



JAVA - String (Different Ways to Print Output in Java)

Here is all list of

Description	Method	Example
Prints text without new line	<code>print()</code>	<code>System.out.print("Hi");</code>
Prints text with new line	<code>println()</code>	<code>System.out.println("Hi");</code>
Prints formatted text	<code>printf()</code>	<code>System.out.printf("%d", 10);</code>
Returns formatted string	<code>String.format()</code>	<code>String.format("%d", 10);</code>
Prints error in red (stderr)	<code>System.err.println()</code>	<code>System.err.println("Error!");</code>
GUI popup message	<code>JOptionPane</code>	<code>JOptionPane.showMessageDialog(null,"Hi");</code>
Professional logging	<code>Logger</code>	<code>logger.info("msg");</code>

```
public class PrintMthod{
    public static void main(String[] args){

        System.out.print("Hi");
        System.out.println("Hi");
        System.out.printf("%d", 10);
        String.format("%d", 10);
        System.err.println("Error!");
        JOptionPane.showMessageDialog(null,"Hi");
        logger.info("msg");
    }
}
```

Java Format Specifiers (printf / format)

Format specifiers are placeholders that tell Java how to format values when printing.

They always start with `%` followed by a **conversion character**.

Commonly Used Specifiers

Specifier	Data Type	Example	Output
%d	Integer (decimal)	System.out.printf("%d", 25);	25
%f	Floating-point (decimal)	System.out.printf("%.2f", 12.3456);	12.35
%c	Character	System.out.printf("%c", 'A');	A
%s	String	System.out.printf("%s", "Hello");	Hello
%b	Boolean	System.out.printf("%b", true);	true

Integer Format Specifiers

Specifier	Meaning	Example	Output
%d	Decimal integer	System.out.printf("%d", 100);	100
%o	Octal integer	System.out.printf("%o", 100);	144
%x	Hexadecimal (lowercase)	System.out.printf("%x", 100);	64
%X	Hexadecimal (uppercase)	System.out.printf("%X", 100);	64

Floating-Point Format Specifiers

Specifier	Meaning	Example	Output
%f	Decimal floating-point	System.out.printf("%f", 12.3);	12.300000
%.nf	n = decimal places	System.out.printf("%.2f", 12.3456);	12.35
%e	Scientific notation (lowercase)	System.out.printf("%e", 12345.67);	1.234567e+04
%E	Scientific notation (uppercase)	System.out.printf("%E", 12345.67);	1.234567E+04
%g	Automatic (uses %f or %e)	System.out.printf("%g", 0.00012345);	0.00012345
%a	Hexadecimal floating-point	System.out.printf("%a", 12.5);	0x1.9p3

Other Specifiers

Specifier	Meaning	Example	Output
%n	Newline (platform-independent)	System.out.printf("Hello%nWorld");	(newline)
%%	Prints % sign	System.out.printf("100%%");	100%