JAVA AWT,SWING BASED- TWITTER DATABASE MANAGEMENT& SIMULATION - SQL CONNECTIVITY USING JDBC

A

Report

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

BY

T.BADRINATH<1602-18-737-066>

Department of Information Technology

Vasavi College of Engineering *(Autonomous)*

Ibrahimbagh, Hyderabad-31

2020

BONAFIDE CERTIFICATE

This is to certify that this project report titled "***Twitter Database Management in a region***" is the bonafide project work of ***Mr.Thodupunuri Badrinath*** bearing hallticket number ***1602-18-737-066***,who carried out the mini project work under my supervision. Certified further that, to the best of my knowledge the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or award was conferred on as earlier occasion or any other candidate.

Signature	Signature of the Examiner
External Examiner	**B.LEELAVATHY**
	Lecturer
	Department of Information Technology

TWITTER DATA MANAGEMENT OF USERS IN A REGION

ABSTRACT:

In social media, information is present in large amount. Extracting information from social media gives us several usage in various fields. In the field of biomedical and healthcare, extracting information from social media is providing number of benefits such as knowledge about the latest technology, updates of current situation in medical field etc.

Twitter is one of the social media which allows the user post tweets of limited number of characters and share the tweet to their followers. Twitter allows application developer to access the tweets for their purpose .Based on the analysis of comments and reply's people can jump into a conclusion and find a solution to a problem based on user's feedback.

INTRODUCTION:

REQUIREMENTS ANALYSIS:

LIST OF TABLES:

>USERS

>POSTS

>TWEET

>HAS

>FOLLOWING

>HAS_A

>RESPONSE

List of attributes with their domain types:

USERS:

USER_NAME	: VARCHAR2(10)
USER_ID	:VARCHAR2(10)
MOB_NO	: NUMBER(10)
VERIFIED_FLAG	: CHAR(1)
EMAIL_ID	: VARCHAR2(25)
PASSWORD	: VARCHAR2(5)

TWEET:

MESSAGE	: VARCHAR2(60)
POLARITY	:CHAR(1)
LIKES	: NUMBER(4)
TWEET_ID	:VARCHAR2(10)
CATEGORY	: VARCHAR2(10)

USERS_TWEET:

USER_ID	:VARCHAR2(10)
TWEET_ID	:VARCHAR2(10)
P_DATE	: DATE

FOLLOWING:

F_USER_ID	: VARCHAR2(10)

USERS_FOLLOWING:

USER_ID	: VARCHAR2(10)
F_USER_ID	:VARCHAR2(10)
SINCE	: DATE

RESPONSE:

R_USER_ID	:VARCHAR2(10)

REPLIED_DATE	:DATE
CATEGORY	: VARCHAR2(6)
MESSAGE	:VARCHAR2(100)
MESSAGE_ID	:VARCHAR2(10)

TWEET_RESPONSE:

TWEET_ID	:VARCHAR2(10)
MESSAGE_ID	:VARCHAR2(10)
R_POLARITY	:CHAR(1)

ARCHITECTURE AND TECHNOLOGY USED

SOFTWARE USED:

Java Eclipse, Oracle 11g Database, Java SE version 7, SQL*Plus.

Java AWT:

Java AWT (Abstract Window Toolkit) is *an API to develop GUI or window-based applications* in java.

Java AWT components are platform-dependent i.e. components are displayed according to the view of operating system. AWT is heavyweight i.e. its components are using the resources of OS.

The java.awt package provides classes for AWT API such as TextField, Label, TextArea, RadioButton, CheckBox, Choice, List etc.

SQL:

Structure Query Language(SQL) is a database query language used for storing and managing data in Relational DBMS. SQL was the first commercial language introduced for E.F Codd's **Relational** model of database. Today almost all RDBMS (MySQL, Oracle, Infomix, Sybase, MS Access) use **SQL** as the standard database query language. SQL is used to perform all types of data operations in RDBMS.

Java-SQL Connectivity using JDBC:

Java Database Connectivity (**JDBC**) is an application programming interface (API) for the programming language Java, which defines how a client may access a database. It is a Java-based data access technology used for Java database connectivity. It is part of the Java Standard Edition platform, from Oracle Corporation. It provides methods to query and update data in a database and is oriented towards relational databases.

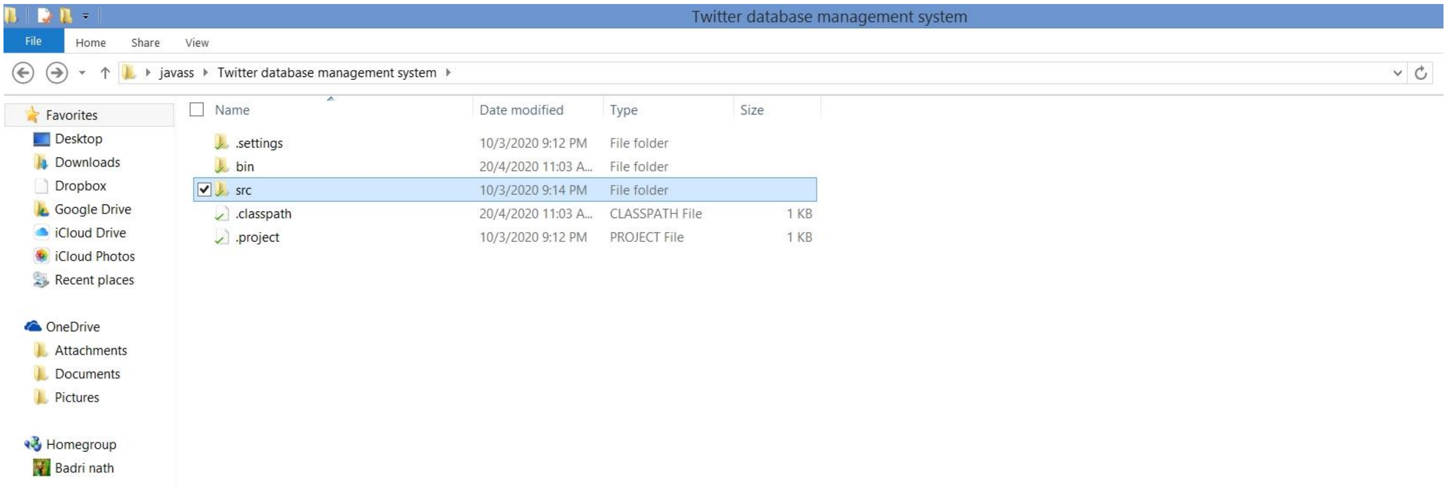
The connection to the database can be performed using Java programming (JDBC API) as:

```
private void connToDb(){
             try {
             Class.forName("oracle.jdbc.driver.OracleDriver");
             connection =
DriverManager.getConnection("jdbc:oracle:thin:@localhost:1521:ORCL","Project","Project");
             statement = connection.createStatement();

           } catch (SQLException connectException) {
             System.out.println(connectException.getMessage());
             System.out.println(connectException.getSQLState());
             System.out.println(connectException.getErrorCode());
             System.exit(1);
           }
           catch (Exception e)
          {
               System.err.println("Unable to find and load driver");
               System.exit(1);
          }
       }
```

Thus, the connection from Java to Oracle database is performed and therefore, can be used for updating tables in the database directly.

FOLDER STRUCTURE:

This project contains a folder named src in which it consists of one folder and inside the folder there are few codes. By this we can navigate easily to reach code and we can make many changes as we can want easily.



Name	Date modified	Type	Size
.settings	10/3/2020 9:12 PM	File folder	
bin	20/4/2020 11:03 A...	File folder	
☑ src	10/3/2020 9:14 PM	File folder	
.classpath	20/4/2020 11:03 A...	CLASSPATH File	1 KB
.project	10/3/2020 9:12 PM	PROJECT File	1 KB

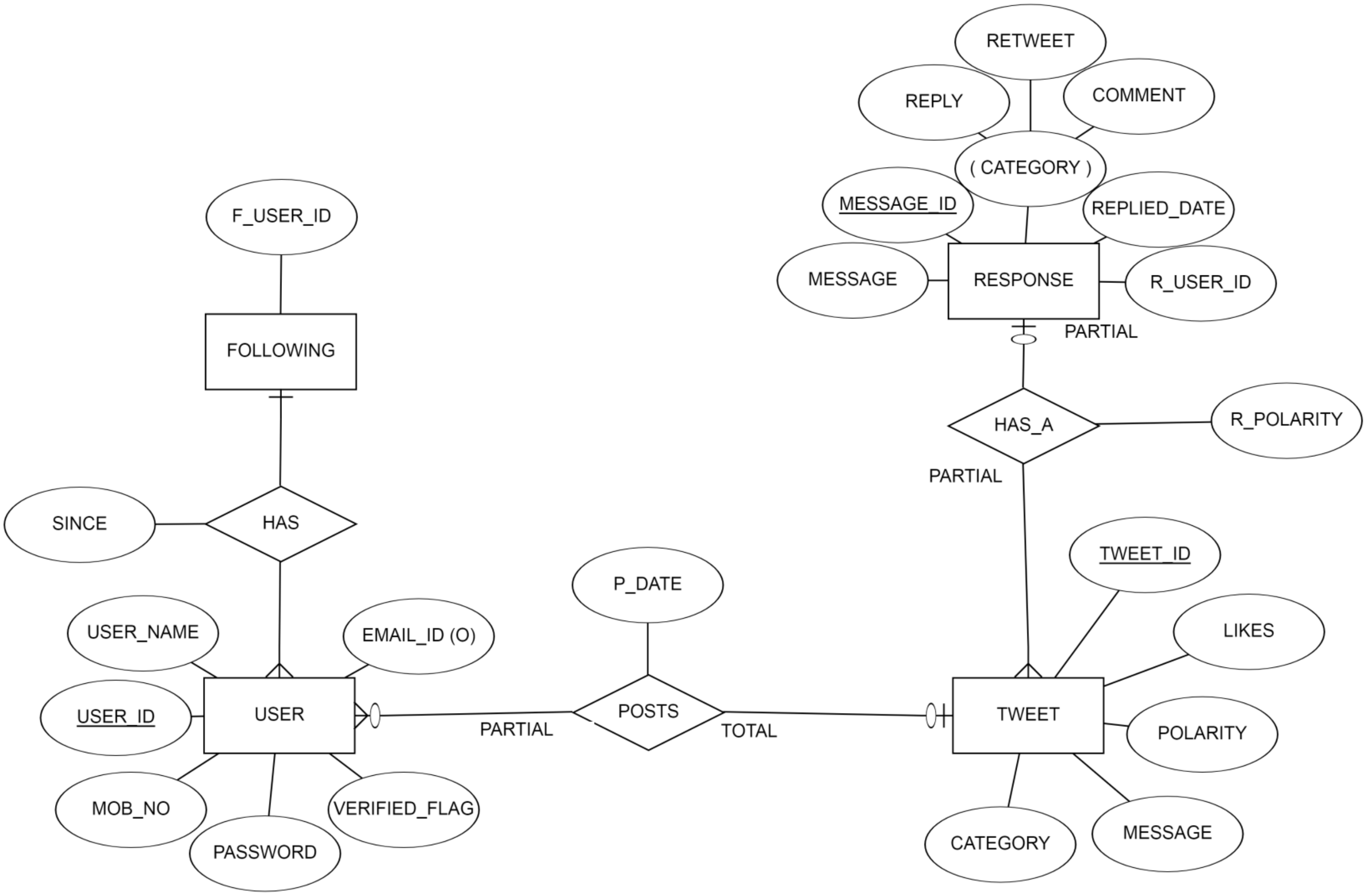
Follow up the below link in order to have a knowledge of how system is being implemented.

https://github.com/Badrinath2428/DBMS-PROJECT/blob/master/Twitter%20database%20management%20system.zip

TESTING

The program executes three basic operations those are insert update and delete on 8 different tables. Along with this, it also has an output column which gives information about how many rows have been edited. Errors syntactical or exceptional will be shown if occurred.

