

## **APPENDIX C - PACKAGES**

Terry Marris 7 June 2001

A package is a collection of related classes and interfaces.

When you place a set of classes in your current working directory, then all those classes form a package with no name, the default package.

Classes in different packages can have the same name without conflict because a package provides a new namespace.

A file may contain several classes but only one of them can be public (the class with the same name as the file). Public classes can be accessed outside the package, other classes are private to the package; objects from any class within the package can use these classes.

To create a package:

**1** Specify that a class belongs to a named package by including a statement of the form

```
package packagename;
```

at the top of the file.

```
/* Counter.java  
   Terry Marris  7 June 2001  
*/
```

```
package counter;
```

```
public interface Counter {  
    public void increment();  
    public String toString();  
}
```

```
/* BoundCounter.java  
   Terry Marris  7 June 2001  
*/
```

```
package counter;
```

```
public class BoundCounter implements Counter {  
    int count;  
    int upperBound;  
  
    public BoundCounter(int upperBound)  
    {  
        count = 0;  
        this.upperBound = upperBound;  
    }  
  
    public void increment()  
    {  
        if (count < upperBound)  
            count++;  
    }  
  
    public String toString()  
    {  
        return "Count: " + count + ", Upper bound: " + upperBound;  
    }  
}
```

2 Put the source file in a directory with the same name as the package.

e.g. in `c:\project\counter`

3 To use a class or an interface that is in a different package import the entire package with

```
import packagename;

/* TestCounter.java
   Terry Marris 7 June 2001
*/

import counter.*;

public class TestCounter {
    public static void main(String[] s)
    {
        Counter c = new BoundCounter(2);
        c.increment();
        c.increment();
        c.increment();
        System.out.println(c);
    }
}
```

4 Set the classpath to the parent of the directory containing the package.

At a DOS prompt enter

```
set classpath=.;c:\project
```

The `.` means the current directory.

The `;` is the separator.

`project` is the parent directory of `counter`.

Program output:

```
Count: 2, Upper bound: 2
```

Common problems:

If the compiler or interpreter cannot find a class

- check that the class has been imported
- check that the classpath has been correctly set
- check the spelling, remember that case matters
- make sure that the classes are in the right subdirectory
- invoke the interpreter from the directory that contains the class.