OBJECT ORIENTED PROGRAMMING

PROJECT REPORT

Name/Surname:Ege Kutay YÜRÜŞEN

No:180316017

Department:Computer Engineering

Subject : Social Network OOP Project

Subject is seperated according to**:**

1)User Class

2)Message Class

3)Post Class

4)Administrator Class

5)UserPanel Class

6) Appendix

1)User Class

Fields:

First,I assigned private fields which are:

Name , email, place of residence (por):(String) type

followed,nonfollowed,userlist: (ArrayList(User) type)

inbox, outbox , timeline, favpost , notifbox: (ArrayList(String) type)

date of birth (dob): (LocalDate type)

Constructor:

User class constructor has 4 paramaters which are:Name, email, date of birth, place of residence these fields are manually set in UserPanel class . Rest of fields(ArrayLists) are filled from terminal or by default values.I used this constructor in Userpanel to create User objects.

Methods:

sendMessage: Gets two paramaters which contains name(User type) and message (String type). Those paramaters are gathered in a string variable(sout) and it contains the message itself and the receiver’s name. Then the string variable is added to the sender’s outbox.

receiveMessage: The difference between sendmessage and receiveMessage is that string variable(sinb) contains sender’s name instead of receiver’s and the string variable is added to the receiver’s inbox.

setFollow: Whenever the user follows any other user, the method adds a notification message to both users’ notification boxes.

getSOutbox: Whenever you send a message to another user it displays your message and the content information. This method is used in message class.

Rest of the methods are getters and setters. Those methods are used to give access of the private fields to other classes.

2)Message Class

Fields:

private long info:Holds date information as Simple date format.
private String sendmessage: Holds messages upcoming from author.
private User sender:Holds message sender as user type.
private User receiver:Holds message receiver as user type.

Constructor:

Message class constructor checks if the users follow each other, holds the date information and there is a function that enables the message to be added into sender’s outbox and receiver’s inbox with spaces between words. If I don’t use this function only the first word of the message is hold in the ArrayList.

3)Post Class

Fields:

private User author: Holds author (post owner) as User type.

Constructor:

It has a similiar job like Message class constructor .

Constructor adds posts to all users' timeline with the content information and also adds post to the favpost which is of an ArrayList(String)type. favpost holds posts that the user shared in past. When favpost is called in Share Post operation, it displays old shared posts by the User.

4)Administrator Class

Fields:

User anyuser: Holds users' as user type.

Constructor:

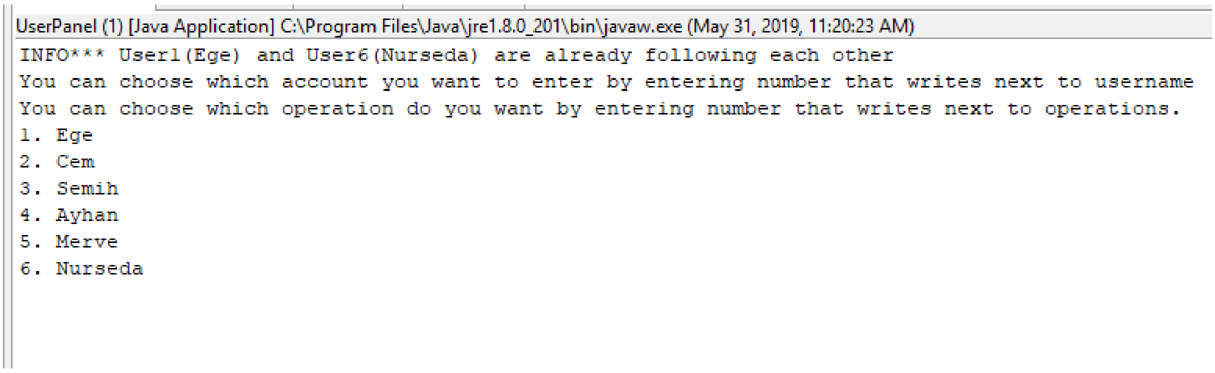
It gets user type paramater to check the user's birthday. I used LocalDate class to make mathmatical operations easier for calculating the birthday. This calculation is done by substracting the day of the current year from the day of the birth year. However, this calculation does not take into account the leap years. So I added two different variables to the calculation which help to fix the issue of the leap years.The result of this calculation is stored in an integer variable called difference. If the difference value is a positive value and less than 30, it sends a notification to the user's notification box which contains the information regarding how many days left to the user's birthday. If the difference value is zero, it sends a celebration message to the user's notification box.