ER DIAGRAM:



Mapping Cardinalities and Participation Constraints:

>One user can post many posts, so one to many cardinality suit posts.

> One user can have many followers, so one to many cardinality

suit has.

>A tweet can have many responses, so one to many cardinality suit has_a.

DDL AND DML COMMANDS:

```
SQL> create table Users(

  2  user_name varchar2(10),

  3  user_id number(5) primary key,

  4  mob_no number(10,0),

  5  verified_flag char(1) check (verified_flag in ('Y','N')),

  6  email_id varchar2(10),

  7  password varchar2(5));

Table created.

SQL> alter table Users modify(user_id varchar2(10));

Table altered.

SQL> desc users;

 Name                                       Null?          Type

 ----------------------------------------   --------       --------------------------

 USER_NAME                                                 VARCHAR2(10)

 USER_ID                                   NOT NULL        VARCHAR2(10)

 MOB_NO                                                    NUMBER(10)

 VERIFIED_FLAG                                             CHAR(1)

 EMAIL_ID                                                  VARCHAR2(10)

 PASSWORD                                   VARCHAR2(5)
```

```
SQL> alter table users modify(email_id varchar2(25));

Table altered.

SQL> desc users

 Name                                      Null?               Type

 ---------------------------------------   --------            --------------------------

 USER_NAME                                                         VARCHAR2(10)

 USER_ID                                    NOT NULL               VARCHAR2(10)

 MOB_NO                                                             NUMBER(10)

 VERIFIED_FLAG                                                         CHAR(1)

 EMAIL_ID                                                            VARCHAR2(25)

 PASSWORD                                                             VARCHAR2(5)

SQL> insert into users
values('&user_name','&user_id',&mob_no,'&verified_flag','&email_id','&password
');

Enter value for user_name: Badrinath

Enter value for user_id: Badri2428

Enter value for mob_no: 9381756470

Enter value for verified_flag: Y

Enter value for email_id: badrinath@gmail.com

Enter value for password: Badri2428
```

```
old   1: insert into users
values('&user_name','&user_id',&mob_no,'&verified_flag','&email_id','&password
')

new   1: insert into users
values('Badrinath','Badri2428',9381756470,'Y','badrinath@gmail.com','Badri2428')


1 row created.

SQL> /

Enter value for user_name: Abhiraj

Enter value for user_id: Abhi5007

Enter value for mob_no: 9949939007

Enter value for verified_flag: Y

Enter value for email_id: dusariabhiraj@gmail.com

Enter value for password: abcd1234@

old   1: insert into users
values('&user_name','&user_id',&mob_no,'&verified_flag','&email_id','&password
')

new   1: insert into users
values('Abhiraj','Abhi5007',9949939007,'Y','dusariabhiraj@gmail.com','abcd1234
@')

1 row created.

SQL> alter table users modify(user_name varchar2(10) not null);
```

```
Table altered.

SQL> alter table users modify(mob_no number(10) not null);

Table altered.

SQL> desc users;

 Name                                      Null?          Type

 ----------------------------------------- --------       ----------------------------

 USER_NAME                                 NOT NULL       VARCHAR2(10)

 USER_ID                                   NOT NULL       VARCHAR2(10)

 MOB_NO                                    NOT NULL       NUMBER(10)

 VERIFIED_FLAG                                            CHAR(1)

 EMAIL_ID                                                 VARCHAR2(25)

 PASSWORD                                                 VARCHAR2(10)

SQL> insert into users
values('&user_name','&user_id','&mob_no','&verified_flag','&email_id','&password');

Enter value for user_name: Yash

Enter value for user_id: Yash6960

Enter value for mob_no: 8686819973

Enter value for verified_flag: N

Enter value for email_id: yasho6960@gmail.com
Enter value for password: 6960118

old   1: insert into users
values('&user_name','&user_id','&mob_no','&verified_flag','&email_id','&password')

new   1: insert into users
values('Yash','Yash6960','8686819973','N','yasho6960@gmail.com','6960118')

1 row created.

SQL> /

Enter value for user_name: pranav

Enter value for user_id: Prana0680

Enter value for mob_no: 9515789639

Enter value for verified_flag: N

Enter value for email_id: pranav23@gmail.com

Enter value for password: pranav23

old   1: insert into users
values('&user_name','&user_id','&mob_no','&verified_flag','&email_id','&password')

new   1: insert into users
values('pranav','Prana0680','9515789639','N','pranav23@gmail.com','pranav23')

1 row created.

SQL> /

Enter value for user_name: Varun
Enter value for user_id: bel007

Enter value for mob_no: 8179700761

Enter value for verified_flag: Y

Enter value for email_id:

Enter value for password: varun115

old   1: insert into users
values('&user_name','&user_id','&mob_no','&verified_flag','&email_id','&passwor
d')

new   1: insert into users values('Varun','bel007','8179700761','Y','','varun115')


1 row created.


SQL> select * from users;


USER_NAME  USER_ID       MOB_NO       V      EMAIL_ID                  PASSWORD

----------           ----------           ----------          -    -------------------------         ----------

Badrinath      Badri2428      9381756470   Y    badrinath@gmail.com       Badri2428

Abhiraj          Abhi5007       9949939007   Y dusariabhiraj@gmail.com   abcd1234@

Yash              Yash6960       8686819973  N      yasho6960@gmail.com       6960118

pranav           Prana0680    9515789639   N      pranav23@gmail.com       pranav23
```

```
Varun            bel007         8179700761 Y                                   Varun115


SQL> create table tweet
  2  (
  3  message varchar2(60),
  4  polarity char(1) check (polarity in ('P','N')),
  5  likes number(4,0),
  6  tweet_id varchar2(10),
  7  twote_date date,
  8  category varchar2(10));


Table created.


SQL> alter table tweet add(primary key(tweet_id));


Table altered.


SQL> alter table tweet drop(twote_date);


Table altered.
```

```
SQL> desc tweet;

 Name                                        Null?              Type

 -----------------------------------------  --------          ---------------------------

 MESSAGE                                                        VARCHAR2(60)

 POLARITY                                                          CHAR(1)

 LIKES                                                            NUMBER(4)

 TWEET_ID                                    NOT NULL            VARCHAR2(10)

 CATEGORY                                                         VARCHAR2(10)


SQL> insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category');

Enter value for message: A new dawn,fresh vigour and renewed hope in Assam!.

Enter value for polarity: P

Enter value for likes: 4024

Enter value for tweet_id: TY1456278

Enter value for category: Social

old   1: insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('A new dawn,fresh vigour and renewed hope in
Assam!.','P',4024,'TY1456278','Social')
```

```
1 row created.


SQL> /

Enter value for message: Strolling in sidney.

Enter value for polarity: P

Enter value for likes: 1234

Enter value for tweet_id: Tx129674

Enter value for category: Fun

old   1: insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('Strolling in sidney.','P',1234,'Tx129674','Fun')


1 row created.


SQL> insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category');

Enter value for message: Sending my best wishes to the Indian U19 Cricket team
ahead of their world cup final.

Enter value for polarity: P

Enter value for likes: 9876
Enter value for tweet_id: QW189528

Enter value for category: Sports

old   1: insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('Sending my best wishes to the Indian U19 Cricket
team ahead of their world cup


final.','P',9876,'QW189528','Sports')



1 row created.



SQL> /

Enter value for message: Stronger together.Happy Republic Day to all of us.

Enter value for polarity: P

Enter value for likes: 9123

Enter value for tweet_id: QP196328

Enter value for category: Social

old   1: insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('Stronger together.Happy Republic Day to all of
us.','P',9123,'QP196328','Social')
```

```
1 row created.


SQL> /

Enter value for message: In such a massive country like India,only 2200 people have declared their annual income over 1cr.

Enter value for polarity: N

Enter value for likes: 1450

Enter value for tweet_id: AS190634

Enter value for category: Social

old   1: insert into tweet values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('In such a massive country like India,only 2200 people have declared their annual income

over 1cr.','N',1450,'AS190634','Social')


SQL> /

Enter value for message: The ganga is at the heart of our civilisation.I am glad you spent time in varanasi @Yasho.

Enter value for polarity: P
Enter value for likes: 7803

Enter value for tweet_id: ZX678310

Enter value for category: Fun

old   1: insert into tweet
values('&message','&polarity',&likes,'&tweet_id','&category')

new   1: insert into tweet values('The ganga is at the heart of our civilisation.I am
glad you spent time in varanasi


@Yasho.','P',7803,'ZX678310','Fun')



1 row created.



SQL> select message,tweet_id from tweet;



MESSAGE                                                          TWEET_ID

--------------------------------------------------------------------  -----------

A new dawn,fresh vigour and renewed hope in Assam!.        TY1456278

Strolling in sidney.                                                  Tx129674

Sending my best wishes to the Indian U19 Cricket team          QW189528

ahead of their world cup  final.

Stronger together.Happy Republic Day to all of us.                QP196328
```

```
In such a massive country like India,only 2200 people              AS190634

 have declared their annual income over 1cr

The ganga is at the heart of our civilisation.I am               ZX678310

 glad you spent time in varanasi @Yasho6960.


6 rows selected.


SQL> select tweet_id,polarity,likes,category from tweet;


TWEET_ID              P     LIKES             CATEGORY

----------            -    ----------          ----------

TY1456278             P    4024                 Social

Tx129674               P    1234                 Fun

QW189528               P    9876                 Sports

QP196328               P    9123                 Social

AS190634              N    1450                 Social

ZX678310               P    7803                  Fun

6 rows selected.


SQL> create table posts(
  2  user_id varchar2(10)  references users(user_id),

  3  tweet_id varchar2(10) references tweet(tweet_id),

  4  p_date date);


Table created.


SQL> insert into posts values('&user_id','&tweet_id', '&p_date');

Enter value for user_id: Badri2428

Enter value for tweet_id: TY1456278

Enter value for p_date: 22-JAN-2018

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')

new   1: insert into posts values('Badri2428','TY1456278', '22-JAN-2018')


1 row created.


SQL> /

Enter value for user_id: Badri2428

Enter value for tweet_id: ZX678310

Enter value for p_date: 09-AUG-2010

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')
```

```
new   1: insert into posts values('Badri2428','ZX678310', '09-AUG-2010')


1 row created.


SQL> /

Enter value for user_id: Yash6960

Enter value for tweet_id: Tx129674

Enter value for p_date: 24-DEC-2017

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')

new   1: insert into posts values('Yash6960','Tx129674', '24-DEC-2017')


1 row created.


SQL> /

Enter value for user_id: Abhi5007

Enter value for tweet_id: QW189528

Enter value for p_date: 14-FEB-2018

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')

new   1: insert into posts values('Abhi5007','QW189528', '14-FEB-2018')
```

```
1 row created.


SQL> /

Enter value for user_id: Prana0680

Enter value for tweet_id: QP196328

Enter value for p_date: 26-JAN-2020

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')

new   1: insert into posts values('Prana0680','QP196328', '26-JAN-2020')


1 row created.


SQL> /

Enter value for user_id: bel007

Enter value for tweet_id: AS190634

Enter value for p_date: 22-JAN-2019

old   1: insert into posts values('&user_id','&tweet_id', '&p_date')

new   1: insert into posts values('bel007','AS190634', '22-JAN-2019')

1 row created.


SQL> alter table posts add(primary key(user_id,tweet_id));
```

```
Table altered.


SQL> desc posts;

 Name                                      Null?    Type

 ----------------------------------------- -------- ----------------------------

 USER_ID                                    NOT NULL VARCHAR2(10)

 TWEET_ID                                   NOT NULL VARCHAR2(10)

 P_DATE                                              DATE


SQL> select * from posts;


USER_ID                TWEET_ID                 P_DATE

----------               ----------                ---------

Badri2428              TY1456278                22-JAN-18

Badri2428              ZX678310                 09-AUG-10

Yash6960               Tx129674                 24-DEC-17

Abhi5007               QW189528                  14-FEB-18

Prana0680              QP196328                 26-JAN-20

bel007                 AS190634                 22-JAN-19
```

```
6 rows selected.


SQL> create table following(

  2  f_user_id references users(user_id));

Table created.


SQL> desc following;

 Name                                      Null?    Type

 ----------------------------------------- -------- ----------------------------

 F_USER_ID                                          VARCHAR2(10)


SQL> insert into following values('&f_user_id');

Enter value for f_user_id: Badri2428

old   1: insert into following values('&f_user_id')

new   1: insert into following values('Badri2428')

1 row created.


SQL> /

Enter value for f_user_id: Abhi5007

old   1: insert into following values('&f_user_id')
```

```
new   1: insert into following values('Abhi5007')


1 row created.


SQL> /

Enter value for f_user_id: Yash6960

old   1: insert into following values('&f_user_id')

new   1: insert into following values('Yash6960')


1 row created.


SQL> /

Enter value for f_user_id: Prana0680

old   1: insert into following values('&f_user_id')

new   1: insert into following values('Prana0680')


1 row created.


SQL> /

Enter value for f_user_id: bel007
```

```
old   1: insert into following values('&f_user_id')

new   1: insert into following values('bel007')


1 row created.


SQL> create table has(

  2  user_id varchar2(10),

  3  f_user_id varchar2(10),

  4  since date);


Table created.


SQL> alter table has add(foreign key(user_id) references users);


Table altered.


SQL> desc has;

 Name                                      Null?    Type

 ----------------------------------------- -------- ----------------------------

 USER_ID                                            VARCHAR2(10)
 F_USER_ID                                  VARCHAR2(10)

 SINCE                                            DATE


SQL> insert into has values('&user_id','&f_user_id','&since');

Enter value for user_id: Badri2428

Enter value for f_user_id: bel007

Enter value for since: 12-JUL-2019

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Badri2428','bel007','12-JUL-2019')


1 row created.


SQL> /

Enter value for user_id: Badri2428

Enter value for f_user_id: Yash6960

Enter value for since: 27-DEC-2016

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Badri2428','Yash6960','27-DEC-2016')


1 row created.
```

```
SQL> /

Enter value for user_id: Abhi5007

Enter value for f_user_id: Badri2428

Enter value for since: 11-JAN-2001

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Abhi5007','Badri2428','11-JAN-2001')


1 row created.


SQL> /

Enter value for user_id: Abhi5007

Enter value for f_user_id: Prana0680

Enter value for since: 12-NOV-2017

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Abhi5007','Prana0680','12-NOV-2017')


1 row created.


SQL> /
```

```
Enter value for user_id: Yash6960

Enter value for f_user_id: Badri2428

Enter value for since: 17-FEB-2014

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Yash6960','Badri2428','17-FEB-2014')


1 row created.


SQL> /

Enter value for user_id: Prana0680

Enter value for f_user_id: Abhi5007

Enter value for since: 12-SEP-2019

old   1: insert into has values('&user_id','&f_user_id','&since')

new   1: insert into has values('Prana0680','Abhi5007','12-SEP-2019')


1 row created.


SQL> select * from has;


USER_ID       F_USER_ID    SINCE
```

```
----------          ----------         ---------

Badri2428     bel007        12-JUL-19

Badri2428     Yash6960     27-DEC-16

Abhi5007     Badri2428     11-JAN-01

Abhi5007     Prana0680    12-NOV-17

Yash6960     Badri2428     17-FEB-14

Prana0680     Abhi5007     12-SEP-19

6 rows selected.


SQL> create table response

  2  (

  3  r_user_id varchar2(10) references users(user_id),

  4  replied_date date,

  5  category varchar2(6),

  6  meassage varchar2(100),

  7  message_id varchar2(10));

Table created.


SQL> alter table response rename column meassage to message;

Table altered.
```

```
SQL> alter table response add(primary key(message_id));

Table altered.


SQL> desc response;

 Name                             Null?          Type

 ------------------------------- --------       ---------------------------

 R_USER_ID                                        VARCHAR2(10)

 REPLIED_DATE                                          DATE

 CATEGORY                                          VARCHAR2(6)

 MESSAGE                                          VARCHAR2(100)

 MESSAGE_ID                      NOT NULL         VARCHAR2(10)


SQL> insert into response
values('&r_user_id','&replied_date','&category','&message','&message_id');

Enter value for r_user_id: Prana0680

Enter value for replied_date: 12-FEB-2020

Enter value for category: Reply

Enter value for message: Yeah its shocking.

Enter value for message_id: QW12345
```

```
old   1: insert into response
values('&r_user_id','&replied_date','&category','&message','&message_id')

new   1: insert into response values('Prana0680','12-FEB-2020','Reply','Yeah its
shocking.','QW12345')


1 row created.


SQL> /

Enter value for r_user_id: Yash6960

Enter value for replied_date: 11-FEB-2020

Enter value for category: REPLY

Enter value for message: Yeah! It was awfull.

Enter value for message_id: QN18756

old   1: insert into response
values('&r_user_id','&replied_date','&category','&message','&message_id')

new   1: insert into response values('Yash6960','11-FEB-2020','REPLY','Yeah! It was
awfull.','QN18756')


1 row created.


SQL> select r_user_id,replied_date,category,message_id from response;
```

```
R_USER_ID  REPLIED_D CATEGO MESSAGE_ID

---------- --------- ------ ----------

Prana0680  12-FEB-20 Reply  QW12345

Yash6960   11-FEB-20 REPLY  QN18756


SQL> select message,message_id from response;


MESSAGE                                                      MESSAGE_ID

--------------------------------------------------             ----------

Yeah its shocking                                                    QW12345

Yeah! It was awfull.                                         QN18756


SQL> create table has_a (

  2  tweet_id varchar2(10) references tweet(tweet_id),

  3  message_id varchar2(10) references response(message_id),

  4  r_polarity char(1));


Table created.
```

