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/ListUtils.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/KeyValue.html)
* [Next Class](http://docs.google.com/org/apache/commons/collections4/ListValuedMap.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/collections4/ListUtils.html)
* [No Frames](http://docs.google.com/ListUtils.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* Constr |
* [Method](#3znysh7)
* Detail:
* Field |
* Constr |
* [Method](#tyjcwt)

org.apache.commons.collections4

## Class ListUtils

* [java.lang.Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)
  + org.apache.commons.collections4.ListUtils
* public class ListUtils  
  extends [Object](https://docs.oracle.com/javase/7/docs/api/java/lang/Object.html?is-external=true)  
  Provides utility methods and decorators for [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true) instances.Since: 1.0

### Method SummaryAll Methods Static Methods Concrete Methods

|  |  |
| --- | --- |
| * + Modifier and Type | * + Method and Description |
| * + static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> | * + [defaultIfNull](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#defaultIfNull-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> defaultList) Returns either the passed in list, or if the list is null, the value of defaultList. |
| * + static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> | * + [emptyIfNull](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#emptyIfNull-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list) Returns an immutable empty list if the argument is null, or the argument itself otherwise. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [fixedSizeList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#fixedSizeList-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list) Returns a fixed-sized list backed by the given list. |
| * + static int | * + [hashCodeForList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#hashCodeForList-java.util.Collection-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list) Generates a hash code using the algorithm specified in [List.hashCode()](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#hashCode--). |
| * + static <E> int | * + [indexOf](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#indexOf-java.util.List-org.apache.commons.collections4.Predicate-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<E> predicate) Finds the first index in the given List which matches the given predicate. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [intersection](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#intersection-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns a new list containing all elements that are contained in both given lists. |
| * + static boolean | * + [isEqualList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#isEqualList-java.util.Collection-java.util.Collection-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list1, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list2) Tests two lists for value-equality as per the equality contract in [List.equals(java.lang.Object)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#equals-java.lang.Object-). |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [lazyList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#lazyList-java.util.List-org.apache.commons.collections4.Factory-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Factory](http://docs.google.com/org/apache/commons/collections4/Factory.html)<? extends E> factory) Returns a "lazy" list whose elements will be created on demand. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [lazyList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#lazyList-java.util.List-org.apache.commons.collections4.Transformer-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Transformer](http://docs.google.com/org/apache/commons/collections4/Transformer.html)<[Integer](https://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html?is-external=true),? extends E> transformer) Returns a "lazy" list whose elements will be created on demand. |
| * + static [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) | * + [longestCommonSubsequence](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#longestCommonSubsequence-java.lang.CharSequence-java.lang.CharSequence-)([CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) a, [CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) b) Returns the longest common subsequence (LCS) of two [CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) objects. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [longestCommonSubsequence](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#longestCommonSubsequence-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> a, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> b) Returns the longest common subsequence (LCS) of two sequences (lists). |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [longestCommonSubsequence](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#longestCommonSubsequence-java.util.List-java.util.List-org.apache.commons.collections4.Equator-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> a, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> b, [Equator](http://docs.google.com/org/apache/commons/collections4/Equator.html)<? super E> equator) Returns the longest common subsequence (LCS) of two sequences (lists). |
| * + static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<[List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T>> | * + [partition](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#partition-java.util.List-int-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list, int size) Returns consecutive [sublists](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#subList-int-int-) of a list, each of the same size (the final list may be smaller). |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [predicatedList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#predicatedList-java.util.List-org.apache.commons.collections4.Predicate-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<E> predicate) Returns a predicated (validating) list backed by the given list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [removeAll](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#removeAll-java.util.Collection-java.util.Collection-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<E> collection, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> remove) Removes the elements in remove from collection. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [retainAll](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#retainAll-java.util.Collection-java.util.Collection-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<E> collection, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> retain) Returns a List containing all the elements in collection that are also in retain. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [select](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#select-java.util.Collection-org.apache.commons.collections4.Predicate-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<? extends E> inputCollection, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<? super E> predicate) Selects all elements from input collection which match the given predicate into an output list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [selectRejected](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#selectRejected-java.util.Collection-org.apache.commons.collections4.Predicate-)([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<? extends E> inputCollection, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<? super E> predicate) Selects all elements from inputCollection which don't match the given predicate into an output collection. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [subtract](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#subtract-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Subtracts all elements in the second list from the first list, placing the results in a new list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [sum](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#sum-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns the sum of the given lists. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [synchronizedList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#synchronizedList-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list) Returns a synchronized list backed by the given list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [transformedList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#transformedList-java.util.List-org.apache.commons.collections4.Transformer-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Transformer](http://docs.google.com/org/apache/commons/collections4/Transformer.html)<? super E,? extends E> transformer) Returns a transformed list backed by the given list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [union](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#union-java.util.List-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns a new list containing the second list appended to the first list. |
| * + static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> | * + [unmodifiableList](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#unmodifiableList-java.util.List-)([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list) Returns an unmodifiable list backed by the given list. |

