

printf()输出格式大全

学习了《[C语言printf\(\)和puts\(\)的简单使用](#)》，大家应该能够使用 printf() 输出简单格式的数据了，%d、%f、%c 和 %s 只是 printf() 的格式控制字符串中转化字符的最简形式，printf() 格式控制字符串的完整形式为：

```
%[标志][输出最小宽度][.精度][数据长度]数据类型字符
```

其中，`[]` 表示可选项。

1. 数据类型字符

用于表示输出数据的类型，其格式字符和意义如表 2 所示。

表 2：C语言中 printf() 函数控制输出数据类型的格式字符

格式字符	输出示例	意义
d	printf("%d",16);	以十进制输出带符号整数（正数不输出符号）
u	printf("%u",16);	以十进制输出无符号整数
o	printf("%o",75);	以八进制输出无符号整数（不输出前缀 0）
x	printf("%x",5B);	以十六进制输出无符号整数（不输出前缀 0x）
f	printf("%f",3.14);	以小数形式输出单、双精度实数
e	printf("%e",30000);	以指数形式输出单、双精度实数
g	printf("%e",30000);	以 %f、%e 中较短的宽度输出单、双精度实数
c	printf("%c",'x');	输出单个字符
s	printf("%s","Chi");	输出字符串

2. 标志

标志字符有 -、+、#、空格、0 五种，其意义如表 3 所示。

表 3：C语言中 printf() 函数控制数据输出形式的标志字符意义

格式字符	意义	↑

-	输出结果左对齐，右边填充格（和输出最小宽度搭配使用）
+	输出结果右对齐，左边填充格（和输出最小宽度搭配使用），同时显示符号
#	对 c、s、d、u 类无影响，对 o 类输出时加前缀 0，x 类输出时加前缀 0x
空格	输出符号，值为正时冠以空格，为负时冠以负号
0	放置在十进制整数的输出最小宽度前，用于当实际位数少于最小宽度时在前面补 0

