

```

// Greeting.java - GREETING GENERATOR CLASS
//
// USAGE
//
// java Greeting namefile boysfile girlsfile
//
// namefile  file name of the file containing the list of names
// boysfile  file name of the file containing names usually given to boys
// girlsfile file name of the file containing names usually given to girls
//
// MAINTENANCE HISTORY
// DATE          PROGRAMMER AND DETAILS
// 25-09-14  MPF      Original
//
//-----

import java.io.*;
import java.util.*;

//-----

// CLASS DECLARATION

public class Greeting {

    //-----

    // GENDER ENUMERATOR VALUES

    private enum Gender {
        UNKNOWN,
        MALE,
        FEMALE,
        MIXED,
    }

    //-----

    // READ NAMES FROM FILE

    private static List<String>
    readNames (
        String      file)
    {
        List<String> names;      // Name list
        BufferedReader  reader;  // Reader
        String          line;    // Single-line data

        names = new ArrayList<String>();

        try {
            reader = new BufferedReader(new InputStreamReader(
                new FileInputStream(file)));
            while ((line = reader.readLine()) != null)
                names.add(line);
        }
    }
}

```

```

        reader.close();
    } catch (IOException ex) {
        ex.printStackTrace();
    }
    return names;
}

//-----

// MAIN LINE

public static void
main (
    String      args[]          // Arguments
{
    List<String> names;         // Names
    List<String> boys;         // Names for boys
    List<String> girls;        // Names for girls
    StringBuildersurname;     // Surname
    StringBuilderscapName;    // Capitalized name
    StringBuildersword;       // Word
    boolean      surnameFlg;   // Surname found flag
    boolean      spaceFlg;    // Space found flag
    Gender       gender;       // Gender

    if (args.length != 3) {
        System.out.println("Invalid number of paramters");
        return;
    }

    names = readNames(args[0]);
    boys = readNames(args[1]);
    girls = readNames(args[2]);

    surname = new StringBuilder();
    capName = new StringBuilder();
    word = new StringBuilder();
    for (String name : names) {

        surname.setLength(0);
        capName.setLength(0);
        word.setLength(0);
        surnameFlg = false;
        spaceFlg = true;
        gender = Gender.UNKNOWN;

        for (char c : name.toCharArray()) {
            if (c == '[')
                surnameFlg = true;
            else if (c == ']')
                surnameFlg = false;
            else if (c == ' ') {
                if (! spaceFlg) {

                    // Check gender

```

```

switch (gender) {
// Set to boy or girl
case UNKNOWN:
    if (boys.contains(word.toString()))
        gender = Gender.MALE;
    else if (girls.contains(word.toString()))
        gender = Gender.FEMALE;
    break;

// Set to mixed if girl name found
case MALE:
    if (girls.contains(word.toString()))
        gender = Gender.MIXED;
    break;

// Set to mixed if boy name found
case FEMALE:
    if (boys.contains(word.toString()))
        gender = Gender.MIXED;
    break;

// Ignore when already mixed
case MIXED:
    break;
}

word.setLength(0);
capName.append(c);
if (surnameFlg)
    surname.append(c);
}
spaceFlg = true;
}
else if (spaceFlg) {
word.append(Character.toUpperCase(c));
capName.append(Character.toUpperCase(c));
if (surnameFlg)
    surname.append(Character.toUpperCase(c));
spaceFlg = false;
}
else {
word.append(Character.toLowerCase(c));
capName.append(Character.toLowerCase(c));
if (surnameFlg)
    surname.append(Character.toLowerCase(c));
}
}

System.out.print ("To: ");
System.out.println (capName.toString());
System.out.print ("Wishing you a Merry Christmas, ");
if (surname.length() == 0)
    System.out.println (capName.toString());
else {

```

```
switch (gender) {  
  case MALE:  
    System.out.print ("Mr ");  
    System.out.println (surname.toString());  
    break;  
  case FEMALE:  
    System.out.print ("Ms ");  
    System.out.println (surname.toString());  
    break;  
  default:  
    System.out.println (capName.toString());  
    break;  
}  
}  
System.out.println ();  
}  
}
```