

JAVA.UTIL.ARRAYLIST.ADDALL METHOD

http://www.tutorialspoint.com/java/util/arraylist_addall_index.htm

Copyright © tutorialspoint.com

Description

The **java.util.ArrayList.addAll**(int index, Collection < ? extends E > c) method inserts all of the elements in the specified collection into this list, starting at the specified position. It shifts the element currently at that position *if any* and any subsequent elements to the right *increases their indices*. The new elements will appear in the list in the order that they are returned by the specified collection's iterator.

Declaration

Following is the declaration for **java.util.ArrayList.addAll**(int index, c) method

```
public boolean addAll(int index, Collection<? extends E> c)
```

Parameters

- **index** -- The index at which to insert the first element from the specified collection.
- **c** -- This is the collection containing elements to be added to this list.

Return Value

This method returns true if this list changed as a result of the call.

Exception

- **IndexOutOfBoundsException** -- If the index is out of range
- **NullPointerException** -- If the specified collection is null

Example

The following example shows the usage of java.util.ArrayList.addAll(index, c) method.

```
package com.tutorialspoint;

import java.util.ArrayList;

public class ArrayListDemo {
    public static void main(String args[]) {

        // create an empty array list1 with an initial capacity
        ArrayList<Integer> arrlist = new ArrayList<Integer>(5);

        // use add() method to add elements in the list
        arrlist.add(12);
        arrlist.add(20);
        arrlist.add(45);

        // let us print all the elements available in list1
        System.out.println("Printing list1:");
        for (Integer number : arrlist) {
            System.out.println("Number = " + number);
        }

        // create an empty array list2 with an initial capacity
        ArrayList<Integer> arrlist2 = new ArrayList<Integer>(5);

        // use add() method to add elements in list2
        arrlist2.add(25);
        arrlist2.add(30);
```

```

arrlist2.add(31);
arrlist2.add(35);

// let us print all the elements available in list2
System.out.println("Printing list2:");
for (Integer number : arrlist2) {
System.out.println("Number = " + number);
}

// inserting all elements of list2 at third position
arrlist.addAll(2, arrlist2);

System.out.println("Printing all the elements");
// let us print all the elements available in list1
for (Integer number : arrlist) {
System.out.println("Number = " + number);
}
}
}

```

Let us compile and run the above program, this will produce the following result:

```

Printing list1:
Number = 12
Number = 20
Number = 45
Printing list2:
Number = 25
Number = 30
Number = 31
Number = 35
Printing all the elements
Number = 12
Number = 20
Number = 25
Number = 30
Number = 31
Number = 35
Number = 45

```

Loading [Mathjax]/jax/output/HTML-CSS/fonts/TeX/fontdata.js