```java
package src;

import java.io.File;
import java.io.FileNotFoundException;
import java.io.IOException;
import java.util.Arrays;
import java.util.InputMismatchException;
import java.util.Scanner;
import javax.swing.JOptionPane;


public class ProjectMain {
    static int floor;
    public static People[] queue = new People[6];

    public static void main(String[] args) throws IOException {
        System.out.println("Welcome to the prototype of the Securavator. The worlds most intelligent and secure elevator. Click here for instructions how this revolutionary technology operates");
        JOptionPane.showMessageDialog(null, "Welcome to the prototype of the Securavator. The worlds most intelligent and secure elevator. Ready to learn how this revolutionary technology operates?");
        JOptionPane.showMessageDialog(null, "Step 1: Scan your badge by inputting numbers onto the touchpad");
        JOptionPane.showMessageDialog(null, "Step 2: If the system verifies your ID, step forward into the queue");
        JOptionPane.showMessageDialog(null, "Step 3: When prompted by the system, enter the elevator and enjoy your ride!");
        JOptionPane.showMessageDialog(null, "Click here to begin. If at any time there is an emergency, please enter 911 into the keypad to stop the elevator.");

        // create an array and fill it with employee objects
```

```
25          People[] employees = new People[50];
26          // read in all the employees from file
27          try {
28              File fileObj = new File("C:\\Users\\cbcha
    \\Downloads\\employees.txt");
29              Scanner scanObj = new Scanner(fileObj);
30              for (int i = 0; i < 50; i++) {
31                  if (scanObj.hasNextLine()) {
32                      String data = scanObj.nextLine();
33                      int idTest;
34                      int depTest;
35                      String nameTest;
36
37                      String idString = data.substring(
    0, 3);
38                      String depString = data.substring
    (data.length() - 1, data.length());
39
40                      idTest = Integer.parseInt(
    idString);
41                      depTest = Integer.parseInt(
    depString);
42                      nameTest = data.substring(3, data
    .length() - 1);
43
44                      People entry = new People(idTest
    , depTest, nameTest);
45                      employees[i] = entry;
46                  }
47              }
48          } catch (FileNotFoundException e) {
49              System.out.println("The file you entered
    in incorrect, please enter a new file:");
50              Scanner keyboard = new Scanner(System.in
    );
51              String newFile = keyboard.nextLine();
52              File fileObj = new File(newFile);
53              Scanner scanObj = new Scanner(fileObj);
54              for (int i = 0; i < 50; i++) {
55                  if (scanObj.hasNextLine()) {
56                      String data = scanObj.nextLine();
```

```
57                      int idTest;
58                      int depTest;
59                      String nameTest;
60
61                      String idString = data.substring(
   0, 3);
62                      String depString = data.substring
   (data.length() - 1, data.length());
63
64                      idTest = Integer.parseInt(
   idString);
65                      depTest = Integer.parseInt(
   depString);
66                      nameTest = data.substring(3, data
   .length() - 1);
67
68                      People entry = new People(idTest
   , depTest, nameTest);
69                      employees[i] = entry;
70                  }
71              }
72          }
73          // get which employee is waiting next in the
   queue and fill up the queue with 6 employees
74          Scanner keyboard = new Scanner(System.in);
75          int counter = 0;
76          while (counter < 6) {
77              try {
78                  System.out.println("Please scan your
   ID number: ");
79                  int scan = keyboard.nextInt();
80                  if(scan == 911){
81                      JOptionPane.showMessageDialog(
   null, "Emergency Button activated and elevator is now
    stopped. Press okay to continue the ride.");
82                  }
83                  for (int i = 0; i < 50; i++) {
84                       if (scan == employees[i].
   getIDnum()) {
85                          System.out.println("This
   VERIFIED employee is going to floor " + employees[i].
```

```
 85 getFloorNum());
 86                         floor = employees[i].
    getFloorNum();
 87                         // add this employee to the
    queue
 88                         queue[counter] = new People(
    employees[i].getIDnum(), employees[i].getFloorNum
    (), employees[i].geteName());
 89                         counter++;
 90                     }
 91                 }
 92             } catch (InputMismatchException e) {
 93                 System.out.println("Input error,
    please input an integer");
 94                 System.out.println("Please scan your
     ID number: ");
 95                 int scan = keyboard.nextInt();
 96                 for (int i = 0; i < 50; i++) {
 97                     if (scan == employees[i].
    getIDnum()) {
 98                         System.out.println("This
    VERIFIED employee is going to floor " + employees[i
    ].getFloorNum());
 99                         floor = employees[i].
    getFloorNum();
100                         // add this employee to the
    queue
101                         queue[counter] = new People(
    employees[i].getIDnum(), employees[i].getFloorNum
    (), employees[i].geteName());
102                         counter++;
103                     }
104                 }
105             }
106         }
107         JOptionPane.showMessageDialog(null, "The
    queue is now full. ");
108         System.out.println("The following employees
    are authorized to ride the elevator: ");
109             System.out.println("One: " + queue[0].
    geteName());
```

```
110             System.out.println("Two: " + queue[1].
    geteName());
111             System.out.println("Three: " + queue[2].
    geteName());
112             System.out.println("Four: " + queue[3].
    geteName());
113             System.out.println("Five: " + queue[4].
    geteName());
114             System.out.println("Six: " + queue[5].
    geteName());
115         // now there are 6 employees ready to be
    brought to their floor
116         JOptionPane.showMessageDialog(null, "All
    approved employees may enter the elevator. ");
117
118         int one = queue[0].getFloorNum();
119         int two = queue[1].getFloorNum();
120         int three = queue[2].getFloorNum();
121         int four = queue[3].getFloorNum();
122         int five = queue[4].getFloorNum();
123         int six = queue[5].getFloorNum();
124         int[] numbers = {one, two, three, four, five
    , six};
125         // sort the floors in ascending order and
    drop off each employee
126         Arrays.sort(numbers);
127         for (int i = 0; i < 6; i++) {
128             // if two employees are getting off at
    the same floor, only create one ride dropping off
    mulitple employees
129             if(i<5 && numbers[i] == numbers[i+1]){
130                 Elevator ride = new Elevator(numbers
    [i]);
131                 System.out.println(ride);
132                 i++;
133             }else{
134                 Elevator ride = new Elevator(numbers
    [i]);
135                 System.out.println(ride);
136             }
137             if (i == 0) {
```

```
138                 JOptionPane.showMessageDialog(null,
    "The ride has terminated and all employees have been
     dropped off. Thank you for riding the Securavator "
    );
139             }
140         }
141     }
142
143
144 }
145
146
147
148
```