**Write a program that uses constructors to initialize objects and demonstrates method overloading.**

**package Basic;**

**// Define the class with constructor overloading and method overloading**

**public class Rectangle {**

**// Attributes for width and height**

**private double width;**

**private double height;**

**// Constructor with no parameters (default constructor)**

**public Rectangle() {**

**this.width = 2.0;**

**this.height = 1.0;**

**}**

**// Constructor with one parameter (for a square)**

**public Rectangle(double side) {**

**this.width = side;**

**this.height = side;**

**}**

**// Constructor with two parameters (for a rectangle)**

**public Rectangle(double width, double height) {**

**this.width = width;**

**this.height = height;**

**}**

**// Method to calculate the area of the rectangle**

**public double calculateArea() {**

**return width \* height;**

**}**

**// Overloaded method to calculate the area with a different signature**

**public double calculateArea(double width, double height) {**

**return width \* height;**

**}**

**// Method to display the dimensions of the rectangle**

**public void displayDimensions() {**

**System.*out*.println("Width: " + width);**

**System.*out*.println("Height: " + height);**

**}**

**// Main method to test the Rectangle class**

**public static void main(String[] args) {**

**// Create Rectangle objects using different constructors**

**Rectangle defaultRectangle = new Rectangle(); // Default constructor**

**Rectangle square = new Rectangle(5.0); // Constructor for a square**

**Rectangle customRectangle = new Rectangle(4.0, 6.0); // Constructor for a rectangle**

**// Display dimensions and area for each rectangle**

**System.*out*.println("Default Rectangle:");**

**defaultRectangle.displayDimensions();**

**System.*out*.println("Area: " + defaultRectangle.calculateArea());**

**System.*out*.println("\nSquare:");**

**square.displayDimensions();**

**System.*out*.println("Area: " + square.calculateArea());**

**System.*out*.println("\nCustom Rectangle:");**

**customRectangle.displayDimensions();**

**System.*out*.println("Area: " + customRectangle.calculateArea());**

**// Demonstrate method overloading**

**System.*out*.println("\nArea calculated using overloaded method:");**

**double area = customRectangle.calculateArea(4.0, 6.0);**

**System.*out*.println("Area: " + area);**

**}**

**}**