迭代

Python Reference

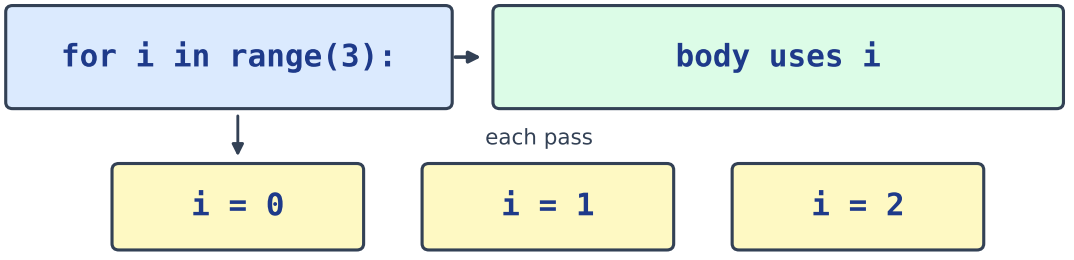
for 循环与 range

循环 (loop) 会重复运行代码。`for` 循环对序列 (sequence) 里的每个元素重复一次。`range(n)` 给出 `0` 到 `n - 1` 的数字。

```
for i in range(5):
    print(i)
# 0, then 1, 2, 3, 4 (each on its own line)
```

- `range(a, b)` 的范围是从 `a` 到（但不包括）`b`。
- `range(a, b, step)` 每次增加一个步长 (step)。

```
for n in range(2, 11, 2):
    print(n)        # 2 4 6 8 10
```



Exam tip: range(n) is 0 … n-1; the loop variable takes one value per pass

for i in range(n): 循环体以 $i = 0 \ldots n\text{-}1$ 运行

累加器模式

要在循环中逐步构建出一个结果，就在循环**之前**先建一个变量，然后每一轮更新 (update) 它。这就是累加器 (accumulator) 模式。

```
total = 0
for n in range(1, 6):
    total = total + n
print(total)        # 15
```

- 同样的方法可以数出有多少个元素符合条件。

```
count = 0
for letter in "banana":
    if letter == "a":
        count = count + 1
print(count)        # 3
```

while 循环

`while` 循环**只要**条件保持为真就一直重复。要在里面改变某些东西，否则它永不停止 — 这叫无限循环 (infinite loop)。

```
n = 1
while n <= 3:
    print(n)
    n = n + 1
# 1 2 3
```

- `break` 会立刻离开循环。

```
total = 0
while True:
    total = total + 10
    if total >= 30:
        break
print(total)        # 30
```

嵌套循环

一个循环放在另一个循环里面, 就是嵌套循环 (nested loop)。外层循环 (outer loop) 每转一轮, 内层循环 (inner loop) 都会完整运行一遍。

```
for row in range(3):
    line = ""
    for col in range(3):
        line = line + "*"
    print(line)
# ***
# ***
# ***
```

常见错误

- `range(n)` 从 `0` 到 `n - 1`, 不是 `1` 到 `n`— 经典的差一 (off-by-one) 错误。
- 一边遍历列表一边删元素会漏掉元素; 要删就遍历一个副本。
- `while` 里忘了改循环变量, 就会无限循环。
- 缩进决定哪些语句在循环内; 缩进错的那行只会在循环结束后运行一次。