

Control Flow

2022年6月9日 13:56

control flow

foreach + continue

```
set list_1 {3 2 1}
foreach i $list_1 {
  if {$i==2} {
    continue
  }
  puts $i
}
3
1
```

while:

```
set i 3
while {$i > 0} {
  puts $i
```

```
  incr i -1 #set i [expr $i - 1]
```

```
}
```

for:

```
for {i 3} {$i > 0} {incr i -1} {
  puts $i
}
```

过程函数

```
proc sample {x y} {
  set a [expr $x + $y]
  return $a
}
sample 3 4
7
```

global 在过程中引用全局变量

```
set a 1
```

```
proc sample {x} {  
  global a  
  incr a -1  
  return [expr $a + $x]  
}  
sample 3  
3
```

正则表达

\w 表示字母、数字、下划线

\d 表示数字

abc123

\w\w\w\d\d\d

简写

\d*\w*\d* *满足 ≥ 0 次匹配, 允许零次匹配

\d+\w+\d+ +满足 ≥ 1 次匹配, 至少有一次匹配

? 满足0/1次匹配

锚位

^ 字符串开头

\$ 字符串末尾

123abc123

^\d\d\d 取字符串开头前三个数字

\d\d\d\$ 取字符串后头三个数字

123 abc 123

\s 表示空格

正则表达式: \d+\s\w+\s\d+

. 表示任意一个字符

.\s+.\s+

匹配一个数字开头, 数字结尾的字符串

regexp {\^d.*d\$} "2 dsk 3"

捕获字符串

()

regexp {\^(\w+)\s\w+\s(\d+).*\$} "snowman is 30 years old" total names age

puts \$total

snowman is 30 years old

```
puts $names
snowman
```

```
puts $age
30
```

文本处理

```
open 打开文件 r只读模式 w
gets
close 关闭文件
```

```
set inputfile [open abc.txt r]
while {[gets $inputfile line]} >= 0 {
```

```
# 读取inputfile对应的abc.txt的下一行，把该行赋值给line，并返回该行的字符数（文件尾输出-1）
puts "$line"
}
```

```
close $inputfile
```

```
set outputfile [open file.txt w]
puts $outputfile "Hello World"
```

```
close $outputfile
```

```
file.txt
```

```
...
Slack = -0.051
Slack = -0.234
Slack = -0.311
Slack = -0.056
...
```

Example:

```
set sum 0
set readfile [open file.txt r]
while {[gets $readfile rows] >= 0} {
if {[regexp {^Slack\s+=\s+(-?\d+\.\d+)} $rows total slacknum]} { #return 1 if
matched
set sum [expr $sum + $slacknum]

}
}
close $readfile
puts $sum
```

all built-in commands