Московский государственный технический университет им. Н.Э. Баумана

Факультет “Радиотехнический”
Кафедра “Системы обработки информации и управления”

Курс «Парадигмы и конструкции языка»

Отчет по лабораторной работе №2 «Объектно-ориентированные возможности языка Python» Вариант №7

Выполнил:
студент группы РТ5-31Б:
Зеленева И.Е.

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2025 г.

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. 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»:
    - 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.

1. 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.

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


def demonstrate_seaborn_simple():
    print("\n" + "=" * 50)
    print("Демонстрация работы с Seaborn:")

    try:
        import seaborn as sns
        import matplotlib.pyplot as plt
        import pandas as pd

        # Создаем данные для гистограммы
        N = 5

        # Создаем фигуры
        rectangle = Rectangle(N, N, "синий")
        circle = Circle(N, "зеленый")
        square = Square(N, "красный")

        # Собираем данные в DataFrame
        data = {
            'Фигура': ['Прямоугольник', 'Круг', 'Квадрат'],
            'Площадь': [rectangle.area(), circle.area(), square.area()],
            'Цвет': ['синий', 'зеленый', 'красный']
        }

        df = pd.DataFrame(data)

        # Создаем гистограмму
        plt.figure(figsize=(10, 6))
        plot = sns.barplot(data=df, x='Фигура', y='Площадь', palette=['blue', 'green', 'red'])

        # Настройки графика
        plot.set_title('Площади геометрических фигур', fontsize=16, fontweight='bold')
        plot.set_xlabel('Фигура', fontsize=12)
        plot.set_ylabel('Площадь', fontsize=12)
```

```
        # Добавляем значения на столбцы
        for i, row in df.iterrows():
            plot.text(i, row['Площадь'] + 0.1, f"{row['Площадь']:.2f}",
                      ha='center', va='bottom', fontweight='bold')

        # Сохраняем график
        plt.tight_layout()
        plt.savefig('figures_histogram.png', dpi=150)
        print("✓ Гистограмма сохранена как 'figures_histogram.png'")

        # Показываем информацию о seaborn
        print(f"✓ Используется Seaborn версии {sns.__version__}")

    except ImportError as e:
        print(f"✗ Ошибка: Не удалось импортировать seaborn")
        print(f"  Установите: pip install seaborn matplotlib pandas")
        print(f"  Детали: {e}")
    except Exception as e:
        print(f"✗ Ошибка при построении графика: {e}")
        # Сохраняем график
    plt.tight_layout()
    plt.savefig('figures_histogram.png', dpi=150)
    print("✓ Гистограмма сохранена как 'figures_histogram.png'")
    plt.show()

def main():
    """
    Основная функция программы
    """
    # Номер варианта
    N = 5

    print("=" * 50)
    print("ГЕОМЕТРИЧЕСКИЕ ФИГУРЫ")
    print("=" * 50)

    # Создаем объекты согласно заданию
    rectangle = Rectangle(N, N, "синий")
    circle = Circle(N, "зеленый")
    square = Square(N, "красный")
```

```
    # Выводим подробную информацию о каждой фигуре
    print("\n1. ПРЯМОУГОЛЬНИК:")
    print(f"   Название: {rectangle.get_figure_name()}")
    print(f"   Цвет: {rectangle.color_obj.color}")
    print(f"   Ширина: {rectangle.width}")
    print(f"   Высота: {rectangle.height}")
    print(f"   Площадь: {rectangle.area():.2f}")
    print(f"   Строковое представление: {rectangle}")

    print("\n2. КРУГ:")
    print(f"   Название: {circle.get_figure_name()}")
    print(f"   Цвет: {circle.color_obj.color}")
    print(f"   Радиус: {circle.radius}")
    print(f"   Площадь: {circle.area():.2f}")
    print(f"   Строковое представление: {circle}")

    print("\n3. КВАДРАТ:")
    print(f"   Название: {square.get_figure_name()}")
    print(f"   Цвет: {square.color_obj.color}")
    print(f"   Сторона: {square.side}")
    print(f"   Площадь: {square.area():.2f}")
    print(f"   Строковое представление: {square}")

