

Java集合框架

Iterator迭代器

提供一种方法访问一个容器对象中各个元素，但又不需要暴露该对象内部细节

安全

当前遍历集合元素被更改时，会抛出ConcurrentModificationException异常

线程不安全的集合

ArrayList

LinkedList

HashMap

HashSet

TreeSet

TreeMap

PriorityQueue

java.util.concurrent包提供了很多并发容器来使用线程安全的集合

快速失败(fail-fast)

错误检测机制

多线程下操作非安全失败(fail-safe)的集合类

安全失败机制

遍历集合时

1. 复制原有集合内容

2. 在拷贝集合上进行遍历

线程1对集合进行遍历

线程2对集合进行修改

线程1抛出ConcurrentModificationException异常

迭代器使用hasNext()/next()遍历下一个元素之前

检测modCount变量

与expectedModCount对比

是，返回遍历

否，终止遍历

触发fail-fast机制

单线程下对集合对象内容进行修改

不要在foreach循环中进行元素的remove/add操作

remove元素使用iterator方式：iterator.remove()

通过iterator修改集合会修改expectedModCount值，不会抛出异常

Arrays.asList()

将数组转换为集合

底层还是数组

传入对象必须是对象数组

集合修改方法：add(), remove(), clear()会抛出异常

返回并不是ArrayList，而是Arrays的一个内部类

这个类没有实现集合的修改方法