8.Packages

Sakib Abrar

CSE

Bangladesh University of Engineering & Technology

sakib.cghs@gmail.com

September 9, 2020

Overview

Package basics

Declaring Package

Package Examples

Access Protection

Package basics

- Java package provides a mechanism for par55oning the class name space into more manageable chunks
  – Both **naming** and **visibility** control mechanism
- Define classes inside a package that are not accessible by code outside that package.
- Define class members that are exposed only to other members of the same package
- This allows classes to have in5mate knowledge of each other
  – Not expose that knowledge to the rest of the world.

Declaring Package

- **package pkg**
  – Here, pkg is the name of the package
- **package MyPackage**
  – creates a package called MyPackage
- The package statement defines a name space in which classes are stored.
- If you omit the package statement, the class names are put into the **default package**, which has no name.

Declaring Package

- Java uses file system directories to store packages
  – the .class files for any classes that are part of MyPackage must be stored in a directory called MyPackage
- More than one file can include the same package statement
- The package statement simply specifies to which package the classes defined in a file belong
- To create hierarchy of packages, separate each package name from the one above it by use of a (.)

Package Examples

AccountBalance.java

```
1  package MyPackage;
2
3  class Balance {
4      String name;
5      double bal;
6
7      Balance(String n, double b) {
8          name = n;
9          bal = b;
10     }
11     void show() {
12         if(bal < 0)
13             System.out.print("--> ");
14         System.out.println(name + ": $" + bal);
15     }
16 }
17
18 public class AccountBalance {
19     public static void main(String[] args) {
20         Balance current[] = new Balance[3];
21         current[0] = new Balance( n: "K. J. Fielding",  b: 123.23);
22         current[1] = new Balance( n: "Will Tell",  b: 157.02);
23         current[2] = new Balance( n: "Tom Jackson",  b: -12.33);
24         for(Balance b : current) {
25             b.show();
26         }
27     }
28 }
```

Access Protection

- Packages act as containers for classes and other subordinate packages
- Classes act as containers for data and code
- The class is Java's smallest unit of abstraction
- Four categories of visibility for class members
  - Subclasses in the same package
  - Non-subclasses in the same package
  - Subclasses in different package
  - Classes that are neither in the same package subclasses

Access Protection

- The three access modifiers provide a variety of ways to produce the many levels of access required

The following applies only to members of classes:

	Private	No Modifier	Protected	Public
Same class	Yes	Yes	Yes	Yes
Same package subclass	No	Yes	Yes	Yes
Same package non-subclass	No	Yes	Yes	Yes
Different package subclass	No	No	Yes	Yes
Different package non-subclass	No	No	No	Yes

Access Protection

- Anything declared **public** can be accessed from anywhere
- Anything declared **private** cannot be seen outside of its class
- When a member does not have an explicit access specifica5on, it is visible to subclasses as well as to other classes in the same package (**default access**)
- If you want to allow an element to be seen outside your current package, but only to classes that subclass the class directly, then declare that element **protected**

Access Protection

- A non-nested class has only two possible access levels – default and public
- When a class is declared as public, it is accessible by any other code
- If a class has default access, then it can only be accessed by other code within its same package
- When a class is public, it must be the only public class declared in the file, and the file must have the same name as the class

THE END