

Credit
3110
Assignment
E4Friends

How has your program changed from planning to coding to now? Please explain?

```
32     int choice = 0;
33
34
35     while (choice != 6)
36     {
37         displayMenu();
38         choice = getChoice(input);
39
40         if (choice == 1)
41         {
42             addFriend(input, database);
43         }
44         else if (choice == 2)
45         {
46             database.sortByLastName();
47             database.display();
48         }
49         else if (choice == 3)
50         {
51             database.sortByFirstName();
52             database.display();
53         }
54         else if (choice == 4)
55         {
56             findFriend(input, database);
57         }
58         else if (choice == 5)
59         {
60             deleteFriend(input, database);
61         }
62         else if (choice == 6)
63         {
64             save(database);
65         }
66     }
67
68     input.close();
69 }
70
71 // shows menu
72 private static void displayMenu()
73 {
74     System.out.println();
75     System.out.println("1. add a friend");
76     System.out.println("2. display friends by last name");
77     System.out.println("3. display friends by first name");
78     System.out.println("4. find a friend");
79     System.out.println("5. delete a friend");
80     System.out.println("6. quit");
81 }
82
```

```

32
33     int choice = 0;
34
35     while (choice != 6)
36     {
37         displayMenu();
38         choice = getChoice(input);
39
40         if (choice == 1)
41         {
42             addFriend(input, database);
43         }
44         else if (choice == 2)
45         {
46             database.sortByLastName();
47             database.display();
48         }
49         else if (choice == 3)
50         {
51             database.sortByFirstName();
52             database.display();
53         }
54         else if (choice == 4)
55         {
56             findFriend(input, database);
57         }
58         else if (choice == 5)
59         {
60             deleteFriend(input, database);
61         }
62         else if (choice == 6)
63         {
64             save(database);
65         }
66     }
67
68     input.close();
69 }
70
71 // shows menu
72 private static void displayMenu()
73 {
74     System.out.println();
75     System.out.println("1. add a friend");
76     System.out.println("2. display friends by last name");
77     System.out.println("3. display friends by first name");
78     System.out.println("4. find a friend");
79     System.out.println("5. delete a friend");
80     System.out.println("6. quit");
81 }
82

```

```

1      System.out.print("enter first name: ");
2      String first = input.nextLine();
3
4      System.out.print("enter last name: ");
5      String last = input.nextLine();
6
7      Friend f = new Friend(first, last);
8      database.addFriend(f);
9  }
10
11  // finds friend
12  private static void findFriend(Scanner input, FriendsDatabase database)
13  {
14      System.out.print("enter first name: ");
15      String first = input.nextLine();
16
17      System.out.print("enter last name: ");
18      String last = input.nextLine();
19
20      database.findFriend(first, last);
21  }
22
23  // deletes friend
24  private static void deleteFriend(Scanner input, FriendsDatabase database)
25  {
26      System.out.print("enter first name: ");
27      String first = input.nextLine();
28
29      System.out.print("enter last name: ");
30      String last = input.nextLine();
31
32      database.deleteFriend(first, last);
33  }
34
35  // saves file
36  private static void save(FriendsDatabase database)
37  {
38      try
39      {
40          database.saveToFile();
41      }
42      catch (IOException e)
43      {
44          System.out.println("error saving file");
45      }
46  }
47  }

```

```

32
33     int choice = 0;
34
35     while (choice != 6)
36     {
37         displayMenu();
38         choice = getChoice(input);
39
40         if (choice == 1)
41         {
42             addFriend(input, database);
43         }
44         else if (choice == 2)
45         {
46             database.sortByLastName();
47             database.display();
48         }
49         else if (choice == 3)
50         {
51             database.sortByFirstName();
52             database.display();
53         }
54         else if (choice == 4)
55         {
56             findFriend(input, database);
57         }
58         else if (choice == 5)
59         {
60             deleteFriend(input, database);
61         }
62         else if (choice == 6)
63         {
64             save(database);
65         }
66     }
67
68     input.close();
69 }
70
71 // shows menu
72 private static void displayMenu()
73 {
74     System.out.println();
75     System.out.println("1. add a friend");
76     System.out.println("2. display friends by last name");
77     System.out.println("3. display friends by first name");
78     System.out.println("4. find a friend");
79     System.out.println("5. delete a friend");
80     System.out.println("6. quit");
81 }
82