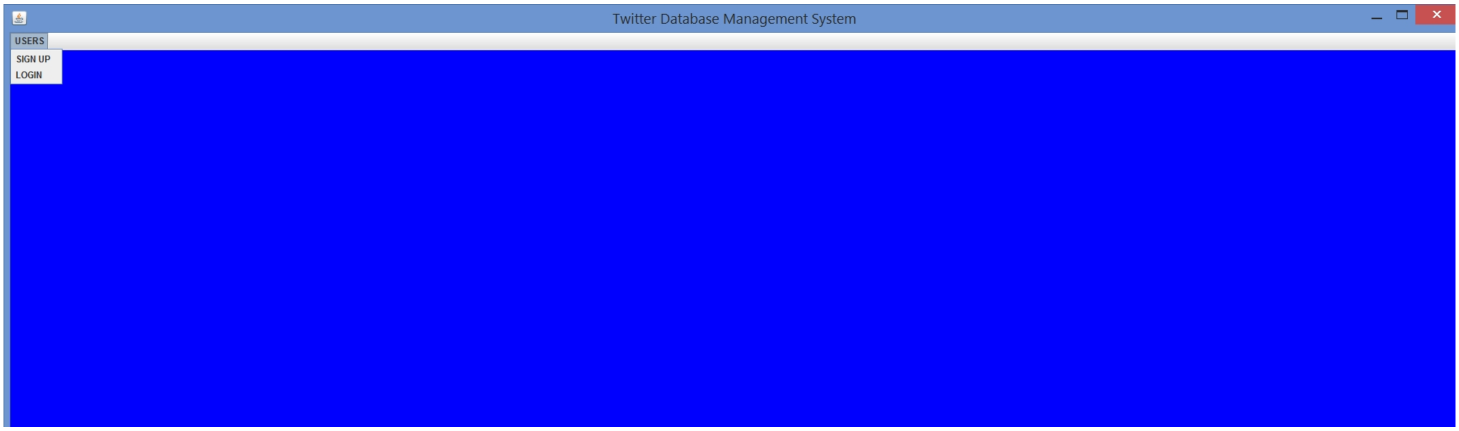
```
SQL> desc has_a

 Name                                      Null?    Type

 ----------------------------------------- -------- ----------------------------

 TWEET_ID                                           VARCHAR2(10)

 MESSAGE_ID                                           VARCHAR2(10)

 R_POLARITY                                          CHAR(1)


SQL> insert into has_a values('&tweet_id','&message_id','&r_polarity');

Enter value for tweet_id: AS190634

Enter value for message_id: QW12345

Enter value for r_polarity: P

old   1: insert into has_a values('&tweet_id','&message_id','&r_polarity')

new   1: insert into has_a values('AS190634','QW12345','P')


1 row created.


SQL> /

Enter value for tweet_id: ZX678310

Enter value for message_id: QN18756

Enter value for r_polarity: P
```

```
old   1: insert into has_a values('&tweet_id','&message_id','&r_polarity')

new   1: insert into has_a values('ZX678310','QN18756','P')


1 row created.


SQL> select * from has_a;


TWEET_ID   MESSAGE_ID R

---------- ---------- -

AS190634   QW12345    P

ZX678310   QN18756    P


TNAME                          TABTYPE       CLUSTERID

------------------------------ -------       ----------

FOLLOWING                      TABLE

HAS                            TABLE

HAS_A                          TABLE

POSTS                          TABLE

RESPONSE                       TABLE

TWEET                          TABLE
```