5)UserPanel Class

Main Method:

I assigned User type ArrayList that holds all User objects which is called "users".I added several user objects into "users"ArrayList manually in order to perform the operations between users. Userpanel contains two segments which is:

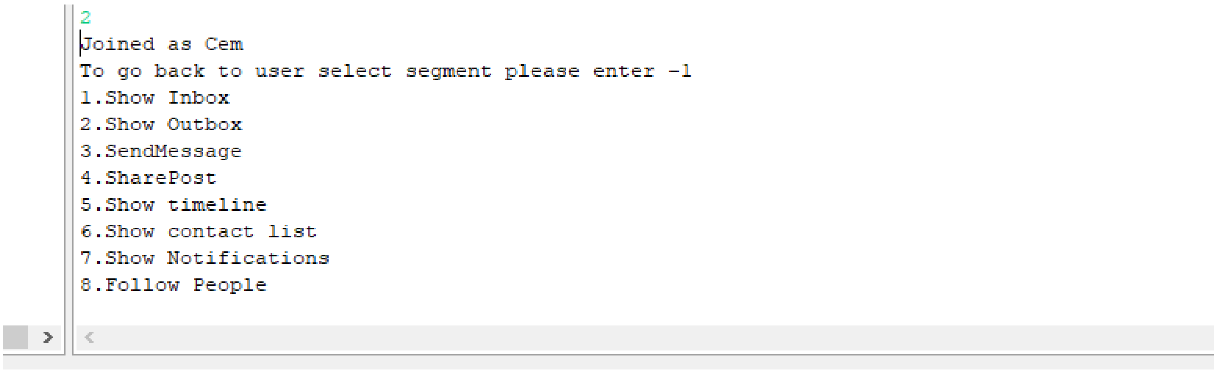
User Select Segment:



UserPanel (1) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (May 31, 2019, 11:20:23 AM)

```
INFO*** User1(Ege) and User6(Nurseda) are already following each other
You can choose which account you want to enter by entering number that writes next to username
You can choose which operation do you want by entering number that writes next to operations.
1. Ege
2. Cem
3. Semih
4. Ayhan
5. Merve
6. Nurseda
```

You can select an account by entering the number which is next to usernames.When you enter any username number you will see:



```
2
Joined as Cem
To go back to user select segment please enter -1
1.Show Inbox
2.Show Outbox
3.SendMessage
4.SharePost
5.Show timeline
6.Show contact list
7.Show Notifications
8.Follow People
```

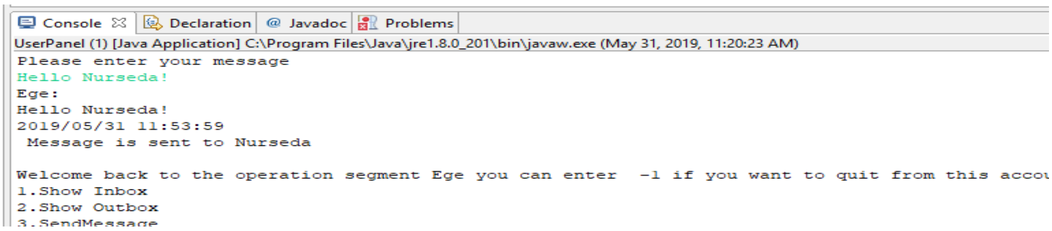
This is the segment that you can test any operation by entering the number next to the operations'. You can quit from the user by entering -1 and go back to the user segment.During the program running, you can enter or switch to any user account as many times you want for testing the operations'.

