CORE JAVA PROGRAMMING ARRAYLIST AND WRAPPER CLASSES (1Z0-808)

By www.HadoopExam.com

Note: These instructions should be used with the HadoopExam Apache OOzie: Professional Trainings. Where it is executed and you can do hands on with trainer.

Array List: An array has one shortcoming: you have to know how many elements will be in the array when you create it.

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

	public static void main(String[] args) {
		ArrayList list1 = new ArrayList();
		ArrayList list2 = new ArrayList(10);
		ArrayList list3 = new ArrayList(list2);

		System.out.println("Size list1 " + list1.size() + " Size list2 " + list2.size() + " Size list3
" +list3.size());

		ArrayList<String> list4 = new ArrayList<String>();
		ArrayList<String> list5 = new ArrayList<>();

		System.out.println("Size list4 " + list4.size() + " Size list5 " + list5.size());

		List<String> list6 = new ArrayList<>();
		//ArrayList<String> list7 = new List<>(); // DOES NOT COMPILE
	}
}
```

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

	public static void main(String[] args) {

		//List add() method
		ArrayList list1 = new ArrayList();
		list1.add("Hadoop");
		list1.add("Exam");
		list1.add(".com");
		list1.add(0,new Integer(1));
		System.out.println(list1);
```

```
		//Safe list
		ArrayList<String> list2 = new ArrayList<String>();
		list2.add("Hadoop");
		list2.add("Exam");
		//list2.add(new Integer(1));
		System.out.println(list2);
	}
}
```

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

	public static void main(String[] args) {

		//List add() method
		ArrayList list1 = new ArrayList();
		list1.add("Hadoop");
		list1.add("Exam");
		list1.add(".com");
		list1.add(0,new Integer(1));
		System.out.println("Before Element removal "+list1);

		list1.remove(0);
		list1.remove(".com");
		System.out.println("After Element removal "+list1);


	}
}
```

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

	public static void main(String[] args) {

		//List add() method
```

```
                ArrayList list1 = new ArrayList();
                list1.add("Hadoop");
                list1.add("Exam");
                list1.add(".com");
                list1.add(0,new Integer(1));
                System.out.println("Before Element removal "+list1 + "Size of list = " + list1.size());

                list1.remove(0);
                list1.remove(".com");
                System.out.println("After Element removal "+list1 + "Size of list = " + list1.size());

                //Check whether list is empty or not.
                System.out.println( list1.isEmpty());

                list1.set(0,"www.");
                list1.set(1,"HadoopExam");
                //list1.set(2,".com"); //java.lang.IndexOutOfBoundsException
                System.out.println("After Element Set "+list1 + "Size of list = " + list1.size());

                ArrayList list2 = new  ArrayList();
                list2.addAll(list1);

                System.out.println(list1.contains("HadoopExam"));
                System.out.println(list1.equals(list2));
                System.out.println(list1 == list2);
        }
}
```

Wrapper Classes

- Each primitive type has a wrapper class.
- Wrapper class is an Object form of primitive types, with having much more functionalities.

Primitive Types	Wrapper Class	Example
boolean	Boolean	new Boolean(false)
byte	Byte	new Byte((byte)2)
short	Short	new Short(1 short)
int	Integer	new Integer(1)
long	Long	new Long(1)

float	Float	new Float(1.0)
double	Double	new Double(1.0)
Char	Character	new Character('C')

- We can convert primitive to wrapper and wrapper to primitive class using methods e.g. intValue()
- String to Wrapper class using valueOf method .
- String to primitive value using parseInt() method.

```
public class Welcome {

	public static void main(String[] args) {
		Integer wInt = new Integer(100);
		System.out.println(wInt.intValue());
		System.out.println(wInt);

		System.out.println(Integer.parseInt("1000")); //String to primitive types
		System.out.println(Integer.valueOf("500").getClass()); //String to Wrapper class
		//System.out.println(Integer.valueOf("Hadoop").getClass()); //You will get an RuntimeException here.
	}
}
```

Wrapper class	Converting `String` to primitive	Converting `String` to wrapper class
Boolean	`Boolean.parseBoolean("true");`	`Boolean.valueOf("TRUE");`
Byte	`Byte.parseByte("1");`	`Byte.valueOf("2");`
Short	`Short.parseShort("1");`	`Short.valueOf("2");`
Integer	`Integer.parseInt("1");`	`Integer.valueOf("2");`
Long	`Long.parseLong("1");`	`Long.valueOf("2");`
Float	`Float.parseFloat("1");`	`Float.valueOf("2.2");`
Double	`Double.parseDouble("1");`	`Double.valueOf("2.2");`
Character	None	None

Autoboxing (Java 1.5 onwards)

- Java automatically convert the primitive data types to wrapper class, wherever it is required.

