PRACTICAL 2 – LIST INTERFACE (ADVANCED JAVA)

AIM

To study and implement List Interface in Java and demonstrate traversal using for-each loop and ListIterator.

SOFTWARE REQUIREMENTS

• Operating System: Windows / Linux
• JDK: 7 or 8
• IDE: NetBeans IDE 7.x

PART A: List using for-each loop

The List interface is a part of Java Collection Framework. It allows duplicate elements and maintains insertion order.

SOURCE CODE

```
package listdemo;

import java.util.*;

public class ListForEachDemo {
    public static void main(String[] args) {
        List<String> items = new ArrayList<String>();
        items.add("Java");
        items.add("Python");
        items.add("C++");
        items.add("Advanced Java");

        for(String item : items) {
            System.out.println(item);
        }
    }
}
```

OUTPUT

```
Java
Python
C++
Advanced Java
```

NETBEANS 7 EXECUTION STEPS

1. Open NetBeans IDE 7
2. File → New Project → Java → Java Application

3. Project Name: ListDemo
4. Uncheck 'Create Main Class'
5. Finish
6. Right-click Source Packages → New → Java Package
7. Package Name: listdemo
8. Right-click package → New → Java Class
9. Class Name: ListForEachDemo
10. Paste code and press Shift + F6

PART B: List using ListIterator (Forward & Backward)

ListIterator allows bidirectional traversal of a List.

SOURCE CODE

```
package listdemo;

import java.util.*;

public class ListIteratorDemo {
    public static void main(String[] args) {
        List<String> items = new ArrayList<String>();
        items.add("Red");
        items.add("Green");
        items.add("Blue");

        ListIterator<String> itr = items.listIterator();

        System.out.println("Forward Direction:");
        while(itr.hasNext()) {
            System.out.println(itr.next());
        }

        System.out.println("Backward Direction:");
        while(itr.hasPrevious()) {
            System.out.println(itr.previous());
        }
    }
}
```

OUTPUT

```
Forward Direction:
Red
Green
Blue
Backward Direction:
Blue
Green
Red
```

RESULT

Thus, List Interface was successfully implemented using for-each loop and ListIterator.