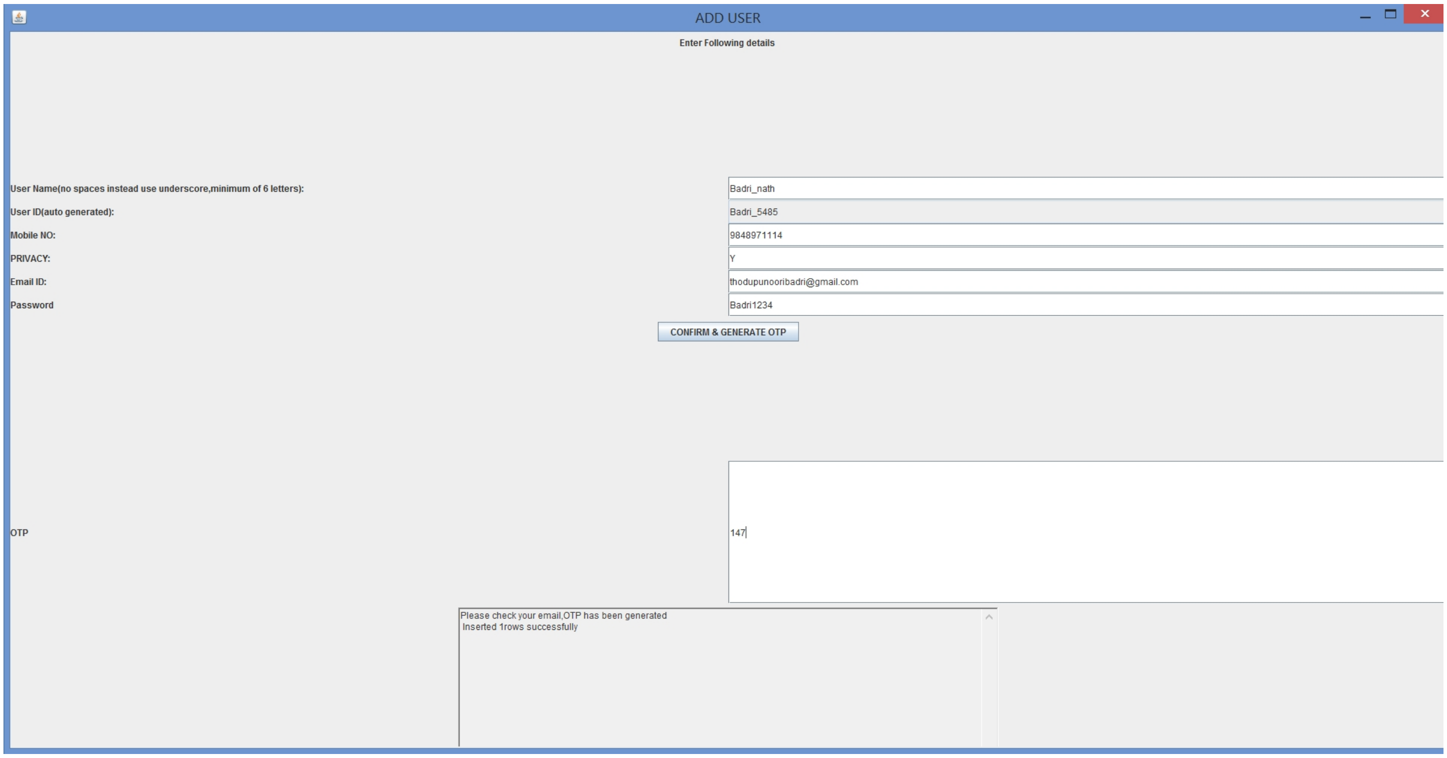
```
USERS                    TABLE


7 rows selected.

SQL> rename has to users_following;

Table renamed.


SQL> desc users_following;

 Name                                      Null?    Type

 ----------------------------------------- -------- ----------------------------

 USER_ID                                            VARCHAR2(10)

 F_USER_ID                                          VARCHAR2(10)

 SINCE                                                      DATE


SQL> rename posts to users_tweet;

Table renamed.


SQL> desc users_tweet;

 Name                                      Null?                        Type

 -----------------------------------------    --------                  ----------------------------

 USER_ID                                   NOT NULL                 VARCHAR2(10)
 TWEET_ID                        NOT NULL             VARCHAR2(10)

 P_DATE                                                  DATE


SQL> rename has_a to tweet_response;


Table renamed.


SQL> select * from tab;


TNAME                          TABTYPE            CLUSTERID

------------------------------  -------           ----------

FOLLOWING                       TABLE

RESPONSE                        TABLE

TWEET                           TABLE

TWEET_RESPONSE                  TABLE

USERS                           TABLE

USERS_FOLLOWING                 TABLE

USERS_TWEET                     TABLE


7 rows selected.
```

Here are few details regarding the system:

1. Initially system prompts the user either to sing up or login(if login account exits in data base).

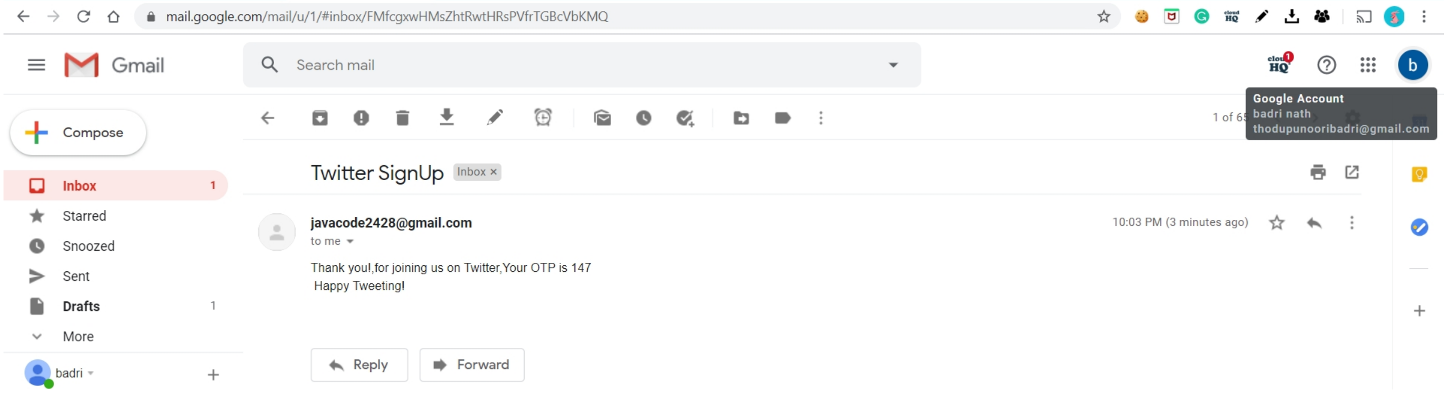


2.If user selects signup he/she has to enter details, once he/she confirms , he/she(user) receives an OTP, by which user has to validate details.



User Name(no spaces instead use underscore,minimum of 6 letters):	Badri_nath
User ID(auto generated):	Badri_5485
Mobile NO:	9848971114
PRIVACY:	Y
Email ID:	thodupunooribadri@gmail.com
Password	Badri1234
OTP	147

```
Please check your email,OTP has been generated
 Inserted 1rows successfully
```

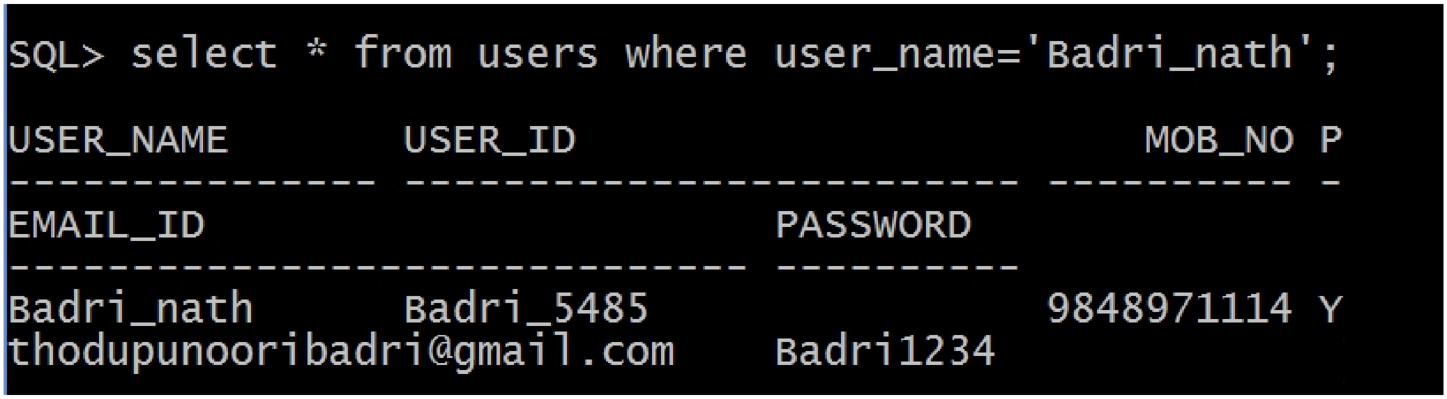


Twitter SignUp

javacode2428@gmail.com
to me

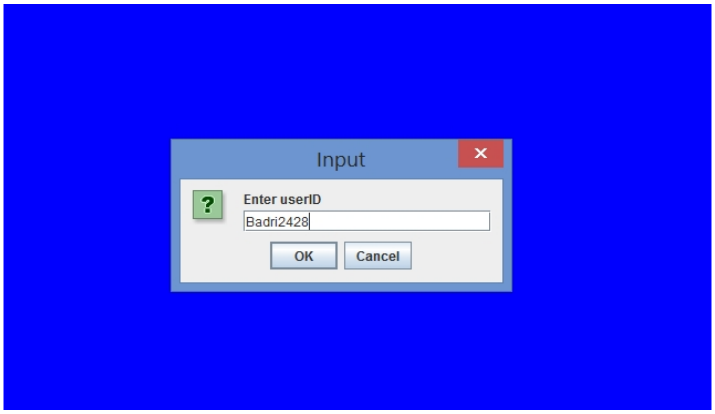
10:03 PM (3 minutes ago)

Thank youl,for joining us on Twitter,Your OTP is 147
Happy Tweeting!

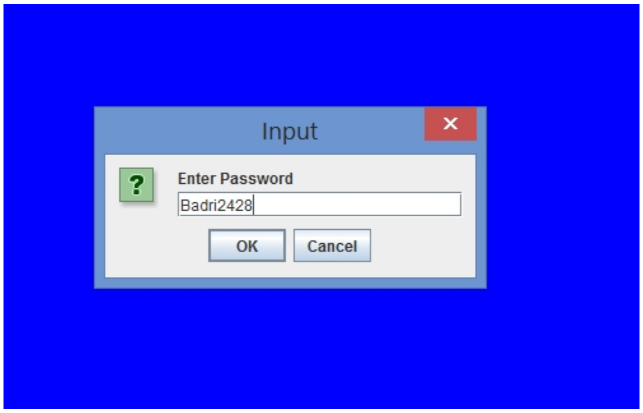


```
SQL> select * from users where user_name='Badri_nath';

USER_NAME       USER_ID                           MOB_NO P
--------------- --------------------------- ---------- -
EMAIL_ID                            PASSWORD
----------------------------------- ----------
Badri_nath      Badri_5485                  9848971114 Y
thodupunooribadri@gmail.com         Badri1234
```

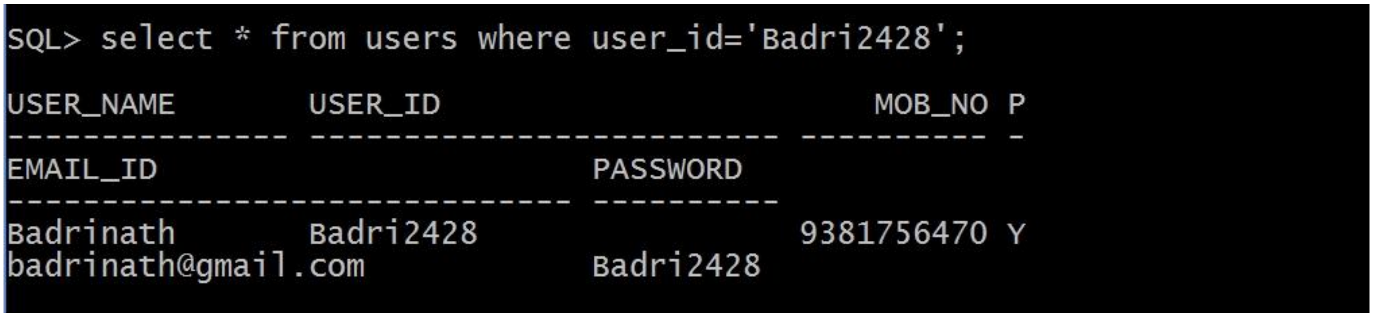
2. A registered user of TDBMS can login to the system by providing his User ID and password as set by him while registering. After successful login, "Home" page for the user is shown from where he can access the different functionalities of ,can avail the tweets made by his followers. Where user can either reply or make a comments to the tweet or user can like and can re tweet a tweet and many more.



