Exercise

Implement a console-based social networking application (similar to Twitter) satisfying the scenarios below.

Features

Posting: **A**lice can publish messages to a personal timeline

```
> Alice -> I love the weather today
> Bob -> Damn! We lost!
> Bob -> Good game though.
```

Reading: I can view **A**lice and **B**ob’s timelines

```
> Alice
I love the weather today (5 minutes ago)
> Bob
Good game though. (1 minute ago)
Damn! We lost! (2 minutes ago)
```

Following: **C**harlie can subscribe to **A**lice’s and **B**ob’s timelines, and view an aggregated list of all subscriptions

```
> Charlie -> I'm in New York today! Anyone want to have a coffee?
> Charlie follows Alice
> Charlie wall
Charlie - I'm in New York today! Anyone want to have a coffee? (2 seconds ago)
Alice - I love the weather today (5 minutes ago)

> Charlie follows Bob
> Charlie wall
Charlie - I'm in New York today! Anyone wants to have a coffee? (15 seconds
ago)
Bob - Good game though. (1 minute ago)
Bob - Damn! We lost! (2 minutes ago)
Alice - I love the weather today (5 minutes ago)
```

Details

- The application must use the console for input and output.
- Users submit commands to the application. There are four commands. “posting”, “reading”, etc. are not part of the commands; commands always start with the user’s name.
    - **posting**: `<user name> -> <message>`
    - **reading**: `<user name>`
    - **following**: `<user name> follows <another user>`
    - **wall**: `<user name> wall`
- Don't worry about handling any exceptions or invalid commands. Assume that the user will always type the correct commands. Just focus on the sunny day scenarios.

- Don’t bother making it work over a network or across processes. It can all be done in memory, assuming that users will all use the same terminal.
- Non-existing users should be created as they post their first message. Application should not start with a pre-defined list of users.
- Exercise should be done either in Java or C#.
- Provide instructions on how to run the application.

IMPORTANT: Focus on writing the best code you can produce. Do not rush. Take as much time as you need; there is no deadline.

Code Submission

Push the code to your own GitHub account, as a public project, and send us the link.