
让程序按规则选择、重复与停止

第 3 章 | 条件语句、循环语句、跳转语句与综合实训

Python 零基础培训

100 分钟课堂

流程控制：让程序不只会从上到下执行

条件决定走哪条路，循环负责重复， 跳转语句决定何时提前停止或跳过。

- ▶ 用 `if/elif/else` 把业务规则写成分支
- ▶ 用 `while/for/range` 完成可控重复
- ▶ 用 `break/continue` 改变循环节奏
- ▶ 完成会员、物流、登录三个实训

先从一个布尔条件开始

```
score = 58  
print(score >= 60)  # False
```

条件表达式的结果只有 **True** 或 **False**。流程控制先看条件，再决定执行哪个代码块。

if: 条件成立才进入缩进代码块

```
if score >= 60:  
    print(" 考试及格")  
print(" 本次评估结束")
```

- ▶ 英文冒号表示分支开始
- ▶ 缩进决定代码块边界
- ▶ 条件不成立时，直接跳过缩进代码块

if-else：两种结果都要给出反馈

```
if score >= 60:  
    result = " 及格 "  
else:  
    result = " 不及格 "
```

适用场景：通过/不通过、库存充足/不足、输入合法/不合法。

if-elif-else：多分支必须从高阈值向下写

任一分支命中后即结束判断，所以不能先写 `score >= 60`。

判断顺序	命中条件	结果
1	<code>score >= 85</code>	优秀
2	<code>score >= 75</code>	良好
3	<code>score >= 60</code>	中等
4	其他	差

边界样例比正常样例更重要

```
for score in [59, 60, 74, 75, 84, 85]:  
    print(score, grade(score))
```

测试每个阈值的前一位、阈值本身和后一位，才能发现 $>$ 与 $\geq$ 写反的问题。

条件嵌套： 只在需要时再检查第二层

```
if month == 2:  
    if is_leap_year:  
        days = 29  
    else:  
        days = 28
```

嵌套适合有先后依赖的规则。 不是所有多个条件都应该嵌套。

先写规则，再写条件

1. 列出所有可能情况，确认是否互斥、是否覆盖完整；
2. 从最具体或最高阈值的规则开始；
3. 统一边界符号： $\geq$ 是否包含阈值；
4. 为非法输入保留明确反馈；
5. 用边界样例验证每个分支。

while：不知道次数，但知道停止条件

```
current = 1
total = 0
while current <= 10:
    total += current
    current += 1
```

循环三要素：初值、条件、更新。 少了更新，循环可能无法停止。

for: 逐个访问; range: 明确次数

```
for letter in "Python":  
    print(letter)  
for number in range(2, 6):  
    print(number)    # 2, 3, 4, 5
```

range 的右边界不包含在结果中。

while 和 for 如何选择？

问题特征	更适合	示例
停止条件明确，次数未知	while	输入直到合法
遍历已有对象 重复固定次数	for for range	逐个处理名单 最多尝试 3 次

循环嵌套：外层控制行，内层控制列

```
for row in range(3):  
    line = []  
    for col in range(4):  
        line.append("*")  
    print(" ".join(line))
```

棋盘、乘法表、二维表格都可按行和列拆成两层循环。

break 和 continue 改变循环节奏

break

```
for x in values:  
    if x == target:  
        break
```

结束当前循环。

continue

```
for x in values:  
    if x < 0:  
        continue
```

跳过当前这一轮。

嵌套循环中的 break 只结束最内层循环。

实训一：会员等级评定

从高等级向下判断；每级金额和积分都必须满足。

等级	消费金额	积分
钻石	≥ 1000	≥ 10000
白金	≥ 500	≥ 5000
黄金	≥ 200	≥ 2000
白银	≥ 100	≥ 1000
青铜	≥ 50	≥ 500

实训二： 物流费用计算

```
price_table = {  
    "01": (13, 3),    # 首重价、续重价  
    "02": (12, 2),  
    "03": (14, 4),  
}
```

- ▶ 先检查地区编号和重量是否合法
- ▶ 不超过 2 kg：只收首重价
- ▶ 超过部分向上按整 kg 计费

登录检测：最多三次机会

```
for attempt in range(3):  
    if user_ok and password_ok:  
        print(" 登录成功")  
        break  
    print(" 登录失败")  
else:  
    print(" 输入错误次数过多")
```

循环的 else：循环没有被 break 打断时才执行。

拓展：房贷估算器不要求手算公式

- ▶ 条件分支：贷款类型和期限决定年利率；
- ▶ 函数封装：将选利率、算月供、算总额拆开；
- ▶ 输入校验：金额和期限都必须大于 0；
- ▶ 课堂重点仍是流程控制，公式直接调用即可。

离场检验：写之前先说清楚

- ▶ 条件是否覆盖完整？多阈值为什么要先判断高阈值？
- ▶ 循环的初值、条件、更新分别是什么？
- ▶ `range(2, 6)` 会输出哪些数？
- ▶ `break` 与 `continue` 分别发生什么？
- ▶ 你测试过哪些边界样例？