Enter userID
Badri2428



Enter Password
Badri2428



```
SQL> select * from users where user_id='Badri2428';

USER_NAME       USER_ID                         MOB_NO P
--------------- ------------------------- ---------- -
EMAIL_ID                       PASSWORD
------------------------------ ----------
Badrinath       Badri2428                 9381756470 Y
badrinath@gmail.com            Badri2428
```

3. Menu bar consists of two menu's named settings and tweet respectively.

Where settings consists of items like,

Update details

Followers

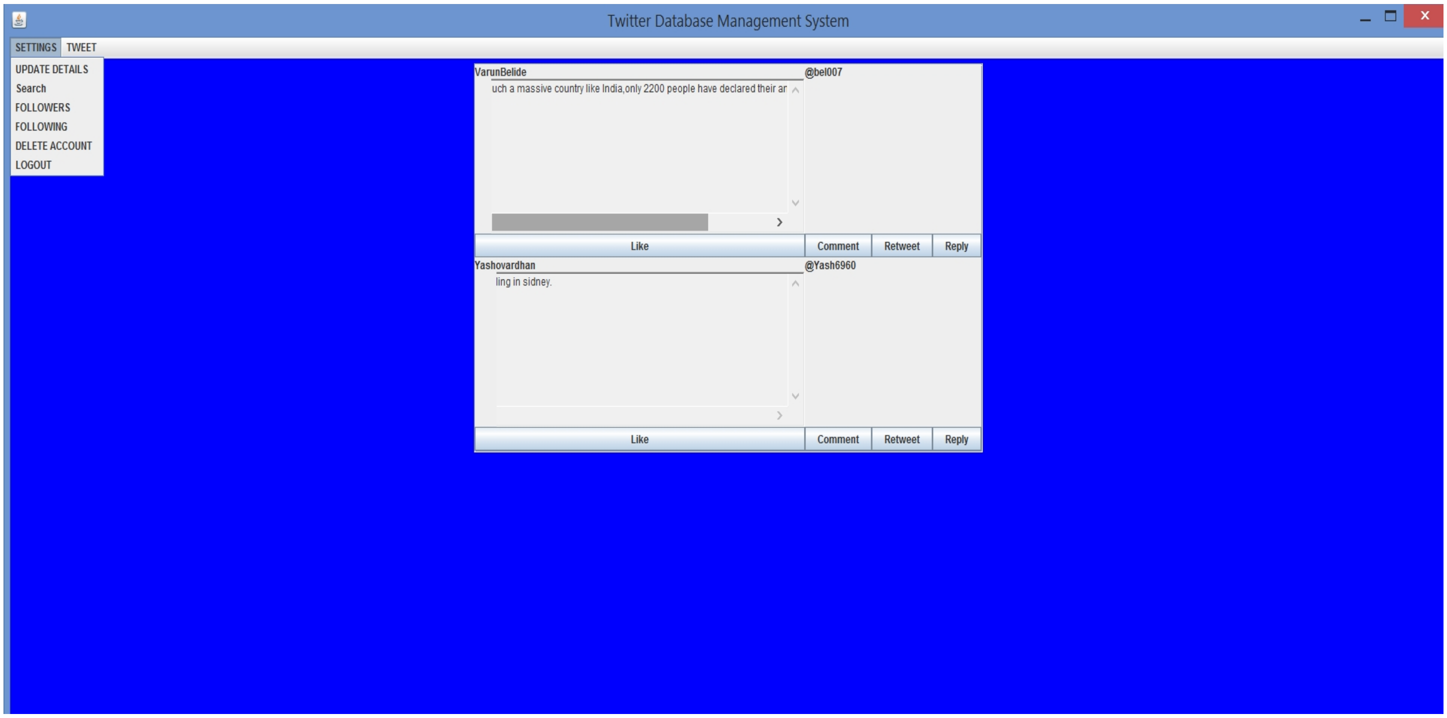
Following

Search

Sign out

Logout

(All these names give their original meaning, So I think there is no need for explaining these terms)



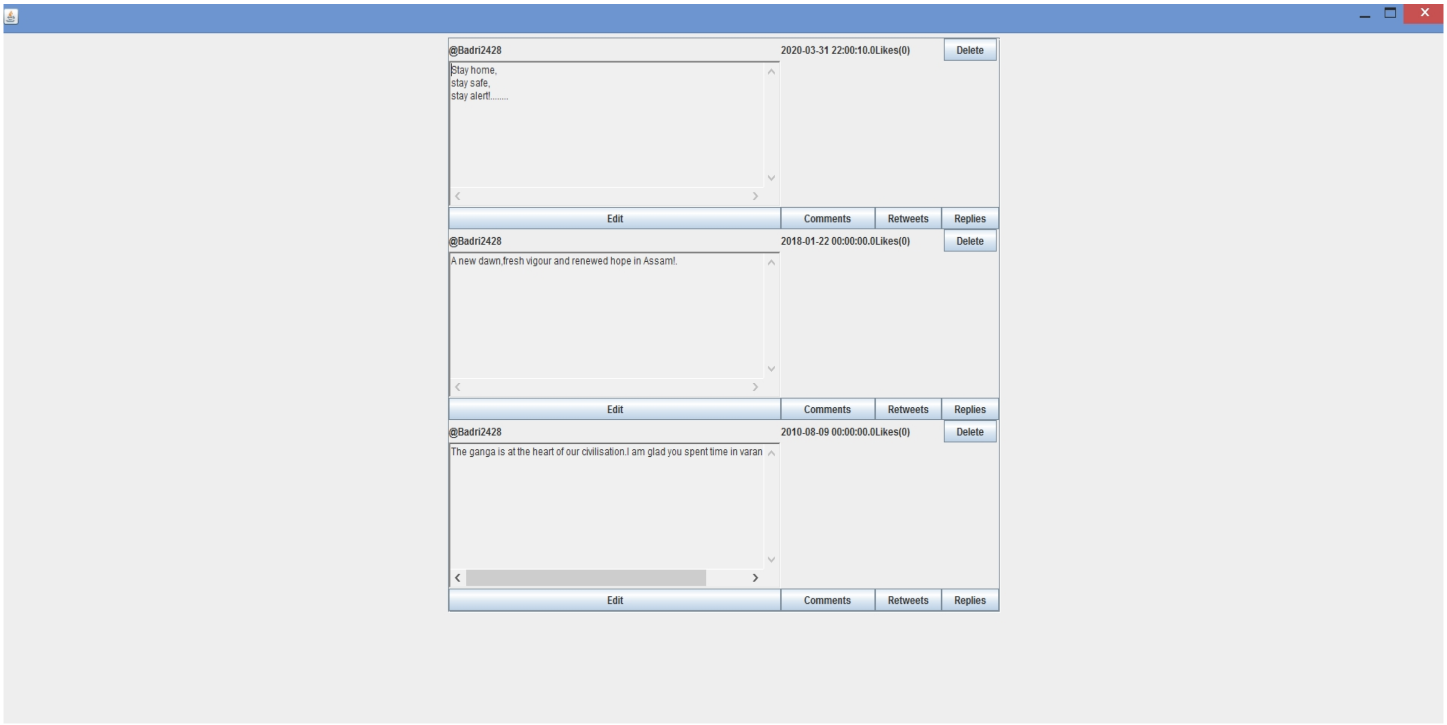
VarunBelide @bel007

uch a massive country like India,only 2200 people have declared their ar

Yashovardhan @Yash6960

ling in sidney.

And tweets consists of items like compose tweet and your tweets where user can see the tweets made by him/her and can check the likes, comments, replies, re tweets to a particular tweet.



@Badri2428 2020-03-31 22:00:10.0Likes(0)

Stay home,
stay safe,
stay alert!........

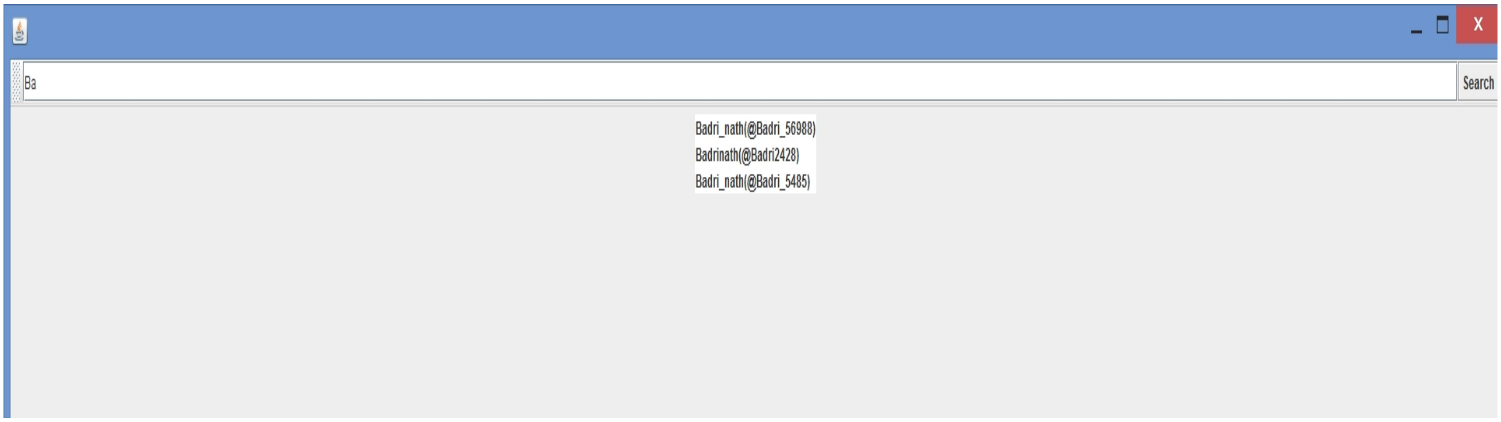
@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 2010-08-09 00:00:00.0Likes(0)

