МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ  
им. Н.Э. Баумана

Факультет “Информатика и системы управления”  
Кафедра “Системы обработки информации и управления”



Дисциплина “Парадигмы и конструкции языков программирования”

Отчет по лабораторной работе №2  
“Объектно-ориентированные возможности языка Python.”

**Выполнил:**Студент группы ИУ5Ц-54Б  
Цурин А.П.  
**Преподаватель:**Гапанюк Ю.Е.

Москва 2025

# Задания для выполнения

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.

# Листинг программы

Файл main.py

from lab\_python\_oop.rectangle import Rectangle

from lab\_python\_oop.circle import Circle

from lab\_python\_oop.square import Square

# Импорт внешнего пакета (colorama для цветного вывода)

try:

    from colorama import init, Fore, Back, Style

    init(autoreset=True)  # Инициализация colorama

    COLORAMA\_AVAILABLE = True

except ImportError:

    COLORAMA\_AVAILABLE = False

    print("Colorama не установлен. Установите: pip install colorama")

def main():

    """Основная функция тестирования"""

    # Номер варианта (замените N на ваш номер)

    N = 4  # вариант 4

    print("Демонстрация работы с геометрическими фигурами")

    print("=" \* 50)

    # Создание объектов

    rectangle = Rectangle(width=N, height=N, color="синего")

    circle = Circle(radius=N, color="зеленого")

    square = Square(side=N, color="красного")

    # Вывод информации о фигурах

    if COLORAMA\_AVAILABLE:

        print(Fore.BLUE + str(rectangle))

        print(Fore.GREEN + str(circle))

        print(Fore.RED + str(square))

        # Демонстрация работы внешнего пакета

        print("\n" + "=" \* 50)

        print(Fore.YELLOW + Back.BLACK + "Демонстрация внешнего пакета colorama:")

        print(Fore.CYAN + "Этот текст выведен с использованием colorama!")

        print(Fore.MAGENTA + Style.BRIGHT + "Яркий цветной текст")

        print(Style.RESET\_ALL + "Текст сброшен к стандартному формату")

    else:

        # Вывод без цветов, если colorama не установлен

        print(rectangle)

        print(circle)

        print(square)

        print("\n" + "=" \* 50)

        print("Colorama не установлен. Для цветного вывода установите: pip install colorama")

    # Дополнительная информация о фигурах

    print("\n" + "=" \* 50)

    print("Дополнительная информация:")

    print(f"Площадь {rectangle.name}: {rectangle.area():.2f}")

    print(f"Площадь {circle.name}: {circle.area():.2f}")

    print(f"Площадь {square.name}: {square.area():.2f}")

if \_\_name\_\_ == "\_\_main\_\_":

    main()

файл lab\_python\_oop/circle.py

import math

from .figure import Figure

from .color import FigureColor

class Circle(Figure):

    """Класс Круг"""

    def \_\_init\_\_(self, radius, color):

        self.radius = radius

        self.color\_property = FigureColor()

        self.color\_property.color = color

        self.\_name = "Круг"

    @property

    def name(self):

        return self.\_name

    def area(self):

        """Вычисление площади круга"""

        return math.pi \* self.radius \*\* 2

    def \_\_repr\_\_(self):

        return "{} {} цвета радиусом {} площадью {:.2f}".format(

            self.name,

            self.color\_property.color,

            self.radius,

            self.area()

        )

файл lab\_python\_oop/color.py

class FigureColor:

    """Класс Цвет фигуры"""

    def \_\_init\_\_(self):

        self.\_color = None

    @property

    def color(self):

        """Getter для цвета"""

        return self.\_color

    @color.setter

    def color(self, value):

        """Setter для цвета"""

        self.\_color = value

файл lab\_python\_oop/figure.py

from abc import ABC, abstractmethod

class Figure(ABC):

    """Абстрактный класс Геометрическая фигура"""

    @abstractmethod

    def area(self):

        """Абстрактный метод для вычисления площади"""

        pass

    @property

    @abstractmethod

    def name(self):

        """Абстрактное свойство для названия фигуры"""

        pass

файл lab\_python\_oop/rectangle.py

from .figure import Figure

from .color import FigureColor

class Rectangle(Figure):

    """Класс Прямоугольник"""

    def \_\_init\_\_(self, width, height, color):

        self.width = width

        self.height = height

        self.color\_property = FigureColor()

        self.color\_property.color = color

        self.\_name = "Прямоугольник"

    @property

    def name(self):

        return self.\_name

    def area(self):

        """Вычисление площади прямоугольника"""

        return self.width \* self.height

    def \_\_repr\_\_(self):

        return "{} {} цвета шириной {} и высотой {} площадью {:.2f}".format(

            self.name,

            self.color\_property.color,

            self.width,

            self.height,

            self.area()

        )

файл lab\_python\_oop/square.py

from .rectangle import Rectangle

class Square(Rectangle):

    """Класс Квадрат (наследуется от Прямоугольника)"""

    def \_\_init\_\_(self, side, color):

        # Вызываем конструктор родительского класса

        super().\_\_init\_\_(side, side, color)

        self.\_name = "Квадрат"

    @property

    def name(self):

        return self.\_name

    def \_\_repr\_\_(self):

        return "{} {} цвета со стороной {} площадью {:.2f}".format(

            self.name,

            self.color\_property.color,

            self.width,  # или self.height, они равны

            self.area()

        )

# Результаты работы программы

(venv) PS C:\Users\student\project> python .\main.py

Демонстрация работы с геометрическими фигурами

==================================================

Прямоугольник синего цвета шириной 4 и высотой 4 площадью 16.00

Круг зеленого цвета радиусом 4 площадью 50.27

Квадрат красного цвета со стороной 4 площадью 16.00

==================================================

Демонстрация внешнего пакета colorama:

Этот текст выведен с использованием colorama!

Яркий цветной текст

Текст сброшен к стандартному формату

==================================================

Дополнительная информация:

Площадь Прямоугольник: 16.00

Площадь Круг: 50.27

Площадь Квадрат: 16.00