

Credit
3110
Assignment
E4Friends

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

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

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

```
        System.out.print("enter first name: ");
        String first = input.nextLine();

        System.out.print("enter last name: ");
        String last = input.nextLine();

        Friend f = new Friend(first, last);
        database.addFriend(f);
    }

    // finds friend
    private static void findFriend(Scanner input, FriendsDatabase database)
    {
        System.out.print("enter first name: ");
        String first = input.nextLine();

        System.out.print("enter last name: ");
        String last = input.nextLine();

        database.findFriend(first, last);
    }

    // deletes friend
    private static void deleteFriend(Scanner input, FriendsDatabase database)
    {
        System.out.print("enter first name: ");
        String first = input.nextLine();

        System.out.print("enter last name: ");
        String last = input.nextLine();

        database.deleteFriend(first, last);
    }

    // saves file
    private static void save(FriendsDatabase database)
    {
        try
        {
            database.saveToFile();
        }
        catch (IOException e)
        {
            System.out.println("error saving file");
        }
    }
}
```

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

```
Purpose: store friend name
Author: Bilal Hajac
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     System.out.println();
74     System.out.println("1. add a friend");
75 }
```

```
50 /*  
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  
130 import java.util.Scanner;  
14 import java.io.IOException;  
15  
16 public class FriendsTester  
17 {  
18  
190     public static void main(String[] args)  
20     {  
21         Scanner input = new Scanner(System.in);  
22         FriendsDatabase database = new FriendsDatabase();  
23  
240         try  
25         {  
26             database.loadFromFile();  
27         }  
280         catch (Exception e)  
29         {  
30             // file not found  
31         }  
32  
33         int choice = 0;  
34  
350         while (choice != 6)  
36         {  
37             displayMenu();  
38             choice = getChoice(input);  
39  
400             if (choice == 1)  
41             {  
42                 addFriend(input, database);  
43             }  
440             else if (choice == 2)  
45             {
```