The ganga is at the heart of our civilisation.I am glad you spent time in varan

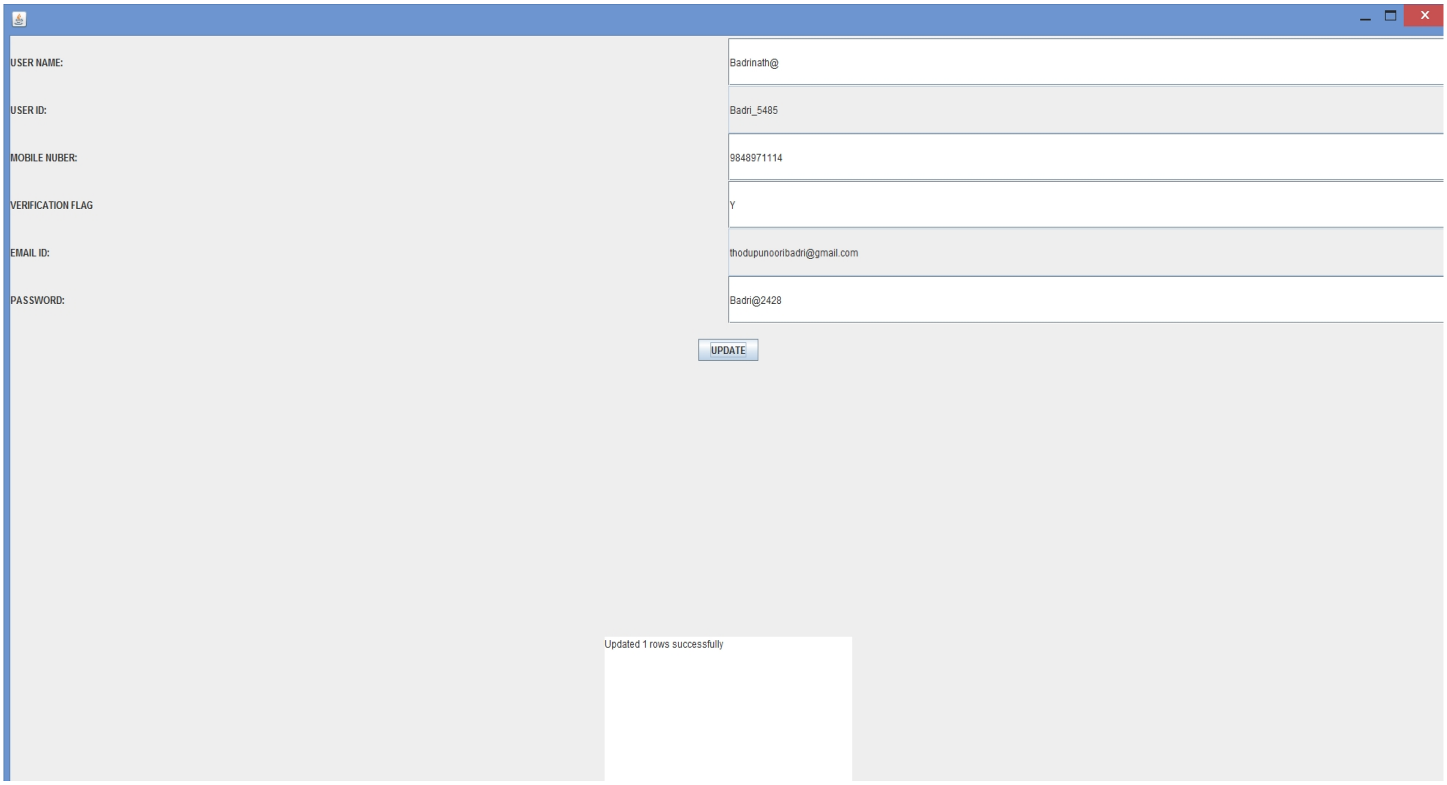
3. Search bar is customized i.e., results change based on text entered which just looks like a search engine in a web browser.



Ba

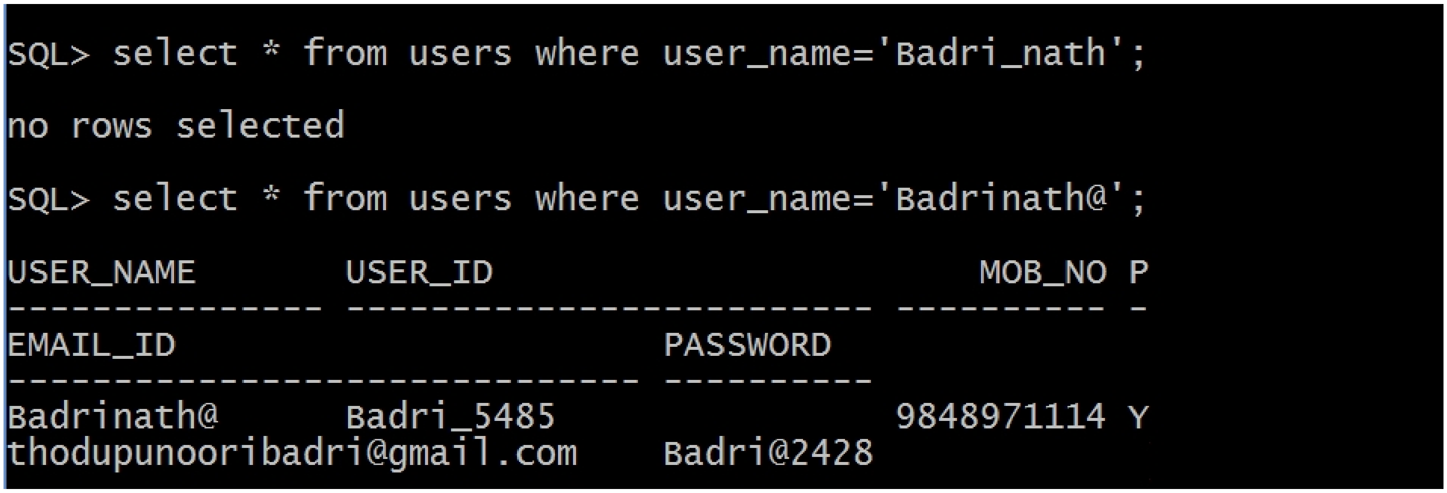
Badri_nath(@Badri_56988)
Badrinath(@Badri2428)
Badri_nath(@Badri_5485)

4.Update Details:



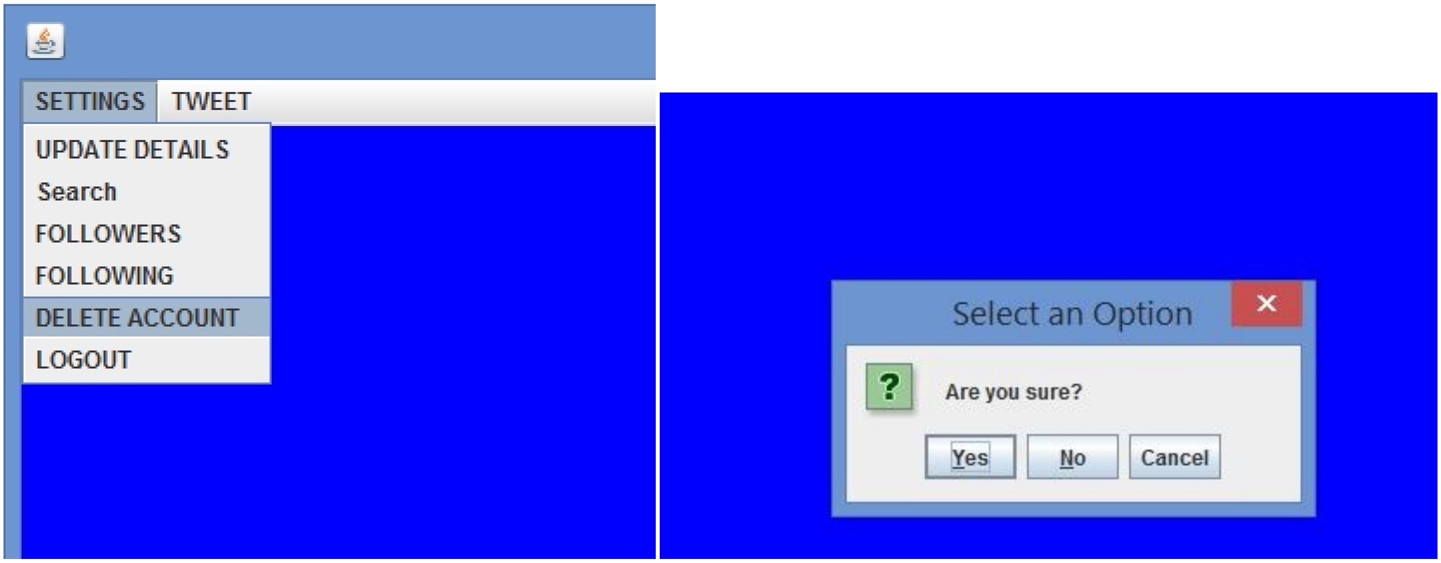
USER NAME:	Badrinath@
USER ID:	Badri_5485
MOBILE NUBER:	9848971114
VERIFICATION FLAG	Y
EMAIL ID:	thodupunooribadri@gmail.com
PASSWORD:	Badri@2428

Updated 1 rows successfully

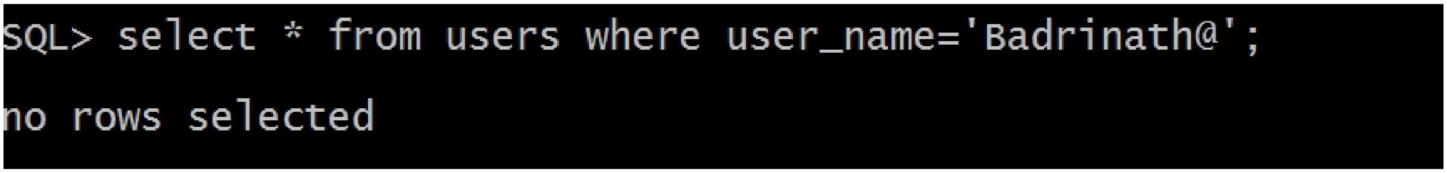


```
SQL> select * from users where user_name='Badri_nath';

no rows selected

SQL> select * from users where user_name='Badrinath@';

USER_NAME       USER_ID                            MOB_NO P
--------------- -------------------------- ---------- -
EMAIL_ID                              PASSWORD
------------------------------ ----------
Badrinath@      Badri_5485                     9848971114 Y
thodupunooribadri@gmail.com    Badri@2428
```

Deleting an account:



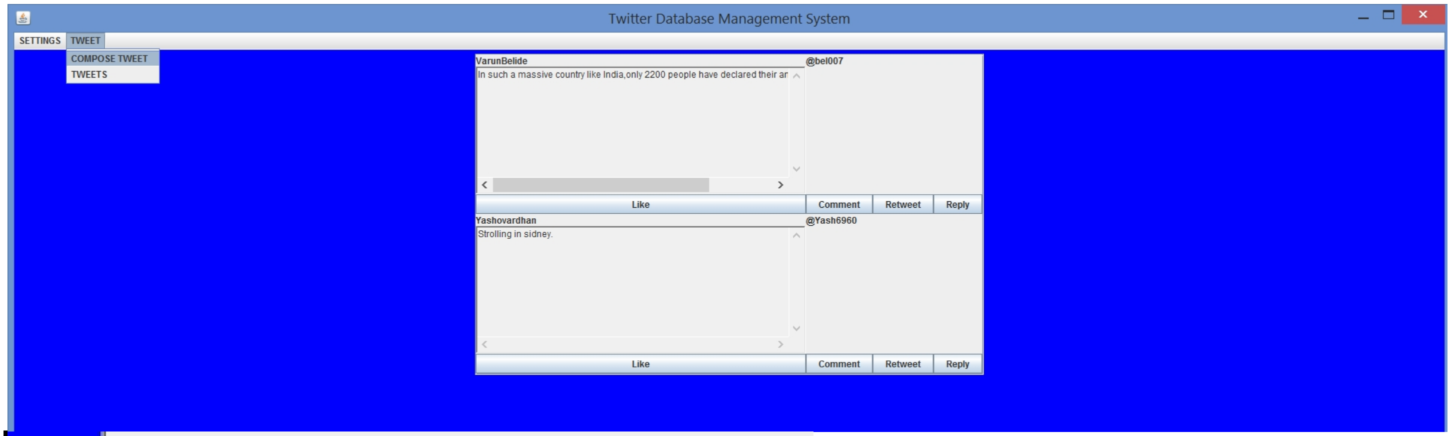
Are you sure?



```
SQL> select * from users where user_name='Badrinath@';

no rows selected
```

TWEETS:

You can even compose a tweet via following , home>tweet>compose tweet, then post the tweet you wanted.

