

python 迭代器与分发器

可迭代对象



- 如果一个对象拥有 `__iter__` 方法，其是可迭代对象
- 如果一个对象拥有 `next` 方法，其是迭代器

1. `__iter__()`

- 返回当前对象的迭代器类的实例

2. `next()`

- 返回迭代的每一步 超届跑出 `stopIteration` 异常

迭代器 `iterator`

- 实现了迭代器协议的对象
- 内置数据类型（列表，字符串，字典等）可以通过 `for` 语句迭代
- 迭代末尾，引发 `stopIteration` 异常

生成器 `generator`

- 通过 `yield` 语句快速生成迭代器
- `yield` 可以让一个普通函数变成一个生成器
- 自动实现迭代器协议