

---

## Experiment 11: Program on packages

---

### Theory :

There are folders or directories in our computers for the classification and accessibility of various files, and in Java, we have packages for the same. In Java, Packages are similar to folders, which are mainly used to organize classes and interfaces.

Packages help us to write better and manageable code by preventing naming conflicts. Java provides some built-in packages which we can use but we can also create our own (user-defined) packages.

A **package** is a collection of similar types of Java entities such as classes, interfaces, subclasses, exceptions, errors, and enums. A package can also contain sub-packages.

### Types of Packages in Java

They can be divided into two categories:

1. Java API packages or built-in packages and
2. User-defined packages.

#### 1. Java API packages or built-in packages

Java provides a large number of classes grouped into different packages based on a particular functionality.

Examples:

**java.lang:** It contains classes for primitive types, strings, math functions, threads, and exceptions.

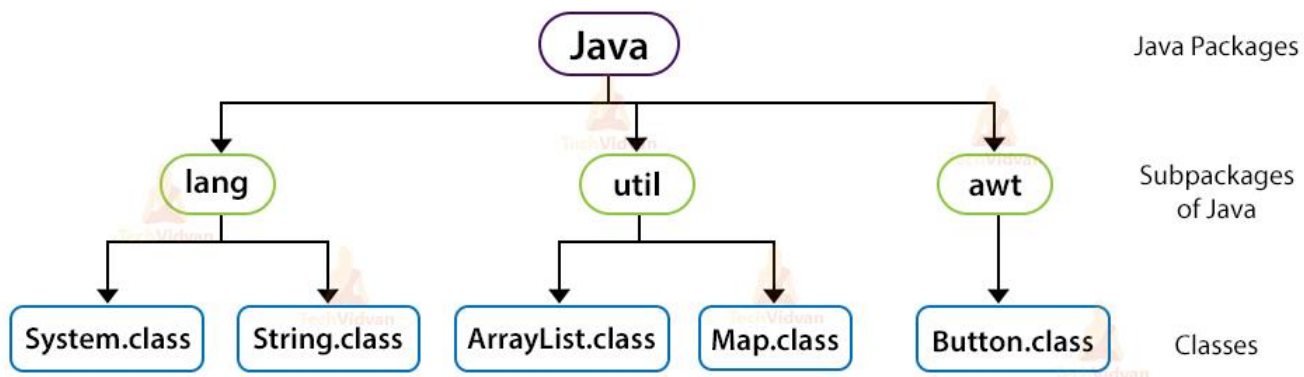
**java.util:** It contains classes such as vectors, hash tables, dates, Calendars, etc.

**java.io:** It has stream classes for Input/Output.

**java.awt:** Classes for implementing Graphical User Interface – windows, buttons, menus, etc.

**java. Applet:** Classes for creating and implementing applets

## Built-in Packages in Java



## 2. User-defined packages

As the name suggests, these packages are defined by the user. We create a directory whose name should be the same as the name of the package. Then we create a class inside the directory.

### Creating a Package in Java

To create a package, we choose a package name and to include the classes, interfaces, enumerations, etc, inside the package, we write the package with its name at the top of every source file.

There can be only one package statement in each type of file. If we do not write class, interfaces, inside any package, then they will be placed in the current default package.

### Accessing Packages or Classes from Another Package

If we want to access all the classes and interfaces of an existing package then we use the **import** statement. We can do it in three different ways:

- **import** package.\*;
- **import** package.classname;
- fully qualified name.

1. By using \* after the import statement, we can access all the classes of the package but not the sub-packages.

**Syntax:**

For importing all the classes:

```
import packageName.*;
```

2. By using a **particular class name** after the import statement, we can access that particular class package but not the sub-packages.

**Syntax:**

For importing a particular class:

```
import packageName.className;
```

3. Using a **Fully qualified name** means we can access the declared class of different packages without using the import statement. But you need to use a fully qualified name every time when you are accessing the class or interface which is present in a different package.

