```java
package Exercise;

import java.util.Arrays;
import java.util.Objects;

class Address{
    String line1;
    String line2;
    String line3;
    char[] city;
    char[] state;
    String pin;

    public Address(String parameter){
        String[] stringArray = parameter.split("\\$");

        this.line1 = stringArray[0];
        this.line2 = stringArray[1];
        this.line3 = stringArray[2];
        this.city = stringArray[3].toCharArray();
        this.state = stringArray[4].toCharArray();
        this.pin = stringArray[5];
    }

    void print(){
        System.out.println(this.line1 + " " + this.line2+ " " + this.line3+ " " + Arrays.toString(this.city)
                + " " + Arrays.toString(this.state) + " " + this.pin);
    }
}

class AddressList{
    public static int countAddressWithCity(Address[] addressList, String city){
        int result = 0;
        for (int i = 0; i < addressList.length; i++) {
            if (String.valueOf(addressList[i].city
```

```
35 ).equals(city)){
36                     result++;
37                 }
38             }
39             if (addressList.length == 0 || city.length
   () == 0){
40                 return -1;
41             }
42             return result;
43         }
44 
45         public static int countAddressWithPin(Address
   [] addressList, String strP){
46             int result = 0;
47             for (int i = 0; i < addressList.length; i
   ++) {
48                 if (Objects.equals(addressList[i].pin,
   strP)){
49                     result++;
50                 }
51             }
52             if (addressList.length == 0 || strP.length
   () == 0){
53                 return -1;
54             }
55             return result;
56         }
57 
58         public static Address[] getAddressWithCity(
   Address[] addressList, String city){
59             Address[] resultArray = new Address[
   addressList.length];
60             int resultCount = 0;
61                 for (int i = 0; i < addressList.length
   ; i++) {
62                     String city1 = new String(
   addressList[i].city);
63                     if (city1.equals(city)){
64                         resultArray[resultCount++] =
   addressList[i];
65                     }
```

```
66              }
67              if (resultCount == 0 || addressList.
   length == 0 || city.length() == 0){
68                  return null;
69              }
70              return resultArray;
71      }
72
73      public static Address[] getAddressWithPin(
   Address[] addressList, String strP){
74          Address[] resultArray = new Address[
   addressList.length];
75          int resultCount = 0;
76          for (int i = 0; i < addressList.length; i
   ++) {
77              if (addressList[i].pin.equals(strP)){
78                  resultArray[resultCount++] =
   addressList[i];
79              }
80          }
81          if (resultCount == 0 || addressList.length
    == 0 || strP.length() == 0){
82              return null;
83          }
84          return resultArray;
85      }
86  }
87
88
89
90  public class Test {
91      public static void main(String[] args) {
92          Address x = new Address("
   line1$line2$line3$city$state$pin");
93          Address y = new Address("
   line1$line2$line3$city$state$pin");
94          Address[] addressList = {x,y};
95
96          System.out.println(AddressList.
   countAddressWithCity(addressList, "city"));
97          System.out.println(AddressList.
```

```
 97 countAddressWithPin(addressList,"pin"));
 98
 99         System.out.println("city");
100         Address[] getAddressByCity = AddressList.
    getAddressWithCity(addressList, "city");
101         for (int i = 0; i < getAddressByCity.
    length; i++) {
102             getAddressByCity[i].print();
103         }
104
105         Address[] getAddressByPin = AddressList.
    getAddressWithPin(addressList, "pin");
106         for (int i = 0; i < getAddressByPin.length
    ; i++) {
107             getAddressByPin[i].print();
108         }
109
110     }
111 }
112
```

```java
package Exercise;

//Exercie 3.1
class RetailStore{
    private int[] itemId;
    private double[] price;
    private String itemName[];
    public RetailStore() {
        itemId = new int[] { 1001, 1002, 1003, 1004, 1005 };
        price = new double[] { 950.00, 750.00, 450.00, 350.00, 250.00 }; itemName = new String[] {
                "Yonex Tennis Racket-950","Yonex Badminton Racket-750",
                "Silvers Badminton Racket-450","Cosco Badminton shuttle-350",
                "Cosco Tennis Racket-250" };
    }

    protected double computePrice(int value) {
        for (int i = 0; i<price.length; ++i){
            if (itemId[i] == value) {
                return price[i];
            }
        }
        return price[value];
    }

    protected String fetchDescription(int value) {
        for (int i = 0; i<price.length; ++i) {
            if (itemId[i] == value) {
                return itemName[i];
            }
        }
        return null;
    }
}

public class RetailStoreExample extends RetailStore{
    public static void main(String[] args) {
```

```
37          int index;
38          RetailStore retailOne = new RetailStore();
39          String description = retailOne.
   fetchDescription(1004);
40          String StringArray[];
41          StringArray = description.split("\\s");
42          String type = StringArray[2];
43          index = type.indexOf('-');
44          String stringFromSubstring = type.substring
   (index + 1);
45          String stringFromDouble = Double.toString(
   new RetailStore().computePrice(1004));
46
47          System.out.println(stringFromDouble ==
   stringFromDouble);
48          System.out.println(stringFromDouble.equals(
   stringFromDouble));
49      }
50  }
```