

Social Network

Background

You and your friend have decided to start a company that hosts a social network site. Your friend will handle the website creation (they know what they are doing, having taken our web development class). However, it is up to you to create the classes that manages the social-network information and to define several methods that operate on the network. In a website, the data is stored in a database. In our case, however, all the information comes in a big string stored in a text file. Each pair of sentences in the text is formatted as follows:

<username> is connected to <name1>, <name2>,...,<nameN>.

Your friend records the information in that string based on user activity on the website and gives it to you to manage. You can think of every pair of sentences as defining a user profile.

For example:

John is connected to Bryant,Debra,Walter

You can assume that <username> is a unique identifier for a user. In other words, there is only one User Object as John in the network. Furthermore, connections are not symmetric - **if John is connected with Alice, it does not mean that Alice is connected with John.**

User Class

The User has the following properties

name of the user, the connections of the user and the number of connections.

SocialNetwork Class:

The SocialNetwork is an array of User objects and size indicates the number of user objects. Create a Class SocialNetwork with the following methods:

1. createDataStructure(); takes a string as a parameter that creates a social network. Do nothing if the string is empty. Please see the Notes section (1) below on how the string is represented.
 - a. Note: While creating the network using the following:
John is connected to Bryant,Debra,Walter
Then: First John should be on the network if John is not in the network.
Then connections also should be on the network if not, add them to the network and then add all connections to the (User) John.
2. addUser(); takes user object as argument and adds them to the network if the user is not in the network. The user should not be added if the user is already in the network.

3. addConnection(); takes two users objects as arguments and connects them on the network. Let userA and userB are the users that are passed as parameters to this function, then add userB to userA's connections.
 - a. Note: Both userA and userB should be there on the network, if anyone of them are not in the network, then do nothing.
4. getConnections(); takes a user object as an argument that returns the connections of this user. Return **null** if the user is not in the network.
5. getCommonConnections(); returns the common connections of userA and userB.
 - a. Note: both userA and userB should be on the network. if anyone of them is not in the network, then return null

Test Cases:

Input:

John is connected to Bryant,Debra,Walter

Mercedes is connected to Walter,Robin,Bryant

Walter is connected to John,Levi,Bryant

1. getConnections("Mercedes"); returns ["Walter", "Robin", "Bryant"]
2. getConnections("MERCEDES"); returns null
3. addConnection("Mercedes", "John");
4. getConnections("Mercedes"); returns ["Walter", "Robin", "Bryant", "John"]
5. addConnection("Mercedes", "JOHN"); do nothing....
6. getCommonConnections("John", "Walter"); returns ["Bryant"]
7. getCommonConnections("John", "WALTER"); returns []
8. toString() prints the network. See the notes section below (2)

Notes Section:

1. The string will be represented as follows:
 - a. **The user and user connections are separated with " is connected to "**
 - b. **User connections are separated with commas (,)**
 - c. **Each user related information is separated by semicolon (;)**

Example:

John is connected to Bryant,Debra,Walter,Mercedes

Mercedes is connected to Walter,Robin,Bryant

The above example is represented in the string as follows:

John is connected to Bryant,Debra,Walter;Mercedes is connected to
Walter,Robin,Bryant

Hint 1: The split(String delimiter) on string is used to split string that returns String array

Example:

String str = "John is connected to Bryant, Debra, Walter;Mercedes is connected to Walter,Robin,Bryant"

```
String[] tokens = str.split(",");
```

Now the tokens array will have the following:

token[0] has "John is connected to Bryant,Debra,Walter"

token[1] has "Mercedes is connected to Walter,Robin,Bryant"

2. The output should be as follows when the toString() is invoked on SocialNetwork.
 userName : connections of this user (Consider only userNames not the User object) and all the users are separated by a new line character

Example: If the network is as follows

John is connected to Bryant,Debra,Walter

Mercedes is connected to Walter,Robin,Bryant

Then the output is:

John : [Bryant, Debra, Walter]

Bryant :

Debra :

Walter :

Mercedes : [Walter, Robin, Bryant]

Robin :

3. <Nothing>

You can download the assignment from the drive [URL](#)

All the best

Note: This assignment has been designed by Praveen Garimella in 2015 and was updated by Siva Sankar in 2019.