## 1.py 条件语句

条件语句是通过一条或多条语句的执行结果（True 或者 False）来决定执行的代码块

if 判断条件：
  
 执行语句...

if 判断条件：
  
 执行语句...
  
else：
  
 执行语句...

if 判断条件1:
  
 执行语句1...
  
elif 判断条件2:
  
 执行语句2...
  
elif 判断条件3:
  
 执行语句3...
  
else:
  
 执行语句4...

## 2.py while 循环语句

while 语句用于循环执行程序，即在某条件下，循环执行某段程序，以处理需要重复处理的相同任务

while 判断条件(condition)：
  
 执行语句(statements)...

### 2.1 continue 跳过该次循环

i = 1
  
while i < 10:
  
 i += 1
  
 if i%2 > 0: # 非双数时跳过输出
  
 continue
  
 print(i) # 输出双数2、4、6、8、10

### 2.2 break 退出循环

i = 1
  
while 1: # 循环条件为1必定成立
  
 print(i) # 输出1~10
  
 i += 1
  
 if i > 10: # 当i大于10时跳出循环
  
 break

### 2.3 无限循环

var = 1
  
while var == 1 : # 该条件永远为true，循环将无限执行下去
  
 print("雪凤小仙女")

### 2.4 循环使用 else 语句

count = 0
  
while count < 5:
  
 print(count), # " 小于 5"
  
 count = count + 1
  
else:
  
 print(count), # " 大于 5"

## 3.py for 循环语句

for 循环可以遍历任何序列的项目，如一个列表或者一个字符串。

for iterating\_var in sequence:
  
 statements(s)

for letter in 'Python':
  
 print("当前字母: %s" % letter)
  
  
# 当前字母: P
  
# 当前字母: y
  
# 当前字母: t
  
# 当前字母: h
  
# 当前字母: o
  
# 当前字母: n

fruits = ['banana', 'apple', 'mango']
  
for fruit in fruits:
  
 print('当前水果: %s'% fruit)
  
  
# 当前水果: banana
  
# 当前水果: apple
  
# 当前水果: mango

### 3.1 for 使用 else 语句

for num in range(0,5):
  
 print(num), # " 小于 5"
  
else:
  
 print(" 大于 5")

## 4.py 循环嵌套

for iterating\_var in sequence:
  
 for iterating\_var in sequence:
  
 statements(s)
  
 statements(s)

while expression:
  
 while expression:
  
 statement(s)
  
 statement(s)

for iterating\_var in sequence:
  
 while expression:
  
 statement(s)
  
 statements(s)

while expression:
  
 for iterating\_var in sequence:
  
 statements(s)
  
 statement(s)

## 5.py pass 语句

pass 是空语句，是为了保持程序结构的完整性。pass 不做任何事情，一般用做占位语句。