

# Java Output

## What is Output?

**Output** is how your Java program displays information on the screen for users to see. It's your program's way of "talking" or communicating results, messages, and data to the person using the program.

### Simple Analogy: Output is Like Speaking

Think of your Java program like a person who wants to share information:

- ☐ **Speaking out loud** = `System.out.println()`
  - You say something, then pause (move to next line)
- **Texting continuously** = `System.out.print()`
  - You keep typing without pressing Enter

**Just like speaking, you want your output to be clear, helpful, and easy to understand!**

## The Three Main Output Methods

Method	What It Does	Cursor Position After	Best Used For
<code>System.out.print()</code>	Prints text, stays on same line	At the end of printed text	Continuing on same line
<code>System.out.println()</code>	Prints text, moves to new line	Beginning of next line	Separate messages

### Example 1: Understanding the Difference

```
System.out.print("Hello ");
System.out.print("beautiful ");
System.out.println("world!");
System.out.println("This is a new line.");
System.out.print("Back to ");
System.out.print("same line.");
```

#### Output:

```
Hello beautiful world!
This is a new line.
Back to same line.
```

**Notice:** `print()` continues on the same line, `println()` moves to the next line.

### Example 2: Printing Variables and Data

```
String studentName = "Maria";
int studentAge = 16;
double testScore = 95.7;
boolean isPassing = true;
```

```
System.out.println("Student Information:");
System.out.println("Name: " + studentName);
System.out.println("Age: " + studentAge + " years old");
System.out.println("Test Score: " + testScore + "%");
System.out.println("Passing: " + isPassing);
```

```
Student Information:
Name: Maria
Age: 16 years old
Test Score: 95.7%
Passing: true
```

## String Concatenation: Joining Text Together

**Concatenation** means connecting strings together using the + operator.

### Good Examples:

```
// Clear and readable
System.out.println("Hello " + name);

// Multiple parts
System.out.println("Score: " + points + "/100");

// With calculations
System.out.println("Total: " + (a + b));
```

### Common Mistakes:

```
// No spaces - looks bad
System.out.println("Hello" + name);

// Confusing math order
System.out.println(a + b + " total");

// Missing context
System.out.println(result);
```

## Example 3: Special Characters and Escape Sequences

```
System.out.println("Line 1\nLine 2"); // \n = new line
System.out.println("Column1\tColumn2"); // \t = tab space
System.out.println("He said \"Hello!\""); // \" = quote mark
System.out.println("File: C:\\MyFiles\\data.txt"); // \\ = backslash
```

```
Line 1
Line 2
Column1      Column2
```

```
He said "Hello!"
File: C:\MyFiles\data.txt
```

## Practical Output Examples

### Example 4: Creating a Simple Menu

```
System.out.println("=====");
System.out.println(" MAIN MENU");
System.out.println("=====");
System.out.println("1. View Account Balance");
System.out.println("2. Make a Deposit");
System.out.println("3. Make a Withdrawal");
System.out.println("4. Exit");
System.out.println("=====");
System.out.print("Please choose an option (1-4): ");
```

### Example 5: Progress Bar Simulation

```
System.out.println("Loading...");
System.out.print("[");

for (int i = 0; i < 10; i++) {
    System.out.print("=");
    // In real program, you'd add a delay here
}
System.out.println("] 100% Complete!");
```

```
Loading...
[=====] 100% Complete!
```

## Best Practices for Output

### Do This:

- Use **clear, descriptive messages**
- Include **units and context** ("5 meters", not just "5")
- Use **consistent formatting**
- Add **visual separators** for sections
- End input prompts with ": " or "? "

### Don't Do This:

- Print **cryptic abbreviations**
- Output numbers **without context**

- Mix **different formatting styles**
- Create **walls of text**
- Forget **spaces around text**
- Use **too many decimal places**

## Common Output Mistakes and Solutions

Problem	Bad Example	Good Solution
No spaces in output	"Hello" + name + "!"	"Hello " + name + "!"
No context for numbers	System.out.println(age);	"Age: " + age + " years"
Unclear prompts	"Enter:"	"Enter your name: "

### Your Turn: Write Your Own Definition

**What is output in Java? How would you explain it to a friend?**

Write your definition in your own words:

**Give three examples of when you would use each output method:**

**System.out.print():**

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_

**System.out.println():**

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_