```
1  /*********************************************************************/
2
3                  /**
4                  * @author                Ehsan Kourkchi
5                  * @lecture               ICS 211 November 2013
6                  * @date                  December 9, 2013
7                  * @class type            public generic class
8                  */
9  /*********************************************************************/
10
11 import java.util.Scanner;
12 import java.util.InputMismatchException;
13 import java.io.BufferedWriter;
14 import java.io.File;
15 import java.io.FileWriter;
16 import java.io.FileOutputStream;
17 import java.io.IOException;
18 import java.io.OutputStreamWriter;
19 import java.io.Writer;
20
21 ////////////////////////////////////////////////
22 ////////////////////////////////////////////////
23 ////////////////////////////////////////////////
24
25 public class AddressBook {
26
27 ////////////////////////////////////////////////
28
29           static int add(OrderedLinkedList<String> myList) {
30
31           String name = " ", tel_number= " ";
32           String output = " ", get_key = " ";
33           Scanner scan = new Scanner(System.in);
34
35
36           System.out.printf("Enter a name to add : ");
37               try {
38                 name = scan.next();
39               } catch(InputMismatchException ex) {
40                 System.out.println("\033[1;31m Error: You did not enter
                  a valid input\033[0m\n");
41                 return 1;
42               } scan.nextLine();
43
44
45           System.out.printf("Enter telephone number for '%s\': ",
              name);
46               try {
47                 tel_number = scan.next();
48               } catch(InputMismatchException ex) {
49                 System.out.println("\033[1;31m Error: You did not enter
                  a valid input\033[0m\n");
50                 return 1;
51               } scan.nextLine();
52
53
54               output = myList.find(name);
55
56
57               if (output == null)
58               {
59                  output = myList.add(name, tel_number);
60                  System.out.printf("\n'%s' added to telephone book,
                    with number '%s'\n", name, tel_number);
61                  }
62                  else
63                  {
```

```
64                      System.out.printf("\n\033[1;31m'%s' already exists
                        with the phone number of '%s'.\033[0m\n", name,
                        output);
65                      System.out.printf("\033[1;32mDo you want to replace
                        '%s' by '%s'? (Y/N):\033[0m ", output, tel_number);
66
67                      try {
68                      get_key = scan.next();
69                      } catch(InputMismatchException ex) {
70                            System.out.println("\033[1;31m Error: You
                              did not enter a valid input\033[0m\n");
71                            System.exit(1);
72                            } scan.nextLine();
73
74                      if (get_key.compareToIgnoreCase("Y") == 0 ||
                        get_key.compareToIgnoreCase("yes") == 0)
75                      {
76                        output = myList.add(name, tel_number);
77                        System.out.printf("\nDone ... \nThe updated phone
                          number for '%s' is '%s'.\n", name, tel_number);
78                      }
79                      else
80                      {
81                        System.out.printf("\nNot changed ...\nThe phone
                          number for '%s' is '%s'.\n", name, output);
82                        return 0;
83                      }
84
85                   }
86
87                return 0;
88
89           }
90
91
92 ////////////////////////////////////////////////
93 ////////////////////////////////////////////////
94 ////////////////////////////////////////////////
95
96
97           static int save(OrderedLinkedList<String> myList) {
98
99           String output = " ";
100          Scanner scan = new Scanner(System.in);
101          String Yes_No = " ";
102          int check = 0;
103
104          do {
105
106                System.out.printf("\nEnter the output file name: ");
107                try {
108                      output = scan.next();
109                }
110                catch(InputMismatchException ex) {
111                      System.out.println("\033[1;31m Error: You did
                        not enter a valid input\033[0m\n");
112                      return 1;
113                } scan.nextLine();
114
115
116                try {
117                      File file = new File(output);
118
119                       // if file doesnt exists, then create it
120                       if (!file.exists()) {
121                          file.createNewFile();
122                          check = 0 ;
```

