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 № 2

Выполнил:
студент группы ИУ5-35Б
Талаев А.П.
Подпись и дата:

Проверил:
Подпись и дата:

Москва, 2024 г

Задание:
Необходимо создать виртуальное окружение и установить в него хотя бы один внешний пакет с использованием pip.

Необходимо разработать программу, реализующую работу с классами. Программа должна быть разработана в виде консольного приложения на языке Python 3.

Все файлы проекта (кроме основного файла main.py) должны располагаться в пакете lab_python_oop.

Каждый из нижеперечисленных классов должен располагаться в отдельном файле пакета lab_python_oop.

Абстрактный класс «Геометрическая фигура» содержит абстрактный метод для вычисления площади фигуры. Подробнее про абстрактные классы и методы Вы можете прочитать здесь.

Класс «Цвет фигуры» содержит свойство для описания цвета геометрической фигуры. Подробнее про описание свойств Вы можете прочитать здесь.

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.

Класс «Круг» создается аналогично классу «Прямоугольник», задается параметр «радиус». Для вычисления площади используется константа math.pi из модуля math.

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 - https://pyformat.info/
Название фигуры («Прямоугольник», «Квадрат», «Круг») должно задаваться в виде поля данных класса и возвращаться методом класса.
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):

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.

Выполнение:

```
use std::f64::consts::PI;

struct Color {
    color: String,
}

impl Color {
    fn new(color: &str) -> Color {
        Color { color: color.to_string() }
    }
}

enum Shape {
    Rectangle(Rectangle),
    Circle(Circle),
    Square(Square),
}

impl Shape {
    fn get_type(&self) -> &str {
        match self {
            Shape::Rectangle(_) => "Прямоугольник",
            Shape::Circle(_) => "Круг",
            Shape::Square(_) => "Квадрат",
        }
    }

    fn get_area(&self) -> f64 {
        match self {
            Shape::Rectangle(rect) => rect.get_area(),
            Shape::Circle(circle) => circle.get_area(),
            Shape::Square(square) => square.get_area(),
        }
    }
}

struct Rectangle {
    width: f64,
```

```
    height: f64,
    color: Color,
}

impl Rectangle {
    fn new(width: f64, height: f64, color: &str) -> Rectangle {
        Rectangle {
            width,
            height,
            color: Color::new(color),
        }
    }

    fn get_area(&self) -> f64 {
        self.width * self.height
    }
}

struct Circle {
    radius: f64,
    color: Color,
}

impl Circle {
    fn new(radius: f64, color: &str) -> Circle {
        Circle {
            radius,
            color: Color::new(color),
        }
    }

    fn get_area(&self) -> f64 {
        PI * self.radius * self.radius
    }
}

struct Square {
    side_length: f64,
    color: Color,
}

impl Square {
    fn new(side_length: f64, color: &str) -> Square {
        Square {
            side_length,
```

```
            color: Color::new(color),
        }
    }

    fn get_area(&self) -> f64 {
        self.side_length * self.side_length
    }
}

fn main() {
    let rectangle = Shape::Rectangle(Rectangle::new(5.0, 10.0, "red"));
    let circle = Shape::Circle(Circle::new(3.0, "blue"));
    let square = Shape::Square(Square::new(4.0, "green"));

    println!("Rectangle: {}, Color: {}, Area: {}", rectangle.get_type(), "red",
rectangle.get_area());
    println!("Circle: {}, Color: {}, Area: {}", circle.get_type(), "blue",
circle.get_area());
    println!("Square: {}, Color: {}, Area: {}", square.get_type(), "green",
square.get_area());
}
```

Результаты:

Rectangle: Прямоугольник, Color: red, Area: 50

Circle: Круг, Color: blue, Area: 28.274333882308138

Square: Квадрат, Color: green, Area: 16