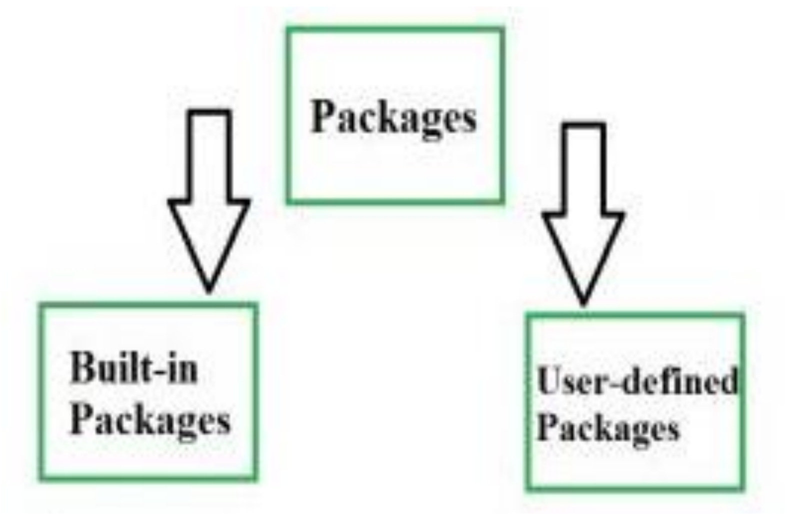
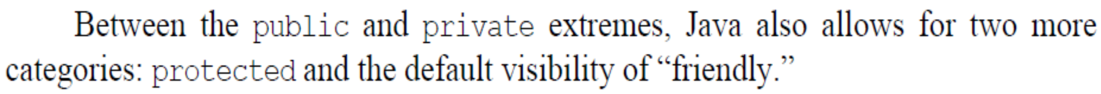
LEC21 OOP 2018-2019

Packages

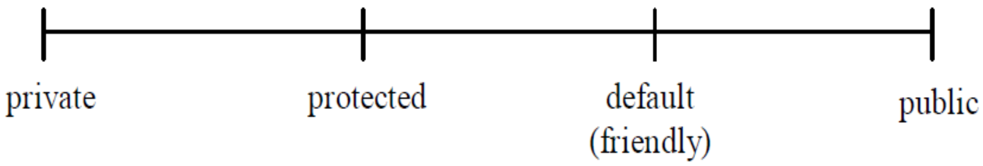


Package in Java is a mechanism to encapsulate a group of classes, sub packages and interfaces. Packages are used for:

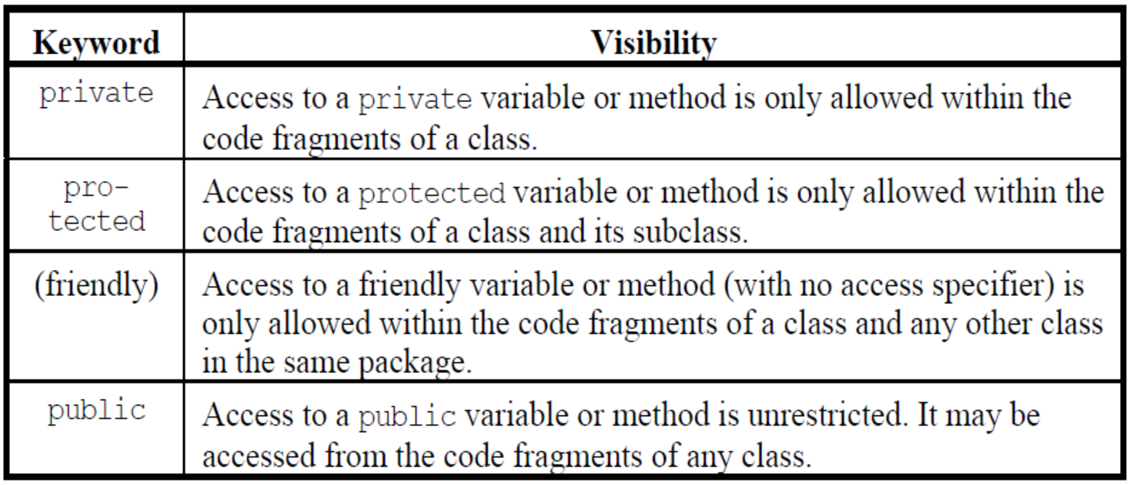
- Preventing naming conflicts. For example there can be two classes with name Employee in two packages, college.staff.cse.Employee and college.staff.ee.Employee
- Making searching/locating and usage of classes, interfaces, enumerations and annotations easier
- Providing controlled access: protected and 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).