### 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-)

### Method Detail

#### emptyIfNull public static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> emptyIfNull([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list) Returns an immutable empty list if the argument is null, or the argument itself otherwise.Type Parameters: T - the element type Parameters: list - the list, possibly null Returns: an empty list if the argument is null

#### defaultIfNull public static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> defaultIfNull([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> defaultList) Returns either the passed in list, or if the list is null, the value of defaultList.Type Parameters: T - the element type Parameters: list - the list, possibly null defaultList - the returned values if list is null Returns: an empty list if the argument is null Since: 4.0

#### intersection public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> intersection([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns a new list containing all elements that are contained in both given lists.Type Parameters: E - the element type Parameters: list1 - the first list list2 - the second list Returns: the intersection of those two lists Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list is null

#### subtract public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> subtract([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Subtracts all elements in the second list from the first list, placing the results in a new list. This differs from [List.removeAll(Collection)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#removeAll-java.util.Collection-) in that cardinality is respected; if list1 contains two occurrences of null and list2 only contains one occurrence, then the returned list will still contain one occurrence.Type Parameters: E - the element type Parameters: list1 - the list to subtract from list2 - the list to subtract Returns: a new list containing the results Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list is null

#### sum public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> sum([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns the sum of the given lists. This is their intersection subtracted from their union.Type Parameters: E - the element type Parameters: list1 - the first list list2 - the second list Returns: a new list containing the sum of those lists Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list is null

#### union public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> union([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list1, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list2) Returns a new list containing the second list appended to the first list. The [List.addAll(Collection)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#addAll-java.util.Collection-) operation is used to append the two given lists into a new list.Type Parameters: E - the element type Parameters: list1 - the first list list2 - the second list Returns: a new list containing the union of those lists Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list is null

#### select public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> select([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<? extends E> inputCollection, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<? super E> predicate) Selects all elements from input collection which match the given predicate into an output list. A null predicate matches no elements.Type Parameters: E - the element type Parameters: inputCollection - the collection to get the input from, may not be null predicate - the predicate to use, may be null Returns: the elements matching the predicate (new list) Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the input list is null Since: 4.0 See Also: [CollectionUtils.select(Iterable, Predicate)](http://docs.google.com/org/apache/commons/collections4/CollectionUtils.html#select-java.lang.Iterable-org.apache.commons.collections4.Predicate-)

#### selectRejected public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> selectRejected([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<? extends E> inputCollection, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<? super E> predicate) Selects all elements from inputCollection which don't match the given predicate into an output collection. If the input predicate is null, the result is an empty list.Type Parameters: E - the element type Parameters: inputCollection - the collection to get the input from, may not be null predicate - the predicate to use, may be null Returns: the elements **not** matching the predicate (new list) Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the input collection is null Since: 4.0 See Also: [CollectionUtils.selectRejected(Iterable, Predicate)](http://docs.google.com/org/apache/commons/collections4/CollectionUtils.html#selectRejected-java.lang.Iterable-org.apache.commons.collections4.Predicate-)

