8 Difference Between Array And ArrayList In Java With Example

Difference between array and arraylist in java is considered as a starting interview question . This question checks whether candidate know about static and dynamic nature of array.We have already discussed other popular java interview questions like difference between comparable and comparator and difference between arraylist and vector . *Difference between array and arraylist in java* include eight points namely Resizable, Performance, Traversal ,Primitives , Length , Type-Safety, Adding elements , Multi-dimensional.

Read Also : Difference between HashSet and TreeSet

Difference between Array and ArrayList in Java with Example

1. Resizable : Array is static in size that is fixed length data structure, One can not change the length after creating the Array object.
ArrayList is dynamic in size . Each ArrayList object has instance variable *capacity* which indicates the size of the ArrayList. As elements are added to an ArrayList its capacity grows automatically.

2. Performance : Performance of Array and ArrayList depends on the operation you are performing :

resize() opertation : Automatic resize of ArrayList will slow down the performance as it will use temporary array to copy elements from the old array to new array.
ArrayList is internally backed by Array during resizing as it calls the native implemented method System.arrayCopy(src,srcPos,dest,destPos,length) .

add() or get() operation : adding an element or retrieving an element from the array or arraylist object has almost same performance , as for ArrayList object these operations run in constant time.

3. Primitives : ArrayList can not contains primitive data types (like int , float , double) it can only contains Object while Array can contain both primitive data types as well as objects.

One get a misconception that we can store primitives(int,float,double) in ArrayList , but it is not true

Suppose we have ArrayList object ,

```
ArrayList  arraylistobject = new ArrayList();
arraylistobject.add(23);  // try to add 23 (primitive)
```

JVM through Autoboxing(converting primitives to equivalent objects internally) ensures that only objects are added to the arraylist object.
thus , above step internally works like this :

```
arraylistobject.add( new Integer(23));
// Converted int primitive to Integer object and added to arraylistobject
```

4. Iterating the values : We can use iterator to iterate through ArrayList . The iterators returned by the ArrayList class's iterator and listiterator method are fail-fast. We can use for loop or for each loop to iterate through array .

5. Type-Safety : In Java , one can ensure Type Safety through Generics. while Array is a homogeneous data structure , thus it will contain objects of specific class or primitives of specific data type. In array if one try to store the different data type other than the specified while creating the array object , ArrayStoreException is thrown.

for example :

```
String temp[] =  new String[2];  // creates a string array of size 2
temp[0] = new Integer(12); // throws ArrayStoreException, trying to add Integer
object in String[]
```

6. Length : Length of the ArrayList is provided by the size() method while Each array object has the length variable which returns the length of the array.

for example :

```
Integer arrayobject[] = new Integer[3];
arraylength= arrayobject.length   ;  //uses arrayobject length variable
```

```
ArrayList  arraylistobject = new ArrayList();
arraylistobject.add(12);
arraylistobject.size();   //uses arraylistobject size method
```

7. Adding elements : We can insert elements into the arraylist object using the add() method while in array we insert elements using the assignment operator.

for example :

```
Integer addarrayobject[] = new Integer[3];
addarrayobject[0]= new Integer(8)   ;  //new object is added to the array object
```

8. Multi-dimensional : Array can be multi dimensional , while ArrayList is always single dimensional.

example of multidimensional array:

```
Integer addarrayobject[][] = new Integer[3][2];
addarrayobject[0][0]= new Integer(8)
```

Example of Array and ArrayList

```
import java.util.ArrayList;
import java.util.Iterator;

public class ArrayArrayListExample {

    public static void main(String[] args) {

        // ArrayList Example

        ArrayList<String> arrlistobj = new ArrayList<String>();
        arrlistobj.add("Alive is awesome");
        arrlistobj.add("Love yourself");
        Iterator it = arrlistobj.iterator();
        System.out.print("ArrayList object output :");
        while(it.hasNext())
          System.out.print(it.next() + " ");



        // Array Example

        String[] arrayobj = new String[3];
        arrayobj[0]= "Love yourself";
        arrayobj[1]= "Alive is awesome";
        arrayobj[2]= "Be in Present";
        System.out.print("Array object output :");
        for(int i=0; i < arrayobj.length ;i++)
        System.out.print(arrayobj[i] + " ");

  }
 }
```

Output : *ArrayList object output :**{ Alive is awesome , Love yourself }***
*Array object output :**{ Love yourself , Alive is awesome, Be in present}***

Similarities Between Array and ArrayList

1. add and get method : Performance of Array and ArrayList are similar for the add and get operations .Both operations runs in constant time.

2. Duplicate elements : Both array and arraylist can contain duplicate elements.

3. Null Values : Both can store null values and uses index to refer to their elements.

4. Unordered : Both does not guarantee ordered elements.

Recap : Difference between Array and ArrayList in Java

	Array	ArrayList
Resizable	No	Yes
Primitives	Yes	No
Iterating values	for, for each	Iterator , for each
Length	length variable	size method
Performance	Fast	Slow in comparision
Multidimensional	Yes	No
Add Elements	Assignment operator	add method