Between the `public` and `private` extremes, Java also allows for two more categories: `protected` and the default visibility of "friendly."



Access Modifiers

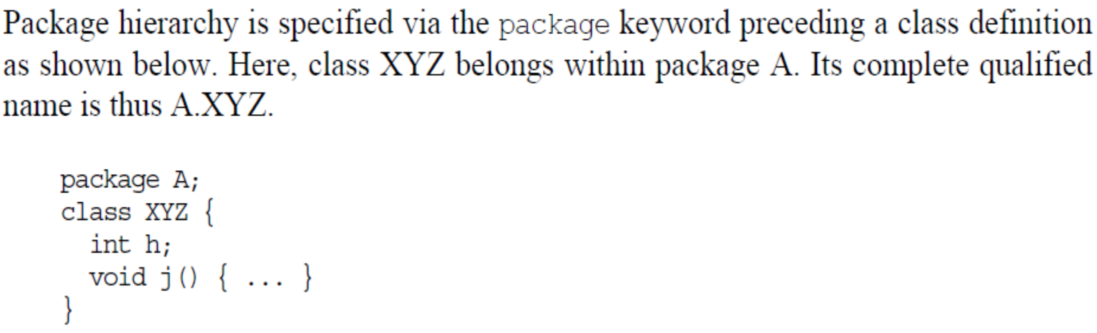


Keyword	Visibility
`private`	Access to a `private` variable or method is only allowed within the code fragments of a class.
`protected`	Access to a `protected` variable or method is only allowed within the code fragments of a class and its subclass.
(friendly)	Access to a friendly variable or method (with no access specifier) is only allowed within the code fragments of a class and any other class in the same package.
`public`	Access to a `public` variable or method is unrestricted. It may be accessed from the code fragments of any class.

All we need to do is put related classes into packages. After that we can simply write a import a class from existing packages and use it in our program. A package is container of 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.

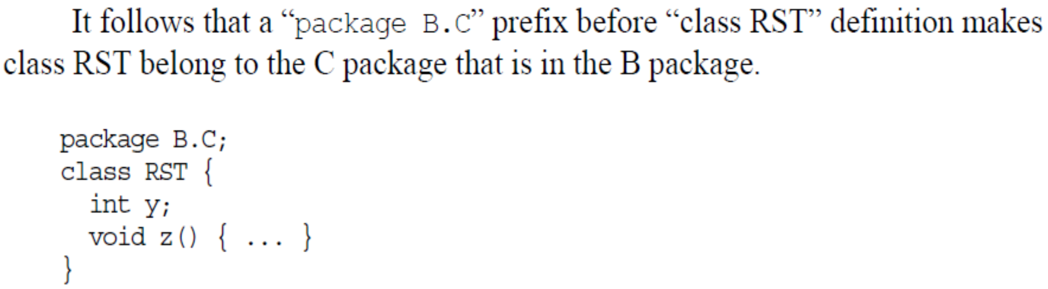
Adding a class to a Package : We can add more classes to an created package by using package name at the top of the java file (class declaration) and saving it in the package directory (example1).

Subpackages: Packages that are inside another package are the **subpackages**. These are not imported by default, they have to imported explicitly. Also, members of a subpackage have no access privileges (الامتيازات), i.e., they are considered as different package for protected and default access specifiers (example2).



