JavaScript is disabled on your browser.

[Skip navigation links](#1fob9te)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/LazyIteratorChain.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/apache/commons/collections4/iterators/IteratorIterable.html)
* [Next Class](http://docs.google.com/org/apache/commons/collections4/iterators/ListIteratorWrapper.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/collections4/iterators/LazyIteratorChain.html)
* [No Frames](http://docs.google.com/LazyIteratorChain.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)
* Detail:
* Field |
* [Constr](#1t3h5sf) |
* [Method](#2s8eyo1)

org.apache.commons.collections4.iterators

## Class LazyIteratorChain<E>

* [java.lang.Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)
  + org.apache.commons.collections4.iterators.LazyIteratorChain<E>
* All Implemented Interfaces: [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<E>  
    
  public abstract class LazyIteratorChain<E>  
  extends [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)  
  implements [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<E>  
  An LazyIteratorChain is an Iterator that wraps a number of Iterators in a lazy manner.  
  This class makes multiple iterators look like one to the caller. When any method from the Iterator interface is called, the LazyIteratorChain will delegate to a single underlying Iterator. The LazyIteratorChain will invoke the Iterators in sequence until all Iterators are exhausted.  
  The Iterators are provided by [nextIterator(int)](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#nextIterator-int-) which has to be overridden by sub-classes and allows to lazily create the Iterators as they are accessed:  
   return new LazyIteratorChain<String>() {  
   protected Iterator<String> nextIterator(int count) {  
   return count == 1 ? Arrays.asList("foo", "bar").iterator() : null;  
   }  
   };  
     
  Once the inner Iterator's [Iterator.hasNext()](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true#hasNext--) method returns false, [nextIterator(int)](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#nextIterator-int-) will be called to obtain another iterator, and so on until [nextIterator(int)](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#nextIterator-int-) returns null, indicating that the chain is exhausted.  
  NOTE: The LazyIteratorChain may contain no iterators. In this case the class will function as an empty iterator.Since: 4.0

### Constructor SummaryConstructors

|  |
| --- |
| * + Constructor and Description |
| * + [LazyIteratorChain](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#LazyIteratorChain--)() |

### Method SummaryAll Methods Instance Methods Abstract Methods Concrete Methods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + boolean | * + [hasNext](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#hasNext--)() Return true if any Iterator in the chain has a remaining element. |
| * + [E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html) | * + [next](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#next--)() Returns the next element of the current Iterator |
| * + protected abstract [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<? extends [E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html)> | * + [nextIterator](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#nextIterator-int-)(int count) Gets the next iterator after the previous one has been exhausted. |
| * + void | * + [remove](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html#remove--)() Removes from the underlying collection the last element returned by the Iterator. |

### Methods inherited from class java.lang.[**Object**](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)[clone](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#clone--), [equals](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#equals-java.lang.Object-), [finalize](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#finalize--), [getClass](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#getClass--), [hashCode](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#hashCode--), [notify](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#notify--), [notifyAll](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#notifyAll--), [toString](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#toString--), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait--), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait-long-), [wait](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true#wait-long-int-)

### Methods inherited from interface java.util.[**Iterator**](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)[forEachRemaining](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true#forEachRemaining-java.util.function.Consumer-)

### Constructor Detail

#### LazyIteratorChain public LazyIteratorChain()

### Method Detail

#### nextIterator protected abstract [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<? extends [E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html)> nextIterator(int count) Gets the next iterator after the previous one has been exhausted. This method **MUST** return null when there are no more iterators.Parameters: count - the number of time this method has been called (starts with 1) Returns: the next iterator, or null if there are no more.

#### hasNext public boolean hasNext() Return true if any Iterator in the chain has a remaining element.Specified by: [hasNext](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true#hasNext--) in interface [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<[E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html)> Returns: true if elements remain

#### next public [E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html) next() Returns the next element of the current IteratorSpecified by: [next](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true#next--) in interface [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<[E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html)> Returns: element from the current Iterator Throws: [NoSuchElementException](https://docs.oracle.com/javase/7/docs/api/java/util/NoSuchElementException.html?is-external=true) - if all the Iterators are exhausted

#### remove public void remove() Removes from the underlying collection the last element returned by the Iterator. As with next() and hasNext(), this method calls remove() on the underlying Iterator. Therefore, this method may throw an UnsupportedOperationException if the underlying Iterator does not support this method.Specified by: [remove](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true#remove--) in interface [Iterator](https://docs.oracle.com/javase/7/docs/api/java/util/Iterator.html?is-external=true)<[E](http://docs.google.com/org/apache/commons/collections4/iterators/LazyIteratorChain.html)> Throws: [UnsupportedOperationException](https://docs.oracle.com/javase/7/docs/api/java/lang/UnsupportedOperationException.html?is-external=true) - if the remove operator is not supported by the underlying Iterator [IllegalStateException](https://docs.oracle.com/javase/7/docs/api/java/lang/IllegalStateException.html?is-external=true) - if the next method has not yet been called, or the remove method has already been called after the last call to the next method.

[Skip navigation links](#44sinio)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/LazyIteratorChain.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-all.html)
* [Help](http://docs.google.com/help-doc.html)
* [Prev Class](http://docs.google.com/org/apache/commons/collections4/iterators/IteratorIterable.html)
* [Next Class](http://docs.google.com/org/apache/commons/collections4/iterators/ListIteratorWrapper.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/collections4/iterators/LazyIteratorChain.html)
* [No Frames](http://docs.google.com/LazyIteratorChain.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)
* Detail:
* Field |
* [Constr](#1t3h5sf) |
* [Method](#2s8eyo1)

Copyright © 2001–2019 [The Apache Software Foundation](https://www.apache.org/). All rights reserved.