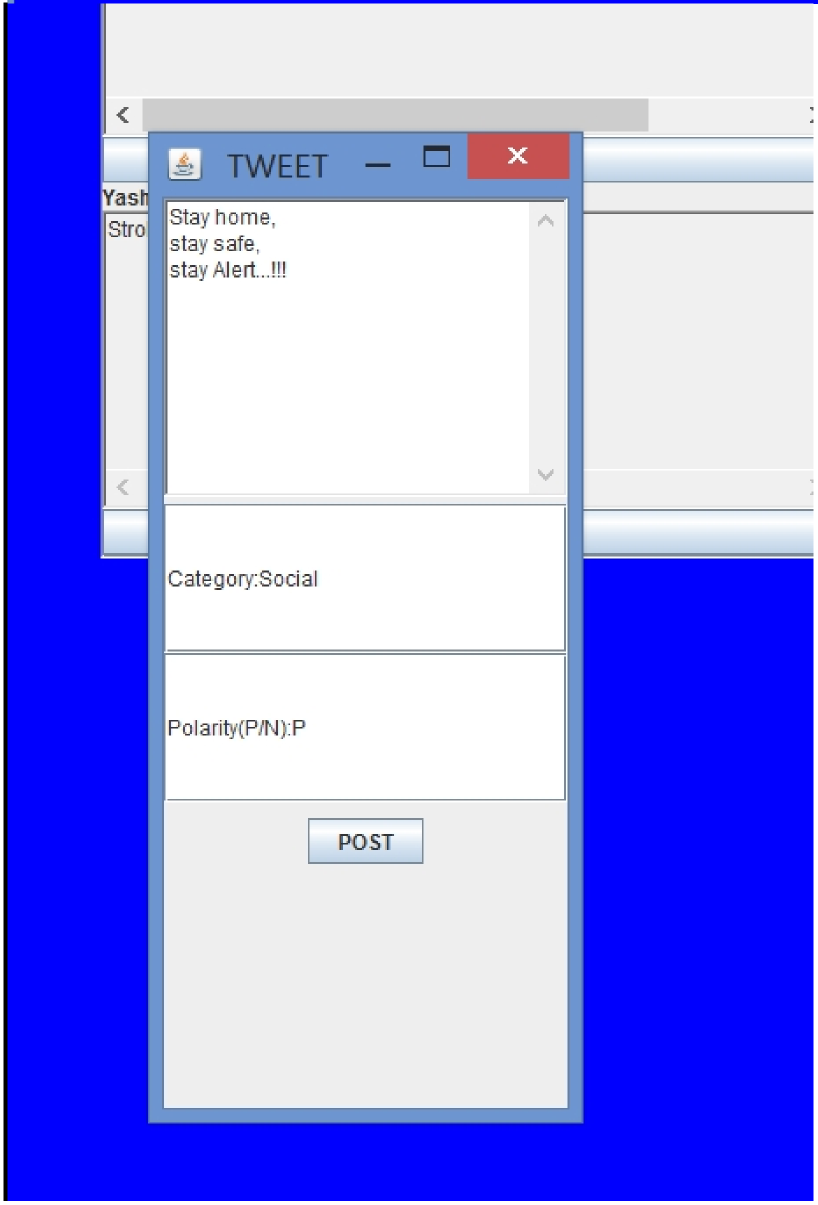


VarunBelide @bel007

In such a massive country like India,only 2200 people have declared their ar

Yashovardhan @Yash6960

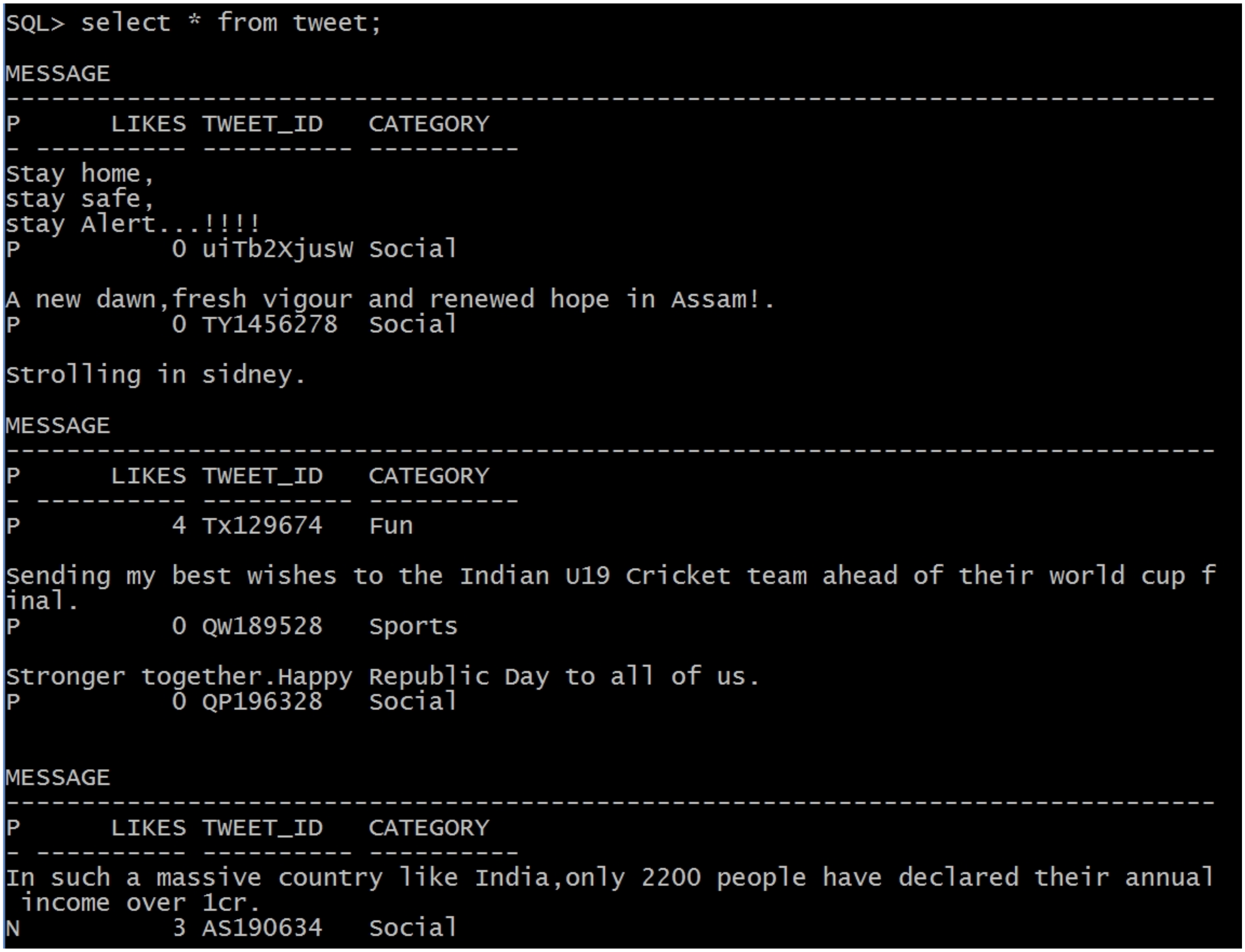
Strolling in sidney.



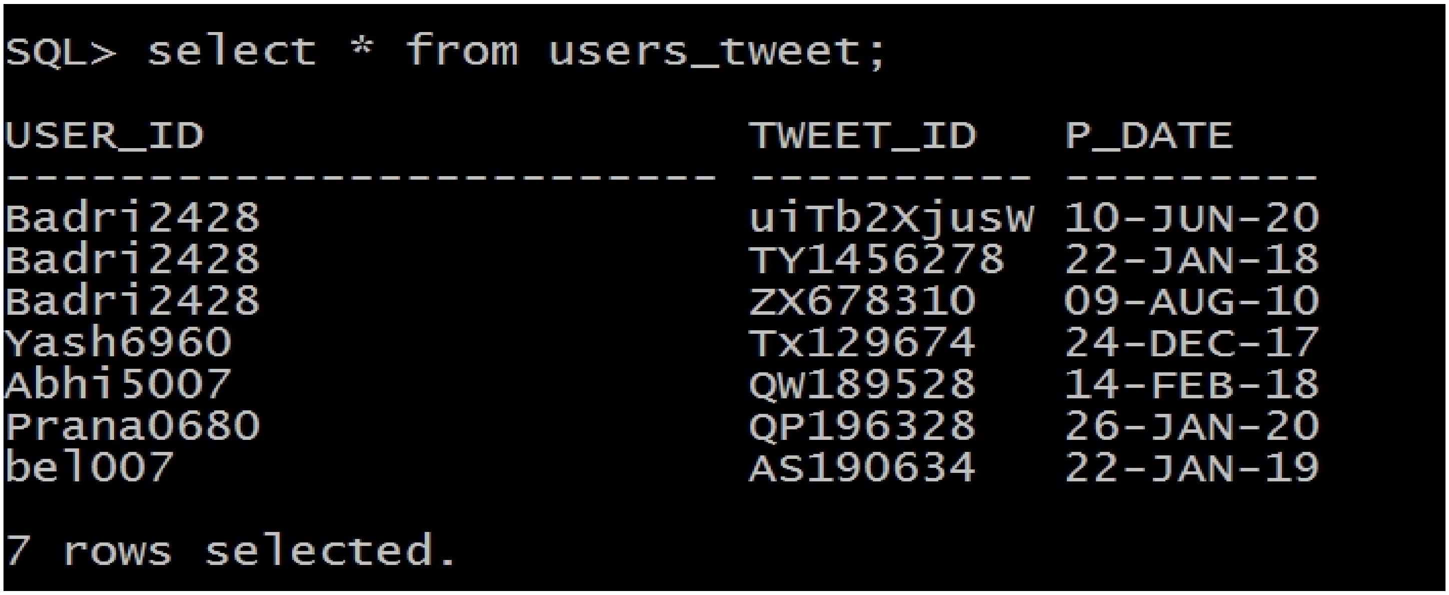
Stay home,
stay safe,
stay Alert...!!!

