**Packages**

**Package** in Java is a mechanism to encapsulate a group of classes, sub packages and interfaces. Packages are used for:

* Preventing naming conflicts. For example there can be two classes with name Employee in two packages, college.staff.cse.Employee and college.staff.ee.Employee
* Making searching/locating and usage of classes, interfaces, enumerations and annotations easier
* Providing controlled access: protected and default have package level access control. A protected member is accessible by classes in the same package and its subclasses. A default member (without any access specifier) is accessible by classes in the same package only.
* Packages can be considered as data encapsulation (or data-hiding).

All we need to do is put related classes into packages. After that, we can simply write an import class from existing packages and use it in our program. A package is a container of a group of related classes where some of the classes are accessible are exposed and others are kept for internal purpose.  
We can reuse existing classes from the packages as many time as we need it in our program.

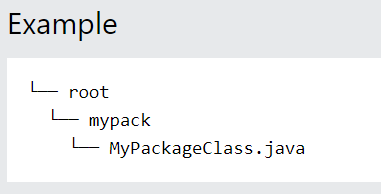
The package statement should be the first line in the source file. There can be only one package statement in each source file, and it applies to all types in the file.

**How to define a Java package?**

To define a package in Java, you use the keyword package

package packageName;

To create your own package, you need to understand that Java uses a file system directory to store them. Just like folders on your computer:



package mypack;

class MyPackageClass {

public static void main(String[] args) {

System.out.println("This is my package!");

}

}

The full path to the classes directory, <path-two>\classes, is called the class path, and is set with the CLASSPATH system variable. Both the compiler and the JVM construct the path to your .class files by adding the package name to the class path.

**Set CLASSPATH System Variable**

To display the current CLASSPATH variable, use the following commands in Windows and UNIX (Bourne shell) −

* In Windows → C:\> set CLASSPATH
* In UNIX → % echo $CLASSPATH

To delete the current contents of the CLASSPATH variable, use −

* In Windows → C:\> set CLASSPATH =
* In UNIX → % unset CLASSPATH; export CLASSPATH

To set the CLASSPATH variable −

* In Windows → set CLASSPATH = C:\users\jack\java\classes
* In UNIX → % CLASSPATH = /home/jack/java/classes; export CLASSPATH