

Collection type:

1)Set

HashSet,TreeSet

2)List

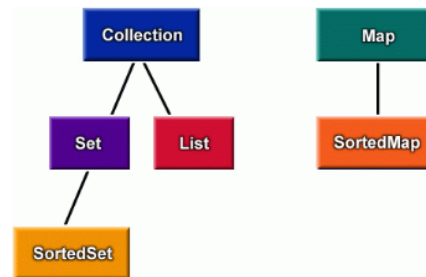
ArraList,LinkedList

3)Map

HashMap,TreeMap

Collection Interfaces

- Collections are primarily defined through a set of interfaces.
 - Supported by a set of classes that implement the interfaces



[Source: java.sun.com]

HashSet example

```
import com.objectspace.jgl.*;
import java.util.Enumeration;
```

```
/**
```

```
 * Construction, enumeration, rejection of duplicates.
```

```
 *
```

```
 * @see com.objectspace.jgl.HashSet
```

```
 * @version 3.0.0
```

```
 * @author ObjectSpace, Inc.
```

```
 */
```

```
public class HashSet1
```

```
{
```

```
    public static void main( String[] args )
```

```
    {
```

```
        HashSet set = new HashSet();
```

```
        set.add( new Integer( 6 ) );
```

```
        set.add( new Integer( 1 ) );
```

```
        set.add( new Integer( 4 ) );
```

```
        System.out.println( set );
```

```
        System.out.println();
```

```
        System.out.println( "Enumerate the HashSet" );
```

```
        Enumeration e = set.elements();
```

```
        while ( e.hasMoreElements() )
```

```
            System.out.println( e.nextElement() );
```

```
        System.out.println();
```

```
        System.out.println( "Iterate through the HashSet" );
```

```
        for ( HashSetIterator i = set.begin(); !i.atEnd(); i.advance() )
```

```
            System.out.println( i.get() );
```

```
        System.out.println();
```

```
        System.out.println( "Show that duplicates cannot be added." );
```

```

Object value = set.add( new Integer( 8 ) );
if ( value != null )
    System.out.println( "Could not add 8." );
else
    {
        System.out.println( "Added 8" );
        System.out.println( "New contents are " + set );
    }

value = set.add( new Integer( 4 ) );
if ( value != null )
    System.out.println( "Could not add 4." );
else
    {
        System.out.println( "Added 4." );
        System.out.println( "New contents are " + set );
    }
}
}

```

Arraylist example

```

import java.util.ArrayList;

public class SimpleArrayListExample {

    public static void main(String[] args) {

        //create an ArrayList object
        ArrayList arrayList = new ArrayList();

        /*
         Add elements to Arraylist using
         boolean add(Object o) method. It returns true as a general behavior
         of Collection.add method. The specified object is appended at the end
         of the ArrayList.
        */
        arrayList.add("1");
        arrayList.add("2");
        arrayList.add("3");

        /*
         Use get method of Java ArrayList class to display elements of ArrayList.
         Object get(int index) returns and element at the specified index in
         the ArrayList
        */
        System.out.println("Getting elements of ArrayList");
        System.out.println(arrayList.get(0));
        System.out.println(arrayList.get(1));
        System.out.println(arrayList.get(2));
    }
}

```

HashMap example

```

import java.util.*;
class HashMapDemo {
public static void main(String args[]) {
    // Create a hash map
    HashMap hm = new HashMap();
    // Put elements to the map
    hm.put("John Doe", new Double(3434.34));
    hm.put("Tom Smith", new Double(123.22));
    hm.put("Jane Baker", new Double(1378.00));
    hm.put("Todd Hall", new Double(99.22));
    hm.put("Ralph Smith", new Double(-19.08));
}
}

```

```
// Get a set of the entries
Set set = hm.entrySet();
// Get an iterator
Iterator i = set.iterator();
// Display elements
while(i.hasNext()) {
Map.Entry me = (Map.Entry)i.next();
System.out.print(me.getKey() + ": ");
System.out.println(me.getValue());
System.out.println();
// Deposit 1000 into John Doe's account
double balance = ((Double)hm.get("John Doe")).doubleValue();
hm.put("John Doe", new Double(balance + 1000));
System.out.println("John Doe's new balance: " +
hm.get("John Doe"));
}
```