```
123                         }
124                         else {
125                           System.out.printf("\n'%s' already exists.
                              \nDo you want to overwrite it? (Y/N): ",
                              output);
126                           Yes_No = scan.next();
127                           if (Yes_No.compareToIgnoreCase("Y") == 0 ||
                              Yes_No.compareToIgnoreCase("yes") == 0) {
128                                file.createNewFile(); check = 0;
129                           }
130                           else {
131                              System.out.printf("'%s' was NOT
                              overwritten ...\n", output);
132                              check = 1;
133                           }
134                         }
135
136                        if (check == 0) {
137                          FileWriter fw = new
                             FileWriter(file.getAbsoluteFile());
138                          BufferedWriter bw = new BufferedWriter(fw);
139                          check = myList.write_file(bw);
140                          bw.close();
141                          System.out.println("Your address book was
                             successfully saved in \'" + output + "\'.\n");
142                        }
143                 }
144                 catch (IOException e) {
145                    e.printStackTrace();
146                    check = 1;
147                 }
148
149           }  while (check != 0);
150
151
152           return 0;
153           }
154
155
156 ////////////////////////////////////////////////
157 ////////////////////////////////////////////////
158 ////////////////////////////////////////////////
159
160           static int find(OrderedLinkedList<String> myList) {
161
162           String name = " ", tel_number= " ";
163           String output = " ";
164           Scanner scan = new Scanner(System.in);
165
166           if (myList.size() == 0) {
167           System.out.printf("\n\033[1;31mThe list is empty.\033[0m\n");
168           return 0;
169           }
170
171           System.out.printf("Enter a name to search for: ");
172
173           try {
174                 name = scan.next();
175               }
176               catch(InputMismatchException ex) {
177                 System.out.println("\033[1;31m Error: You did not enter
                  a valid input\033[0m\n");
178                 return 1;
179               } scan.nextLine();
180
181
182           output = myList.find(name);
```

```
183           if (output != null)
184                 System.out.printf("\n\033[1;32m'%s' was found, number
                  is %s\033[0m\n", name, output);
185                    else
186                      System.out.printf("\n\033[1;31m'%s' was not found.
                       \033[0m\n", name);
187
188
189           return 0;
190           }
191
192 ////////////////////////////////////////////////
193 ////////////////////////////////////////////////
194 ////////////////////////////////////////////////
195
196           static int remove(OrderedLinkedList<String> myList) {
197
198           String name = " ", tel_number= " ";
199           String output = " ", get_key= " ";
200           Scanner scan = new Scanner(System.in);
201
202           if (myList.size() == 0) {
203           System.out.printf("\n\033[1;31mThe list is empty.\033[0m\n");
204           return 0;
205           }
206
207           System.out.printf("Enter a name to remove: ");
208
209           try {
210                 name = scan.next();
211           }
212           catch(InputMismatchException ex) {
213             System.out.println("\033[1;31m Error: You did not enter a
                valid input\033[0m\n");
214             return 1;
215           } scan.nextLine();
216
217           output = myList.find(name);
218
219           if (output != null)  {
220             System.out.printf("\n\033[1;32m'%s' was found, number is
                %s\033[0m\n", name, output);
221             System.out.printf("\033[1;32mDo you want to remove it?
                (Y/N):\033[0m ");
222
223             try {
224               get_key = scan.next();
225             }
226             catch(InputMismatchException ex) {
227               System.out.println("\033[1;31m Error: You did not enter a
                  valid input\033[0m\n");
228               System.exit(1);
229             } scan.nextLine();
230
231             if (get_key.compareToIgnoreCase("Y") == 0 ||
                get_key.compareToIgnoreCase("yes") == 0) {
232               output = myList.remove(name);
233               System.out.printf("\nName: '%s' Telephone:'%s'.
                  \nsuccessfully removed ...\n", name, output);
