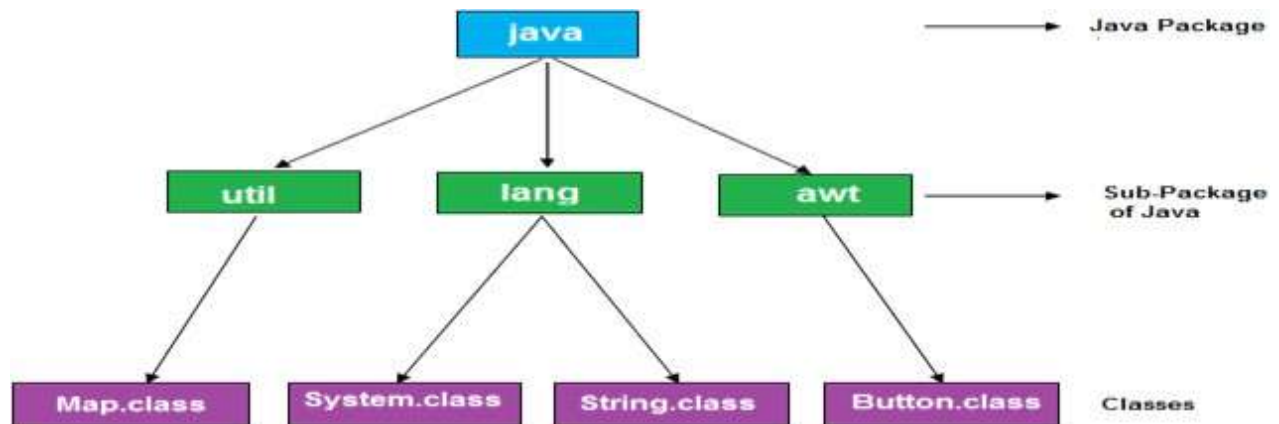


# Packages

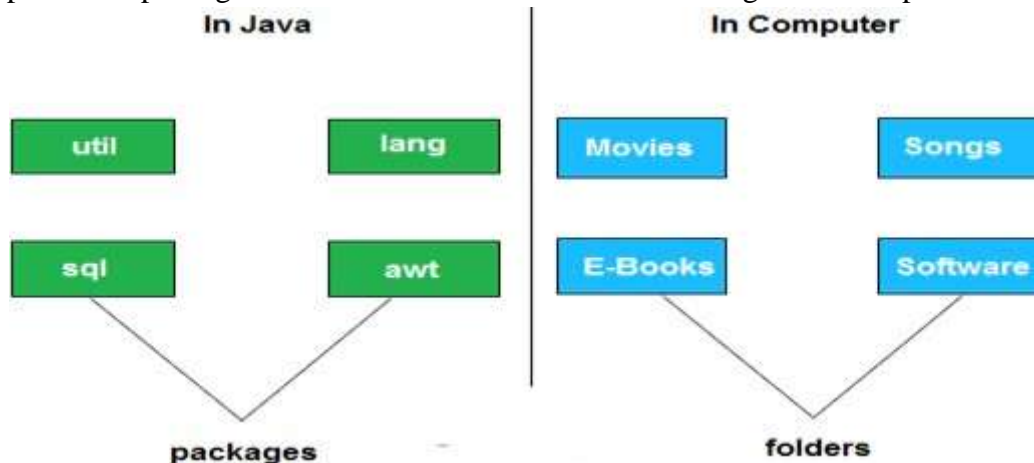
A package is a collection of similar types of classes, interfaces and sub-packages.

## Purpose of package

The purpose of package concept is to provide common classes and interfaces for any program separately. In other words if we want to develop any class or interface which is common for most of the java programs than such common classes and interfaces must be placed in a package.



Packages in Java are the way to organize files when a project has many modules. Same like we organized our files in Computer. For example we store all movies in one folder and songs in other folder, here also we store same type of files in a particular package for example in awt package have all classes and interfaces for design GUI components.



## Advantage of package

- Package is used to categorize the classes and interfaces so that they can be easily maintained
- Application development time is less, because reuse the code
- Application memory space is less (main memory)
- Application execution time is less
- Application performance is enhance (improve)
- Redundancy (repetition) of code is minimized
- Package provides access protection.
- Package removes naming collision.

## Types of package

Packages are classified into two types which are given below.

1. Predefined or built-in package

## 2. User defined package

### Predefined or built-in package

These are the packages which are already designed by the Sun Microsystems and supply as a part of java API, every predefined package is collection of predefined classes, interfaces and sub-package.

### User defined package

If any package is designed by the user that is known as a user defined package. User defined package are those which are developed by java programmer and supply as a part of their project to deal with common requirement.

### Rules to create user defined package

- package statement should be the first statement of any package program.
- Choose an appropriate class name or interface name and whose modifier must be public.
- Any package program can contain only one public class or only one public interface but it can contain any number of normal classes.
- Package program should not contain any main class (that means it should not contain any main())
- modifier of constructor of the class which is present in the package must be public. (This is not applicable in case of interface because interface has no constructor.)
- The modifier of method of class or interface which is present in the package must be public (This rule is optional in case of interface because interface methods by default public)
- Every package program should be save either with public class name or public Interface name

<code>//Sum.java</code>	→	Save package program with 'public' class name
<code>package MyPackage</code>	→	First statement of java program is package
<code>public class Sum</code>	→	class modifier must be 'public'
<code>{</code>		
<code>public Sum()</code>	→	constructor modifier must be 'public'
<code>{</code>		
<code>System.out.println("Sum class constructor");</code>		
<code>}</code>		
<code>public void show()</code>	→	method modifier must be 'public'
<code>{</code>		
<code>System.out.println("Sum class method");</code>		
<code>}</code>		
<code>}</code>		

### Compile package programs

For compilation of package program first we save program with public className.java and it compile using below syntax:

#### Syntax

`javac -d . className.java`

#### Syntax

`javac -d path className.java`

Explanations: In above syntax "-d" is a specific tool which is tell to java compiler create a separate folder for the given package in given path. When we give specific path then it create a new folder at that location and when we use . (dot) then it crate a folder at current working directory.

Note: Any package program can be compile but can not be execute or run. These program can be executed through user defined program which are importing package program.

Example of package program

Package program which is save with A.java and compile by javac -d . A.java

Example

```
package mypack;
public class A
{
    public void show()
    {
        System.out.println("Sum method");
    }
}
```

Import above class in below program using import packageName.className

Example

```
import mypack.A;
public class Hello
{
    public static void main(String arg[])
    {
        A a=new A();
        a.show();
        System.out.println("show() class A");
    }
}
```

Explanations: In the above program first we create Package program which is save with A.java and compiled by "javac -d . A.java". Again we import class "A" in class Hello using "import mypack.A;" statement.

Difference between Inheritance and package

Inheritance concept always used to reuse the feature within the program between class to class, interface to interface and interface to class but not accessing the feature across the program.

Package concept is to reuse the feature both within the program and across the programs between class to class, interface to interface and interface to class.

Difference between package keyword and import keyword

Package keyword is always used for creating the undefined package and placing common classes and interfaces.

import is a keyword which is used for referring or using the classes and interfaces of a specific package.