

## 4. break 和 continue

通过对循环结构的学习，我们已经了解了在执行循环时要进行条件判断。只有在条件为 **False** 或遍历表达式失败时，才能结束循环。但是，有时根据实际情况需要停止整个循环或者跳到下一次循环，这些都可以由跳转语句来完成。接下来我们学习 **Python** 中的跳转语句：**break**、**continue**。

### (1) break 语句

**break** 语句用于终止某个循环，使程序跳到循环体外。在循环中位于 **break** 后的语句将不再执行，循环也全部停止执行。举个例子感受一下。

**示例：**输出 1~10 这是十个数字，如遇到数字 5 则程序自动退出。

输出 1~10 这是十个数字的关键代码如下：

```
for i in range(1, 11):  
    print(i, end = ' ')
```

如遇到数字 5 则程序自动退出，该怎么实现呢，这里就用到 **break** 语句了：

```
for i in range(1, 11):  
    if i == 5 :  
        break  
    print(i, end = ' ')
```

输出结果就是：1 2 3 4

判断数字是否为 5 后，使用 **break** 语句，整个 **for** 循环退出，也就是后面的数字都不再输出了。

### 小贴士

- **break** 语句不仅可以用在 **for** 循环结构中，也可以用在 **while** 循环结构中。
- **break** 语句通常与 **if** 条件语句一起使用。

### (2) continue 语句

根据要求，在循环结构中的 **if** 语句中使用 **break** 语句退出循环。但是，可能有时会遇到这样的问题：在某次循环中，不执行完所有的循环体，就跳出本次循环，开始执行下一次循环。我们看下面这个示例。

**示例：**输出 1~10 这是十个数字，如遇到数字 5 则不输出该数字。

```
for i in range(1,11):  
    if i==5 :  
        continue  
    print(i,end=' ')
```

输出结果就是：1 2 3 4 6 7 8 9 10

`continue` 语句可以用于 `for` 循环结构，也可以用于 `while` 循环结构。在 `for` 循环结构中，`continue` 语句使程序先跳转到迭代部分，然后判断循环条件。如果为 `true`，则继续下一次循环；否则终止循环。在 `while` 循环结构中，`continue` 语句执行完毕后，程序将直接判断循环条件。

#### 小贴士

在循环结构中：

- `break` 语句用于终止某个循环，程序跳转到循环体外的下一条语句。
- `continue` 语句用于跳出本次循环，进入下一次循环的语句。