1.

// Book.java

public class Book {

private String title;

private String author;

private int yearPublished;

// Constructor to initialize attributes

public Book(String title, String author, int yearPublished) {

this.title = title;

this.author = author;

this.yearPublished = yearPublished;

}

// Main method to create and display Book objects

public static void main(String[] args) {

Book book1 = new Book("To Kill a Mockingbird", "Harper Lee", 1960);

Book book2 = new Book("1984", "George Orwell", 1949);

System.out.println("Book 1: " + book1.title + " by " + book1.author + " (" + book1.yearPublished + ")");

System.out.println("Book 2: " + book2.title + " by " + book2.author + " (" + book2.yearPublished + ")");

}

}

2. // Student.java

public class Student {

private String name;

private int age;

private String major;

// Constructor to initialize all fields

public Student(String name, int age, String major) {

this.name = name;

this.age = age;

this.major = major;

}

// Method to print student information

public void printStudentInfo() {

System.out.println("Student Name: " + name);

System.out.println("Age: " + age);

System.out.println("Major: " + major);

}

// Main method to create and display Student object

public static void main(String[] args) {

Student student = new Student("Alice", 21, "Computer Science");

student.printStudentInfo();

}

}

3. // Shape.java

abstract class Shape {

protected String name;

// Constructor to initialize name

public Shape(String name) {

this.name = name;

}

// Abstract method to calculate area

public abstract double calculateArea();

}

// Circle.java

class Circle extends Shape {

private double radius;

// Constructor to initialize name and radius

public Circle(String name, double radius) {

super(name);

this.radius = radius;

}

// Implement calculateArea() for Circle

@Override

public double calculateArea() {

return Math.PI \* radius \* radius;

}

}

// Rectangle.java

class Rectangle extends Shape {

private double length;

private double width;

// Constructor to initialize name, length, and width

public Rectangle(String name, double length, double width) {

super(name);

this.length = length;

this.width = width;

}

// Implement calculateArea() for Rectangle

@Override

public double calculateArea() {

return length \* width;

}

}

// Main method to create instances of Circle and Rectangle

public class Main {

public static void main(String[] args) {

Shape circle = new Circle("Circle", 5.0);

Shape rectangle = new Rectangle("Rectangle", 4.0, 6.0);

System.out.println(circle.name + " Area: " + circle.calculateArea());

System.out.println(rectangle.name + " Area: " + rectangle.calculateArea());

}

}

4. // Car.java

public class Car {

private String brand;

private String model;

private int year;

// Default constructor

public Car() {

this.brand = "Unknown";

this.model = "Unknown";

this.year = 2000;

}

// Constructor with brand and model, year set to 2024 by default

public Car(String brand, String model) {

this.brand = brand;

this.model = model;

this.year = 2024;

}

// Constructor with brand, model, and year

public Car(String brand, String model, int year) {

this.brand = brand;

this.model = model;

this.year = year;

}

// Main method to create and display Car objects

public static void main(String[] args) {

Car car1 = new Car();

Car car2 = new Car("Tesla", "Model S");

Car car3 = new Car("Ford", "Mustang", 2023);

System.out.println("Car 1: " + car1.brand + " " + car1.model + " (" + car1.year + ")");

System.out.println("Car 2: " + car2.brand + " " + car2.model + " (" + car2.year + ")");

System.out.println("Car 3: " + car3.brand + " " + car3.model + " (" + car3.year + ")");

}

}