

~\OneDrive - VietNam National University - HCM INTERNATIONAL UNIVERSITY\Desktop\DSA\DSA LAB NEW\Lab 2  
Simple sorting\ITITSB22029\_DoMinhDuy\_Lab2\Problem 5\ObjectSortApp.java

```

1  class Person {
2      private String lastName;
3      private String firstName;
4      private int age;
5      private double grade; // new field to store grade
6      // -----
7
8      public Person(String last, String first, int a, double g) { // constructor
9          lastName = last;
10         firstName = first;
11         age = a;
12         grade = g;
13     }
14
15     // -----
16     public void displayPerson() {
17         System.out.print("    Last name: " + lastName);
18         System.out.print(", First name: " + firstName);
19         System.out.print(", Age: " + age);
20         System.out.println(", Grade: " + grade);
21     }
22
23     // -----
24     public String getLast() {
25         return lastName;
26     }
27
28     public String getFirst() {
29         return firstName;
30     }
31
32     public int getAge() {
33         return age;
34     }
35
36     public double getGrade() {
37         return grade;
38     }
39 } // end class Person
40
41 class ArrayInOb {
42     private Person[] a; // reference to array a
43     private int nElems; // number of data items
44     // -----
45
46     public ArrayInOb(int max) { // constructor
47         a = new Person[max]; // create the array

```

```
48     nElems = 0; // no items yet
49 }
50
51 // -----
52 public void insert(String last, String first, int age, double grade) {
53     a[nElems] = new Person(last, first, age, grade);
54     nElems++; // increment size
55 }
56
57 // -----
58 public void display() { // displays array contents
59     for (int j = 0; j < nElems; j++) { // for each element,
60         a[j].displayPerson(); // display it
61     }
62 }
63
64 // -----
65 public void sortByLastName() { // Insertion sort by last name
66     int in, out;
67     for (out = 1; out < nElems; out++) {
68         Person temp = a[out];
69         in = out;
70         while (in > 0 && a[in - 1].getLast().compareTo(temp.getLast()) > 0) {
71             a[in] = a[in - 1];
72             --in;
73         }
74         a[in] = temp;
75     }
76 }
77
78 // -----
79 public void sortByFirstName() { // Insertion sort by first name
80     int in, out;
81     for (out = 1; out < nElems; out++) {
82         Person temp = a[out];
83         in = out;
84         while (in > 0 && a[in - 1].getFirst().compareTo(temp.getFirst()) > 0) {
85             a[in] = a[in - 1];
86             --in;
87         }
88         a[in] = temp;
89     }
90 }
91
92 // -----
93 public void sortByAge() { // Insertion sort by age
94     int in, out;
95     for (out = 1; out < nElems; out++) {
96         Person temp = a[out];
97         in = out;
```

```
98         while (in > 0 && a[in - 1].getAge() > temp.getAge()) {
99             a[in] = a[in - 1];
100             --in;
101         }
102         a[in] = temp;
103     }
104 }
105 }
106
107 class ObjectSortApp {
108     public static void main(String[] args) {
109         int maxSize = 10; // array size
110         ArrayInOb arr; // reference to array
111         arr = new ArrayInOb(maxSize); // create the array
112
113         // Inserting 10 people with last name, first name, age, and grade
114         arr.insert("Evans", "Patty", 24, 3.6);
115         arr.insert("Smith", "Doc", 59, 2.9);
116         arr.insert("Smith", "Lorraine", 37, 3.7);
117         arr.insert("Smith", "Paul", 37, 3.2);
118         arr.insert("Yee", "Tom", 43, 3.1);
119         arr.insert("Hashimoto", "Sato", 21, 3.9);
120         arr.insert("Stimson", "Henry", 29, 2.8);
121         arr.insert("Velasquez", "Jose", 72, 3.5);
122         arr.insert("Vang", "Minh", 22, 3.3);
123         arr.insert("Creswell", "Lucinda", 18, 3.8);
124
125         System.out.println("Before sorting:");
126         arr.display(); // display items
127
128         // Sort by last name
129         arr.sortByLastName();
130         System.out.println("\nAfter sorting by last name:");
131         arr.display();
132
133         // Sort by first name
134         arr.sortByFirstName();
135         System.out.println("\nAfter sorting by first name:");
136         arr.display();
137
138         // Sort by age
139         arr.sortByAge();
140         System.out.println("\nAfter sorting by age:");
141         arr.display();
142     }
143 }
144
```