

## Lab – Working with Java packages

**15 points – See Canvas for due date**

### Exercise 1 – Working with the default package.

If you do not specify a package in a java source code file, a package designation of 'default' is assumed. When this is the case, the package identified can be considered to be pointed to the current directory.



- Create a new folder named tempFolder on your C drive.
- Start notepad.
- Key in the following code and save the file as CallingClass.java in tempFolder. (Be sure to defeat notepad's default extension name of .txt.)

```
public class CallingClass
{
    public static void main(String[] args)
    {
        int x = CalledClass.method1();
        System.out.println(x);
    }
}
```

- Start another instance of notepad, key in the following code and save in tempFolder as CalledClass.java.

```
public class CalledClass
{
    public static int method1()
    {
        return 1001;
    }
}
```

- Start a console session and navigate to tempFolder. Compile CallingClass.java.  
javac CallingClass.java
- Key dir (directory listing.) Notice that CalledClass.class is also generated. Know why?
- Run your java program. You should see:

```
C:\tempFolder>java CallingClass
1001
```

- Now, insert a package statement at the top of CalledClass.java ('package' statement must be first:)

```
package pack1;
public class CalledClass
{
    public static int method1()
    {
        return 1002; // ← Changed this line, too...
    }
}
```

- Save the CalledClass.java file.
- Delete your class files with `C:\tempFolder>del *.class` and recompile:

```
javac CallingClass.java
```

- You should see an error saying:

```
CallingClass.java:7: cannot access CalledClass
bad class file: .\CalledClass.java
file does not contain class CalledClass
Please remove or make sure it appears in the correct subdirectory of the
classpath.
```

```
        int x = CalledClass.method1();
                ^
```

```
1 error
```

- The package statement you added indicated that the file CalledClass.class must be stored in a package (directory) named “pack1”.
- Create a new folder off tempFolder named pack1.
- `C:\tempFolder> md pack1` (“md” stands for “make directory”)

- Move CalledClass.java to pack1:

```
C:\tempfolder>move CalledClass.java c:\tempfolder\pack1\CalledClass.java
```

- Navigate to pack1 and compile with

```
javac CalledClass.java
```

- Change CallingClass.java to indicate the package name of pack1 on the method1() call:

```
public class CallingClass
{
    public static void main(String[] args)
    {
        int x = pack1.CalledClass.method1();
        System.out.println(x);
    }
}
```

- Return to your tempFolder directory and compile CallingClass.java.

- Run your program.

- 
- Change CallingClass to use the import statement instead of the explicit pack1.CalledClass reference:

```
import pack1.CalledClass;
```

```
public class CallingClass
{
    public static void main(String[] args)
    {
        int x = CalledClass.method1(); // <- Note change here...
        System.out.println(x);
    }
}
```

---

**Exercise 2** – Running a single .class file that contains a package statement.

- Navigate to your base directory (C:\tempFolder) and create a new folder named pack2:
- C:\tempFolder> md pack2
- Navigate to pack2.
- Open notepad and add the following code:

```
package pack2;
public class MyClass
{
    public static void main(String[] args)
    {
        System.out.println("Hello from MyClass");
    }
}
```

- Save and compile MyClass. This class specifies a package of pack2, so you cannot run the program from the pack2 folder:

```
C:\tempFolder\pack2>java MyClass
Exception in thread "main" java.lang.NoClassDefFoundError: MyClass (wrong
name:
pack2/MyClass)
    at java.lang.ClassLoader.defineClass1(Native Method)
    at java.lang.ClassLoader.defineClass(Unknown Source)
```

```
at java.security.SecureClassLoader.defineClass(Unknown Source)
at java.net.URLClassLoader.defineClass(Unknown Source)
at java.net.URLClassLoader.access$000(Unknown Source)
at java.net.URLClassLoader$1.run(Unknown Source)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)
at sun.misc.Launcher$AppClassLoader.loadClass(Unknown Source)
at java.lang.ClassLoader.loadClass(Unknown Source)
at java.lang.ClassLoader.loadClassInternal(Unknown Source)
```

- To fix the problem, go to your base directory (up one level) and specify the package/folder name:

```
cd ..
C:\tempFolder>java pack2.MyClass
Hello from MyClass
```

- 
- Print a hard-copy listing of your program with your name on it and submit in class to receive lab credit. Do not submit to Canvas.