**ASSIGNMENT -11**

**Session 2**

1. Implement Queue data structure using LinkedList and Queue collection.

**package** my.collections;

**import** java.util.Queue;

**import** java.util.LinkedList;

**public** **class** queue {

**public** **static** **void** main(String[] args) {

Queue<Integer> q= **new** LinkedList();

**for**(**int** i=0;i<=5;i++) {

q.add(i);

}

System.***out***.println("Values in Queue:"+q);

System.***out***.println("Size of queue:"+q.size());

System.***out***.println("Head of Queue:"+q.peek());

System.***out***.println("Remove element of Queue:"+q.remove());

System.***out***.println("Size of Queue:"+q.size());

}

}

OUTPUT:

Values in Queue:[0, 1, 2, 3, 4, 5]

Size of queue:6

Head of Queue:0

Remove element of Queue:0

Size of Queue:5

1. Perform various operations by using methods of queue.

**package** my.collections;

**import** java.util.PriorityQueue;

**import** java.util.Iterator;

**public** **class** priorityqueue {

**public** **static** **void** main(String[] args) {

PriorityQueue pq =**new** PriorityQueue();

pq.add("Rose");

pq.add("Sunflower");

pq.add("Lily");

pq.add("Lotus");

pq.add("Daffodil");

System.***out***.println("Queue is :"+pq);

System.***out***.println("remove element:"+pq.remove());

System.***out***.println("After Poll operation:"+pq.poll());

System.***out***.println("After Operations :"+pq);

System.***out***.println();

System.***out***.println("Elements After Iterating the queue.......");

Iterator itr = pq.iterator();

**while**(itr.hasNext()) {

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

}

}

}

OUTPUT:

Queue is :[Daffodil, Lily, Rose, Sunflower, Lotus]

remove element:Daffodil

After Poll operation:Lily

After Operations :[Lotus, Sunflower, Rose]

Elements After Iterating the queue.......

Lotus

Sunflower

Rose

1. Create a of HasSet add the heterogenous objects print all the objects.

**package** my.collections;

**import** java.util.HashSet;

**public** **class** hashset {

**public** **static** **void** main(String[] args) {

HashSet h=**new** HashSet();

h.add(20);

h.add(50.3);

h.add('c');

h.add("niharika");

h.add(79.54f);

h.add(null);

System.***out***.println("Values of Hash Set is : "+h);

}

}

OUTPUT:

Values of Hash Set is : [null, niharika, c, 20, 79.54, 50.3]

1. Create a of Linked HashSet add the heterogenous objects print all the objects.

**package** my.collections;

**import** java.util.LinkedHashSet;

**public** **class** linkedHashset {

**public** **static** **void** main(String[] args) {

LinkedHashSet ls=**new** LinkedHashSet();

ls.add(20);

ls.add(50.3);

ls.add('c');

ls.add("niharika");

ls.add(79.54f);

ls.add(**null**);

System.***out***.println("Values:"+ls);

}

}

Output:

Values:[20, 50.3, c, niharika, 79.54, null]

1. By using Generics create Tree Set and print the all the values.

**package** my.collections;

**import** java.util.TreeSet;

**public** **class** Tree {

**public** **static** **void** main(String[] args) {

TreeSet ts=**new** TreeSet();

ts.add("Niharika");

ts.add("Manish");

ts.add("Dev");

ts.add("Ritu");

ts.add("Varsha");

ts.add("Arun");

ts.add("Fairy");

System.***out***.println("Values of Tree is :"+ts);

}

}

OUTPUT:

Values of Tree is :[Arun, Dev, Fairy, Manish, Niharika, Ritu, Varsha]