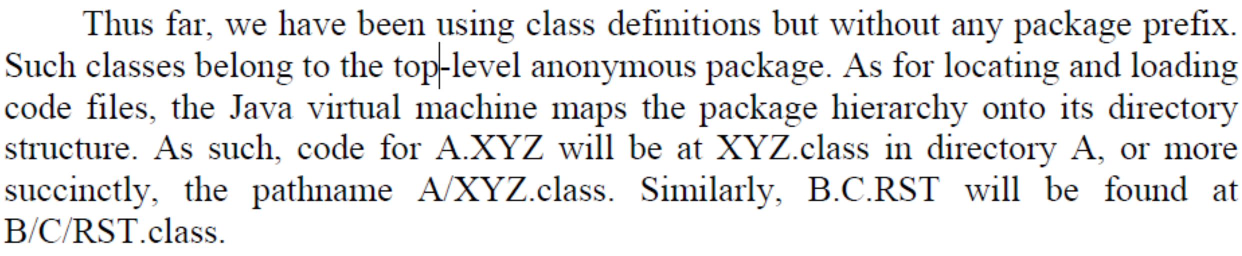
Package hierarchy is specified via the package keyword preceding a class definition as shown below. Here, class XYZ belongs within package A. Its complete qualified name is thus A.XYZ.

```
package A;
class XYZ {
  int h;
  void j() { ... }
}
```



It follows that a "package B.C" prefix before "class RST" definition makes class RST belong to the C package that is in the B package.

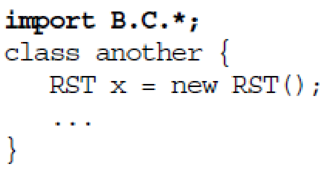
```
package B.C;
class RST {
  int y;
  void z() { ... }
}
```



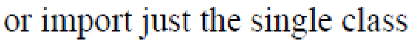
Thus far, we have been using class definitions but without any package prefix. Such classes belong to the top-level anonymous package. As for locating and loading code files, the Java virtual machine maps the package hierarchy onto its directory structure. As such, code for A.XYZ will be at XYZ.class in directory A, or more succinctly, the pathname A/XYZ.class. Similarly, B.C.RST will be found at B/C/RST.class.

The import Keyword

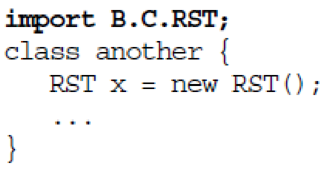
The import keyword provides the complement function of the package facility. While the package keyword places succeeding class definitions in a separate name space of the said package, the import keyword makes package constituents visible.



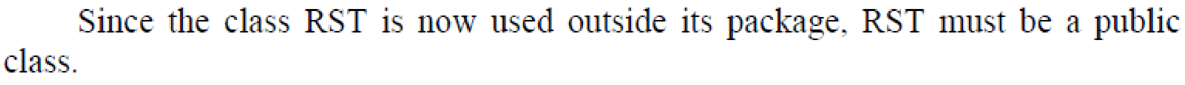
```
import B.C.*;
class another {
   RST x = new RST();
   ...
}
```



