

迭代器 and 生成器

迭代器是python中的一个集合元素的一种方式，并且迭代器对象从集合的一个元素开始访问，直到所有的元素访问后就会结束，而迭代器不会后退，只能向下进行。

主要函数有 `iter()` and `next()`

创建与输出迭代器

```
# -----+
# 新建了列表 and 迭代器 /
# -----+
var = [1,2,3,4,5,6,7,8]
iter = iter(var)

# -----+
# 输出迭代器 /
# -----+
print('& ',next(iter))
print('& ',next(iter))
print('& ',next(iter))
print('& ',next(iter))
```

使用for循环输出迭代器

```
var = [1,2,3,4,5,6,7]

hor = iter(var)

for a in hor:
    print('[~]:',a ,end=" ")
```

输出结果

```
[~]: 1 [~]: 2 [~]: 3 [~]: 4 [~]: 5 [~]: 6 [~]: 7
```

生成器

使用yield的函数被成为生成器 (generator)

生成器是一个返回迭代器的函数只能用于迭代操作，而生成器也可以理解为一个迭代器。在调用生成器运行的过程中，每次遇到yield时函数会暂停并保存当前运行的信息，返回yield的值并在下次执行next()方法时从当前位置继续运行。

```
import sys

def fibo(n):                                # 生成斐波那契
    a, b, counter = 0, 1, 0
    while True:
        if (counter > n):
            return
        yield a
        a, b = b, a + b
        counter += 1
f = fibo(10)                                # 迭代器

while True:
    try:
        print(next(f),end=" ")
    except StopIteration:
        sys.exit()
```

输出结果

```
0 1 1 2 3 5 8 13 21 34 55
```