DAY-43

------

IMPORTING THE PACKAGES IN THE JAVA:

There are 3 ways to import the packages into the java program they are:

1. import package.\*; --> implicit import statement

2. import package.className; --> explicit import statement

3. using fully qualified name --> 'fully qualified name'

EXAMPLE:

---------

Student.java

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

package pack2;

public class Student

{

String name;

int roll\_no;

public Student(String name,int roll\_no)

{

this.name=name;

this.roll\_no=roll\_no;

}

public void getData()

{

System.out.println("name is :"+name+" roll\_no is : "+roll\_no);

}

}

StudentApp.java

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

package pack1;

//import pack2.\*; // implicit import statement

//import pack2.Student; // explicit import statment

class StudentApp

{

public static void main(String[] args)

{

pack2.Student s = new pack2.Student("ramu",24); // fully qualified name

s.getData();

}

}

OUTPUT:

--------

D:\sep\_batch\javaPrograms\DAY-43>javac -d . Student.java

D:\sep\_batch\javaPrograms\DAY-43>javac -d . StudentApp.java

D:\sep\_batch\javaPrograms\DAY-43>java pack1.StudentApp

name is :ramu roll\_no is : 24

PRE-DEFIEND PACKAGES :

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

These packages are provided by JAVA programming language along woth java software.

examples:

----------

1. java.lang

--> This package is default package in java application, and need no to import this package to the present java file, when we save java file with

.java extension then automatically java.lang package will be imported internally.

This package will provide funadamental classes and interfaces which is required to crearte java programs.

eg: String.class,StringBugffer,StringBuilder,System,class,Int,Float,,,,,,,

2. java.io

--> This package is able to provide predefiend classes and interface inorder to perform the INPUT AND OUTPUT operations in java.

eg: InputStream,Serializable,FileWriter,FileOutputStream.......................................

3. java.util

--> This package is able to provide all pre-defiend classes and interface to represent the DATA STRUCTURS

eg: Collection,List,Set,Queue,Map,HashMap,LinkedList,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,,

OTHER IMPORTENT SUB-PACKAGES IN JAVA PACKAGE ARE:

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

java.awt

javax.swing

java.net

java.sql

java.rmi

programming format:

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

The order of programming format is :

1.comment section

2.packages

3.import

4.classes / interface / Enum

TYPES OF COMMENTS IN JAVA :

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

1. Single line comment

2. multiline comment

3. Documentation comment

EXAMPLE FOR SINGLE LINE COMMENT:

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

// This program add two numbers

class Demo

{

public static void main(String[] args)

{

int a;

int b;

int c;

a=20;

b=5;

c=a+b;

System.out.println("the sum is:"+c);

}

}

OUTPUT:

-------

the sum is:25

EXAMPLE FOR MULTI-LINE COMMENT:

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

/\* This program add two numbers

also subtarct two numbers

multiply two numbers \*/

class Demo

{

public static void main(String[] args)

{

int a;

int b;

int c;

a=20;

b=5;

c=a+b;

System.out.println("the sum is:"+c);

c=a-b;

System.out.println("the difference is:"+c);

c=a\*b;

System.out.println("the product is:"+c);

}

}

OUTPUT:

-------

the sum is:25

the difference is:15

the product is:100

EXAMPLE FOR JAVA DOCUMENTATION :

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

/\*\*

\*This is documentation comment

\*/

public class Demo

{

/\*\*

\* @param a first value

\* @param a first value

\* @return sum of a and b

\*/

public int sum(int a, int b)

{

return a+b;

}

}

NOTE: JAVADOC COMMAND WILL CREATE A HTML PAGE IN WHICH THE DOCUMENTATED DETAILS WILL BE PRESENT.

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