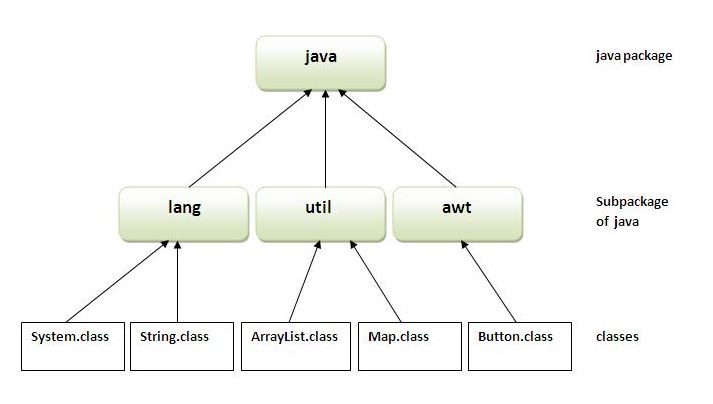
**Java Package**

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

Package in java can be categorized in two form, built-in package and user-defined package.

There are many built-in packages such as java, lang, awt, javax, swing, net, io, util, sql etc.



**Advantage of Java Package**

**1)** Java package is used to categorize the classes and interfaces so that they can be easily maintained.

**2)** Java package provides access protection.

**3)** Java package removes naming collision.

Simple example of java package

**The package keyword is used to create a package in java.**

**//save as Simple.java**

package mypack;

public class Simple{

public static void main(String args[]){

System.out.println("Welcome to package");

}

}

**To Compile**: javac -d . Simple.java

**To Run:** java mypack.Simple

**To access package from another package**

There are three ways to access the package from outside the package.

**1)**import package.\*;

**2)**import package.classname;

**3)**fully qualified name.

**1) Using packagename.\***

If you use package.\* then all the classes and interfaces of this package will be accessible but not the subpackages. The **import** keyword is used to make the classes and interface of another package accessible to the current package.

**Example:**

**//save by A.java**

package pack;

public class A{

public void msg()

{

System.out.println("Hello");

}

}

**//save by B.java**

package mypack;

import pack.\*;

class B{

public static void main(String args[])

{

A obj = new A();

obj.msg();

}

}

=====================================================================================

**2) Using packagename.classname**

If we import package.classname then only declared class of this package will be accessible.

Example:

**//save by A.java**

package pack;

public class A{

public void msg()

{

System.out.println("Hello");

} }

------------------------------------------------------------------------------------------------------------------------------------------

**//save by B.java**

package mypack;

import pack.A;

class B{

public static void main(String args[]){

A obj = new A();

obj.msg();

}

}

----------------------------------------------------------------------------------------------------------------------------------------

**3) Using fully qualified name**

If we use fully qualified name then only declared class of this package will be accessible. Now there is no need to import. But we need to use fully qualified name every time when we are accessing the class or interface.

**It is generally used when two packages have same class name .**

**//save by A.java**

package pack;

public class A{

public void msg(){

System.out.println("Hello");

}

}

------------------------------------------------------------------------------------------------------------------------------------------

**//save by B.java**

package mypack;

class B{

public static void main(String args[])

{

pack.A obj = new pack.A();//using fully qualified name

obj.msg();

}

}

=====================================================================================

**Note: If we import a package, subpackages will not be imported.**

=====================================================================================

**How to put two public classes in a package?**

If we want to put two public classes in a package, keep the package name same.

=====================================================================================

**Access Modifiers in java**

The access modifiers in java specifies accessibility (scope) of a data member, method, constructor or class.

There are 4 types of java access modifiers:

**1)private**

**2)default**

**3)protected**

**4)public**

**1) private access modifier**

**The private access modifier is accessible only within class.**

**Example of private access modifier:**

class A{

private int data=40;

private void msg()

{

System.out.println("Hello java");

}

}

---------------------------------------------------------------------------------------- --------------------------------------------------

public class Simple

{

public static void main(String args[])

{

A obj=new A();

System.out.println(obj.data); //Compile Time Error

obj.msg(); //Compile Time Error

}

}

================================================================================================

**note**:If we make any class constructor private, we cannot create the instance of that class from outside the class.

**2) default access modifier**

If you don't use any modifier, it is treated as “default” by default. The default modifier is accessible only within package.

**//save by A.java**

package pack;

class A{

void msg()

{

System.out.println("Hello");

}

}

------------------------------------------------------------------------------------------------------------------------------------------

**//save by B.java**

package mypack;

import pack.\*;

class B{

public static void main(String args[]){

A obj = new A(); //Compile Time Error

obj.msg(); //Compile Time Error

}

}

=====================================================================================

**3) protected access modifier**

The protected access modifier is accessible within package and outside the package but through inheritance only.

The protected access modifier can be applied on the data member, method and constructor. It can't be applied on the class.

**Example of protected access modifier**

**//save by A.java**

package pack;

public class A

{

protected void msg()

{

System.out.println("Hello");

}

}

------------------------------------------------------------------------------------------------------------------------------------------

**//save by B.java**

package mypack;

import pack.\*;

class B extends A{

public static void main(String args[]){

B obj = new B();

obj.msg();

} }

===============================================================================================

**4) public access modifier**

The public access modifier is accessible everywhere. It has the widest scope among all other modifiers.

Example of public access modifier

**//save by A.java**

package pack;

public class A

{

public void msg()

{

System.out.println("Hello");

}

}

------------------------------------------------------------------------------------------------------------------------------------------**//save by B.java**

package mypack;

import pack.\*;

class B

{

public static void main(String args[]){

A obj = new A();

obj.msg();

} }