**ArrayList**

import java.util.\*;

public class TestJavaCollection1{

public static void main(String args[]){

ArrayList<String> LL=new ArrayList <String>();

LL.add("Ayush");

LL.add("Amit");

LL.add("Ashish");

LL.add("Garima");

Iterator<String> itr= LL.iterator();

while(itr.hasNext()){

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

}

}

}

**Linked List**

import java.util.\*;

public class TestJavaCollection2{

public static void main(String args[]){

LinkedList<String> LL=new LinkedList <String>();

LL.add("Ayush");

LL.add("Amit");

LL.add("Ashish");

LL.add("Garima");

Iterator<String> itr= LL.iterator();

while(itr.hasNext()){

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

}

}

}

**Vector**

import java.util.\*;

public class TestJavaCollection3{

public static void main(String args[]){

Vector<String> v=new Vector<String>();

v.add("Ayush");

v.add("Amit");

v.add("Ashish");

v.add("Garima");

Iterator<String> itr=v.iterator();

while(itr.hasNext()){

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

}

}

}

**Stack**

import java.util.\*;

public class TestJavaCollection4{

public static void main(String args[]){

Stack<String> stack = new Stack<String>();

stack.push("Ayush");

stack.push("Garvit");

stack.push("Amit");

stack.push("Ashish");

stack.push("Garima");

stack.pop();

Iterator<String> itr=stack.iterator();

while(itr.hasNext()){

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

}

}

}

**Queue**:

import java.util.\*;

public class TestJavaCollection5{

public static void main(String args[]){

PriorityQueue<String> queue=new PriorityQueue<String>();

queue.add("Amit Sharma");

queue.add("Vijay Raj");

queue.add("JaiShankar");

queue.add("Raj");

System.out.println("head:"+queue.element());

System.out.println("head:"+queue.peek());

System.out.println("iterating the queue elements:");

Iterator itr=queue.iterator();

while(itr.hasNext()){

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

}

queue.remove();

queue. poll();

System.out.println("after removing two elements:");

Iterator<String> itr2=queue.iterator();

while(itr2.hasNext()){

System.out.println(itr2.next());

}

}

}

**HashSet**

import java.util.\*;

public class TestJavaCollection{

public static void main(String args[]){

//Creating HashSet and adding elements

HashSet<String> set=new HashSet<String>();

set.add("Ravi");

set.add("Vijay");

set.add("Ravi");

set.add("Ajay");

//Traversing elements

Iterator<String> itr=set.iterator();

while(itr.hasNext()){

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

}

}

}

**HashMap**

import java.util.HashMap;

public class SimpleHashMapExample {

public static void main(String[] args) {

// Create a HashMap

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

// Add key-value pairs

map.put(1, "Apple");

map.put(2, "Banana");

map.put(3, "Cherry");

// Print the HashMap

System.out.println("HashMap: " + map);

// Access a value using key

System.out.println("Value for key 2: " + map.get(2));

// Remove a key-value pair

map.remove(3);

// Check if a key exists

System.out.println("Contains key 1? " + map.containsKey(1));

// Iterate through keys and values

System.out.println("Iterating through map:");

for (Integer key : map.keySet()) {

System.out.println(key + " -> " + map.get(key));

}

}

}

**Comparable**

class Student implements Comparable<Student> {

int rollNo;

String name;

Student(int rollNo, String name) {

this.rollNo = rollNo;

this.name = name;

}

// Natural order: by rollNo

@Override

public int compareTo(Student s) {

return this.rollNo - s.rollNo;

}

}

**Comparator**

import java.util.\*;

class Student {

int rollNo;

String name;

Student(int rollNo, String name) {

this.rollNo = rollNo;

this.name = name;

}

}

// Comparator for sorting by name

class NameComparator implements Comparator<Student> {

public int compare(Student s1, Student s2) {

return s1.name.compareTo(s2.name);

}

}