JAVA Textbook

Chapter 10

Additional Topics

Objectives

In this chapter, you will learn about:

- Creating a simple programmer-defined class
- Creating a simple graphical user interface (GUI)

Creating a Programmer-Defined Class

- A class created by the programmer is referred to as a programmer-defined class.
- Procedural programming declares data and defines methods separate from the data, and then calls those methods to operate on the data.
  - This is the style of programming used in Chapters 1 through 9 of the textbook.
- Object-oriented programming encapsulates data and the methods needed to manipulate that data within a class.
  - Objects are created as an instance of a class.
  - The program tells an object to perform tasks by passing to it messages consisting of instructions to execute the class' methods.
  - The class' methods then manipulate the data, which is part of

Creating a Programmer-Defined Class

```
// Employee class
public class Employee
{
   private String lastName;
   private double hourlyWage;
   private double weeklyPay;

   public void setLastName(String name)
   {
      lastName = name;
      return;
   }

   public void setHourlyWage(double wage)
   {
      hourlyWage = wage;
      calculateWeeklyPay();
      return;
   }
```

```
   public String getLastName()
   {
      return lastName;
   }

   public double getHourlyWage()
   {
      return hourlyWage;
   }

   public double getWeeklyPay()
   {
      return weeklyPay;
   }

   private void calculateWeeklyPay()
   {
      final int WORK_WEEK_HOURS = 40;
      weeklyPay = hourlyWage * WORK_WEEK_HOURS;
      return;
   }
} // End Employee class
```

- Access specifiers: *public* vs. *private*
- Three types of methods: set methods, get methods, work methods
- Nonstatic vs. static methods
- No *main*() method: not an application

Using a Programmer-Defined Class

- A *main*() method must be included, because the class is an application.
  - It is a static method.
  - No need to create an *EmployeeWages* object to call it.
- Can use a prewritten default constructor.
  - Expects no arguments.
  - Created automatically by the compiler.
- Syntax used to invoke a method with an instance (an

```
// This program uses the programmer-defined Employee class.

public class EmployeeWages
{
   public static void main(String args[])
   {
      final double LOW = 9.00;
      final double HIGH = 14.65;
      // Instantiate an Employee object
      Employee myGardener = new Employee();

      // Use the get and set methods
      myGardener.setLastName("Greene");
      myGardener.setHourlyWage(LOW);
      System.out.println("My gardener makes " +
            myGardener.getWeeklyPay() + " per week.");

      // Use the get and set methods
      myGardener.setHourlyWage(HIGH);
      System.out.println("My gardener makes " +
            myGardener.getWeeklyPay() + " per week.");
      System.exit(0);
   }
}
```

Thank You!