

**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:

студент группы ИУ5-32
Вольвач Михаил

Подпись и дата: 20.10.2022

Проверил:

преподаватель каф. ИУ5
Гапанюк Ю. Е.

Подпись и дата: 20.10.2022

Москва, 2022

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

Текст программы

Файл Figure.py

```
from abc import ABC, abstractmethod

class Figure:
    """
    Абстрактный класс "Геометрическая фигура"
    """

    @abstractmethod
    def square(self):
        """
        Виртуальный метод вычисления площади фигуры
        """
        pass
```

Файл FigureColor.py

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

    def __init__(self):
        self._color = None

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

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

Файл Rectangle.py

```
from lab_python_oop.Figure import Figure
from lab_python_oop.FigureColor import FigureColor

class Rectangle(Figure):
    """
    Класс "Прямоугольник"
    """
    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.color = color_param
self.width = width_param
self.height = height_param
self.fc = FigureColor()
self.fc.color_property = color_param

def square(self):
    """
    Вычисление площади фигуры
    """
    return self.width * self.height

def __repr__(self):
    return '{} {} цвета, шириной {} и высотой {}, площадью {:.3f}.'.format(
        Rectangle.get_figure_type(),
        self.fc.color_property,
        self.width,
        self.height,
        self.square()
    )

```

Файл Square.py

```
from lab_python_oop.Rectangle import Rectangle
```

```

class Square(Rectangle):
    """
    Класс "Квадрат"
    """
    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 "цвет" и "сторона"
        """
        self.side = side_param
        super().__init__(color_param, self.side, self.side)

    def __repr__(self):
        return '{} {} цвета со стороной {}, площадью {:.3f}.'.format(
            self.get_figure_type(),
            self.fc.color_property,
            self.side,
            self.square()
        )

```

Файл Circle.py

```
import math
```

```

from lab_python_oop.Figure import Figure
from lab_python_oop.FigureColor import FigureColor

```

```

class Circle(Figure):
    """

```

```

Класс "Круг"
"""

FIGURE_TYPE = "Круг"

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

def __init__(self, color_param, r_param):
    self.fc = FigureColor()
    self.fc.color_property = color_param
    self.r = r_param

def square(self):
    return 2 * math.pi * self.r ** 2

def __repr__(self):
    return '{} {} цвета, радиуса {}, площадью {:.3f}'.format(
        self.get_figure_type(),
        self.fc.color_property,
        self.r,
        self.square()
    )

```

Файл main.py

```

from prettytable import PrettyTable

from lab_python_oop.Rectangle import Rectangle
from lab_python_oop.Circle import Circle
from lab_python_oop.Square import Square

def main():
    r = Rectangle("синего", 3, 2)
    c = Circle("зеленого", 5)
    s = Square("красного", 5)

    x = PrettyTable()
    x.field_names = ["Название фигуры", "Цвет фигуры", "Площадь фигуры"]
    x.add_rows([
        [r.get_figure_type(), r.color, r.square()],
        [c.get_figure_type(), c.fc.color_property, c.square()],
        [s.get_figure_type(), s.color, s.square()]
    ])

    print(r)
    print(c)
    print(s)
    print(x)

if __name__ == "__main__":
    main()

```

Пример выполнения программы

Прямоугольник синего цвета, шириной 3 и высотой 2, площадью 6.000.

Круг зеленого цвета, радиуса 5, площадью 157.080.

Квадрат красного цвета со стороной 5, площадью 25.000.

+-----+-----+-----+			
Название фигуры	Цвет фигуры	Площадь фигуры	
+-----+-----+-----+			
Прямоугольник	синего	6	
Круг	зеленого	157.07963267948966	
Квадрат	красного	25	
+-----+-----+-----+			