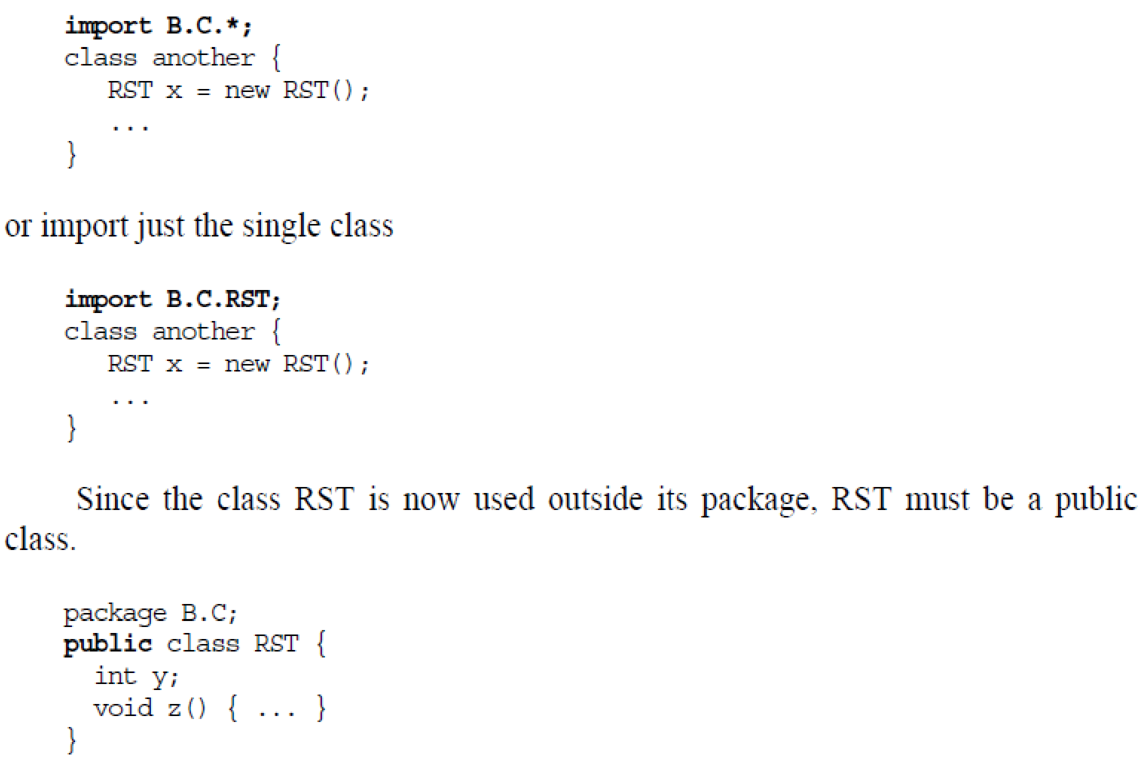
or import just the single class



```
import B.C.RST;
class another {
   RST x = new RST();
   ...
}
```



Since the class RST is now used outside its package, RST must be a public class.



```
import B.C.*;
class another {
   RST x = new RST();
   ...
}
```

or import just the single class

```
import B.C.RST;
class another {
   RST x = new RST();
   ...
}
```

Since the class RST is now used outside its package, RST must be a public class.

```
package B.C;
public class RST {
  int y;
  void z() { ... }
}
```

Keyboard Input with the Scanner Class

Java programmers no longer need to wade through a design pattern or use a third-party toolkit for keyboard input. The **Scanner** class of Java 5.0 supports easy keyboard input of all primitive data types as well as lines of text. The next code segment shows how to set up a scanner and input data of various types:

Example 1

```
package javaapplication60;
 import java.util.Scanner;

public class JavaApplication60 {
    public static void main(String[] args) {

 Scanner reader = new Scanner(System.in);

System.out.print("Enter a line of text followed by <enter>: ");
String str = reader.nextLine();

System.out.print("Enter an integer followed by <enter>: ");
int i = reader.nextInt();

System.out.print("Enter a floating-point number followed by <enter>: ");
double d = reader.nextDouble();   reader.close(); } }
```

```
run:
Enter a line of text followed by <enter>: YAZAN
Enter an integer followed by <enter>: 2006
Enter a floating-point number followed by <enter>: 12.11
```

Example 2

```
import java.util.*;

    public class ScannerClassExample1 {
        public static void main(String args[]){

            System.out.println("--------Enter Your Details-------- ");
            Scanner in = new Scanner(System.in);

            System.out.print("Enter your name: ");
            String name = in.next();
            System.out.println("Name: " + name);

            System.out.print("Enter your age: ");
            int i = in.nextInt();
            System.out.println("Age: " + i);

           double d = in.nextDouble();
           System.out.print("Enter your salary: ");
           System.out.println("Salary: " + d);

            in.close();          }      }
```

The Output will be:

```
-------Enter Your Details---------
Enter your name: Abhishek
Name: Abhishek
Enter your age: 23
Age: 23
Enter your salary: 25000
Salary: 25000.0
```