# while循环语句

## 课程目标

* 能够使用while结构完成循环语句的书写
* 能够使用while循环完成计算1到100的累加求和案例
* 能够使用break和continue语句控制循环结构的执行顺序
* 能够使用while循环嵌套完成九九乘法表的输出

## 课程内容

* 循环语句
* 循环嵌套

### while循环语句

#### 1.1循环概念

循环是程序设计语言中反复执行某些代码的一种计算机处理过程

#### 1.2 while循环的作用

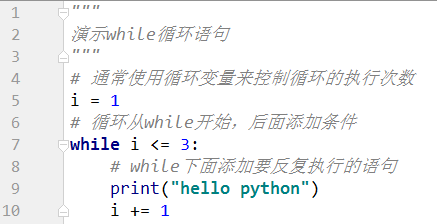
重复执行某些代码

#### 1.3 while循环的基本语法

**while** 条件：

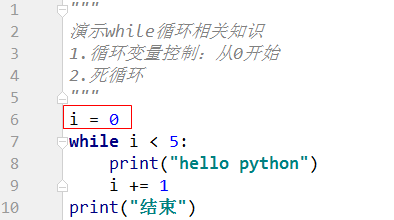
\_\_\_\_反复执行的代码

……



#### 1.4 循环变量

循环变量是用于控制循环执行次数的变量，通常从0开始计数。



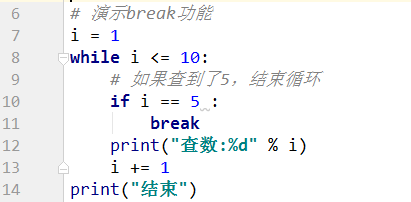
#### 1.5 死循环

如果循环语句未设置合理的退出条件，则循环语句将构成死循环，程序将永久执行下去。

### 循环的终止与跳出

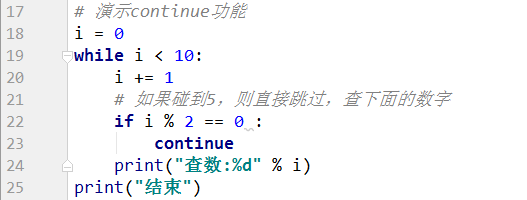
#### 2.1break

break可以用于终止循环的执行，当执行到break语句后，程序将跳出循环，执行循环语句后面的代码



#### 2.2continue

continue可以用于提前结束本轮循环的执行，当执行到continue语句后，程序将跳到判定循环是否向下执行的条件处，进行下一轮循环是否执行的判定与后续代码执行



#### 2.3注意事项

break与continue必须出现在循环体中

### 循环嵌套

#### 3.1循环嵌套概念

while循环中重复执行的代码又包含了另一个while循环。

#### 3.2 循环嵌套的基本语法

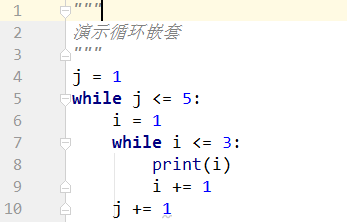
**while** 条件：

\_\_\_\_**while** 条件：

\_\_\_\_\_\_\_\_反复执行的代码

……

……



#### 3.3注意事项

break与continue仅能控制其所在的循环结构，不能控制其他循环结构

### 案例：数字累加求和

"""  
案例：数字累加求和  
要求：求100以内的所有数字和  
"""  
# 分析  
# 1.100以内的数字是1到100  
# 2.求和要有一个变量来保存最终的计算结果 sum  
# 3.使用while循环来构造反复加的操作  
# 定义一个变量用于描述1到100  
num = 1  
sum = 0  
while num <= 100:  
 sum = sum + num  
 # 让num值每次变化增大1  
 num += 1  
print("1到100的和是：%d" % sum)

### 案例：回文数

"""  
案例：回文数  
要求：打印所有3位回文数  
回文数：如果一个数字从左侧读和从右侧读是同一个数，则该数字即为。例如121,777  
"""  
# 分析  
# 1. 3位回文数从100到999  
# 2.回文数特征，百位数字和个位的数字相同  
# 3.打印满足特征的数字  
  
# # 求一个数字的百位数字  
# x = 789  
# print(x // 100)  
# # 求一个数字的个位数字  
# print(x % 10)  
  
i = 100  
while i <= 999 :  
 # 取出数字的百位和个位进行比较，如果相同就打印  
 a = i // 100  
 b = i % 10  
 # 判断a和b是否相同，如果相同打印  
 if a == b :  
 print(i)  
 i += 1

### 案例：九九乘法表

"""  
演示九九乘法表案例  
"""  
# 分析  
# 1.结构：循环嵌套（外层，9次，内层，依次增加1-9）  
# 2.打印内容： X \* Y = Z "%d\*%d=%d" % (X,Y,Z)  
# 3.打印对其：使用指标位符号\t  
  
# 外层循环  
j = 1  
while j <= 9:  
 #内层循环  
 i = 1  
 while i <= j:  
 print("%d\*%d=%d" % (i , j , i\*j ),end="\t")  
 i += 1  
 #外层循环折行  
 print()  
 j += 1