Operations:

1) Show Inbox: Checks if the user's inbox is empty. If it is empty,it displays "Your Inbox is empty.". If it is not, it calls getInbox from User class to display the user's inbox.

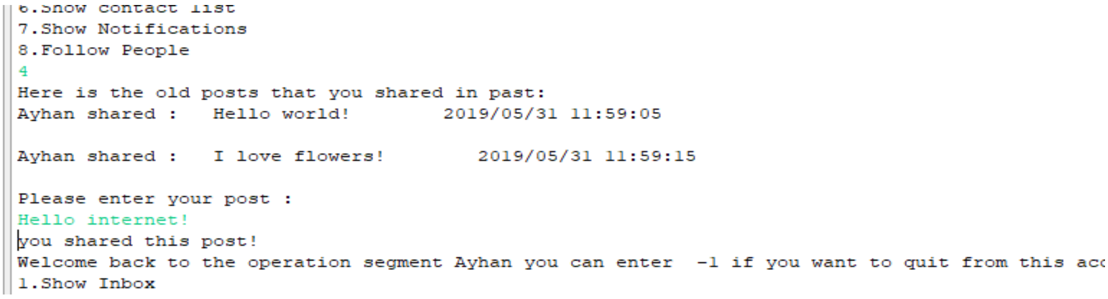
2)Show Outbox: Checks if the user's inbox is empty. If it is empty it displays "Your Outbox is empty" if it is not, it calls getOutbox from User class to display the user's inbox.

3)Send Message: Firstly, it checks if the user follows anyone. If the user follows anyone,the followed people are displayed in message sent segment. However, the user cannot send message if the user is not followed by whom the user selects. If users follow each other,the users can send message to each other. The message is added to the sender's outbox and receiver's inbox along with the content information.



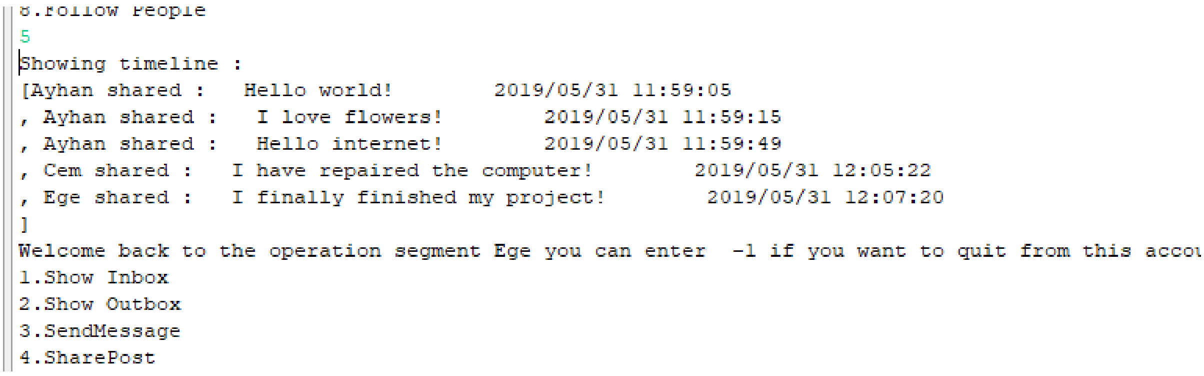
```
UserPanel (1) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (May 31, 2019, 11:20:23 AM)
Please enter your message
Hello Nurseda!
Ege:
Hello Nurseda!
2019/05/31 11:53:59
 Message is sent to Nurseda

Welcome back to the operation segment Ege you can enter  -1 if you want to quit from this accou
1.Show Inbox
2.Show Outbox
3.SendMessage
```

4)Share Post: Share post operation has two functions which are storing the user's old posts into favposts and sharing new post by creating an object of Post class.