#### isEqualList public static boolean isEqualList([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list1, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list2) Tests two lists for value-equality as per the equality contract in [List.equals(java.lang.Object)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#equals-java.lang.Object-). This method is useful for implementing List when you cannot extend AbstractList. The method takes Collection instances to enable other collection types to use the List implementation algorithm. The relevant text (slightly paraphrased as this is a static method) is:Compares the two list objects for equality. Returns true if and only if both lists have the same size, and all corresponding pairs of elements in the two lists are *equal*. (Two elements e1 and e2 are *equal* if (e1==null ? e2==null : e1.equals(e2)).) In other words, two lists are defined to be equal if they contain the same elements in the same order. This definition ensures that the equals method works properly across different implementations of the List interface.**Note:** The behaviour of this method is undefined if the lists are modified during the equals comparison.Parameters: list1 - the first list, may be null list2 - the second list, may be null Returns: whether the lists are equal by value comparison See Also: [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)

#### hashCodeForList public static int hashCodeForList([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> list) Generates a hash code using the algorithm specified in [List.hashCode()](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#hashCode--). This method is useful for implementing List when you cannot extend AbstractList. The method takes Collection instances to enable other collection types to use the List implementation algorithm.Parameters: list - the list to generate the hashCode for, may be null Returns: the hash code See Also: [List.hashCode()](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#hashCode--)

#### retainAll public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> retainAll([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<E> collection, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> retain) Returns a List containing all the elements in collection that are also in retain. The cardinality of an element e in the returned list is the same as the cardinality of e in collection unless retain does not contain e, in which case the cardinality is zero. This method is useful if you do not wish to modify the collection c and thus cannot call collection.retainAll(retain);. This implementation iterates over collection, checking each element in turn to see if it's contained in retain. If it's contained, it's added to the returned list. As a consequence, it is advised to use a collection type for retain that provides a fast (e.g. O(1)) implementation of [Collection.contains(Object)](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true#contains-java.lang.Object-).Type Parameters: E - the element type Parameters: collection - the collection whose contents are the target of the #retailAll operation retain - the collection containing the elements to be retained in the returned collection Returns: a List containing all the elements of c that occur at least once in retain. Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either parameter is null Since: 3.2

#### removeAll public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> removeAll([Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<E> collection, [Collection](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true)<?> remove) Removes the elements in remove from collection. That is, this method returns a list containing all the elements in collection that are not in remove. The cardinality of an element e in the returned collection is the same as the cardinality of e in collection unless remove contains e, in which case the cardinality is zero. This method is useful if you do not wish to modify collection and thus cannot call collection.removeAll(remove);. This implementation iterates over collection, checking each element in turn to see if it's contained in remove. If it's not contained, it's added to the returned list. As a consequence, it is advised to use a collection type for remove that provides a fast (e.g. O(1)) implementation of [Collection.contains(Object)](https://docs.oracle.com/javase/7/docs/api/java/util/Collection.html?is-external=true#contains-java.lang.Object-).Type Parameters: E - the element type Parameters: collection - the collection from which items are removed (in the returned collection) remove - the items to be removed from the returned collection Returns: a List containing all the elements of c except any elements that also occur in remove. Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either parameter is null Since: 3.2

#### synchronizedList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> synchronizedList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list) Returns a synchronized list backed by the given list. You must manually synchronize on the returned list's iterator to avoid non-deterministic behavior: List list = ListUtils.synchronizedList(myList); synchronized (list) { Iterator i = list.iterator(); while (i.hasNext()) { process (i.next()); } } This method is just a wrapper for [Collections.synchronizedList(List)](https://docs.oracle.com/javase/7/docs/api/java/util/Collections.html?is-external=true#synchronizedList-java.util.List-).Type Parameters: E - the element type Parameters: list - the list to synchronize, must not be null Returns: a synchronized list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the list is null

