Different ways to print in Java

We can use the following ways to print in Java:

1. System.out.print() → No newline at the end!
2. System.out.println() → Prints a new line at the end
3. System.out.printf()
4. System.out.format()

System.out.printf("%c", ch)

↳ %d for int
%f for float
%c for char
%s for string

Printing

1. System.out.print( )
2. System.out.println( )
3. System.out.printf( )
4. System.out.format( )

```
int x=10, y=20;
```

```
S.o.p(x+y+" sum");
```

30 + "sum"

Output: 30 Sum

```
S.o.p("sum "+(x+y));
```

"sum 10" + y

Output: "sum 1020

※. Println mai ham sirf single command pass kar sakta hai.

※. Agar hma ek sa jayda chij print karna hoto hma "concatenation" karna hoga.

※. During concatenation, ham first object ka baad "+" ka use karta hai aur jitna baar concatenate karna hota hai utna baar "+" ka use karta hai after each object.

Ya par both given example meaning is different.

※. Ise condition mai phela sum hoga then concatenate hoga.

※. Ise condition mai series of concatination hoga.

Printing

1. System.out.print( )
2. System.out.println( )
3. System.out.printf( )
4. System.out.format( )

int x=10, y=20;

"sum of 10 and 20 is 30

"sum of "+x+" and "+y+" is "+(x+y)

Agar hma add karna ho then (x+y) ka use karna hoga kyuki "()" ka priority order high hota hai.

Printing

Argument index ka use ham tab karta hai jab kisi specific index ka item ko print karna ho.

Width ka use tab hota hai jab kuch specific space ka baad kisi item ko print karna hota hai.

Flag ka use different trah sai hot hai. Some common as follow:

For integer:

i) 0 → iseka use jo blank space aaya hai width ki wja sai usko zero sai fill karna ka karta hai.

ii) + → iska use kisi bhi digit ka sign ko show karna ka liya karta hai.

iii) (→ iska use agar input -ve hai to usa +ve mai convert kai bracket mai show karna ka liya karta hai.

For string:

i) + → align toward right after sapce as per width given.

ii) - → align toward left after sapce as per width given.

Precision ka use ham mostly float data type mai karta hai. Iska help sai ham decimal ka baad jitna value chahiya utna value print karwa sakta hai.

Format Specifier

%[argument_index$][flags][width][.precision]conversion

argument index - 1$, 2$, 3$ .....

flag '-' , '+' , '0' , ' ' , '('

conversion -

char	c
int	d, o, x
float	f, e, g
String	s

```java
package printing;

public class Printing {

    public static void main(String[] args) {

        int x=10;
        float y=12.56f;
        char z='A';

       System.out.printf("Hello %d %f %c World\n",x,y,z);
    }

}
```

Output - Printing (run)

```
ant -f /Users/abdulbari/NetBeansProjects/Printing -Dnb.internal.action.name=run run
init:
Deleting: /Users/abdulbari/NetBeansProjects/Printing/build/built-jar.properties
deps-jar:
Updating property file: /Users/abdulbari/NetBeansProjects/Printing/build/built-jar.properties
Compiling 1 source file to /Users/abdulbari/NetBeansProjects/Printing/build/classes
compile:
run:
Hello 10 12.560000 A World
BUILD SUCCESSFUL (total time: 0 seconds)
```

```java
package printing;

public class Printing {

    public static void main(String[] args) {

        int x=10;
        float y=0.0012f;
        char z='A';
        String str="Java Program";

       System.out.printf("%3$s %2$f %1$d",x,y,str);
    }

}
```

Output - Printing (run)

```
ant -f /Users/abdulbari/NetBeansProjects/Printing -Dnb.internal.action.name=run run
init:
Deleting: /Users/abdulbari/NetBeansProjects/Printing/build/built-jar.properties
deps-jar:
Updating property file: /Users/abdulbari/NetBeansProjects/Printing/build/built-jar.properties
Compiling 1 source file to /Users/abdulbari/NetBeansProjects/Printing/build/classes
compile:
run:
Java Program 0.001200 10BUILD SUCCESSFUL (total time: 0 seconds)
```

Format Specifier	Data Type	Output
%a	floating point (except BigDecima l)	Returns Hex output of floating-point number.
%b	Any type	" true " if non-null, " false " if null
%c	Character	Unicode character
%d	integer (incl. byte, short, int, long, bigint)	Decimal Integer
%e	floating point	Decimal number in scientific notation
%f	floating point	Decimal number
%g	floating point	Decimal number, possibly in scientific notation depending on the precision and value.
%h	any type	Hex String of value from hashCode( ) method.
%n	None	Platform-specific line separator.
%o	integer (incl. byte, short, int, long, bigint)	Octal number
%s	any type	String value
%t	Date/Time (incl. long, Calendar, Date and TemporalAccessor)	%t is the prefix for Date/Time conversions. More formatting flags are needed after this. See Date/Time conversion below.
%x	integer (incl. byte, short, int, long, bigint)	Hex string.