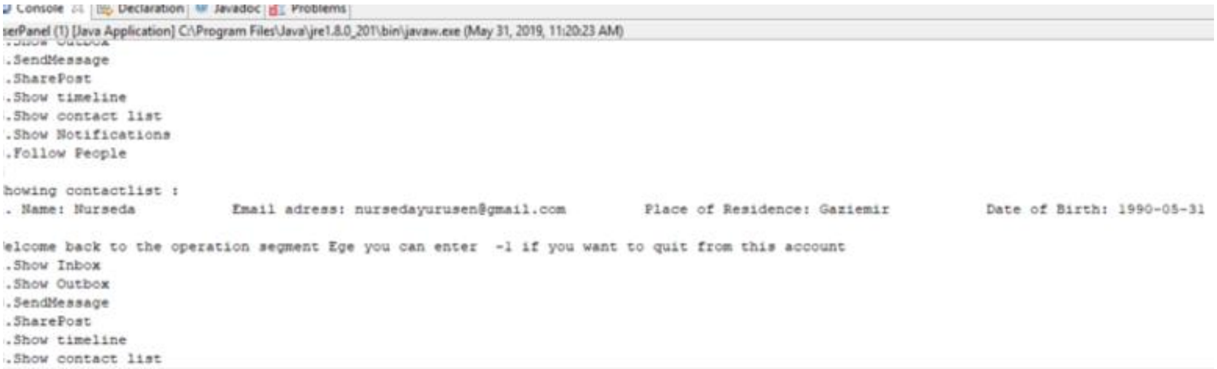
```
6.Show contact list
7.Show Notifications
8.Follow People
4
Here is the old posts that you shared in past:
Ayhan shared :   Hello world!        2019/05/31 11:59:05

Ayhan shared :   I love flowers!        2019/05/31 11:59:15

Please enter your post :
Hello internet!
you shared this post!
Welcome back to the operation segment Ayhan you can enter  -1 if you want to quit from this acc
1.Show Inbox
```

5)Show timeline: If the user’s timeline is empty it displays “your timeline is empty”message. If it is not, it displays everyone’s posts that are shared:



```
8.Follow People
5
Showing timeline :
[Ayhan shared :   Hello world!        2019/05/31 11:59:05
, Ayhan shared :   I love flowers!        2019/05/31 11:59:15
, Ayhan shared :   Hello internet!        2019/05/31 11:59:49
, Cem shared :   I have repaired the computer!        2019/05/31 12:05:22
, Ege shared :   I finally finished my project!        2019/05/31 12:07:20
]
Welcome back to the operation segment Ege you can enter  -1 if you want to quit from this accou
1.Show Inbox
2.Show Outbox
3.SendMessage
4.SharePost
```

6)Show Contact list: If the user follow another user it displays user account information.

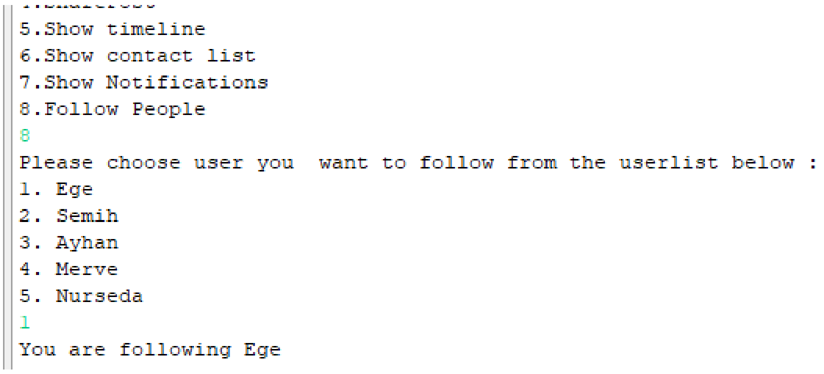


```
serPanel (1) [Java Application] C:\Program Files\Java\jre1.8.0_201\bin\javaw.exe (May 31, 2019, 11:20:23 AM)
.Show Outbox
.SendMessage
.SharePost
.Show timeline
.Show contact list
.Show Notifications
.Follow People

howing contactlist :
. Name: Nurseda          Email adress: nursedayurusen@gmail.com          Place of Residence: Gaziemir          Date of Birth: 1990-05-31

elcome back to the operation segment Ege you can enter  -1 if you want to quit from this account
.Show Inbox
.Show Outbox
.SendMessage
.SharePost
.Show timeline
.Show contact list
```

