

JAVA INPUT OUTPUT

Java provides different ways to get input from the user.

However, in this tutorial, you will learn to get input from user using the object of `Scanner` class.

In order to use the object of Scanner, we need to import `java.util.Scanner` package.

```
import java.util.Scanner;
```

Then, we need to create an object of the `Scanner` class. We can use the object to take input from the user.

```
// create an object of Scanner  
Scanner input = new Scanner(System.in);  
  
// take input from the user  
int number = input.nextInt();
```

Example: Get Integer Input From the User

```
import java.util.Scanner;
```

```
class Input {  
    public static void main(String[] args) {  
  
        Scanner input = new Scanner(System.in);  
  
        System.out.print("Enter an integer: ");  
        int number = input.nextInt();  
        System.out.println("You entered " + number);  
  
    }  
}
```

Output:

```
Enter an integer: 23  
You entered 23
```

In the above example, we have created an object named `input` of the `Scanner` class. We then call the `nextInt()` method of the `Scanner` class to get an integer input from the user.

Similarly, we can use `nextLong()`, `nextFloat()`, `nextDouble()`, and `next()` methods to get `long`, `float`, `double`, and `string` input respectively from the user.

Java Output

In Java, you can simply use to send output to standard output (screen).

```
System.out.println(); or
```

```
System.out.print(); or
```

```
System.out.printf();
```

Here,

- `System` is a class
- `out` is a `public static` field: it accepts output data.

Let's take an example to output a line.

```
class Topperworld {  
    public static void main(String[] args) {  
  
        System.out.println("Java programming is interesting.");  
    }  
}
```

Output:

```
Java programming is interesting.
```

Here, we have used the `println()` method to display the string.

Difference between println(), print().

- `print()` - It prints string inside the quotes.
- `println()` - It prints string inside the quotes similar like `print()` method. Then the cursor moves to the beginning of the next line.

Example: print() and println()

```
class Output {  
    public static void main(String[] args) {  
  
        System.out.println("1. println ");  
        System.out.println("2. println ");  
  
        System.out.print("1. print ");  
        System.out.print("2. print");  
    }  
}
```

Output:

```
1. println  
2. println  
1. print 2. print
```