

```
1 import java.time.LocalDate;
2 import java.util.ArrayList;
3 import java.util.List;
4 import java.util.Scanner;
5
6 class Contact {
7     private String name;
8     private String company;
9     private String title;
10    private String mobile;
11    private String alternateMobile;
12    private String email;
13    private LocalDate dateCreated;
14
15    public Contact(String name, String company,
16        String title, String mobile, String alternateMobile,
17        String email, LocalDate dateCreated) {
18        this.name = name;
19        this.company = company;
20        this.title = title;
21        this.mobile = mobile;
22        this.alternateMobile = alternateMobile;
23        this.email = email;
24        this.dateCreated = dateCreated;
25    }
26
27    public String getName() {
28        return name;
29    }
30
31    public String getCompany() {
32        return company;
33    }
34
35    public String getTitle() {
36        return title;
37    }
38
39    public String getMobile() {
40        return mobile;
41    }
42}
```

```
40
41     public String getAlternateMobile() {
42         return alternateMobile;
43     }
44
45     public String getEmail() {
46         return email;
47     }
48
49     public LocalDate getDateCreated() {
50         return dateCreated;
51     }
52
53     @Override
54     public String toString() {
55         return String.format("%-15s %-15s %-20s %-15s
56                               %-20s %-15s %s",
57                               name, company, title, mobile,
58                               alternateMobile, email, dateCreated);
59     }
60 }
61
62 class ContactList {
63     private List<Contact> contacts;
64
65     public ContactList() {
66         contacts = new ArrayList<>();
67     }
68
69     public void addContact(Contact contact) {
70         contacts.add(contact);
71     }
72
73     public List<Contact> searchByName(List<String>
74     names) {
75         List<Contact> results = new ArrayList<>();
76         for (Contact contact : contacts) {
77             if (names.contains(contact.getName())) {
78                 results.add(contact);
79             }
80         }
81     }
82 }
```

```
78         return results;
79     }
80
81     public List<Contact> searchByDateCreated(
        LocalDate date) {
82         List<Contact> results = new ArrayList<>();
83         for (Contact contact : contacts) {
84             if (contact.getDateCreated().equals(date
85             )) {
86                 results.add(contact);
87             }
88         }
89         return results;
90     }
91
92     public List<Contact> searchByEmailDomain(String
93     domain) {
94         List<Contact> results = new ArrayList<>();
95         for (Contact contact : contacts) {
96             if (contact.getEmail().endsWith(domain
97             )) {
98                 results.add(contact);
99             }
100         }
101
102     public class Main {
103         public static void main(String[] args) {
104             ContactList contactList = new ContactList();
105             Scanner scanner = new Scanner(System.in);
106
107             System.out.print("Enter the number of
108             contact details: ");
109             int numContacts = Integer.parseInt(scanner.
110             nextLine());
111
112             for (int i = 1; i <= numContacts; i++) {
113                 System.out.print("Enter contact details
114                 for contact " + i + ": ");
```

```

112         String[] contactDetails = scanner.
            nextLine().split(",");
113
114         String name = contactDetails[0];
115         String company = contactDetails[1];
116         String title = contactDetails[2];
117         String mobile = contactDetails[3];
118         String alternateMobile = contactDetails[
119             4];
119         String email = contactDetails[5];
120         LocalDate dateCreated = LocalDate.parse(
            contactDetails[6]);
121
122         Contact contact = new Contact(name,
            company, title, mobile, alternateMobile, email,
            dateCreated);
123         contactList.addContact(contact);
124     }
125
126     while (true) {
127         System.out.println("\nEnter a search
            type:");
128         System.out.println("1. Name");
129         System.out.println("2. Date created");
130         System.out.println("3. Email domain");
131         System.out.println("4. Exit");
132         System.out.print("Enter your choice: ");
133         int searchType = Integer.parseInt(
            scanner.nextLine());
134
135         switch (searchType) {
136             case 1:
137                 System.out.print("Enter the
                    names: ");
138                 String[] nameInputs = scanner.
                    nextLine().split(",");
139                 List<String> nameList = new
                    ArrayList<>();
140                 for (String name : nameInputs) {
141                     nameList.add(name.trim());
142                 }

```

```

143         List<Contact> nameResults =
        contactList.searchByName(nameList);
144         printContactList(nameResults);
145         break;
146         case 2:
147             System.out.print("Enter the date
        created (yyyy-MM-dd): ");
148             LocalDate date = LocalDate.parse
        (scanner.nextLine());
149             List<Contact> dateResults =
        contactList.searchByDateCreated(date);
150             printContactList(dateResults);
151             break;
152             case 3:
153                 System.out.print("Enter the
        email domain: ");
154                 String domain = scanner.nextLine
        ().trim();
155                 List<Contact> domainResults =
        contactList.searchByEmailDomain(domain);
156                 printContactList(domainResults);
157                 break;
158             case 4:
159                 System.out.println("Exiting..."
        );
160                 System.exit(0);
161             default:
162                 System.out.println("Invalid
        search type. Please try again.");
163                 break;
164         }
165     }
166 }
167
168     private static void printContactList(List<
        Contact> contacts) {
169         System.out.format("%-15s %-15s %-20s %-15s
        %-20s %-15s %s\n",
170             "Name", "Company", "Title", "Mobile"
        , "Alternate Mobile", "Email", "Date Created");
171         for (Contact contact : contacts) {

```

```
172         System.out.println(contact);  
173     }  
174 }  
175 }  
176
```