

```

1 package Assign_7;
2
3 import BasicIO.*;           // for IO classes
4 import static BasicIO.Formats.*; // for field formats
5 import static java.lang.Math.*; // for math constants and functions
6
7 /** This class uses a GUI to make census forms. It reads address, constructs forms,
8  * of the amount of people residing at an address, their sex, and their spoken
9  * language.
10  * @author Sawyer Fenwick st# 6005011
11  * @version 1.0 December 1 2016
12  */
13 public class Census {
14
15     // instance variables
16     private BasicForm houseDisplay;
17     private BasicForm personDisplay;
18     private ASCIIDataFile addressReader;
19     private ASCIIOutputFile outputFile;
20     int numMale = 0; // num male
21     int numFemale = 0; // num female
22     int numEng = 0; // num eng
23     int numFr = 0; // num french
24     int numOth = 0; // oth lang
25
26     /** This constructor calls the buildFormHouse and buildFormPerson methods,
27     creating the forms for each, then
28     * though a series of for loops and other methods, reads a file (the address
29     file) and populates the
30     * household field, then spawns a new individual field (if OK pressed) or
31     skips ahead to the next address
32     * (if Skip pressed). The individual field prompts the user for information on
33     a singular person, the information
34     * is read (as well as the informatoin from the household form) and written to
35     a new file (outputFile). */
36
37     public Census ( ) {
38
39         // local variables
40         int houseButton; //houseButton pressed
41         int p; // person
42         String address; // address read
43
44         //Household
45         houseDisplay = new BasicForm("OK", "Skip");
46         houseDisplay.setTitle("Household");
47
48         //Individual
49         personDisplay = new BasicForm();
50         personDisplay.setTitle("Individual");
51
52         //Address Reader
53         addressReader = new ASCIIDataFile();
54         addressReader.getFile();
55
56         //Output File
57         outputFile = new ASCIIOutputFile();
58         outputFile.getFile();
59
60         buildFormHouse();
61         buildFormPerson();
62
63         //statements
64
65         for ( ; ; ){ //start of for loop 1
66
67             //household form

```

```

63         address = addressReader.readLine();
64
65         if(addressReader.isEOF()) break; //checking for EOF
66
67         fillHouseDisplay(address);
68         houseButton = houseDisplay.accept();
69
70         //if skip button is pressed
71         if( houseButton == 1 ){
72             address = addressReader.readLine();
73             fillHouseDisplay(address);
74             houseButton = houseDisplay.accept();
75         }
76
77         p = houseDisplay.readInt("amount");
78
79         //individual form
80         for(int i = 1; i <= p; i++){ // start of for loop 2
81
82             houseDisplay.hide();
83             fillPersonDisplay(i);
84             personDisplay.accept();
85
86             int s = personDisplay.readInt("sex");
87
88             if (s == 0){
89                 numMale = numMale + 1;
90             }else{
91                 numFemale = numFemale + 1;
92             }
93
94             int l = personDisplay.readInt("lang");
95
96             if (l == 0){
97                 numEng = numEng + 1;
98             }else if (l== 1){
99                 numFr = numFr + 1;
100             }else{
101                 numOth = numOth + 1;
102             }
103
104         } // end of for loop 2
105
106         personDisplay.hide();
107         writeReport(address, p, numMale, numFemale, numEng, numFr, numOth);
108
109         reset();
110
111     } // end of for loop 1
112
113     outputFile.close();
114     addressReader.close();
115     houseDisplay.close();
116     personDisplay.close();
117
118 }; // constructor
119
120 // methods
121
122 /* This method constructs the BasicForm for the household.*/
123 private void buildFormHouse(){
124
125     houseDisplay.addTextField("address", "Address", 20);
126     houseDisplay.setEditable("address", false);
127
128     houseDisplay.addTextField("amount", "# People", 3);
129
130 } //buildFormHouse
131
132 /* This method fills the BasicForm for the household with the address read by
    C:\Users\sawye\Documents\_BrockU\COSC1P02\Assignments\Assign_7\Census.java

```

```

the addressReader, and makes
133     * sure that the number of people text field is clear.*/
134     private void fillHouseDisplay(String address){
135
136         houseDisplay.writeString("address", address);
137         houseDisplay.clear("amount");
138
139     } //fillHouseDisplay
140
141     /* This method constructs the BasicForm for the individual.*/
142     private void buildFormPerson(){
143
144         personDisplay.addTextField("person", "Person", 1);
145         personDisplay.setEditable("person", false);
146
147         personDisplay.addTextField("name", "Name", 20);
148         personDisplay.addTextField("age", "Age", 3);
149         personDisplay.addRadioButtons("sex", "Sex", false, "Male", "Female");
150         personDisplay.addRadioButtons("lang", "Language", false, "English", "French",
151         "Other");
152     } //buildFormPerson
153
154     /* This method fills the BasicForm for the individual with the person number
155     (person number is changing based on
156     * what person the census taker is currently filling out information for). It
157     makes sure that the name, age, sex,
158     * and language fields are clear.*/
159     private void fillPersonDisplay(int p){
160
161         personDisplay.writeInt("person", p);
162         personDisplay.clear("name");
163         personDisplay.clear("age");
164         personDisplay.clear("sex");
165         personDisplay.clear("lang");
166
167     } //fillPersonDisplay
168
169     /*This method resets the count of number of males, females etc etc.*/
170     private void reset(){
171
172         numMale = 0;
173         numFemale = 0;
174         numEng = 0;
175         numFr = 0;
176         numOth = 0;
177
178     } //reset
179
180     /*This method writes the report of the census data, taking the address and
181     total number of people from the
182     * house hold form, as well as the number of male, female, english speakers,
183     french speakers and other language
184     * speakers from the individual form.*/
185     private void writeReport(String address, int numPeople, int male, int female,
186     int eng, int fr, int oth){
187
188         outputFile.writeString(address);
189         outputFile.writeInt(numPeople);
190         outputFile.writeInt(male);
191         outputFile.writeInt(female);
192         outputFile.writeInt(eng);
193         outputFile.writeInt(fr);
194         outputFile.writeInt(oth);
195         outputFile.newLine();
196
197     } //writeReport
198
199     public static void main ( String[] args ) { Census c = new Census(); };

```

```
196  
197 } // Census
```