**Shell printf 命令**

上一章节我们学习了 Shell 的 echo 命令，本章节我们来学习 Shell 的另一个输出命令 printf。

printf 命令模仿 C 程序库（library）里的 printf() 程序。

printf 由 POSIX 标准所定义，因此使用 printf 的脚本比使用 echo 移植性好。

printf 使用引用文本或空格分隔的参数，外面可以在 printf 中使用格式化字符串，还可以制定字符串的宽度、左右对齐方式等。默认 printf 不会像 echo 自动添加换行符，我们可以手动添加 \n。

**printf 命令的语法：**

printf format-string [arguments...]

参数说明：

format-string: 为格式控制字符串

arguments: 为参数列表。

**实例如下：**

$ echo "Hello, Shell"

Hello, Shell

$ printf "Hello, Shell\n"

Hello, Shell

$

接下来,我来用一个脚本来体现printf的强大功能：

#!/bin/bash

# author:教程

# url:www.runoob.com

printf "%-10s %-8s %-4s\n" 姓名 性别 体重kg

printf "%-10s %-8s %-4.2f\n" 郭靖 男 66.1234

printf "%-10s %-8s %-4.2f\n" 杨过 男 48.6543

printf "%-10s %-8s %-4.2f\n" 郭芙 女 47.9876

执行脚本，输出结果如下所示：

姓名 性别 体重kg

郭靖 男 66.12

杨过 男 48.65

郭芙 女 47.99

%s %c %d %f都是格式替代符

%-10s 指一个宽度为10个字符（-表示左对齐，没有则表示右对齐），任何字符都会被显示在10个字符宽的字符内，如果不足则自动以空格填充，超过也会将内容全部显示出来。

%-4.2f 指格式化为小数，其中.2指保留2位小数。

**更多实例：**

#!/bin/bash

# author:教程

# url:www.runoob.com

# format-string为双引号

printf "%d %s\n" 1 "abc"

# 单引号与双引号效果一样

printf '%d %s\n' 1 "abc"

# 没有引号也可以输出

printf %s abcdef

# 格式只指定了一个参数，但多出的参数仍然会按照该格式输出，format-string 被重用

printf %s abc def

printf "%s\n" abc def

printf "%s %s %s\n" a b c d e f g h i j

# 如果没有 arguments，那么 %s 用NULL代替，%d 用 0 代替

printf "%s and %d \n"

执行脚本，输出结果如下所示：

1 abc

1 abc

abcdefabcdefabc

def

a b c

d e f

g h i

j

and 0

printf的转义序列

序列 说明

\a 警告字符，通常为ASCII的BEL字符

\b 后退

\c 抑制（不显示）输出结果中任何结尾的换行字符（只在%b格式指示符控制下的参数字符串中有效），而且，任何留在参数里的字符、任何接下来的参数以及任何留在格式字符串中的字符，都被忽略

\f 换页（formfeed）

\n 换行

\r 回车（Carriage return）

\t 水平制表符

\v 垂直制表符

\\ 一个字面上的反斜杠字符

\ddd 表示1到3位数八进制值的字符。仅在格式字符串中有效

\0ddd 表示1到3位的八进制值字符

**实例**

$ printf "a string, no processing:<%s>\n" "A\nB"

a string, no processing:<A\nB>

$ printf "a string, no processing:<%b>\n" "A\nB"

a string, no processing:<A

B>

$ printf "www.runoob.com \a"

www.runoob.com $ #不换行