

## Understanding Scanner Class

- In java using different classes we can collect input from the users.
- Scanner class is available in 'util' package.
- We can read input from the console using scanner class
- It has been introduced in java 1.5 version.
- Scanner class is capable of reading information from different sources.

Steps:

1. import java.util.Scanner
2. create object of Scanner class
3. use pre-defined methods in the Scanner class to take input  
--->nextInt();  
--->next();

## Scanner Class Methods

Method	Description
nextByte()	Accepts a byte
nextShort()	Accepts a short
nextInt()	Accepts an int
nextLong()	Accepts a long
next()	Accepts a single word
nextLine()	Accept a line of String
nextBoolean()	Accepts a boolean
nextFloat()	Accepts a float
nextDouble()	Accepts a double

Scanner class is used to accept input at run time.

```
1 package com.pack1;
2
3 public class ClassA
4 {
5     void meth1()
6     {
7         System.out.println("meth1() called\n");
8
9         int x=10;
10        int y=20;
11        int z=x+y;
12        System.out.println("z value : "+z);
13    }
14    public static void main(String[] args)
15    {
16        ClassA aobj=new ClassA();
17        aobj.meth1();
18    }
19 }
```

meth1() called  
z value : 30

Except java.lang package, if we want to use any other predefined package we definitely have to import it.

```

1 package com.pack1;
2
3
4 public class ClassA
5 {
6     void meth1()
7     {
8         System.out.println("meth1() called\n");
9
10        Scanner sc=new Scanner(); error bec import util package
11        System.out.println("Enter 1st Number");
12
13        int x=10;
14        int y=20;
15        int z=x+y;
16        System.out.println("z value : "+z);
17    }
18    public static void main(String[] args)
19    {
20        ClassA aobj=new ClassA();
21        aobj.meth1();
22    }
23 }

```

```

1 package com.pack1;
2
3 import java.util.Scanner;
4 public class ClassA
5 {
6     void meth1()
7     {
8         System.out.println("meth1() called\n");
9
10        Scanner sc=new Scanner(); need to initialize with
11        System.out.println("Enter 1st Number"); parameterized constructor
12
13        int x=10;
14        int y=20;
15        int z=x+y;
16        System.out.println("z value : "+z);
17    }
18    public static void main(String[] args)
19    {
20        ClassA aobj=new ClassA();
21        aobj.meth1();
22    }
23 }

```

This is parameterized constructor of a scanner class, which accepts input stream(it is a static variable) as a parameter, input stream is present in input stream class.

```
128     public static final InputStream in = null;
129
130     /**
131      * The "standard" output stream. This stream is already
132      * open and ready to accept output data. Typically this stream
133      * corresponds to display output or another output destination
134      * specified by the host environment or user. The encoding used
135      * in the conversion from characters to bytes is equivalent to
136      * {@link Console#charset()} if the {@code Console} exists,
137      * {@link Charset#defaultCharset()} otherwise.
138      * <p>
139      * For simple stand-alone Java applications, a typical way to write
140      * a line of output data is:
141      * <blockquote><pre>
142      *     System.out.println(data)
143      * </pre></blockquote>
144      * <p>
```

in is static variable  
name

Scanner sc=new Scanner(System.in); calling a static variable by using  
class name

Sysytem is class  
name

```

1 package com.pack1;
2
3 import java.util.Scanner;
4 public class ClassA
5 {
6     void meth1()
7     {
8         System.out.println("meth1() called\n");
9
10        Scanner sc=new Scanner(System.in);
11        System.out.println("Enter 1st Number");
12
13        int x=10;
14        int y=20;
15        int z=x+y;
16        System.out.println("z value : "+z);
17    }
18    public static void main(String[] args)
19    {
20        ClassA aobj=new ClassA();
21        aobj.meth1();
22    }
23 }

```

What is input stream, in java we have a concept known as IO streams (input output streams). Io streams are used to read the data from a file or write data into a file.

What is a stream

A stream is a continuous uninterrupted flow.

Scanner class has a parameterized constructor which accepts input stream as its parameter.

Input stream means we will be giving input from the keyboard, that input will be accepted by input stream and that is static variable which is present in system class



```
1 package com.pack1;
2
3 import java.util.Scanner;
4 public class ClassA
5 {
6     void meth1()
7     {
8         System.out.println("meth1() called\n");
9
10        Scanner sc=new Scanner(System.in);
11
12        System.out.println("Enter 1st Number");
13        int x=sc.nextInt();
14        System.out.println("x : "+x);
15
16        System.out.println("\nEnter 2nd Number");
17        int y=sc.nextInt();
18        System.out.println("y : "+y);
19        int z=x+y;
20        System.out.println("\nz value : "+z);
21    }
22    public static void main(String[] args)
23    {
```

meth1() called  
Enter 1st Number  
99  
x : 99  
Enter 2nd Number  
1  
y : 1  
z value : 100

