

# python

李晨辉

2019 年 5 月 5 日

## 1. 生成器与迭代器的区别

- a. 有`iter()`和`next()`魔法方法的对象，都是迭代器(可以为你的类添加迭代器行为)；
- b. 生成器是一个用于创建迭代器的工具，它们的写法类似标准的函数，但当它们要返回数据时会使用`yield`语句。每次对生成器调用`next()`时，它会从上次离开位置恢复执行
- c. 用生成器来完成的操作同样可以用基于类的迭代器来完成，但生成器的写法更为紧凑，因为它会自动创建`iter()`和`next()`方法。
- d. 局部变量和执行状态会在每次调用之间自动保存，当生成器终结时，它们还会自动引发`StopIteration`。

## 2. Python中，函数名为什么可以当作参数用？

- python中函数是第一等对象，第一等对象的一般特征：
  - a. 运行时（runtime）创建
  - b. 将变量或者元素赋值在一个数据结构当中
  - c. 可以作为一个参数传递给一个函数
  - d. 可以作为函数的结果返回
- python 中一切皆对象，`int`是对象，函数是对象，`class`也是一种对象，跟其它对象一样是最终继承自`PyObject`，函数可以像任何对象那样进行赋值、传递、名字重绑定、赋值、装进容器、垃圾回收

如何追女孩的建议		
建议 \ 颜值	帅	挫
财气		
有钱花	没啥好说的	没啥好说的
叮当响	没啥好说的	没啥好说的

Table 1: 装模做样的表格

这是一个表格

你好， $\text{\LaTeX}$

**Data:** this text

**Result:** how to write algorithm with L<sup>A</sup>T<sub>E</sub>X2e  
initialization;

**for** *not at end of this document* **do**

    read current;

**if** *understand* **then**

        go to next section;

        current section becomes this one;

**else**

        go back to the beginning of current section;

**end**

**end**

**Algorithm 1:** How to write algorithms