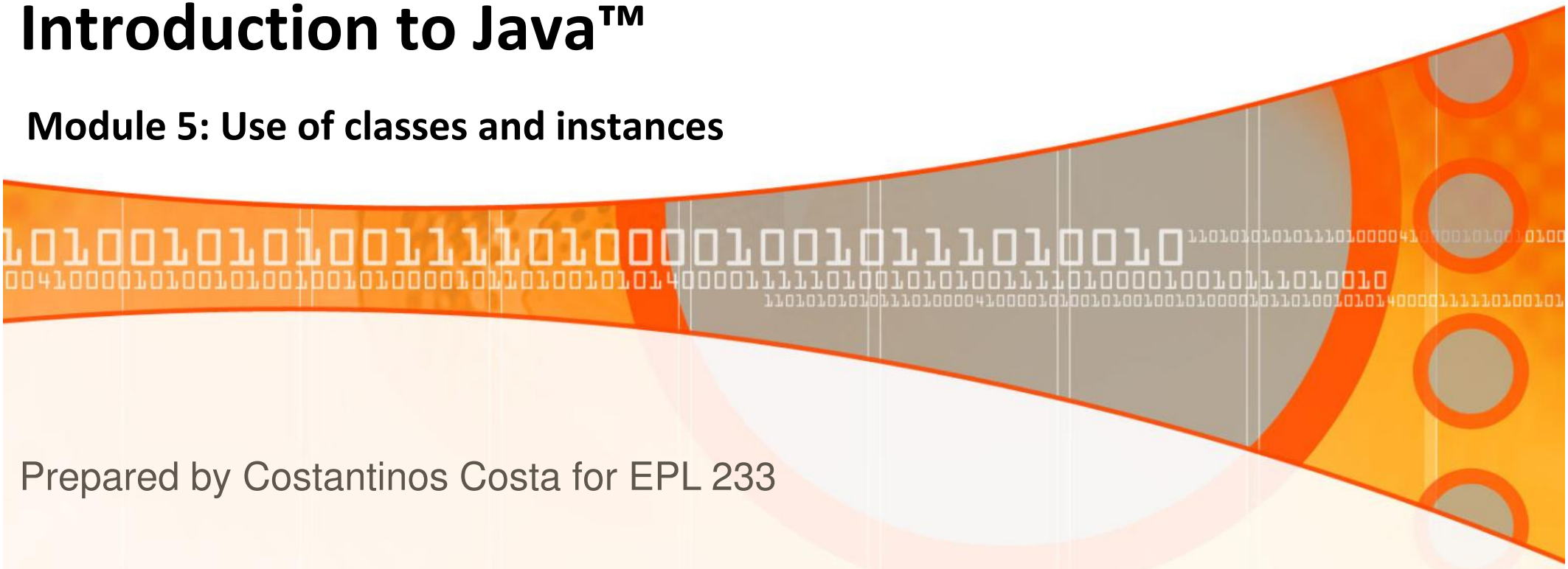
Introduction to Java™

Module 5: Use of classes and instances



Prepared by Costantinos Costa for EPL 233

Objective

- Introduction to creating and use of classes and instances, methods and constructors, modifiers. You will see how to use your own, user-defined classes as well as built-in classes as String, Date, etc.

Task1

- This task will show you how to use **constructors** and **private modifiers**, how to pass objects as arguments as well as how to use **getters** and **setters** to access and modify private fields of an object.
- ***Note:*** The return type of setters is usually *void* while getters should have return type.
- **Do it:** Read carefully, understand and finally store the following text as a file named
- ***TestChangingObjectPrivateData.java***. Compile and run the application.

TestChangingObjectPrivateData.java

```
public class TestChangingObjectPrivateData {
     public static void main(String[] args)
     // create object with specific color and radius
     {
     Circle myCircle = new Circle(5.0, "white");
     printCircle(myCircle);
     myCircle.setColor(myCircle, "black");
     printCircle(myCircle); // passing object as argument (by reference)
     }

     public static void printCircle(Circle c) {
     System.out.println("The area of the circle of the radius "
     + c.getRadius() + " is " + c.findArea());
     System.out.println("The color of the circle is " + c.getColor());
     }
}
```

Circle.java

```
class Circle {
      private double radius; // private modifier
      private String color;
      public Circle(double radius, String color) // constructor
      {
      this.radius = radius;
      this.color = color;
      }
      public Circle() // constructor with no argument
      {
      radius = 1.0;
      color = "white";
      }
      public double getRadius() // implementation of getter
      {
      return radius;
      }
      public String getColor() // getter
      {
      return color;
      }
      public void setColor(Circle c, String color) // implementation of setter
      {
      c.color = color;
      }
      public double findArea() {
      return radius * radius * Math.PI;
      } // Math – built-in class; PI - constant
}
```

Task2

- Write a class named ***Rectangle***, which represents different rectangles. The private data fields are ***width, length, area*** and ***color***. Use **double** for ***width*** and ***length*** and **String** for ***color***. The methods are ***getWidth(), getLenght(), getColor()*** and ***findArea()***. Use a **class variable** for ***color***.
- Include the above class in a program that uses it.

Task3

- Create an application that computes mortgage payments. The program should let the user enter the **interest rate**, **years**, and **loan amount** (Principal) as **input parameters**. Then compute and display the monthly and total payment (i.e. for all years).
- The formula to compute the **monthly** payment is as follows:

$$\frac{\text{Principal x monthly Interest}}{(\ 1 - (1/\ (1 + \text{monthly Interest})^{\text{years x 12}}\ )\ )}$$

- ***Hint:*** To make your formula calculation easier and more professional use the built-in class **Math** and more specifically its **method for calculating the power** of two numbers. Find yourself information about that class.

Solution

```
import java.util.Scanner;

public class TestMortgage {
      public static void main(String[] args) {
      double interestRate;
      int year;
      double loan;
      double monthly;
      double totalPay;

      Scanner MyInput = new Scanner(System.in);
      // enter input
      System.out.println("Enter yearly interest rate, for example 8.25: ");
      interestRate = MyInput.nextDouble();
      System.out
      .println("Enter number of years as an integer, for example 5: ");
      year = MyInput.nextInt();
      System.out.println("Enter loan amount, for example 120000.95: ");
      loan = MyInput.nextDouble();
      // creating Mortgage object
      monthly=loan * interestRate
      / (1 - (Math.pow(1 / (1 + interestRate), year * 12)));
      totalPay= monthly * year * 12;
      // display results
      System.out.println("The monthly pay is " + monthly);
      System.out.println("The total paid is " + totalPay);
      }
}
```

Task4

- In Task 3 you should create an application that computes mortgage payments. I don’t know how your basic class looks like but below you’ll see another class **Mortgage** with all specific characteristics of an ADT and used in true object-oriented style.
- A specific mortgage can be viewed as an object of a Mortgage class. Interest rate, loan amount, and loan period are its data properties, and computing monthly payment and total payment are its methods. When you buy a house for example, a mortgage object is created with its properties and methods.
- **Do it:** In the following example, pay attention to ***Mortgage*** class with properties
- ***interest rate, loan amount*** and ***total payment*** and methods ***monthlyPayment ()*** and ***totalPayment().***
- Understand it and create a file **TestMortgageClass.java.** Compile and run the application.
- ***Hint:*** The mortgage class contains a constructor, three getters, and methods for finding the monthly payment. You’ve constructed a mortgage object by using 3 parameters: ***interest rate, payment years*** and ***loan amount***. The 3 getters ***interest(), year(),*** and l***oan()*** return ***interest rate, payment years*** and ***loan amount*** respectively.