    # Итоговая таблица
    print("\n" + "=" * 50)
    print("ИТОГОВАЯ ТАБЛИЦА:")
    print("-" * 50)
    print(f"{'Фигура':<15} {'Цвет':<10} {'Параметр':<10} {'Значение':<10} {'Площадь':<10}")
    print("-" * 50)
    print(f"{'Прямоугольник':<15} {'синий':<10} {'ширина':<10} {N:<10}
{rectangle.area():<10.2f}")
    print(f"{'Круг':<15} {'зеленый':<10} {'радиус':<10} {N:<10} {circle.area():<10.2f}")
    print(f"{'Квадрат':<15} {'красный':<10} {'сторона':<10} {N:<10} {square.area():<10.2f}")
    print("-" * 50)

    # Сумма площадей
    total_area = rectangle.area() + circle.area() + square.area()
    print(f"\nСуммарная площадь всех фигур: {total_area:.2f}")


    demonstrate_seaborn_simple()
```

```
    print("\n" + "=" * 50)
    print("ПРОГРАММА ЗАВЕРШЕНА")
    print("=" * 50)


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

color.py

```
class Color:
    def __init__(self, color):
        self._color = color

    @property
    def color(self):
        return self._color

    @color.setter
    def color(self, value):
        if not isinstance(value, str):
            raise ValueError("Цвет должен быть строкой")
        self._color = value

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

rectangle.py

```
rom lab_python_oop.figure import Figure
from lab_python_oop.color import Color


class Rectangle(Figure):
    """
    Класс 'Прямоугольник' наследуется от класса 'Геометрическая фигура'
    """

    FIGURE_NAME = "Прямоугольник"

    def __init__(self, width, height, color):
```

```
        """
        Конструктор класса Rectangle

        Args:
            width (float): Ширина прямоугольника
            height (float): Высота прямоугольника
            color (str): Цвет прямоугольника
        """
        self.width = width
        self.height = height
        self.color_obj = Color(color)

    def area(self):
        """
        Метод для вычисления площади прямоугольника

        Returns:
            float: Площадь прямоугольника
        """
        return self.width * self.height

    def get_figure_name(self):
        """
        Метод для получения названия фигуры

        Returns:
            str: Название фигуры
        """
        return Rectangle.FIGURE_NAME

    def __str__(self):
        return "{0} {1} шириной {2}, высотой {3} и площадью {4:.2f}".format(
            self.get_figure_name(),
            self.color_obj.color,
            self.width,
            self.height,
            self.area()
        )
```

circle.py

```
from lab_python_oop.figure import Figure
from lab_python_oop.color import Color
```

```
import math


class Circle(Figure):

    FIGURE_NAME = "Круг"

    def __init__(self, radius, color):
        self.radius = radius
        self.color_obj = Color(color)

    def area(self):
        return math.pi * (self.radius ** 2)

    def get_figure_name(self):

        return Circle.FIGURE_NAME

    def __str__(self):