#### unmodifiableList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> unmodifiableList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<? extends E> list) Returns an unmodifiable list backed by the given list. This method uses the implementation in the decorators subpackage.Type Parameters: E - the element type Parameters: list - the list to make unmodifiable, must not be null Returns: an unmodifiable list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the list is null

#### predicatedList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> predicatedList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<E> predicate) Returns a predicated (validating) list backed by the given list. Only objects that pass the test in the given predicate can be added to the list. Trying to add an invalid object results in an IllegalArgumentException. It is important not to use the original list after invoking this method, as it is a backdoor for adding invalid objects.Type Parameters: E - the element type Parameters: list - the list to predicate, must not be null predicate - the predicate for the list, must not be null Returns: a predicated list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the List or Predicate is null

#### transformedList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> transformedList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Transformer](http://docs.google.com/org/apache/commons/collections4/Transformer.html)<? super E,? extends E> transformer) Returns a transformed list backed by the given list. This method returns a new list (decorating the specified list) that will transform any new entries added to it. Existing entries in the specified list will not be transformed. Each object is passed through the transformer as it is added to the List. It is important not to use the original list after invoking this method, as it is a backdoor for adding untransformed objects. Existing entries in the specified list will not be transformed. If you want that behaviour, see [TransformedList.transformedList(java.util.List<E>, org.apache.commons.collections4.Transformer<? super E, ? extends E>)](http://docs.google.com/org/apache/commons/collections4/list/TransformedList.html#transformedList-java.util.List-org.apache.commons.collections4.Transformer-).Type Parameters: E - the element type Parameters: list - the list to predicate, must not be null transformer - the transformer for the list, must not be null Returns: a transformed list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the List or Transformer is null

#### lazyList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> lazyList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Factory](http://docs.google.com/org/apache/commons/collections4/Factory.html)<? extends E> factory) Returns a "lazy" list whose elements will be created on demand. When the index passed to the returned list's [get](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#get-int-) method is greater than the list's size, then the factory will be used to create a new object and that object will be inserted at that index. For instance: Factory<Date> factory = new Factory<Date>() { public Date create() { return new Date(); } } List<Date> lazy = ListUtils.lazyList(new ArrayList<Date>(), factory); Date date = lazy.get(3); After the above code is executed, date will refer to a new Date instance. Furthermore, that Date instance is the fourth element in the list. The first, second, and third element are all set to null.Type Parameters: E - the element type Parameters: list - the list to make lazy, must not be null factory - the factory for creating new objects, must not be null Returns: a lazy list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the List or Factory is null

#### lazyList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> lazyList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Transformer](http://docs.google.com/org/apache/commons/collections4/Transformer.html)<[Integer](https://docs.oracle.com/javase/7/docs/api/java/lang/Integer.html?is-external=true),? extends E> transformer) Returns a "lazy" list whose elements will be created on demand. When the index passed to the returned list's [get](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#get-int-) method is greater than the list's size, then the transformer will be used to create a new object and that object will be inserted at that index. For instance: List<Integer> hours = Arrays.asList(7, 5, 8, 2); Transformer<Integer,Date> transformer = input -> LocalDateTime.now().withHour(hours.get(input)); List<LocalDateTime> lazy = ListUtils.lazyList(new ArrayList<LocalDateTime>(), transformer); Date date = lazy.get(3); After the above code is executed, date will refer to a new Date instance. Furthermore, that Date instance is the fourth element in the list. The first, second, and third element are all set to null.Type Parameters: E - the element type Parameters: list - the list to make lazy, must not be null transformer - the transformer for creating new objects, must not be null Returns: a lazy list backed by the given list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the List or Transformer is null

#### fixedSizeList public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> fixedSizeList([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list) Returns a fixed-sized list backed by the given list. Elements may not be added or removed from the returned list, but existing elements can be changed (for instance, via the [List.set(int, Object)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#set-int-E-) method).Type Parameters: E - the element type Parameters: list - the list whose size to fix, must not be null Returns: a fixed-size list backed by that list Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if the List is null

