

**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»

Отчёт по лабораторной работе №2
«Объектно-ориентированные возможности языка Python»

Выполнил:
студент группы ИУ5-256
Бикматов Д. А.

Проверил:
преподаватель
Гапанюк Ю. В.

Москва, 2024 г.

Постановка задачи:

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 `getr`, 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:

`geometric_figure.py`

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

```
class geometric_figure(ABC):  
    @abstractmethod  
    def get_area(self):  
        pass
```

```
figure_color.py
```

```
class figure_color:  
    def __init__(self):  
        self._color = None  
    @property  
    def color(self):  
        return self._color  
    @color.setter  
    def color(self, color):  
        self._color = color
```

```
rectangle.py
```

```
from lab_python_oop.geometric_figure import geometric_figure  
from lab_python_oop.figure_color import figure_color
```

```
class Rectangle(geometric_figure):  
    figure_type = "Прямоугольник"  
    def __init__(self, width, height, color_param):  
        self.width = width  
        self.height = height  
        self.color = figure_color()  
        self.color.color = color_param  
    def get_area(self):  
        return self.width * self.height  
    @classmethod  
    def get_type(cls):  
        return cls.figure_type  
    def __repr__(self):  
        return 'Тип: {}, Ширина: {}, Высота: {}, Цвет: {}, Площадь:  
{}, {}'.format(Rectangle.get_type(), self.width, self.height, self.color.color,  
self.get_area())  
square.py
```

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

```
class Square(Rectangle):
```

```

figure_type = "Квадрат"
def __init__(self, width, color):
    super().__init__(width, width, color)
@classmethod
def get_type(cls):
    return cls.figure_type
def __repr__(self):
    return 'Тип: {}, Сторона: {}, Цвет: {}, Площадь: {}'.format(Square.get_type(),
self.width, self.color.color, super().get_area())

```

circle.py

```

from lab_python_oop.geometric_figure import geometric_figure
from lab_python_oop.figure_color import figure_color
import math

```

```

class Circle(geometric_figure):
    figure_type = "Круг"
    def __init__(self, radius, color):
        self.radius = radius
        self.color = figure_color()
        self.color.color = color
    def get_area(self):
        return math.pi*self.radius*self.radius
@classmethod
def get_type(cls):
    return cls.figure_type
def __repr__(self):
    return 'Тип: {}, Радиус: {}, Цвет: {}, Площадь: {}'.format(Circle.get_type(),
self.radius, self.color.color, self.get_area())

```

main.py

```

from lab_python_oop.rectangle import Rectangle
from lab_python_oop.circle import Circle
from lab_python_oop.square import Square
from sympy import Symbol, posify, exp

```

```

def main():
    rect = Rectangle(3, 3, "синий")
    circle = Circle(3, "зелёный")
    square = Square(3, "красный")
    print(rect)
    print(circle)
    print(square)

```

```
x = Symbol('x')
expr = exp(x**2)
print(posify(expr))
```

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

Пример выполнения:

| Входные данные | Выходные данные |
|----------------|---|
| - | <pre> • hunter@MacBook-Pro-Andrey sem3 % /usr/local/bin/python3 /Users/hunter/sem3/lab2/main.py Тип: Прямоугольник, Ширина: 3, Высота: 3, Цвет: синий, Площадь: 9 Тип: Круг, Радиус: 3, Цвет: зелёный, Площадь: 28.274333882308138 Тип: Квадрат, Сторона: 3, Цвет: красный, Площадь: 9 (exp(_x**2), {_x: x}) </pre> |