

Java Packages

Organizing classes into namespaces

No Global Variables

- Every Java field and method is part of a class
- Every class is part of a package
- A fully-qualified name consists of the package name, the class name, and the member (field or method) name
- Package name components are separated by periods
- Package names are usually made of multiple period-separated names

Why packages

- Java was designed to support dynamic loading of modules
 - Any Java class can be loaded into a running Java interpreter at any time.
 - Dynamically loaded classes can then be dynamically instantiated
- Packages help avoid namespace conflicts

package statement

- One package statement per .java file
- Usually the first line in a source file

- Example declaration

```
package edu.und.hw2;  
public class Fork {  
...  
}
```

- Example reference:

```
import edu.und.hw2.*;    // don't use *  
import edu.und.hw2.Fork;
```

Java Filenames and Directory Structure

- A Java source file must have the same name as the public class or interface defined in the file, plus the *.java* extension
- The package statement is optional*
- Class files are stored in a directory that has the same components as the package name
- By convention, source tree has same directory structure

[project name]/build/edu/und/hw2/Fork.class
[project name]/src/edu/und/hw2/Fork.java
[project name]/build.xml