3. 输出最小宽度

用十进制整数表示输出的最少位数，若实际位数多于定义的宽度，则按实际位数输出，实际位数少于定义的宽度则补以空格或 0。

4. 精度

精度格式符以小数点 . 开头，后跟十进制整数。如果输出数字，则表示其小数位数，如果输出字符，则表示输出字符的个数。若实际位数大于定义的精度，则截去超出的部分。

5. 数据长度

长度格式符有 h 和 l 两种。h 表示以短整型输出整数或以单精度输出浮点数，l 表示以长整型输出整数或以双精度输出浮点数。

汇总和示例

图 2 展示了使用 printf() 函数进行格式化输出数据的方式。

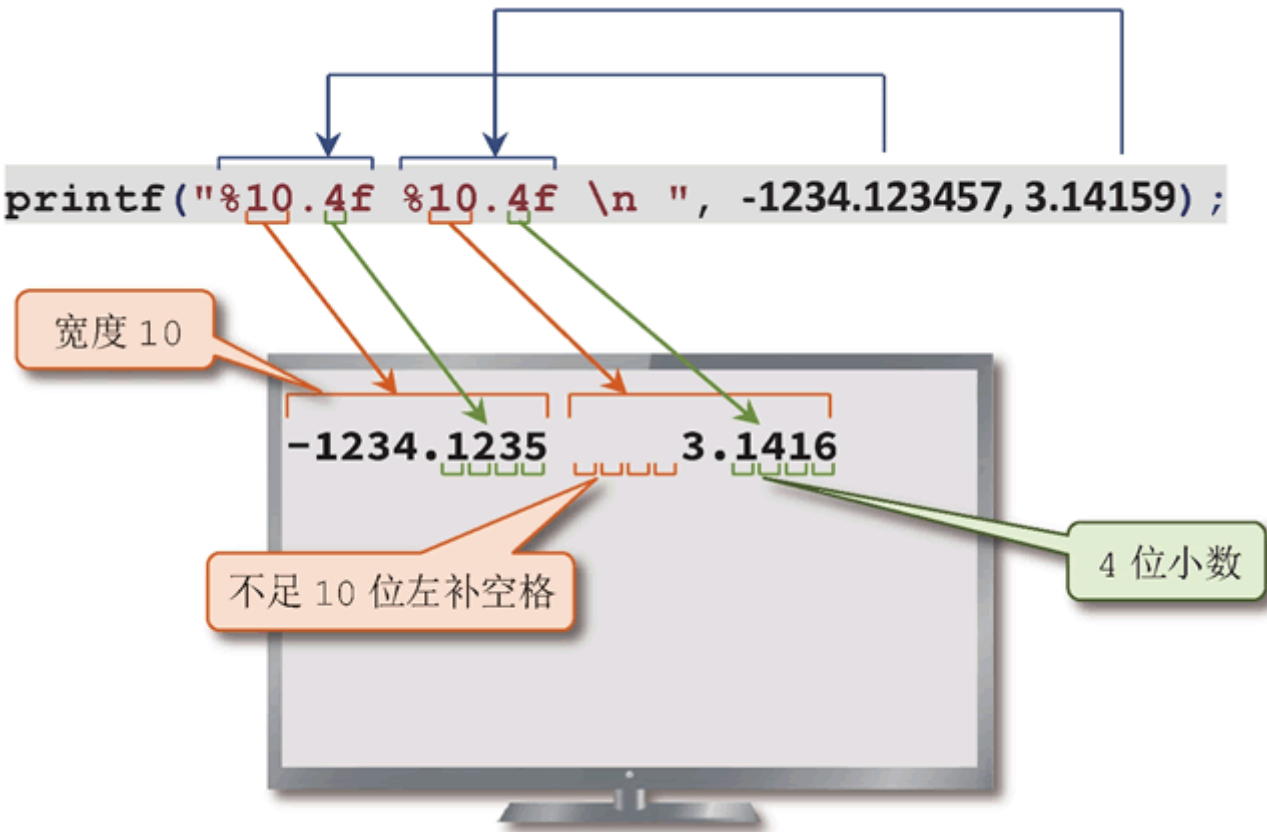


图 2：printf() 函数格式输出图例详解



printf()函数进行格式化输出数据的两个实例如代码清单 1 和代码清单 2 所示。

代码清单 1: printf() 函数格式输出示例

```
01. #include <stdio.h>
02. #include <stdlib.h>
03. int main( )
04. {
05.     printf("\n “+” 标志输出右对齐的带符号数据: \n\n");
06.     printf("%+10d %+10.2f %+10s\n", 56, -809.56, "American");
07.     printf("%+10d %+10.2f %+10s\n", 5668, -23.5, "USA");
08.     printf("\n “-” 标志输出左对齐的数据: \n\n");
09.     printf("%-10d %-10.2f %-10s\n", 56, -809.56, "American");
10.     printf("%-10d %-10.2f %-10s\n", 5668, -23.5, "USA");
11.     printf("\n “#” 标志输出带前缀的八进制数和十六进制数: \n\n");
12.     printf("%#o %#x\n", 0546, 0x5B);
13.     printf("\n输出不带前缀的八进制数和十六进制数: \n\n");
14.     printf("%o %x\n\n", 0546, 0x5B);
15.     system("pause");
16.     return 0;
17. }
```

运行结果:

```
“+” 标志输出右对齐的带符号数据:
          +56          -809.56      American
        +5668          -23.50              USA

“-” 标志输出左对齐的数据:
56          -809.56      American
5668        -23.50      USA

“#” 标志输出带前缀的八进制数和十六进制数:
0546 0x5b

输出不带前缀的八进制数和十六进制数:
546 5b
```

代码清单 2: printf() 函数格式输出示例

```
01. #include <stdio.h>
02. #include <stdlib.h>
03. int main( )
```

```
04.  {
05.     printf("“%10.4f” 输出宽度10带4位小数的单精度浮点数: \n");
06.     printf("%10.4f %10.4f\n", -1234.123457, 3.14159);
07.     printf("\n“%15.8lf” 输出宽度15带8位小数的双精度浮点数: \n");
08.     printf("%15.8lf %15.8lf\n", -123.123456789, 3.14159);
09.     printf("\n“%10.4s” 输出宽度10含4个字符的字符串: \n");
10.     printf("%10.4s %10.4s", "ABCDEFGH", "english");
11.     system("pause");
12.     return 0;
13. }
```

运行结果:

```
“%10.4f” 输出宽度10带4位小数的单精度浮点数:
-1234.1235      3.1416

“%15.8lf” 输出宽度15带8位小数的双精度浮点数:
-123.12345679   3.14159000

“%10.4s” 输出宽度10含4个字符的字符串:
  ABCD          enli
```

优秀文章

Shell命令的标准格式

Go语言浮点类型（小数）

Shell while循环的用法

Tkinter键盘事件处理

Python pickle模块的用法

Nginx server_name: 配置主机名称

C语言的特点汇总

C语言函数参数及其传递方式

再谈Java包

Python类的继承（非常详细）

weixueyuan.net

