PROGRAM 6

```
import java.util.ArrayList;
import java.util.Iterator;
import java.util.List;
import java.util.ListIterator;
public class IteratorExample {
    public static void main(String[] args) {
        List<Integer> numbers = new ArrayList<>();
        numbers.add(1);
        numbers.add(2);
        numbers.add(3);
        numbers.add(4);
        // Using Iterator
        System.out.println("Using Iterator:");
        Iterator<Integer> iterator = numbers.iterator();
        while (iterator.hasNext()) {
            System.out.println(iterator.next());
        }
        // Using ListIterator
        System.out.println("\nUsing ListIterator (forward):");
        ListIterator<Integer> listIterator =
numbers.listIterator();
        while (listIterator.hasNext()) {
            System.out.println(listIterator.next());
        }
        // Using ListIterator in reverse
        System.out.println("\nUsing ListIterator
(backward):");
        while (listIterator.hasPrevious()) {
            System.out.println(listIterator.previous());
        }
    }
}
```

PROGRAM 7

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
public class MergeFilesAndDisplay {
    public static void main(String[] args) {
        // Specify the paths of the two input files
        String inputFile1 = "C:\Users\Admin\Desktop\input1.txt";
        String inputFile2 = "C:\Users\Admin\Desktop\input2.txt";
        // Specify the path of the output file
        String outputFile = "C:\Users\Admin\Desktop\output.txt";
        {
            // Read data from the first file
            BufferedReader reader1 = new BufferedReader(new FileReader(inputFile1));
            String data1 = "";
            String line1;
            while ((line1 = reader1.readLine()) != null) {
                data1 += line1 + "\n";
            }
            reader1.close();
            // Read data from the second file
            BufferedReader reader2 = new BufferedReader(new FileReader(inputFile2));
            String data2 = "";
            String line2;
            while ((line2 = reader2.readLine()) != null) {
                data2 += line2 + "\n";
            }
            reader2.close();
            // Merge data from both files
            String mergedData = data1 + data2;
            // Write the merged data into the output file
            BufferedWriter writer = new BufferedWriter(new FileWriter(outputFile));
            writer.write(mergedData);
            writer.close();
            System.out.println("Merged data written to " + outputFile);
            // Display the contents of the output file
            System.out.println("Contents of the merged file:");
            BufferedReader mergedReader = new BufferedReader(new
FileReader(outputFile));
            String line;
            while ((line = mergedReader.readLine()) != null) {
                System.out.println(line);
            }
            mergedReader.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```

PROGRAM 3

```
import java.util.*;
public class ArrayL
{
ArrayList<String> list=new ArrayList<String>(); //Creating
arraylist
public void arraydisplay(){
list.add("CSE");//Adding object in arraylist
list.add("ISE");
list.add("ME");
System.out.println("ArrayList element are");
System.out.println(list);
System.out.println("");
}
public void appendatend(){
System.out.println("Enter the element to append at end");
Scanner scob1=new Scanner(System.in);
String ele=scob1.next();
list.add(ele);
System.out.println(list);
System.out.println("");
}
public void insertatpos(){
System.out.println("Enter the position and element to insert");
Scanner scob1=new Scanner(System.in);
int posind=scob1.nextInt();
String ele=scob1.next();
list.add(posind,ele);
System.out.println(list);
System.out.println("");
}
public void searchele(){
System.out.println("Enter the Array element to search");
Scanner scobj=new Scanner(System.in);
String arele=scobj.next();
int in=list.indexOf(arele);
if(in==-1){
System.out.println("Element not found");
}
else{
System.out.println("Element found at "+in);
}
}
void print(){
System.out.println("Enter the starting charecter to print strings");
Scanner nip=new Scanner(System.in);
char inputc=nip.next().charAt(0);
String strc=Character.toString(inputc);
System.out.println("String starting with character "+strc);
for(int i=0;i<list.size();i++){
if(list.get(i).startsWith(strc)){
System.out.println(list.get(i));
}
}
}
public static void main(String args[]){
ArrayL obj=new ArrayL();
obj.arraydisplay();
obj.appendatend();
obj.insertatpos();
obj.searchele();
obj.print();
}
}
```

PROGRAM 8

```
import java.io.ByteArrayInputStream;
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.util.Scanner;

public class ByteArrayExample {
    public static void main(String[] args) {
        // Read a string from the user
        String userInput = getUserInput();

        // Write the string to a byte array
        byte[] byteArray = writeToByteArray(userInput);

        // Read from the byte array and display content
        readAndDisplayFromByteArray(byteArray);
    }

    private static String getUserInput() {
        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter a string: ");
        return scanner.nextLine();
    }

    private static byte[] writeToByteArray(String inputString) {
        try (ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream()) {

            byte[] bytes = inputString.getBytes();
            byteArrayOutputStream.write(bytes);

            System.out.println("String has been written to the byte array.");

            return byteArrayOutputStream.toByteArray();

        } catch (IOException e) {
            System.out.println("An error occurred while writing to the byte array: " + e.getMessage());
            return new byte[0];
        }
    }

    private static void readAndDisplayFromByteArray(byte[] byteArray) {
        try (ByteArrayInputStream byteArrayInputStream = new ByteArrayInputStream(byteArray)) {

            byte[] buffer = new byte[1024];
            int bytesRead = byteArrayInputStream.read(buffer);

            String content = new String(buffer, 0, bytesRead);

            System.out.println("Content read from the byte array: " + content);

        } catch (IOException e) {
            System.out.println("An error occurred while reading from the byte array: " + e.getMessage());}}}
```