**How to synchronize ArrayList in java with example**

[**JAVA COLLECTIONS**](http://beginnersbook.com/category/java-collections/)

We have already discussed a bit about synchronization when we shared the tutorial on [**Vector vs ArrayList**](http://beginnersbook.com/2013/12/difference-between-arraylist-and-vector-in-java/). As we are aware that ArrayList is non-synchronized and should not be used in multi-thread environment without explicit synchronization. This post is to discuss how to synchronize ArrayList in Java.

**There are two ways to synchronize explicitly:**

1. Using Collections.synchronizedList() method
2. Using thread-safe variant of ArrayList: CopyOnWriteArrayList

**Example 1: Collections.synchronizedList() method for Synchronizing ArrayList**

In this example we are using [**Collections.synchronizedList()**](http://docs.oracle.com/javase/6/docs/api/java/util/Collections.html#synchronizedList(java.util.List)) method. The important point to note here is that iterator should be in synchronized block in this type of synchronization as shown in the below example.

package beginnersbook.com;

import java.util.ArrayList;

import java.util.Iterator;

import java.util.List;

import java.util.Collections;

public class Details {

public static void main(String a[]){

List<String> syncal =

Collections.synchronizedList(new ArrayList<String>());

//Adding elements to synchronized ArrayList

syncal.add("Pen");

syncal.add("NoteBook");

syncal.add("Ink");

System.out.println("Iterating synchronized ArrayList:");

synchronized(syncal) {

Iterator<String> iterator = syncal.iterator();

while (iterator.hasNext())

System.out.println(iterator.next());

}

}

}

Output:

Iterating synchronized ArrayList:

Pen

NoteBook

Ink

**Method 2: Using CopyOnWriteArrayList**

[**CopyOnWriteArrayList**](http://docs.oracle.com/javase/6/docs/api/java/util/concurrent/CopyOnWriteArrayList.html) is a thread-safe variant of ArrayList.

package beginnersbook.com;

import java.util.concurrent.CopyOnWriteArrayList;

import java.util.Iterator;

public class Details {

public static void main(String a[]){

CopyOnWriteArrayList<String> al = new CopyOnWriteArrayList<String>();

//Adding elements to synchronized ArrayList

al.add("Pen");

al.add("NoteBook");

al.add("Ink");

System.out.println("Displaying synchronized ArrayList Elements:");

//Synchronized block is not required in this method

Iterator<String> iterator = al.iterator();

while (iterator.hasNext())

System.out.println(iterator.next());

}

}

Output:

Displaying synchronized ArrayList Elements:

Pen

NoteBook

Ink