцент кафедры ИУ5

Гапанюк Ю.Е.

“___” _____ 2021 г.

Содержание

1. Описание задания.....	3
2. Текст программы.....	4
3. Экранные формы с примерами выполнения программы.....	7

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.
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.
10. Для классов «Прямоугольник», «Квадрат», «Круг»:
 - Определите метод `getr`, который возвращает в виде строки основные параметры фигуры, ее цвет и площадь. Использовать метод `format`.
 - Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
11. В корневом каталоге проекта создать файл `main.py` для тестирования классов. Создать следующие объекты и вывести о них информацию в консоль ($N = 11$ - номер 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`.

2. Текст программы

main.py

```
from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
import math
```

```
def main():
    r = Rectangle("синего", 11, 10)
    c = Circle("зеленого", 11)
    s = Square("красного", 11)
    print(r)
    print(c)
    print(s)
    print(math.factorial(12))
```

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

figure.py

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

```
class Figure(ABC):
    """
    Абстрактный класс «Геометрическая фигура»
    """
    @abstractmethod
    def square(self):
        """
        содержит виртуальный метод для вычисления площади фигуры.
        """
        pass
```

color.py

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

    def __init__(self):
        self._color = None

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

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

```

"""
Set-аксесор
"""
self._color = value

```

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()
        )

```

circle.py

```

from lab_python_oop.figure import Figure
from lab_python_oop.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 «радиус» и «цвет». В
    конструкторе создается объект класса «Цвет фигуры» 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()
    )

```

square.py

```

from lab_python_oop.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 «сторона» и «цвет».
        """
        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()
        )

```

3. Экранные формы с примерами выполнения программы

```
Прямоугольник Синего цвета шириной 11 и высотой 10 площадью 110.  
Круг Зеленого цвета радиусом 11 площадью 380.132711084365.  
Квадрат Красного цвета со стороной 11 площадью 121.  
479001600
```