234             }
235           }
236           else
237             System.out.printf("\n\033[1;31m'%s' was not found.
                \033[0m\n", name);
238
239           return 0;
240           }
```

```
241
242 ////////////////////////////////////////////////////
243 ////////////////////////////////////////////////////
244 ////////////////////////////////////////////////////
245
246           static void print(OrderedLinkedList<String> myList) {
247
248           myList.printList();
249
250           }
251
252
253 ////////////////////////////////////////////////////
254 ////////////////////////////////////////////////////
255 ////////////////////////////////////////////////////
256 /*
257 //   THIS IS THE MAIN METHOD
258 //   By: Ehsan Kourkchi
259 //   September 2013
260 //   Testing the database
261 */
262 ////////////////////////////////////////////////////
263 ////////////////////////////////////////////////////
264 ////////////////////////////////////////////////////
265
266
267  public static void main(String []args) {
268
269  Scanner scan = new Scanner(System.in);
270  OrderedLinkedList<String> myList = new OrderedLinkedList<String>();
271  String menu = " ";
272  int menu_index = 100;
273  int check = 1;
274
275   while ( menu_index != 0) {
276
277           if (check == 1)
278             System.out.println("\n\033[1;34mWelcome ....\nThe address
                book is empty.\033[0m ");
279
280           if (check == 0)
281             System.out.printf("\nThe address book has %d entry. \n",
                myList.size());
282
283           System.out.printf("Enter one of <add, remove, find, print,
                save, quit> : ");
284
285           try {
286             menu = scan.next();
287           }
288           catch(InputMismatchException ex) {
289             System.out.println("\033[1;31m Error: You did not enter a
                valid input\033[0m\n");
290             System.exit(1);
291           }
292
293           scan.nextLine();
294
295           if (menu.compareToIgnoreCase("add") == 0)
296             menu_index = 1 ;
297           else if (menu.compareToIgnoreCase("remove") == 0)
298             menu_index = 2 ;
299           else if (menu.compareToIgnoreCase("find") == 0)
300             menu_index = 3 ;
301           else if (menu.compareToIgnoreCase("print") == 0)
302             menu_index = 4 ;
303           else if (menu.compareToIgnoreCase("save") == 0)
```

```
304             menu_index = 5 ;
305           else if (menu.compareToIgnoreCase("quit") == 0)
306             menu_index = 0 ;
307           else {
308             menu_index = 100;
309             System.out.println("\n\033[1;31mWarning: wrong entry, try
                again ...\033[0m");
310           }
311
312           switch (menu_index) {
313
314             case 1:  add(myList);
315                 break;
316
317             case 2:  remove(myList);
318                 break;
319
320             case 3:  find(myList);
321                 break;
322
323             case 4:  print(myList);
324                 break;
325
326             case 5:  save(myList);
327                 break;
328
329           }
330
331           check =0;
332
333           if (menu_index == 0) {
334                 System.out.printf("\nAre you sure you want to exit?
                  (Y/N): ");
335
336                 try {
337                   menu = scan.next();
338                 }
339                 catch(InputMismatchException ex) {
340                   System.out.println("\033[1;31m Error: You did not
                    enter a valid input\033[0m\n");
341                   System.exit(1);
342                 }
343
344                 scan.nextLine();
345
346                 if (menu.compareToIgnoreCase("Y") == 0 ||
                  menu.compareToIgnoreCase("yes") == 0)
347                 {
348                         System.out.printf("\nDo you want to save the
                          address book? (Y/N): ");
349
350                         try {
351                           menu = scan.next();
352                         }
353                         catch(InputMismatchException ex) {
354                           System.out.println("\033[1;31m Error: You did
                            not enter a valid input\033[0m\n");
355                           System.exit(1);
356                         }
357
358                         scan.nextLine();
359
360                         if (menu.compareToIgnoreCase("Y") == 0 ||
                          menu.compareToIgnoreCase("yes") == 0) {
361                           save(myList);
362                         }
363                         else if (menu.compareToIgnoreCase("N") == 0 ||
```

```
363                         menu.compareToIgnoreCase("No") == 0) {
364                         }
365                         else
366                           menu_index = 100;
367
368                 }
369                 else
370                   menu_index = 100;
371
372           }
373
374   }
375
376   System.out.println("\n\033[1;34m  Quit, closing the address book.\n
      Goodbye ...\n\033[0m\n");
377
378  }
379
380
381 ////////////////////////////////////////////////////
382
383 }  // end of the class
384 ////////////////////////////////////////////////////
385 ////////////////////////////////////////////////////
386 ////////////////////////////////////////////////////
```