7)Show Notifications: When a user follows another one, it sends a notification to both users’notification box.

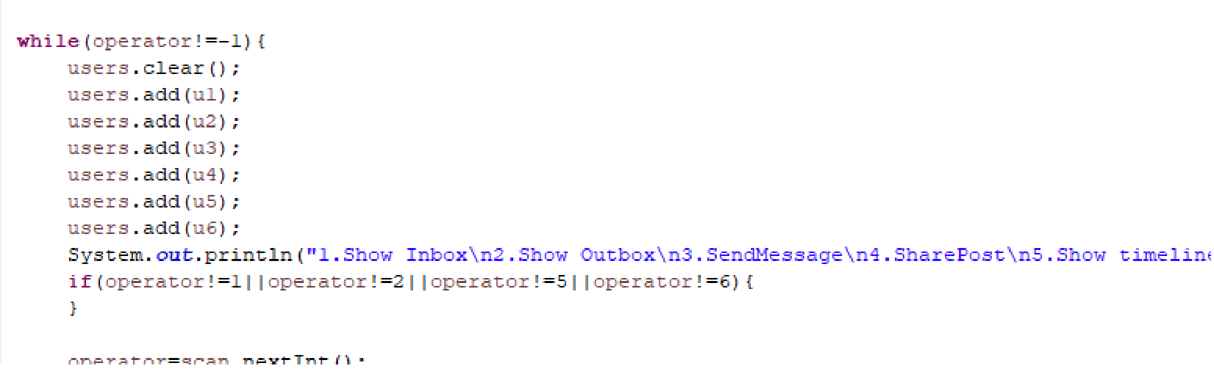
Also if any user has birthday within 30 days, it displays the number of remaining days until the user’s birthday.If the birthday is today, a celebration message is added to the user’s notification box.

8)Follow People:It lists users that the user hasn’t followed yet.The user can enter the number which is next to the user’s name to follow other users. If there is no one that the user can follow its displays a message “You are already follow everyone in the userlist!”.



```
4.SharePost
5.Show timeline
6.Show contact list
7.Show Notifications
8.Follow People
8
Please choose user you  want to follow from the userlist below :
1. Ege
2. Semih
3. Ayhan
4. Merve
5. Nurseda
1
You are following Ege
```

Appendix



```
while(operator!=-1){
    users.clear();
    users.add(u1);
    users.add(u2);
    users.add(u3);
    users.add(u4);
    users.add(u5);
    users.add(u6);
    System.out.println("1.Show Inbox\n2.Show Outbox\n3.SendMessage\n4.SharePost\n5.Show timelin
    if(operator!=1||operator!=2||operator!=5||operator!=6){
    }

    operator=scan.nextInt();
```

In UserPanel you may consider why the users ArrayList elements is added again in operator segment’s while loop.Its purpose is refreshing the users ArrayList elements.If this piece of code is not added, whenever the user follows other users, the ones that are followed are removed not only from nonfollowed(user type ArrayList) list but also from “users” Arraylist,.although there are no visible interactions between those ArrayLists.Although,I don’t understand the reason behind it and wasn’t able to figure it out,I was able to solve the problem this way.

I manually added user1(Ege) and user6(Nurseda) users each other’s followlists .So you can test sendMessage outbox and inbox in short time.