

1) Get a string + Add string to arrayList

Client program-

```
20 public static void main(String[] args) {
21
22     ArrayList<String> nameList = new ArrayList<String>(); //Create arrayList of String
23     Scanner keyboard = new Scanner(System.in);
24     String name;
25
26     do {
27         System.out.println("Enter a name: ");
28         name = keyboard.nextLine(); //Get string from client
29
30         if(!name.equalsIgnoreCase("quit") && !name.equalsIgnoreCase("")) {
31             nameList.add(name); //Add string to arrayList of string
32         }
33
34     } while(!name.equalsIgnoreCase("quit") && !name.equalsIgnoreCase(""));
35
36
37
38     for(String person : nameList) {
39         System.out.println(person);
40     }
41
42
43 }
```

2) Get arrayList of strings + Determine first full name and last full name sorted in alphabetical order

Client program (method)-

```
37     fName = nameList.get(0);
38     lName = nameList.get(0);
39
40     for(String person : nameList) {
41         newFname = person;
42
43         int fnameCompared = fName.compareTo(newFname); //Return either fName is before (< 0) OR after (> 0)
44
45
46         if(fnameCompared < 0) { //fName is smaller than newFname. Therefore newFname is larger
47             lName = newFname; //Think what happens to newFname
48         }
49
50
51         if(fnameCompared > 0) { //fName is larger than newFname. Therefore newFname is smaller
52             fName = newFname; //Think what happens to newFname
53         }
54
55
56     }
57
58     System.out.printf("The first name in the list in alphabetical order is: %s \n", fName);
59     System.out.printf("The last name in the list in alphabetical order is: %s \n", lName);
60 }
```

### 3) Get arrayList of strings + Sort by alphabetical order (ascending order) through Selection sort

How did I do this? I copied and pasted the selection sort template. And made changes from there. As you can see it's very similar

```
39
40 for (int i = 0; i < nameList.size() - 1 - 1; i++) { //Selection Sort ArrayList Ascending
41
42     int indexOfUnsortedSmallest = i;
43
44     for (int j = i + 1; j < nameList.size(); j++) {
45
46         String unsortedSmallestName = nameList.get(indexOfUnsortedSmallest);
47         String nextName = nameList.get(j);
48
49         int fnameCompared = unsortedSmallestName.compareTo(nextName); //Return either nextName is before ( < 0) OR after ( > 0)
50
51         if (fnameCompared > 0) { //unsortedSmallestName is larger than nextName. Therefore nextName is smaller
52             indexOfUnsortedSmallest = j;
53         }
54         //if (smallestFName < 0) for descending
55
56     }
57
58     String temp = nameList.get(indexOfUnsortedSmallest); //Gets the value of unsortedSmallestNumber
59     String temp2 = nameList.get(i); //NOTE: Required for my array
60     nameList.set(indexOfUnsortedSmallest, temp2);
61
62     nameList.set(i, temp);
63 }
64
```

Client program (method)-