```

```
Purpose: store friend name
Author: Bilal Hajar
School: CHHS
Course: Computer Programming 30
*/

import java.io.Serializable;

public class Friend implements Comparable<Friend>, Serializable
{
    private String firstName;
    private String lastName;

    public Friend(String first, String last)
    {
        firstName = first;
        lastName = last;
    }

    public String getFirstName()
    {
        return firstName;
    }

    public String getLastName()
    {
        return lastName;
    }

    // compares by last name
    public int compareTo(Friend other)
    {
        return lastName.compareToIgnoreCase(other.lastName);
    }

    public String toString()
    {
        return firstName + " " + lastName;
    }
}
```

```

05
06 // adds friend
07 private static void addFriend(Scanner input, FriendsDatabase database)
08 {
09     System.out.print("enter first name: ");
10     String first = input.nextLine();
11
12     System.out.print("enter last name: ");
13     String last = input.nextLine();
14
15     Friend f = new Friend(first, last);
16     database.addFriend(f);
17 }
18
19 // finds friend
20 private static void findFriend(Scanner input, FriendsDatabase database)
21 {
22     System.out.print("enter first name: ");
23     String first = input.nextLine();
24
25     System.out.print("enter last name: ");
26     String last = input.nextLine();
27
28     database.findFriend(first, last);
29 }
30
31 // deletes friend
32 private static void deleteFriend(Scanner input, FriendsDatabase database)
33 {
34     System.out.print("enter first name: ");
35     String first = input.nextLine();
36
37     System.out.print("enter last name: ");
38     String last = input.nextLine();
39
40     database.deleteFriend(first, last);
41 }
42
43 // saves file
44 private static void save(FriendsDatabase database)
45 {
46     try
47     {
48         database.saveToFile();
49     }
50     catch (IOException e)
51     {
52         System.out.println("error saving file");
53     }
54 }

```

```

23
24     try
25     {
26         database.loadFromFile();
27     }
28     catch (Exception e)
29     {
30         // file not found
31     }
32
33     int choice = 0;
34
35     while (choice != 6)
36     {
37         displayMenu();
38         choice = getChoice(input);
39
40         if (choice == 1)
41         {
42             addFriend(input, database);
43         }
44         else if (choice == 2)
45         {
46             database.sortByLastName();
47             database.display();
48         }
49         else if (choice == 3)
50         {
51             database.sortByFirstName();
52             database.display();
53         }
54         else if (choice == 4)
55         {
56             findFriend(input, database);
57         }
58         else if (choice == 5)
59         {
60             deleteFriend(input, database);
61         }
62         else if (choice == 6)
63         {
64             save(database);
65         }
66     }
67
68     input.close();
69 }
70
71 // shows menu
72 private static void displayMenu()
73 {
74     System.out.println();
75     System.out.println("1. add a friend");

```

```

3 // ~
4 Program: FriendsTester.java
5 Last Date of this Revision: January 29, 2026
6
7 Purpose: run friends database
8 Author: Bilal Hajar
9 School: CHHS
10 Course: Computer Programming 30
11 */
12
13 import java.util.Scanner;
14 import java.io.IOException;
15
16 public class FriendsTester
17 {
18
19     public static void main(String[] args)
20     {
21         Scanner input = new Scanner(System.in);
22         FriendsDatabase database = new FriendsDatabase();
23
24         try
25         {
26             database.loadFromFile();
27         }
28         catch (Exception e)
29         {
30             // file not found
31         }
32
33         int choice = 0;
34
35         while (choice != 6)
36         {
37             displayMenu();
38             choice = getChoice(input);
39
40             if (choice == 1)
41             {
42                 addFriend(input, database);
43             }
44             else if (choice == 2)
45             {

```