How Integer.parseInt(String x) works internally ?

```
class Arithmetic
{
    public static int doSum(int x, int y)
    {
        return (x+y);
    }
}

public class Calculate
{
    public static void main(String[] args)
    {
        int result = Arithmetic.doSum(12,24);
        System.out.println("Sum is :"+result);
    }
}
```

Note : Arithmetic is a User-defined class which contains a static method doSum(), this method is accepting two parameters of type int, It will accept two parameters, perform the addition and return the result to the caller method.

ECLIPSE IDE :
----

* IDE stands for "**I**ntegrated **D**evelopement **E**nvironment". By using Eclipse IDE, We can develop, cpmpile and execute java programs in a single window.

* By using Eclise IDE we can reduce the development time, Once developement time will be reduced then automatically the cost of the project will also be reduced.

Steps to create a Project in Eclipse IDE :

File -> new -> Project OR Java Project -> Provide the name for the project (Batch 46) -> Finish

* When we expand the project (Batch 46) then we will get **src** (source) folder, Here we can store our java programs **using package.**

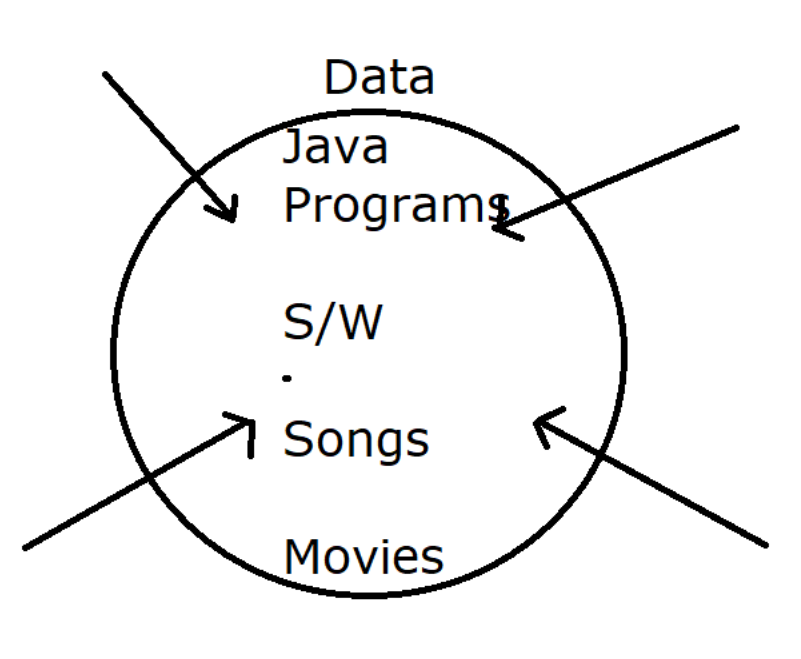
* In order to create a package, Right click on src -> New -> package -> provide name of the package

---

What is a Package ?
----

* A package is a folder in windows. It is used to arrange our java programs so we can get the following two advantages :

    1) Fast Searching of our java classes will become easy
    2) We can reuse the name of java classes into multiple packages (folders)



In this approach we have two drawbacks :
----

1) Fast Searching is not possible

2) Name cannot be reusable

WAP that describes package is a folder in java :
----

* A package is a keyword in java which is used to arrange our java programs in an efficient way.
* Package statement must be first line of any Java program.
* If a java program contains package statement then we have the following command to compile the code

    javac -d . FileName.java
    [javac space -d space . space FileName.java]

Program Example :

```
package com.ravi.arithmetic;

public class Calculate
{
    public static void main(String[] args)
    {
        System.out.println("Welcome to Calculate class");
    }
}
```

Compilation :
----

javac -d . Calculate.java

The above command will compile Calculate.java file, It will create 3 folders

com -> ravi -> arithmetic -> Calculate.class

To execute the program we should write the following command :

java com.ravi.arithmetic.Calculate

Types of Packages :
----

1) Predefined OR Built-in package : The packages which are created by java software people for arranging the programs are called predefined package.

Example : java.lang, java.util, java.io, java.sql, java.net and so on

2) Userdefined Package OR Custom package : The packages which are created by user for arranging the user-defined programs are called user-defined package.

Example :
basic;
com.ravi.basic;
com.tcs.online_shopping;

How to execute command line argument program by using Eclipse IDE :

WAP to find out the area of Rectangle by using Commad Line Argument :

```
package com.ravi.command_line_arg;

public class AreaOfRectangle
{
    public static void main(String[] args)
    {
        int length = Integer.parseInt(args[0]);
        int breadth = Integer.parseInt(args[1]);

        int area = length*breadth;
        System.out.println("Area of Rectangle is :"+area);
    }

}
```

Steps to execute the command Line Argument Program using Eclipse IDE

Right click on the program -> Run As -> Run configuration -> Verify your main class name -> select argument tab -> pass the appropriate value -> Run

WAP to find out the area of the Circle :

```
package com.ravi.command_line_arg;

public class AreaOfCircle
{
    public static void main(String[] args)
    {
      final   double PI = 3.14;
      //String to double
      double radius = Double.parseDouble(args[0]);

      double area = PI * radius * radius;
      System.out.println("Area of Circle is :"+area);

    }
}
```

How to convert String into double value :

Double.parseDouble(String x);

How to convert String into float value :

Float.parseFloat(String x);

Tokens in java :
