

# Difference between ArrayList and LinkedList

ArrayList and LinkedList both implements List interface and maintains insertion order. Both are non synchronized classes.

But there are many differences between ArrayList and LinkedList classes that are given below.

ArrayList	LinkedList
1) ArrayList internally uses <b>dynamic array</b> to store the elements.	LinkedList internally uses <b>doubly linked list</b> to store the elements.
2) Manipulation with ArrayList is <b>slow</b> because it internally uses array. If any element is removed from the array, all the bits are shifted in memory.	Manipulation with LinkedList is <b>faster</b> than ArrayList because it uses doubly linked list so no bit shifting is required in memory.
3) ArrayList class can <b>act as a list</b> only because it implements List only.	LinkedList class can <b>act as a list and queue</b> both because it implements List and Deque interfaces.
4) ArrayList is <b>better for storing and accessing</b> data.	LinkedList is <b>better for manipulating</b> data.

## Example of ArrayList and LinkedList in Java

Let's see a simple example where we are using ArrayList and LinkedList both.

```
import java.util.*;

class TestArrayLinked{

    public static void main(String args[]){

        List<String> al=new ArrayList<String>();//creating arraylist
        al.add("Ravi");//adding object in arraylist
        al.add("Vijay");
        al.add("Ravi");
        al.add("Ajay");

        List<String> al2=new LinkedList<String>
        ();//creating linkedlist
        al2.add("James");//adding object in linkedlist
        al2.add("Serena");
        al2.add("Swati");
        al2.add("Junaid");

        System.out.println("arraylist: "+al);
```

```
System.out.println("linkedlist: "+al2);  
}  
}
```

### Test it Now

Output:

```
arraylist: [Ravi,Vijay,Ravi,Ajay]  
linkedlist: [James,Serena,Swati,Junaid]
```

[< prev](#)[next >](#)

## Share this page



## Latest 4 Tutorials



CouchDB



Docker



Rails



RichFaces