

python中break,continue,pass,else的用法...

python中有两个循环保留字：**break**和**continue**。

作用：辅助的控制程序的循环执行。

break语句和continue语句的区别：break语句是结束整个循环的过程，不在判断执行循环的条件是否成立；continue语句是只结束本次循环，并不终止整个循环的执行。

1. break

作用：用来跳出最内层的for循环或者while循环，脱离该循环后程序从循环代码后面继续执行。即break语句只能跳出当前层次的循环。

例子：

```
for i in "python":
    for j in range(5):
        print(i, end="")
        if i == "t":
            break
#程序执行结果为：pppppyyyt hhhhhooooonnnn
```

这个例子说明了，break语句跳出了最内层的for循环，但还可以执行外层循环。

2. continue

作用：结束当前当次循环，即跳出循环体中还没有执行的语句，但是并不跳出当前循环。

例子：

```
for i in "python":
    if i == "t":
        continue
    print(i, end="")
#程序执行结果为：pyhon
```

```
for i in "python":
    if i == "t":
        break
    print(i, end="")
#程序执行结果为：py
```

3. pass

作用：不做任何作用，只起到占位的作用。循环中使用 pass 不会跳出循环

例子：
for i in "python":
 if i == "t":
 pass
 print(i, end="")
#程序执行结果为：python

4. for循环和while循环中的else扩展用法

说明：else中的程序只在一种条件下执行，即循环正常遍历所有内容或者由于条件不成立而结束循环，没有因break或者return而退出循环。continue对else没有影响。

例子：
for i in "python":
 if i == "t":
 continue
 print(i, end="")
else:
 print("程序正常退出")
#程序执行结果为: pyhon程序正常退出

for i in "python":
 if i == "t":
 break
 print(i, end="")
else:
 print("程序正常退出")
#程序执行结果为: py

注意事项：一定要牢记break语句和continue语句的区别