#### indexOf public static <E> int indexOf([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> list, [Predicate](http://docs.google.com/org/apache/commons/collections4/Predicate.html)<E> predicate) Finds the first index in the given List which matches the given predicate. If the input List or predicate is null, or no element of the List matches the predicate, -1 is returned.Type Parameters: E - the element type Parameters: list - the List to search, may be null predicate - the predicate to use, may be null Returns: the first index of an Object in the List which matches the predicate or -1 if none could be found

#### longestCommonSubsequence public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> longestCommonSubsequence([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> a, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> b) Returns the longest common subsequence (LCS) of two sequences (lists).Type Parameters: E - the element type Parameters: a - the first list b - the second list Returns: the longest common subsequence Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list is null Since: 4.0

#### longestCommonSubsequence public static <E> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> longestCommonSubsequence([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> a, [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<E> b, [Equator](http://docs.google.com/org/apache/commons/collections4/Equator.html)<? super E> equator) Returns the longest common subsequence (LCS) of two sequences (lists).Type Parameters: E - the element type Parameters: a - the first list b - the second list equator - the equator used to test object equality Returns: the longest common subsequence Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either list or the equator is null Since: 4.0

#### longestCommonSubsequence public static [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) longestCommonSubsequence([CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) a, [CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) b) Returns the longest common subsequence (LCS) of two [CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) objects. This is a convenience method for using [longestCommonSubsequence(List, List)](http://docs.google.com/org/apache/commons/collections4/ListUtils.html#longestCommonSubsequence-java.util.List-java.util.List-) with [CharSequence](https://docs.oracle.com/javase/7/docs/api/java/lang/CharSequence.html?is-external=true) instances.Parameters: a - the first sequence b - the second sequence Returns: the longest common subsequence as [String](https://docs.oracle.com/javase/7/docs/api/java/lang/String.html?is-external=true) Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if either sequence is null Since: 4.0

#### partition public static <T> [List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<[List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T>> partition([List](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true)<T> list, int size) Returns consecutive [sublists](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#subList-int-int-) of a list, each of the same size (the final list may be smaller). For example, partitioning a list containing [a, b, c, d, e] with a partition size of 3 yields [[a, b, c], [d, e]] -- an outer list containing two inner lists of three and two elements, all in the original order. The outer list is unmodifiable, but reflects the latest state of the source list. The inner lists are sublist views of the original list, produced on demand using [List.subList(int, int)](https://docs.oracle.com/javase/7/docs/api/java/util/List.html?is-external=true#subList-int-int-), and are subject to all the usual caveats about modification as explained in that API. Adapted from http://code.google.com/p/guava-libraries/Type Parameters: T - the element type Parameters: list - the list to return consecutive sublists of size - the desired size of each sublist (the last may be smaller) Returns: a list of consecutive sublists Throws: [NullPointerException](https://docs.oracle.com/javase/7/docs/api/java/lang/NullPointerException.html?is-external=true) - if list is null [IllegalArgumentException](https://docs.oracle.com/javase/7/docs/api/java/lang/IllegalArgumentException.html?is-external=true) - if size is not strictly positive Since: 4.0

[Skip navigation links](#3o7alnk)

* [Overview](http://docs.google.com/overview-summary.html)
* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/ListUtils.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/KeyValue.html)
* [Next Class](http://docs.google.com/org/apache/commons/collections4/ListValuedMap.html)
* [Frames](http://docs.google.com/index.html?org/apache/commons/collections4/ListUtils.html)
* [No Frames](http://docs.google.com/ListUtils.html)
* [All Classes](http://docs.google.com/allclasses-noframe.html)
* Summary:
* Nested |
* Field |
* Constr |
* [Method](#3znysh7)
* Detail:
* Field |
* Constr |
* [Method](#tyjcwt)

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