1. **Group members**

Group members: Ziqi Liu 9364-4917

Enyang Wang 6494-1124

1. **How to run**

Steps: 1. Enter root folder (project 4.1).

2. Type commend as: mix test

1. **What is working**

The goal of our project is to implement a Twitter-like engine and use actor model to simulate several clients and a single-engine process.

We implement the engine and client by using actor model (GenServer) and several functionalities.

1. **Implementation**

We implement 8 functionalities, which are *register, login, delete, send tweet, subscribe, re-tweet, query and logout*. Besides, we use ETS table as our data storage.

* ETS table:

User: [user\_id (string), user\_pwd (string), user\_ip (pid), connect (Boolean)]

Subscribe: [user\_id (string), user\_id2 (string)]

Tweet: [tweet\_id (integer), publisher (string), content (string), retweet\_id (integer)]

Mention: [tweet\_id (integer), user\_id (string)]

HashTags: [hashtag (string), tweet\_id (integer)]

* Functionalities:

**Register** (*user\_id, password*):

Used to create an account in the ETS data storage. If there is already an account in the data storage with same user\_id, the function will return false, else it will return true.

**Login** (*user\_id, password*):

Used to login an account with the user\_id and password. If system finds an account with same user\_id and password exist in data storage, it will return true, else it will return false.

**Delete** (*user\_id, password*):

Used to delete an account from the data storage. If the user\_id and password of the account is equal to the user\_id and password which stored in the ETS, it will be deleted from the ETS data storage and function will return true, else the function will return false and don’t change the ETS.

**Send\_tweet** (*user\_id, content, mention, hashtags*):

Used to publish a tweet to the account which follows the publisher and the account which the publisher mentions in the content. Words start with ‘#’ and end with a space will be recognized as hashtags. Words start with ‘@’ and end with a space will be recognized as mention.

**Subscribe** (*user\_id, follow*):

If user *user\_id* subscribe user *follow*, user *user\_id* will receive all the tweets that user *follow* sends.

**Re\_tweet** (*user\_id, content, mention, hashTags, retweetID*):

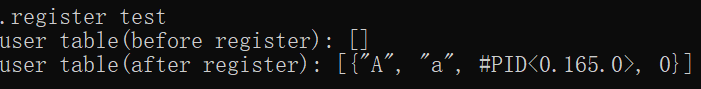
Users can re-tweet other users’ tweet while adding new *content*. Words start with “#” and end with a space will be recognized as “*hashTags*”, which make the tweet searchable by “query” function. Words start with “@” end with a space will be recognized as “*mention*”, which will notify the user with the mentions name. *retweetID* indicates the tweet\_id of original tweet.

**Query** (*type*, *content*):

Users can search for tweets with specific hashTags, tweets subscribed to and tweets in which the user is mentioned.

1. **Test case**

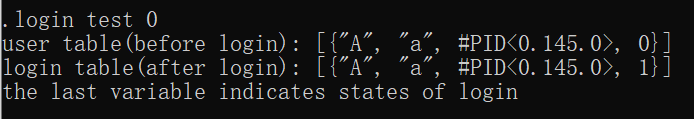
**Register:**



Process <0.165.0> registers an account with user name “A” and password “a”. The last element represents the login status of user, now it is “offline”.

{user\_id, password, ip, status}

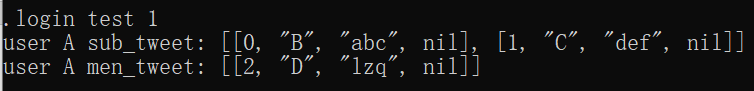
**Login0:**



Process<0.145.0> login as user “A” with password “a”. The last variable indicated the status. 1 means “online”, 0 means “offline”.

{user\_id, password, ip, status}

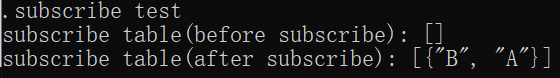
**Login1:**



User A subscribed user B and C. User D sent a tweet mentioned user A. User A receive 3 tweets when he logins.

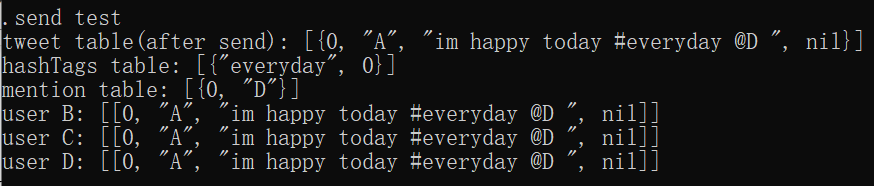
[tweet\_id, publisher, tweet\_content, re\_tweetID]

**Subscribe:**



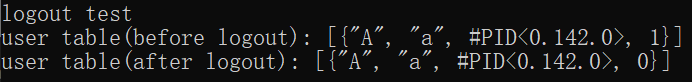
User B subscribes user A.

**Send:**



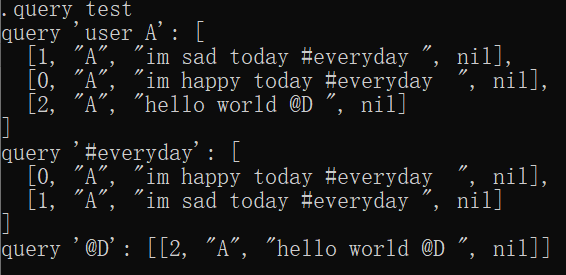
User A send a tweet “im happy today” with hashtag “everyday” mentions user D. Since user B, C subscribed user A, they could receive the tweet that A sent. User D can also get the tweet since user A mentions D in the tweet (@D). The last element in tweet is “nil”, which means this is the original tweet instead of a repost one.

**Logout:**



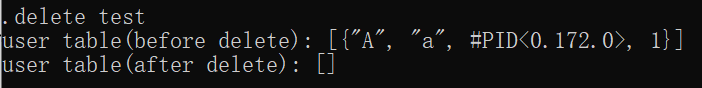
The last attribute in the user table represents the login status of an account, 1 means online and 0 means offline.

**Query:**



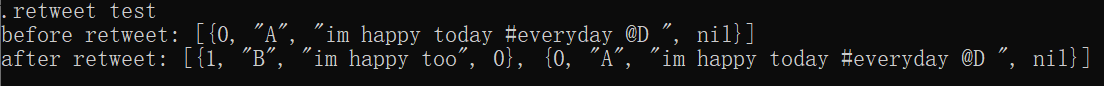
‘Query user A’ means querying all tweets which user A published before. Word start with ‘#’ and end with a space will be recognized as hashtags. ‘Query hashtag’ means querying all tweets that contain that hashtag. Word start with ‘@’ and end with a space will be recognized as mention. ‘Query mention’ means querying all tweets that contain that mention.

**Delete:**



In the test case, there is one account in the data storage before do the deletion and the data storage are empty after deletion.

**Retweet:**



The last attribute in the retweet table means the tweet number of another tweet which retweeted by this tweet. In the test case, user B retweets a tweet of user A. The last attribute of the new tweet of user B will contain the tweet number of user A, which is 0.