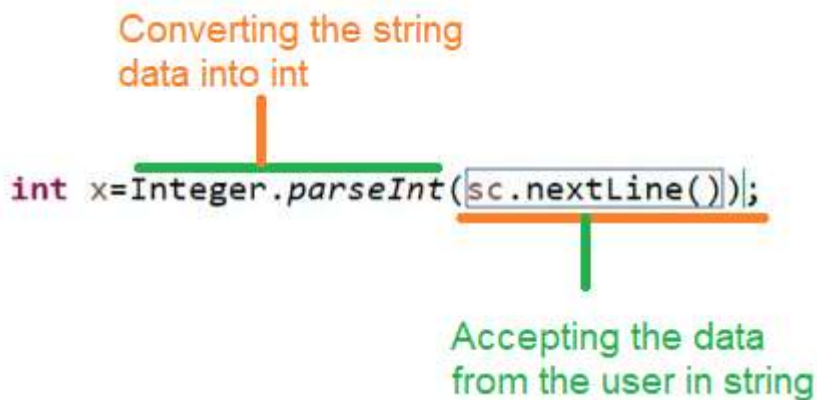
```
1 package com.pack1;
2
3 import java.util.Scanner;
4 public class ClassA
5 {
6     void meth1()
7     {
8         System.out.println("meth1() called\n");
9
10        Scanner sc=new Scanner(System.in);
11        warning because resource not close
12        System.out.println("Enter 1st Number");
13        int x=sc.nextInt();
14        System.out.println("x : "+x);
15
16        System.out.println("\nEnter 2nd Number");
17        int y=sc.nextInt();
18        System.out.println("y : "+y);
19        int z=x+y;
20        System.out.println("\nz value : "+z);
21    }
22    public static void main(String[] args)
23    {
```

meth1() called  
Enter 1st Number  
99  
x : 99  
Enter 2nd Number  
1  
y : 1  
z value : 100

```
6= void meth1()
7 {
8     System.out.println("meth1() called\n");
9
10    Scanner sc=new Scanner(System.in);
11
12    System.out.println("Enter 1st Number");
13    int x=sc.nextInt();
14    System.out.println("x : "+x);
15
16    System.out.println("\nEnter 2nd Number");
17    int y=sc.nextInt();
18    System.out.println("y : "+y);
19    int z=x+y;
20    System.out.println("\nz value : "+z);
21    sc.close();
22 }
23= public static void main(String[] args)
24 {
25     ClassA aobj=new ClassA();
26     aobj.meth1();
27 }
28 }
```

meth1() called  
Enter 1st Number  
99  
x : 99  
Enter 2nd Number  
1  
y : 1  
z value : 100

Parseint method present in integer class it is a static method which is parameterized



```

6= void meth1()
7 {
8     System.out.println("meth1() called\n");
9
10    Scanner sc=new Scanner(System.in);
11
12    System.out.println("Enter 1st Number");
13    //int x=sc.nextInt();
14    int x=Integer.parseInt(sc.nextLine());
15    System.out.println("x : "+x);
16
17    System.out.println("\nEnter 2nd Number");
18    //int y=sc.nextInt();
19    int y=Integer.parseInt(sc.nextLine());
20    System.out.println("y : "+y);
21
22    int z=x+y;
23    System.out.println("\nz value : "+z);
24    sc.close();
25 }
26= public static void main(String[] args)
27 {
28     ClassA aobj=new ClassA();

```

```

meth1() called
Enter 1st Number
99
x : 99
Enter 2nd Number
1
y : 1
z value : 100

```

```

25 }
26= String meth2(int i)
27 {
28     System.out.println("meth2() called");
29     if(i%2==0)
30     {
31         System.out.println("If block executed");
32         System.out.println("Please give your result");
33         return sc.nextLine();
34     }
35     else
36     {
37         System.out.println("else block executed");
38         System.out.println("Please give your result");
39         return sc.nextLine();
40     }
41 }
42= public static void main(String[] args)
43 {
44     ClassA aobj=new ClassA();
45     //aobj.meth1();
46     System.out.println("meth2() is returning : ");
47 }

```

```

meth2() called
else block executed
Please give your result
The given parameter id ODD number
meth2() is returning : The given parameter id ODD number

```



Whenever we are writing scanner class in a method we should close it, out of the method it is fine

```
3 import java.util.Scanner;
4
5 public class ClassA
6 {
7     static Scanner sc=new Scanner(System.in); // Instance Variable
8
9     void meth1()
10    {
11        System.out.println("meth1() called\n");
12
13        System.out.println("Enter 1st Number");
14        //int x=sc.nextInt();
15        int x=Integer.parseInt(sc.nextLine());
16        System.out.println("x : "+x);
17
18        System.out.println("\nEnter 2nd Number");
19        //int y=sc.nextInt();
20        int y=Integer.parseInt(sc.nextLine());
21        System.out.println("y : "+y);
22
23        int z=x+y;
24        System.out.println("\nz value : "+z);
25    }
26    String meth2(int i)
27    {
28        System.out.println("meth2() called");
29        if(i%2==0)
30        {
31            System.out.println("If block executed");
32            System.out.println("Please give your result");
33            return sc.nextLine();
34        }
35        else
36        {
37            System.out.println("else block executed");
38            System.out.println("Please give your result");
39            return sc.nextLine();
40        }
41    }
42 }
```

```

41     }
42     void checkEligibility(String name, int age)
43     {
44         if(age>=18)
45             System.out.println(name+" you are eligible to vote");
46         else
47             System.out.println(name+" you are eligible to vote after "+(18-age)+" years");
48     }
49     public static void main(String[] args)
50     {
51         ClassA aobj=new ClassA();
52         //aobj.meth1();
53         //System.out.println("meth2() is returning : "+aobj.meth2(5));
54
55         System.out.println("Enter your name");
56         //String uname=aobj.sc.next(); // this is the way to access a NON-static variable in side a static method
57         String uname=sc.next();
58
59         System.out.println("Enter your age");
60         //int age=aobj.sc.nextInt();
61         int uage=sc.nextInt();
62
63         aobj.checkEligibility(uname, uage);
64
65     }
66 }

```

**Enter EmpName :**

XXXXX

**Enter EmpId :**

XXXXX

**Enter EmpSal :**

XXXXX

**Enter EmpAddress :**

XXXXX XXXXX XXXXX

**Details Entered by the user**

**EmpName :** XXXXX

**EmpId :** XXXXX

**EmpSal :** XXXXX

**EmpAddress :** XXXXX XXXXX XXXXX