

Experiment No. 05

Title: Implementing the concept of package.
<ol style="list-style-type: none">1. Aim: To study how to create own package.How to import classes from the packageHow to create hierarchy of classes in package.

Theory:

Packages

Java provides a mechanism for partitioning the class name space into more manageable chunks. This mechanism is the package. The package is both a naming and a visibility control mechanism. You can define classes inside a package that are not accessible by code outside that package. You can also define class members that are only exposed to other members of the same package.

Defining a Package

To create a package is quite easy: simply include a package command as the first statement in a Java source file. Any classes declared within that file will belong to the specified package. The package statement defines a name space in which classes are stored. If you omit the package statement, the class names are put into the default package, which has no name. While the default package is fine for short, sample programs, it is inadequate for real applications. Most of the time, you will define a package for your code.

This is the general form of the **package** statement:

```
package pkg;
```

Here, *pkg* is the name of the package.

More than one file can include the same **package** statement. The **package** statement simply specifies to which package the classes defined in a file belong. It does not exclude other classes in other files from being part of that same package. Most real-world packages are spread across many files.

The general form of a
multileveled package
statement is shown here:

```
package pkg1[.pkg2[.pkg3]];
```

For example, a package declared
as
package
java.awt.image;

needs to be stored in **java/awt/image**, **java\awt\image**, or **java:awt:image**
on your UNIX, Windows, or Macintosh file system, respectively. Be sure to
choose your package names carefully. You cannot rename a package without
renaming the directory in which the classes are stored.

Finding Packages and CLASSPATH

Packages are mirrored by directories.

How does the Java run-time system know where to look for packages that you create?

The answer has two parts. First, by default, the Java run-time system uses the current
working directory as its starting point. Thus, if your package is in the current directory, or a
subdirectory of the current directory, it will be found. Second, you can specify a directory
path or paths by setting the **CLASSPATH** environmental variable. For example, consider
the following package specification.

```
package MyPack;
```

In order for a program to find **MyPack**, one of two things must be true. Either the program
is executed from a directory immediately above **MyPack**, or **CLASSPATH** must be set to
include the path to **MyPack**.

Program:

Dell.java

```
package  
mypack;  
public  
class  
Dell  
{  
    public void disp()  
    {  
        System.out.println("dell class");  
    }  
}
```

Hp.java

package mypack;

```
public class Hp  
{  
    public void display()  
    {  
        System.out.println("Hp class");  
    }  
}
```

Subpackage

Intell.java

```
package  
mypack.intel;  
public class  
Intell  
{  
    public void view()  
    {  
        System.out.println("Intell class");  
    }  
}
```

Main class

Laptopshows.java

```
import mypack.*;  
import mypack.intel.*;class Laptopshows  
{
```

```
public static void main(String[]args)
{
    Hp obj=new
    Hp(); Dell
    obj1=new
    Dell();
    Intel1 iobj=new
    Intel1();
    obj.display();
    obj1.disp();
    iobj.view();
}
}
```

Statement:

Write a Java program to perform employee payroll processing using packages. In the java file, Emp.java creates a package employee and creates a class Emp. Declare the variables name,empid, category, bpay, hra, da, npay, pf, grosspay, incometax, and allowance. Calculate the values in methods. Create another java file Emppay.java. Create an object e to call the methods to perform and print values.

Program:

```
//SAVE AS Emp.java
package employee;
public class Emp{
    String name,empid, category;
    int bpay;
    double hra,da, npay,pf,grosspay,incometax,allowance;
    public Emp(String n, String id, String c, int b)
    {
        name = n;
        empid = id;
        category = c;
        bpay = b;
    }
    public void call()
    {

        da = bpay*0.05;
        hra = bpay*0.09;
        pf = bpay*0.11;
        allowance = bpay*0.10;
        grosspay = bpay+da+hra+allowance-pf;
        incometax = 0.75*grosspay;
        npay = grosspay- incometax;
    }

    public void display()
    {
        System.out.println("/n/n Employee Details");
        System.out.println("/n/n Name:"+name);
        System.out.println("/n/n Empid:"+empid);
        System.out.println("/n/n Category:"+category);
        System.out.println("/n/n bpay:"+bpay);
        System.out.println("/n/n da:"+da);
        System.out.println("/n/n hra:"+hra);
        System.out.println("/n/n pf:"+pf);
        System.out.println("/n/n all:"+allowance);
        System.out.println("/n/n gs:"+grosspay);
        System.out.println("/n/n Incometax:"+incometax);
        System.out.println("/n/n npay:"+npay);
    }
}

//SAVE as Emppay.java
```

```
import java.io.*;
import employee.Emp;
class Emppay
{

    public static void main(String args[])
    {

        Emp e = new Emp("ANU","23","Female",12000);
        e.call();
        e.display();
    }
}
```

```
/*OUTPUT:
Employee Details
Name: ANU
Empid: 23
Category: Female
bpay: 12000
da: 600.0
hra: 1080.0
pf: 1320.0
allowance: 1200.0
grosspay: 13560.0
Incometax: 10170.0
npay: 3390 */
```