

```
import itertools
```

导入itertools模块

itertools模块提供了一些迭代器

```
counter = itertools.count(xxx, xxx)
```

生成无限个元素的迭代器，

第一个参数为起始位置

第二个参数为步长，默认步长为1

该迭代器会一直生成，可以用for循环输出

```
for i in counter:
```

```
    print(i)
```

或者取他的有限个数

```
for i in range(1, 10):
```

```
    print(next(counter))
```

这样可以取出这个迭代器的有限个数字

```
b = list(itertools.islice(counter, 90, 100))
```

从无限个数的迭代器中生成有限个序列

第一个参数是迭代器对象，

第二个与第三个参数是从第几个到第几个

注意：是从迭代器中的第几个到第几个，而不是从一个元素到另一个元素

```
cycle_xxx = itertools.cycle(xxx)
```

生成一个周而复始的无限循环的迭代器

这个函数接受一个可迭代对象，然后无限循环对象中的元素

可以用for循环将结果输出

```
for i in cycle_xxx:
```

```
    print(i)
```

同理也可以输出其中有限个数

```
for i in range(4,10):
```

```
    print(next(cycle_xxx))
```

即可将其中有限个数字输出

```
from itertools import repeat
```

```
#重复第一个参数的第二个参数次
```

```
r = repeat('ten', 8)
```

```
list(r)
```

既可以生成八个ten字符串的列表

# chain 函数 相当于连接两个可迭代的对象

```
from itertools import chain
```

```
l1 = ['ten', 'gavin']
```

```
l2 = ['anny', 'aaron']
```

```
l3 = list(chain(l1, l2))
```

```
print(l3)
```

```
for i in chain(l1, l2):
```

```
    print(i)
```