



Packages

Reading: Savitch&Carrano chapter 6.7
Programming Course: Computational Linguistics – Verena Henrich



Packages and Importing

- A **package** is a **named collection of related classes** that can **serve as a library of classes**
- With packages you do not need to place all classes in the same directory as your program
- In order to use classes from packages that have already been defined, such as **Scanner** or **File**, we need to import them:
 - Import a single class: **import java.util.Scanner;**
 - Import **all** classes from a package: **import java.io.*;**



Defining your own Packages

- A package groups a set of classes together into a directory
- The name of the folder is the name of the package
- The classes in the package folder are each placed in a separate file (as usual)
- Each class in the package has **package *Package_Name*;** as the first statement, like this:

```
/** Description of the class */  
package lib.helpers;  
// rest of class definition...
```



Package Names

- A package name tells the compiler the path (divided by dots) to the directory that contains the classes in the package
- For example: our package will be named **lib.helpers**, so we will store the package in the directory **lib/helpers**
- Put all Java files that should be included in the package in the package directory (**lib/helpers**)
- Our package has only one source file, **ListHelper.java** (from selftest 1), but we can add more later
- Don't put any source files that are not part of the package in this directory (no junit tests, for example)



Setting the Classpath

- You need to tell Java where to find the **lib** directory by setting your classpath
- **Setting the classpath in NetBeans:**
 - Right-click on the project → select “Properties”
 - Choose the “Libraries” tab → “Add Library” button
 - Navigate to the directory ABOVE the **lib** directory and single-click on the **lib** directory, so that it is selected, but you are not in it
 - Click “Choose”; you should see the path to the **lib** directory under “Libraries” now



Using the Package

- Now you can use the package (in junit tests, demo programs, etc.) by importing it:
 - Either: **`import lib.helpers.ListHelper;`**
 - Or: **`import lib.helpers.*;`**



Name Clashes

- Packages help in dealing with name clashes, i.e., when two classes have the same name
- Problem: different programmers writing different packages have used the same name for a class
- Solution: ambiguity can be resolved by using the package name before the class name (“fully qualified class name”)

```
lib.helpers.ListHelper helper1 = new lib.helpers.ListHelper();  
fantasy.ListHelper helper2 = new fantasy.ListHelper();
```

- Since fully qualified name includes the package name, there is no need to import the package