```
 1: // J.J. Collins
 2: // CS4125 / CS4135 / CS5721
 3:
 4: // Name: prog1_d-interfaces.java
 5: // Intent: (a) move towards a realisation of the importance of "Program to
         Interfaces, NOT Implementation"
 6: //         (b) a person has an address
 7: // Note: replaces inheritance with association.
 8: // Introduced two new operations into class person in lines 30 and 31 due
 9: // to private visibility for attribute a. However, additional invocations
10: // required to set and display tel. no. - not as efficient. Illustrates
11: // performance versus information hiding (supported with indirection) conflict
12: // that often arises in distributed applications.
13:
14: import java.util.*;
15:
16: /*********************************************************/
17: class address
18: {
19:    // attributes
20:    private int tel;
21:
22:    // operations / methods
23:    public void set_tel_no(int number) {tel = number;}
24:    public void display_tel_no() { System.out.println(" Telephone number is: " + tel)
          ;}
25: };
26: /*********************************************************/
27: // model association with an attribute, a person has an address
28:
29: class person
30: {
31:      private address a = new address();
32:      private int dob;
33:
34:      public void set_tel(int t) { a.set_tel_no(t); }
35:      public void display_tel() {a.display_tel_no();}
36:      public void set_dob(int date) { dob = date;}
37:      public void display_dob() { System.out.println(" Date of birth is: " +  dob);}
38: }
39: /*********************************************************/
40: class client {
41:
42:    public static void main(String args[]) {
43:
44:      // create p, an object that is an instance of class person
45:      person p = new person();
46:
47:      // invoke some operations on p
48:      p.set_tel(12345);
49:      p.set_dob(121212);
50:      p.display_tel();
51:      p.display_dob();
52:    }
53: }
54: /*********************************************************/
55:
```