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

   public static void main(String[] args) {
         List<Double> doubleList = new ArrayList<Double>();

         double val = 26.3;
         System.out.println("Is value added " + doubleList.add(val) + "Content of list" + doubleList);

         //Check below how float is converted into double and underline value also changed :(
         float f = 2.1f;
         System.out.println("Is value added " + doubleList.add((double) f) + "Content of list" + doubleList);
         System.out.println("Is value added " + doubleList.add((double) f) + "Content of list" + doubleList);

         //Check by removing cast here
```

```
        System.out.println("Is value removed " + doubleList.remove((double) f) + "Content of list" + doubleList);

        //Now add null value in list
        System.out.println("Is value added " + doubleList.add(null) + "Content of list" + doubleList);
        System.out.println("Is value added " + doubleList.add(null) + "Content of list" + doubleList);

        //Now add null value in list
        System.out.println("Is value removed " + doubleList.remove(null) + "Content of list" + doubleList);


    }
}
```

- Autoboxing and methods in an Integer ArrayLists (Check Index position or actual value is being removed ?)

```
import java.util.ArrayList;
import java.util.List;

public class Welcome {

   public static void main(String[] args) {

        List<Integer> list = new ArrayList<Integer>();

        list.add(1);
        list.add(2);

        System.out.println("Contents in the list = " + list);

        //Check the surprise by un-commenting below methods
        //list.remove(1);
        list.remove(new Integer(1));
        System.out.println("Contents in the list = " + list);


    }
}
```

Convert Array and List

- When you convert array to list, lists are backed by arrays. Hence, whatever operation you will apply on arrays will also be reflected In lists.
- You can also convert List to arrays back.

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Collections;
import java.util.List;

public class Welcome {

   public static void main(String[] args) {

         String[] s1 = new String[5];
         s1[0] = "Hadoop";
         s1[1] = "Exam";
         s1[2] = ".com";

         System.out.println(s1.length);

         List<String> sList = new ArrayList<String>();
         sList = Arrays.asList(s1);
         System.out.println(sList);

         s1[0] = "Training4"; // Chenage 1st element in array
         System.out.println(sList);

         System.out.println(sList.toArray());
         //Below changes will be applied to both array and lists
         sList.set(0, "www.");
         sList.set(1, "HadoopExam");
         sList.set(3, "Learning");
         sList.set(4, "Resources");
         System.out.println(sList);
         //Sorting the list
         Collections.sort(sList);
         System.out.println(sList);

   }
}
```

All Products List of www.HadoopExam.com

TRAINING'S

- Hadoop BigData Professional Training (3500INR/$79)
- HBase (NoSQL) Professional Training (3500INR/$79)
- Apache Spark Professional Training (~~3900INR/$89~~ for a week 3500INR/$75)
- Apache OOZie (Hadoop workflow) Professional Training
- Beginner AWS Training Course- **(HETRNAWS101)**
- Core Java 1z0-808 Exam training

MAPR HADOOP AND NOSQL CERTIFICATION

- **MapR Hadoop Developer Certification**
- **MapR HBase NoSQL Certification**
- **MapR Spark Developer Certification (In Progress)**

CLOUDERA HADOOP AND NOSQL CERTIFICATION

- CCA50X : Hadoop Administrator
- CCA-175 Cloudera® (Hadoop and Spark Developer)
- CCP:DE575 : Cloudera® Data Engineer Certification

DATABRICKSA OREILLY SPARK CERTIFICATION

- Apache Spark Developer

AWS: AMAZON WEBSERVICE CERTIFICATION

- AWS Solution Architect : Associate
- AWS Solution Architect: Professional
- AWS Developer : Associate
- AWS Sysops Admin : Associate

MICROSOFT AZURE CERTIFICATION

- Azure 70-532
- Azure 70-533

DATA SCIENCE CERTIFICATION

- EMC E20-007

EMC CERTIFICATIONS

- EMC E20-007

SAS ANALYTICS CERTIFICATION

- SAS Base A00-211
- SAS Advanced A00-212
- SAS Analytics : A00-240
- SAS Administrator : A00-250

ORACLE JAVA CERTIFICATION

- Java 1z0-808
- Java 1z0-809

ORACLE DATABASE CLOUD CERTIFICATION

- 1z0-060 (Oracle 12c)
- 1z0-061 (Oracle 12c)

Subscribe Here for Regular Updates: Like New Training Module launched

Become Author and Trainer: We are looking for Author (Writing Technical Books) and Trainer (Creating Training Material): **No Compromise on Quality.**

Benefit: You will get very good revenue sharing. Please drop us an email to hadoopexam@gmail.com (For the skills, you feel you are master)

We are sure, you are good at least one technology. Don't limit your potential, contact us immediately with your skill. Our expert team will contact you with more detail. You training and Books will reach to all our existing network and with our expert marketing team we will help you to reach as much as technical professional, with our Smart Advertising network. Contact us with sending an email hadoopexam@gmail.com

Opportunity to share your knowledge with all learners who are in need. We are helping 1000's of learners since last 4 years and established ourselves with Quality low cost material.