

# 7. 循环

循环是编程中非常基础且重要的概念。在 C 语言中，主要有两种循环：for 循环和 while 循环。for 循环适用于已知循环次数的情况，而 while 循环适用于未知循环次数的情况。

## For 循环

"for" 循环的语法如下：

```
#!/bin/sh
for i in 1 2 3 4 5
do
    echo "Looping ... number $i"
done
```

在上面的例子中，for 循环遍历了 1 到 5 的数字，并打印了每个数字。do 和 done 是循环的结束标志。

```
#!/bin/sh
for i in hello 1 * 2 goodbye
do
    echo "Looping ... i is set to $i"
done
```

在上面的例子中，for 循环遍历了 hello、1、\*、2 和 goodbye。需要注意的是，\* 在 shell 中是一个通配符，会匹配当前目录下的所有文件。因此，上面的例子可能会输出很多内容。为了避免这种情况，可以使用 \\* 来转义 \*。

在上面的例子中，for 循环遍历了 hello、1、\\*、2 和 goodbye。输出结果如下：

```
Looping .... number 1
Looping .... number 2
Looping .... number 3
Looping .... number 4
Looping .... number 5
```

以上就是 for 循环的基本用法。



Looping ... i is set to goodbye

, .

While ☐ ☐

```
"while"  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
1  2  3  4  5  6  7  8  9  10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
... )
```

```
#!/bin/sh

INPUT_STRING=hello

while [ "$INPUT_STRING" != "bye" ]
do
    echo "Please type something in (bye to quit)"
    read INPUT_STRING
    echo "You typed: $INPUT_STRING"
done
```

```
00000000  00 00 00 00 00 00 "bye" 00 00 00 00 00 00 00 00  
00000008  00 00 . 0000 00 '0' - 1 '(4)' 00 00 INPUT_STRING=hello 00 00  
00000010  0000 . 00 00 00 00 00 00 00 00 00 00 .
```

1. 凡在本市行政区域内从事经营活动的个体工商户，其经营场所应当符合下列条件：
   
 (一) 经营场所应当符合城乡规划、土地利用总体规划、环境保护、消防安全、卫生防疫、治安管理等法律法规的要求；
   
 (二) 经营场所应当符合所在地人民政府或者街道办事处、乡镇人民政府的要求；
   
 (三) 经营场所应当符合所在地村民委员会、居民委员会的要求；
   
 (四) 经营场所应当符合所在地市场监督管理部门的要求；
   
 (五) 经营场所应当符合所在地其他有关部门的要求。

```
#!/bin/sh

while :
do
    echo "Please type something in (^C to quit)"
    read INPUT_STRING
    echo "You typed: $INPUT_STRING"
done
```



我们使用 `while read` 循环来读取文件内容。在 `case` 语句中，我们根据输入文本匹配不同的语言并输出相应的问候语。如果输入文本与任何已知语言不匹配，则输出 "Unknown Language"。

```
#!/bin/sh
while read input_text
do
  case $input_text in
    hello)      echo English ;;
    howdy)     echo American ;;
    gday)      echo Australian ;;
    bonjour)   echo French ;;
    "guten tag") echo German ;;
    *)         echo Unknown Language: $input_text ;;
  esac
done < myfile.txt
```

```
#!/bin/sh
while read input_text
do
  case $input_text in
    hello)      echo English ;;
    howdy)     echo American ;;
    gday)      echo Australian ;;
    bonjour)   echo French ;;
    "guten tag") echo German ;;
    *)         echo Unknown Language: $input_text ;;
  esac
done < myfile.txt
```

"myfile.txt" 文件内容如下：

```
this file is called myfile.txt. It is an example text file.
hello
gday
bonjour
hola
```

我们使用 `cat` 命令来查看文件内容：

```
Unknown Language: this file is called myfile.txt. It is an example text file.
English
Australian
French
Unknown Language: hola
```

我们使用 `mkdir` 命令来创建目录。在 Bash 中，我们可以使用以下命令：

```
mkdir rc{0,1,2,3,4,5,6,S}.d
```