        return "{0} {1} радиусом {2} и площадью {3:.2f}".format(
            self.get_figure_name(),
            self.color_obj.color,
            self.radius,
            self.area()
        )
```

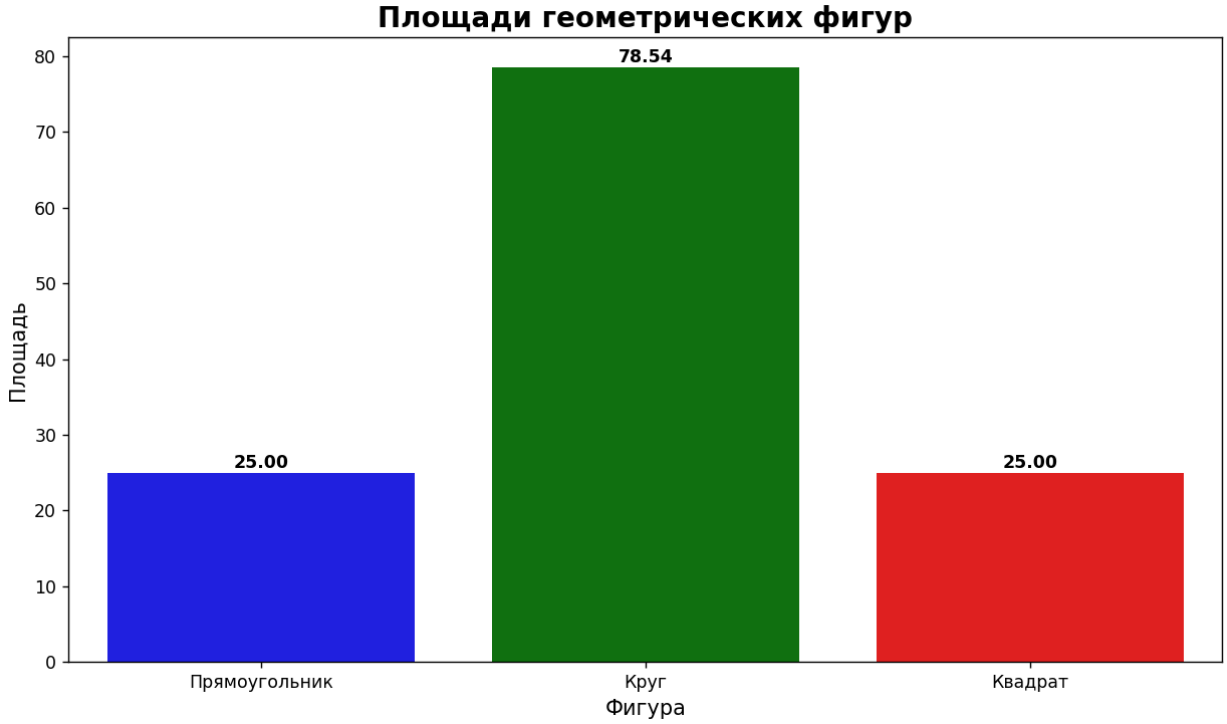
square.py

```
from lab_python_oop.rectangle import Rectangle


class Square(Rectangle):
    FIGURE_NAME = "Квадрат"

    def __init__(self, side, color):
        super().__init__(side, side, color)
        self.side = side

    def get_figure_name(self):
        return Square.FIGURE_NAME

    def __str__(self):
```

```
return "{0} {1} со стороной {2} и площадью {3:.2f}".format(
    self.get_figure_name(),
    self.color_obj.color,
    self.side,
    self.area()
)
```

Примеры выполнения программы

```
C:\Users\User\Репозиторий\lab_2>python main.py
==================================================
ГЕОМЕТРИЧЕСКИЕ ФИГУРЫ
==================================================

1. ПРЯМОУГОЛЬНИК:
   Название: Прямоугольник
   Цвет: синий
   Ширина: 5
   Высота: 5
   Площадь: 25.00
   Строковое представление: Прямоугольник синий шириной 5, высотой 5 и площадью 25.00

2. КРУГ:
   Название: Круг
   Цвет: зеленый
   Радиус: 5
   Площадь: 78.54
   Строковое представление: Круг зеленый радиусом 5 и площадью 78.54

3. КВАДРАТ:
   Название: Квадрат
   Цвет: красный
   Сторона: 5
   Площадь: 25.00
   Строковое представление: Квадрат красный со стороной 5 и площадью 25.00

==================================================
ИТОГОВАЯ ТАБЛИЦА:
--------------------------------------------------
Фигура          Цвет       Параметр   Значение   Площадь
--------------------------------------------------
Прямоугольник   синий      ширина     5          25.00
Круг            зеленый    радиус     5          78.54
Квадрат         красный    сторона    5          25.00
--------------------------------------------------

Суммарная площадь всех фигур: 128.54
```


Площади геометрических фигур
Площадь
Фигура
Прямоугольник
Круг
Квадрат
25.00
78.54
25.00
0
10
20
30
40
50
60
70
80