

## 6. ?????? ??

我们使用 `echo` 命令来输出字符串。在终端中，我们可以输入以下命令：

```
$ echo Hello World
Hello World
$ echo "Hello World"
Hello World
```

但是，如果我们输入 `echo Hello "World"`，我们会得到以下输出：

```
$ echo "Hello \"World\""
```

这是因为 `echo` 命令默认使用双引号来包裹字符串。如果我们希望使用单引号来包裹字符串，我们可以使用 `echo` 命令的 `-n` 选项：

```
$ echo "Hello " World ""
```

或者，我们可以使用 `printf` 命令来输出字符串：

- `"Hello "`
- `World`
- `""`

但是，如果我们输入 `echo Hello World`，我们会得到以下输出：

```
Hello World
```

这是因为 `echo` 命令默认使用双引号来包裹字符串。如果我们希望使用单引号来包裹字符串，我们可以使用 `echo` 命令的 `-n` 选项：

或者，我们可以使用 `printf` 命令来输出字符串：

```
$ echo "Hello "World""
```

但是，如果我们输入 `echo "Hello "World""`，我们会得到以下输出：

这是因为 `echo` 命令默认使用双引号来包裹字符串。如果我们希望使用单引号来包裹字符串，我们可以使用 `echo` 命令的 `-n` 选项：

```
$ echo *
case.shtml escape.shtml first.shtml
functions.shtml hints.shtml index.shtml
ip-primer.txt raid1+0.txt
$ echo *txt
ip-primer.txt raid1+0.txt
$ echo "*"
*
$ echo "*txt"
*txt
```

\* 是通配符，匹配任意文件。
   
 在 shell 中，\* 是通配符，匹配任意文件。
   
 在 shell 中，\* 是通配符，匹配任意文件。

" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。
   
 在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。
   
 在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。

A quote is ", backslash is \, backtick is `.  
 A few spaces are and dollar is \$. \$X is 5.

在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。

```
$ echo "A quote is \", backslash is \\, backtick is \`."
A quote is ", backslash is \, backtick is `.
$ echo "A few spaces are    ; dollar is \$. \ $X is ${X}."
A few spaces are    ; dollar is $. $X is 5.
```

" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。
   
 在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。
   
 在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。

```
$ echo "This is \ a backslash"
This is \ a backslash
$ echo "This is \" a quote and this is \\ a backslash"
This is " a quote and this is \ a backslash
```

在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。
   
 在 shell 中，" 是双引号，\$ 是美元符号，\ 是反斜杠，` 是反引号。

Revision #1

Created 17 January 2024 04:13:14 by 肉肉 (MeatDumpling)

Updated 17 January 2024 04:13:28 by 肉肉 (MeatDumpling)