

《Java与模式》学习笔记：设计模式迭代子(Iterator)模式 PDF  
转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/264/2021\\_2022\\_\\_E3\\_80\\_8AJava\\_E4\\_B8\\_8E\\_c104\\_264313.htm](https://www.100test.com/kao_ti2020/264/2021_2022__E3_80_8AJava_E4_B8_8E_c104_264313.htm)

迭代子(Iterator)模式迭代子模式又叫游标模式，是对象的行为模式。1.定义：提供遍历一个聚集的方法接口，而不需要了解聚集的内部结构(说白了就是为不同的聚集提供相同的遍历方式，使得客户端只需要关注迭代接口，而不需要关注所迭代的聚集的具体结构。)2.为什么需要Iterator 迭代子相当于在客户端和聚集之间加了一层，这样一来，迭代子就可以缓冲客户端或聚集的变化。使得聚集的变化对客户端透明(由于客户端是对迭代子编程的，只要迭代子的接口不变，聚集怎么变都不会影响到客户端)，同样的客户端的需求变化对聚集也是透明的(客户端需求变了，只需要修改迭代子的接口和实现，不需要修改聚集的结构)。这样就将可变的封装到了迭代子中，是符合"开-闭"原则的。3.内禀子，外禀子1)定义：内禀子:聚集本身不提供访问其内部元素的方法，只有通过聚集内部的迭代子来遍历聚集，这时迭代子是个内部类，是聚集的一部分。外禀子:聚集本身提供访问其内部元素的方法，可以通过外部的迭代子来遍历聚集，这时迭代子是个外部类，只维持对聚集的一个引用。2)安全性区别：内禀子:安全，因为聚集不提供访问其内部元素的方法，在迭代过程中，聚集的元素不会被客户端绕过迭代子修改，所以不会导致迭带过程崩溃。外禀子:不安全，因为聚集提供访问其内部元素的方法，在迭代过程中，聚集的元素会被客户端绕过迭代子修改，所以可能会导致迭带过程崩溃。3)何时使用外禀子：当聚集会被几个不同的方法和

对象同时共享和控制，需要这个聚集的多个迭代子分别维护各自的游标时，使用外稟子4)何时使用内稟子：当对安全方面的要求比较高，而且又不需要共享时。

4.一个好的Iterator应该注意的方面

1)主动迭代子和被动迭代子：主动迭代子：客户端用next()推动迭代过程被动迭代子：迭代子自行推进迭代过程

2)静态迭代子和动态迭代子(都是对外稟子来说的)静态迭代子：迭代子内部维持一份聚集的元素的克隆，不会出现迭代崩溃，但是对性能有影响。动态迭代子：迭代子内部只维持一份聚集的引用，可能出现迭代崩溃

3)Fail Fast如果当一个算法开始之后，他的运算环境发生了变化，使得算法无法进行必需的调整时，这个算法就应当立即发出故障信号

5.Iterator的优缺点

优点：迭代子简化了聚集的接口，因为迭代子提供了遍历接口，所以聚集就可以不用提供遍历接口。对于外稟子而言，一个聚集可以提供多个迭代子，而且这些迭代子的迭代状态都是相互独立的，这样一个聚集可以在多个方法中被同时遍历。

缺点：迭代子给客户端一个聚集被顺序化的错觉 迭代子给出的聚集元素没有类型特征，统一都是Object。

备注：在真正的开发中，不需要我们自己去实现Iterator模式，因为Java以及许多第三方库都已经提供了性能优秀的集合类，我们所需要做的就是将元素装进特定的集合类，然后用Java提供的Iterator机制去遍历聚集。就是如此简单。研究这个模式的目的是对Java的实现有更好的理解。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)