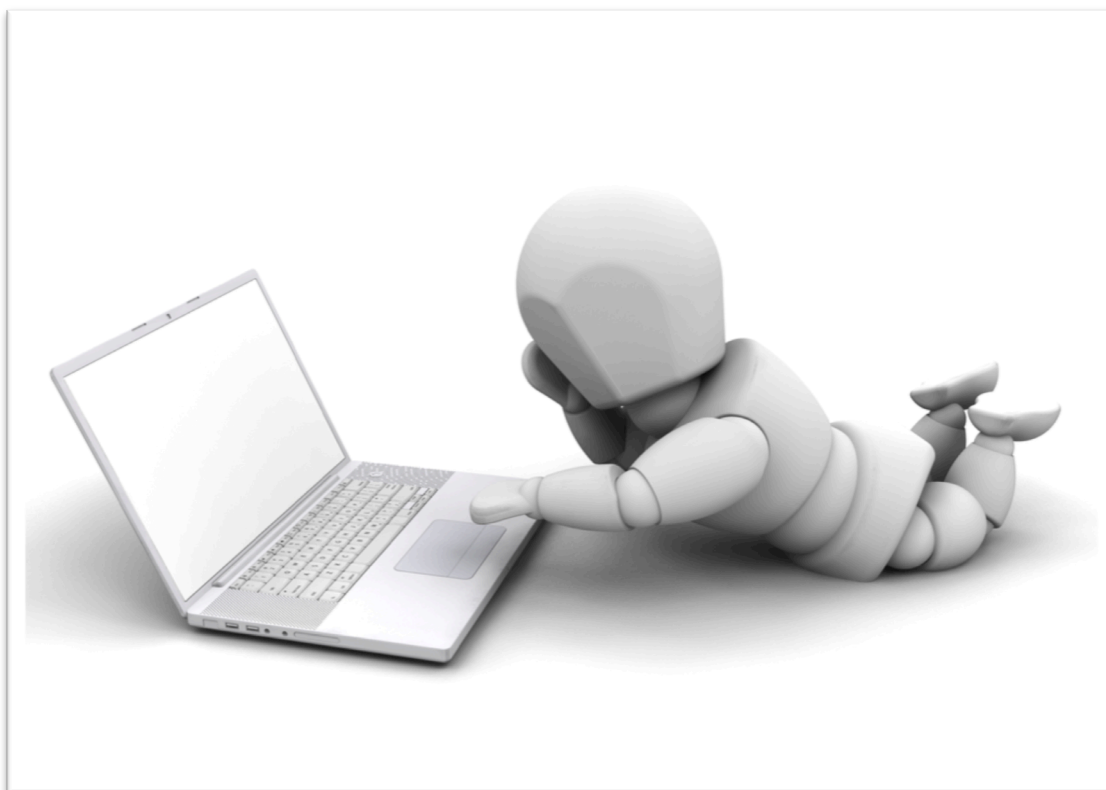


Christa Fox

Basic Linked Collection

Test Case Plan



Explanation

The BasicLinkedCollection class extends AbstractCollection. It has to override and implement some methods from it. The following is the test plan.

Test Case Plan

Constructors and setUp

Operation	Purpose	Object State	Expected Result
BasicLinkedCollection<String> cEmpty = new BasicLinkedCollection<>()	Creates an empty collection.	cEmpty.isEmpty = true	A new Basic Linked collection for Strings was created that was empty.
BasicLinkedCollection<String> c = new BasicLinkedCollection<>()	Creates an empty collection for String items to be added to.	cEmpty.isEmpty = true	A new Basic Linked collection for Strings to be added to was created.
cEmpty.size()	Verify that an empty collection has a size of zero.	cEmpty.size() = 0;	0

Add methods

Operation	Purpose	Object State	Expected Result
c.add(new String("A"));	Add to the collection c.	c.size = 1; "A"	True
c.add(new String("B"));	Add to the collection c.	c.size = 2; "B"	True
c.add(new String("C"));	Add to the collection c.	c.size = 3; "C"	True
c.size	Verify collection state		3
c.isEmpty	Verify collection state		False
c.contains("A");	Verify collection state		True
c.contains("B");	Verify collection state		True
c.contains("C");	Verify collection state		True
c.contains("missing");	Verify collection state		False

Remove

Operation	Purpose	Object State	Expected Result
c.remove("A")	To remove the element at the head.	c.size = 2; "B" "C"	true
c.remove("B")	To remove the element at middle.	c.size = 2; "A" "C"	true
c.remove("C")	To remove the element at the tail.	c.size = 2; "A" "B"	true
c.remove("missing")	To remove an item that is not contained in the collection.	c.size = 3; "A" "B" "C"	true

Abstract Collection

Boolean isEmpty() is a method in AbstractCollection, so Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is written there that it is to return true if no elements are present in the collection.

Since Boolean addAll() is an optional method to instantiate, Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is written there such that all elements provided will be added to the collection.

Boolean removeAll() is an optional method to instantiate, so Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is written there such that all elements in the collection will be removed.

Object[] toArray is an optional method to instantiate, so Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is found in AbstractCollection, and it states that arrays of type object may be transferred to list collection arrays.

T[] toArray(T[]) is an optional method to instantiate, so Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is found in AbstractCollection, and it states that arrays of type list or collection may be transferred to arrays.

Clear() is an optional method to instantiate, so Java looks for the method in BasicLinkedList first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is found in AbstractCollection, and it states all the elements in a collection will be removed.

Contains is an optional method to instantiate, so Java looks for the method in BasicLinkedCollection first. I did override this class in my BasicLinkedCollection.

ContainAll is an optional method to instantiate, so Java looks for the method in BasicLinkedCollection first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is found in AbstractCollection, and it states all the elements in a collection must match all the elements in the other collection.

RetainAll is an optional method to instantiate, so Java looks for the method in BasicLinkedCollection first. If Java can't find the method there, it goes to the parent class—AbstractCollection. The method is found in AbstractCollection, and it states that all of the methods in that collection will be retained.