**ArrayList**

package com.collections.demo;

import java.util.ArrayList;

import java.util.Iterator;

/\* Collection : To represent group of elements/objects/data into a single entity

\* "collection" is an interface available in java.util

\* ArrayList is a class which is implemented List interface.

\* 1) Heterogeneous data --> allowed ( different type of data)

\* 2)Insertion order --> preserved(index)

\* 3) Duplicate elements --> allowed

\* 4) multiple nulls --> allowed

\*

\*/

public class ArrayListDemo {

public static void main(String[] args) {

// Declaration

ArrayList myList = new ArrayList();

//List mylist = new ArrayList();

//ArrayList <String> mylist = new ArrayList<String>();

// adding data into ArrayList.

myList.add(100);

myList.add("sagar");

myList.add(true);

myList.add('A');

myList.add(100);

myList.add(null);

myList.add(null);

// size of an arraylist

System.out.println(myList.size());

// printing data from an arraylist.

System.out.println("Data of arraylist :"+myList);

// Remove the element

myList.remove(4); // 4 is index of element

System.out.println(" After Removing Data of arraylist :"+myList);

// Insert element in ArrayList

myList.add(4,200); // add(index,value)

System.out.println("After Insertion :"+myList);

// modify element in ArrayList (modify/change.replace)

myList.set(1,"Mangesh");

System.out.println("After Replacing :"+myList);

// access specific element from arraylist

System.out.println(myList.get(3)); // 3 is index

// Reading all the elements from ArrayList

// using normal for loop

/\*for(int i=0; i<myList.size();i++)

{

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

}

\*/

//using inhanced for loop or for each loop

/\*for(Object x:myList)

{

System.out.println(x);

}\*/

// using Iterator

Iterator it = myList.iterator();

while (it.hasNext())

{

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

}

// checking ArrayList is empty or not

System.out.println(myList.isEmpty());

// Remove all element from ArrayList

ArrayList myList2 = new ArrayList();

myList2.add(100);

myList2.add(null);

myList.removeAll(myList2);

System.out.println(myList);

// Remove all the ArrayList

myList.clear();

System.out.println(myList.isEmpty());

}

}

**HashMap**

package com.collections.demo;

import java.util.HashMap;

import java.util.Iterator;

/\* HashMap : is a class implemented Map interface.

\* Data can be stored in the form of key,value pairs.

\* Key is unique. But we can have duplicate values.

\* Insertion order not preserved ( Index not followed)

\*

\*/

public class HashMapDemo {

public static void main(String[] args) {

// declaration

//HashMap hm = new HashMap();

//Map hm = new HashMap();

HashMap <Integer,String> hm = new HashMap<Integer,String>();

// adding values

hm.put(101,"John");

hm.put(102,"David");

hm.put(103,"ram");

hm.put(102,"lop");

// printing

System.out.println(hm);

// size

System.out.println(hm.size());

// remove pair

hm.remove(103); //103 is key of pair

System.out.println(hm);

// access value

System.out.println(hm.get(101));

// get all keys from hashmap

System.out.println(hm.keySet());//only keys

System.out.println(hm.values());//only values

System.out.println(hm.entrySet());//both key and value

// reading data

// using for each

/\*

for(int k:hm.keySet())

{

System.out.println(k+" "+hm.get(k));

}\*/

// using Iterator

Iterator it=hm.entrySet().iterator();

while(it.hasNext())

{

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

}

}

}

**HashSet**

package com.collections.demo;

import java.util.ArrayList;

import java.util.HashSet;

import java.util.Iterator;

import java.util.Set;

/\* HashSet : a class implemented set interface

\* 1) Heterogeneous Data --> allowed

\* 2) Insertion order --> Not preserved ( Index not supported)

\* 3) Duplicate elements --> Not allowed

\* 4) Multiple nulls Not allowed/ only single null is allowed

\*

\*

\*/

public class HashSetDemo {

public static void main(String[] args) {

// Declaration

HashSet myset = new HashSet();

//Set myset = new HashSet();

//HashSet <String> myset = new HashSet<String>();

// adding elements

myset.add(100);

myset.add("sagar");

myset.add('A');

myset.add(12.5);

myset.add(true);

myset.add(100);

myset.add(null);

myset.add(null);

// printing

System.out.println(myset); //[null, A, 100, sagar, 12.5, true]

// remove element

myset.remove(12.5); // 12.5 is value not index

System.out.println("After removing :"+myset);

// Insertion is not possible

// accessing specific element not possible

// convert HashSet --> ArrayList

ArrayList arr = new ArrayList(myset);

System.out.println(arr.get(2));

// Read all elements using only For..each loop

for(Object a:myset)

{

System.out.println(a);

}

// using Iterator

Iterator it = myset.iterator();

while(it.hasNext())

{

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

}

// clearing all elements

myset.clear();

System.out.println(myset.isEmpty());

}

}