

Java printf para dar formato a los datos de salida

Vamos a ver como utilizar printf para dar formato a los datos que se imprimen por pantalla en Java.

Este problema se nos plantea por ejemplo cuando queremos mostrar un número de tipo float o double con un número determinado de decimales y no con los que por defecto muestra Java.

A partir de la versión Java 5 se incorporan los métodos format y printf que permiten aplicar un formato a la salida de datos por pantalla.

Ambos realizan la misma función, tienen exactamente el mismo formato y emulan la impresión con formato printf() de C.

Veamos primero varios ejemplos de printf en Java y después explicaremos en detalle la sintaxis de printf.

Si queremos mostrar el número 12.3698 de tipo double con dos decimales:

```
System.out.printf("%.2f", 12.3698);
```

El primer % indica que en esa posición se va a escribir un valor. El valor a escribir se encuentra a continuación de las comillas.

.2 indica el número de decimales.

La f indica que el número es de tipo float o double. En la tabla que aparece más adelante podeis ver todos los caracteres de conversión para todos los tipos de datos.

La salida por pantalla es:

12,37

Comprobamos que printf realiza un redondeo para mostrar los decimales indicados.

Lo más común será que tengamos el valor en una variable, en ese caso si queremos escribir el valor de n con tres decimales:

```
double n = 1.25036;
```

```
System.out.printf("%.3f", n);
```

La salida por pantalla es:

1,250

Para mostrar el signo + en un número positivo:

```
double n = 1.25036;
```

```
System.out.printf("%+.3f ", n);
```

La salida por pantalla es:

+1.250

Si el número a mostrar es un entero se utiliza el carácter d:

```
int x = 10;
```

```
System.out.printf("%d", x);
```

La salida por pantalla es:

10

Para mostrarlo con signo:

```
int x = 10;
```

```
System.out.printf("%+d %n", x);
```

La salida por pantalla es:

+10

Para mostrar varias variables pondremos tantos % como valores vamos a mostrar. Las variables se escriben a continuación de las comillas separadas por comas:

```
double n = 1.25036;
```

```
int x = 10;
```

```
System.out.printf("n = %.2f x = %d", n, x);
```

La salida por pantalla es:

n = 1,25 x = 10

Cuando hay varias variables podemos indicar de cual de ellas es el valor a mostrar escribiendo 1\$, 2\$, 3\$, indicando que el valor a mostrar es el de la primera variable que aparece a continuación de las comillas, de la segunda, etc.

La instrucción anterior la podemos escribir así:

```
System.out.printf("n = %1$.2f x = %2$d %n", n, x);
```

Este número es opcional, si no aparece se entenderá que el primer valor proviene de la primera variable, el segundo de la segunda, etc.

Si queremos mostrar el número 123.4567 y su cuadrado ambos con dos decimales debemos escribir:

```
double n = 123.4567;
```

```
System.out.printf("El cuadrado de %.2f es %.2f\n", n, n*n);
```

La salida por pantalla es:

El cuadrado de 123,46 es 15241,56

printf permite mostrar valores con un ancho de campo determinado.

Por ejemplo, si queremos mostrar el contenido de n en un ancho de campo de 10 caracteres escribimos:

```
double n = 1.25036;
```

```
System.out.printf("%+10.2f %n", n);
```

La salida por pantalla es:

bbbbbb+1.25

Donde cada b indica un espacio en blanco.

El 10 indica el tamaño en caracteres que ocupará el número en pantalla. Se cuentan además de las cifras del número el punto decimal y el signo si lo lleva. En este caso el número ocupa un espacio de 5 caracteres (3 cifras, un punto y el signo) por lo tanto se añaden 5 espacios en blanco al principio para completar el tamaño de 10.

Si queremos que en lugar de espacios en blancos nos muestre el número completando el ancho con ceros escribimos:

```
System.out.printf("%+010.2f %n", n);
```

La salida por pantalla es:
+000001.25

Más ejemplos de printf:

Mostrar el número 1.22 en un ancho de campo de 10 caracteres y con dos decimales.

```
double precio = 1.22;
```

```
System.out.printf("%10.2f", precio);
```

La salida por pantalla es:
bbbbbb1.22
(el carácter b indica un espacio en blanco)

El número ocupa un espacio total de 10 caracteres incluyendo el punto y los dos decimales.

Mostrar la cadena "Total:" con un ancho de 10 caracteres alineada a la izquierda:

```
System.out.printf("%10s", "Total:");
```

La salida por pantalla es:
Total:bbbb

El carácter s indica que se va a mostrar una cadena de caracteres. El signo indica alineación a la izquierda.

Mostrar la cadena "Total:" con un ancho de 10 caracteres y alineada a la derecha:

```
System.out.printf("%10s", "Total:");
```

La salida por pantalla es:
bbbbTotal:

CARACTERES DE CONVERSIÓN			
Carácter	Tipo	Carácter	Tipo
d	Número entero en base decimal	X, x	Número entero en base hexadecimal
f	Número real con punto fijo	s	String
E, e	Número real notación científica	S	String en mayúsculas
g	Número real. Se representará con notación científica si el número es muy grande o muy pequeño	C, c	Carácter Unicode. C: en mayúsculas