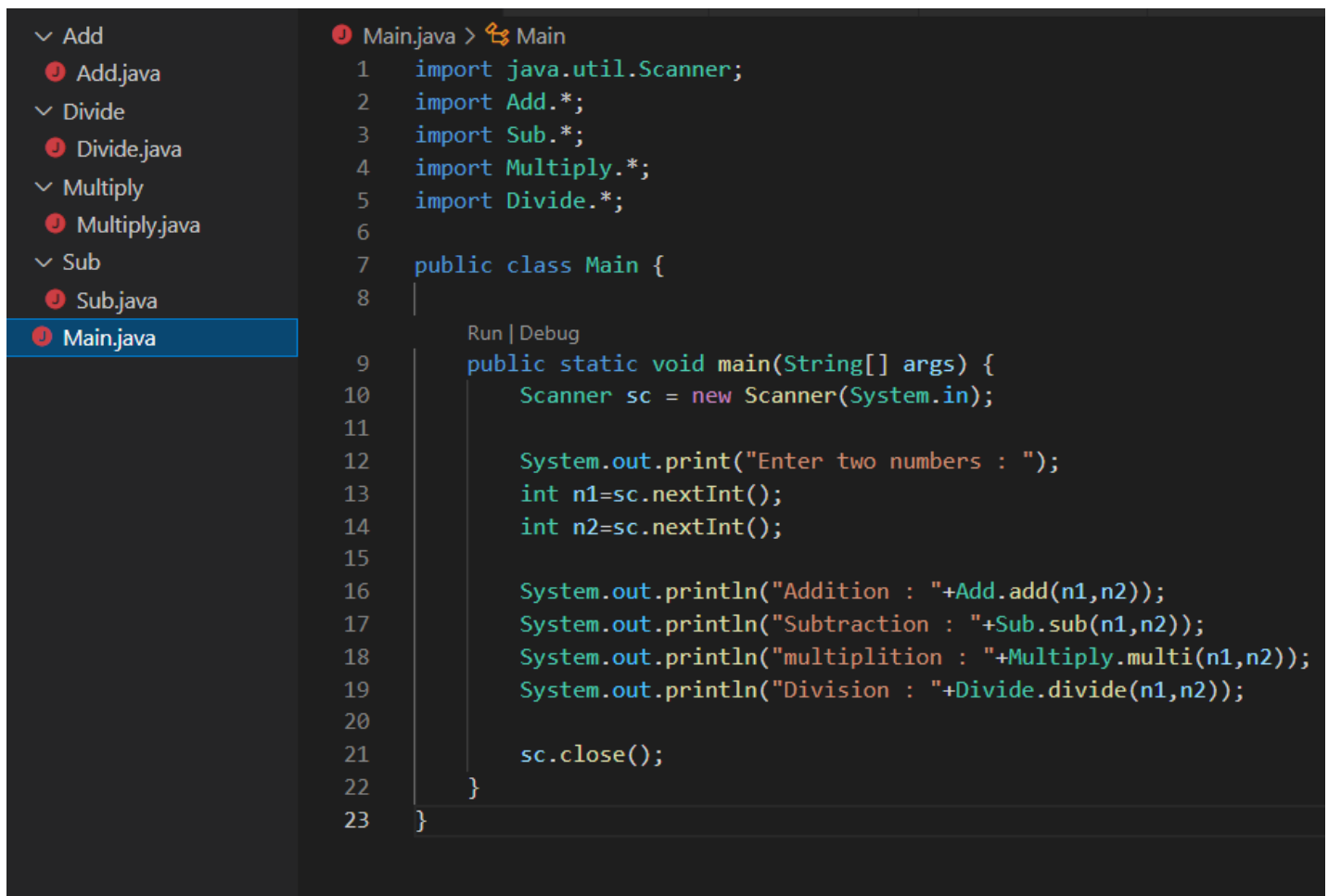
This type of technique is generally used when two packages have the same class name example class Date is present in both the packages **java.util** and **java.sql**.

A.

**Aim :** Write a program to perform four basic Arithmetic Operations using packages  
Each operation should be a part of each class inside the package  
Main class should import all the packages and perform operations .

**Program :**

**Packages paths:**



The screenshot shows an IDE with a project structure on the left and the code for Main.java in the center. The project structure includes a package named 'Add' with classes 'Add.java', 'Divide.java', 'Multiply.java', and 'Sub.java'. The 'Main.java' file is selected. The code in Main.java imports the 'Add', 'Sub', 'Multiply', and 'Divide' packages and implements a 'main' method that prompts the user for two numbers and performs addition, subtraction, multiplication, and division using the respective classes.

```
1  import java.util.Scanner;
2  import Add.*;
3  import Sub.*;
4  import Multiply.*;
5  import Divide.*;
6
7  public class Main {
8
9      Run | Debug
10     public static void main(String[] args) {
11         Scanner sc = new Scanner(System.in);
12
13         System.out.print("Enter two numbers : ");
14         int n1=sc.nextInt();
15         int n2=sc.nextInt();
16
17         System.out.println("Addition : "+Add.add(n1,n2));
18         System.out.println("Subtraction : "+Sub.sub(n1,n2));
19         System.out.println("multiplition : "+Multiply.multi(n1,n2));
20         System.out.println("Division : "+Divide.divide(n1,n2));
21
22         sc.close();
23     }
24 }
```

----- Main class -----

```
import java.util.Scanner;
import Add.*;
import Sub.*;
import Multiply.*;
import Divide.*;

public class Main {

    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);

        System.out.print("Enter two numbers : ");
        int n1=sc.nextInt();
        int n2=sc.nextInt();

        System.out.println("Addition : "+Add.add(n1,n2));
        System.out.println("Subtraction : "+Sub.sub(n1,n2));
        System.out.println("multiplition : "+Multiply.multi(n1,n2));
        System.out.println("Division : "+Divide.divide(n1,n2));

        sc.close();
    }
}
```

----- Code for add class in package -----

```
package Add;

public class Add {
    public static int add(int a,int b){
        return a+b;
    }
}
```

----- Code for subtract class in package -----

```
package Sub;

public class Sub {
    public static int sub(int a,int b){
        return b-a;
    }
}
```

----- Code for multiply class in package -----

```
package Multiply;

public class Multiply {
    public static int multi(int a,int b){
        return a*b;
    }
}
```

----- Code for divide class in package -----

```
package Divide;

public class Divide {
    public static float divide(int a,int b){
        return (float)b/a;
    }
}
```

## Output :

```
PS C:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt11> c::; cd 'c:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt11'; & 'C:\Program Files\Java\jdk-16.0.2\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\IsmailRatlamwala\AppData\Roaming\Code\User\workspaceStorage\6bc1a0e23374ad28359f97af4134a140\redhat.java\jdt_ws\expt11_ad8f9b9b\bin' 'Main'
Enter two numbers : 12 48
Addition : 60
Subtraction : 36
multiplition : 576
Division : 4.0
PS C:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt11> c::; cd 'c:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt11'; & 'C:\Program Files\Java\jdk-16.0.2\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'C:\Users\IsmailRatlamwala\AppData\Roaming\Code\User\workspaceStorage\6bc1a0e23374ad28359f97af4134a140\redhat.java\jdt_ws\expt11_ad8f9b9b\bin' 'Main'
Enter two numbers : 34 467
Addition : 501
Subtraction : 433
multiplition : 15878
Division : 13.735294
PS C:\Users\IsmailRatlamwala\Documents\College prog\Oops Labs\expt11> █
```