Category:Social

Polarity(P/N):P



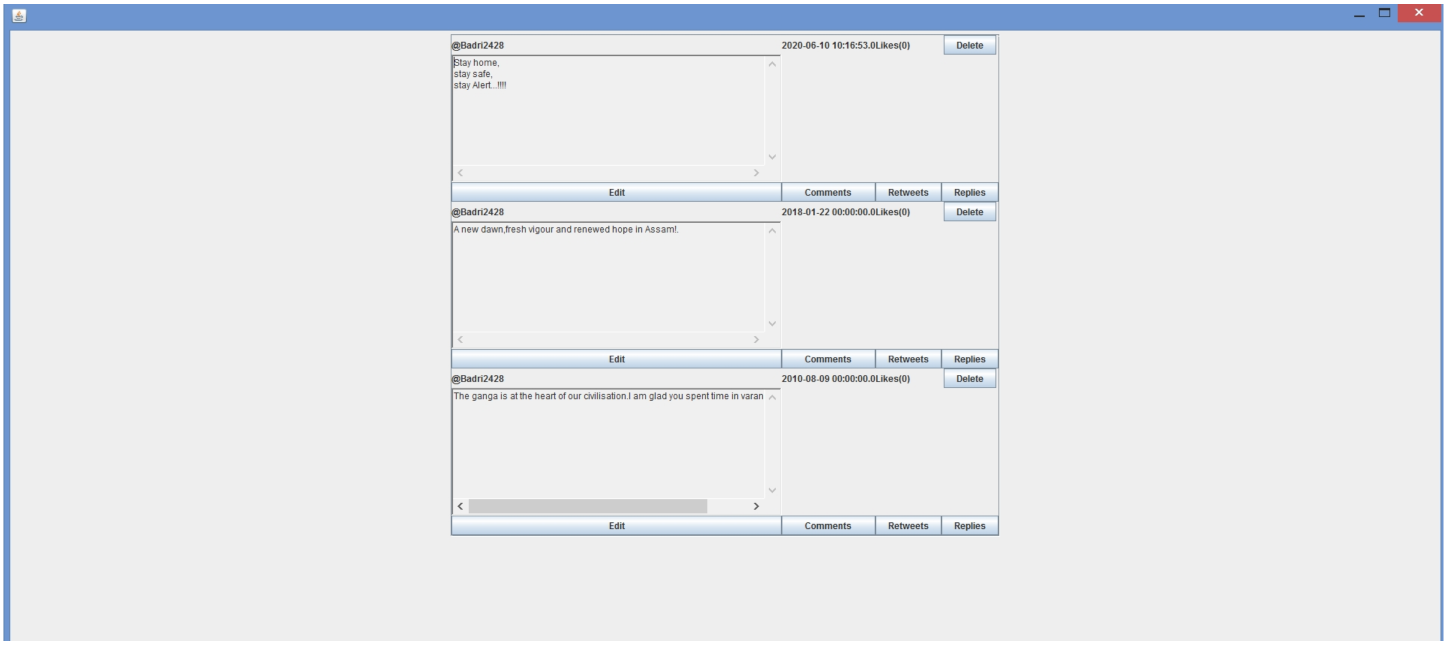
```
SQL> select * from tweet;

MESSAGE
--------------------------------------------------------------------------------
P      LIKES TWEET_ID   CATEGORY
- ---------- ---------- ----------
Stay home,
stay safe,
stay Alert...!!!!
P          0 uiTb2XjusW Social

A new dawn,fresh vigour and renewed hope in Assam!.
P          0 TY1456278  Social

Strolling in sidney.

MESSAGE
--------------------------------------------------------------------------------
P      LIKES TWEET_ID   CATEGORY
- ---------- ---------- ----------
P          4 Tx129674   Fun

Sending my best wishes to the Indian U19 Cricket team ahead of their world cup f
inal.
P          0 QW189528   Sports

Stronger together.Happy Republic Day to all of us.
P          0 QP196328   Social


MESSAGE
--------------------------------------------------------------------------------
P      LIKES TWEET_ID   CATEGORY
- ---------- ---------- ----------
In such a massive country like India,only 2200 people have declared their annual
 income over 1cr.
N          3 AS190634   Social
```



```
SQL> select * from users_tweet;

USER_ID                        TWEET_ID   P_DATE
------------------------------ ---------- ---------
Badri2428                      uiTb2XjusW 10-JUN-20
Badri2428                      TY1456278  22-JAN-18
Badri2428                      ZX678310   09-AUG-10
Yash6960                       Tx129674   24-DEC-17
Abhi5007                       QW189528   14-FEB-18
Prana0680                      QP196328   26-JAN-20
bel007                         AS190634   22-JAN-19

7 rows selected.
```

User can check his tweets via,

home>tweet>tweets, here say user_id is Badri2428.



@Badri2428 2020-06-10 10:16:53.0Likes(0)

Stay home,
stay safe,
stay Alert...!!!!

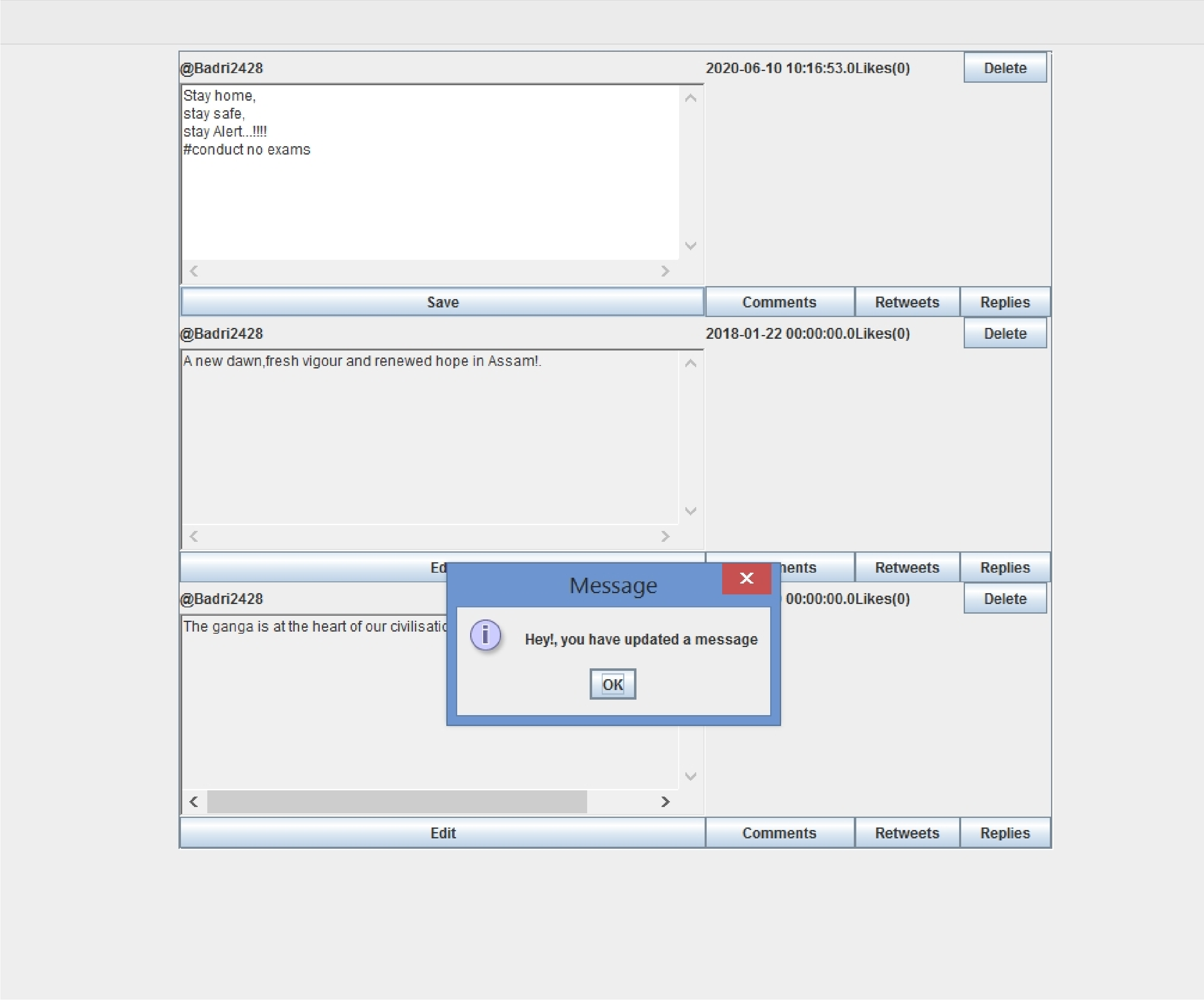
@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 2010-08-09 00:00:00.0Likes(0)

The ganga is at the heart of our civilisation.I am glad you spent time in varan

User can even edit his tweet.



@Badri2428 2020-06-10 10:16:53.0Likes(0)

Stay home,
stay safe,
stay Alert...!!!!
#conduct no exams

@Badri2428 2018-01-22 00:00:00.0Likes(0)

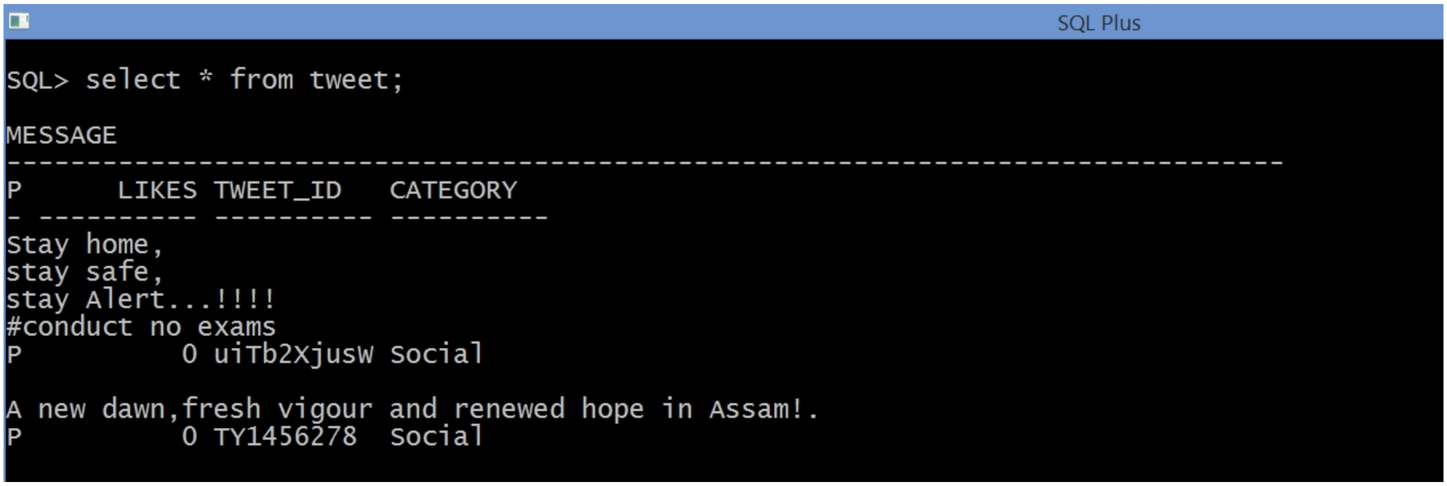
A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 00:00:00.0Likes(0)

The ganga is at the heart of our civilisatic

Message

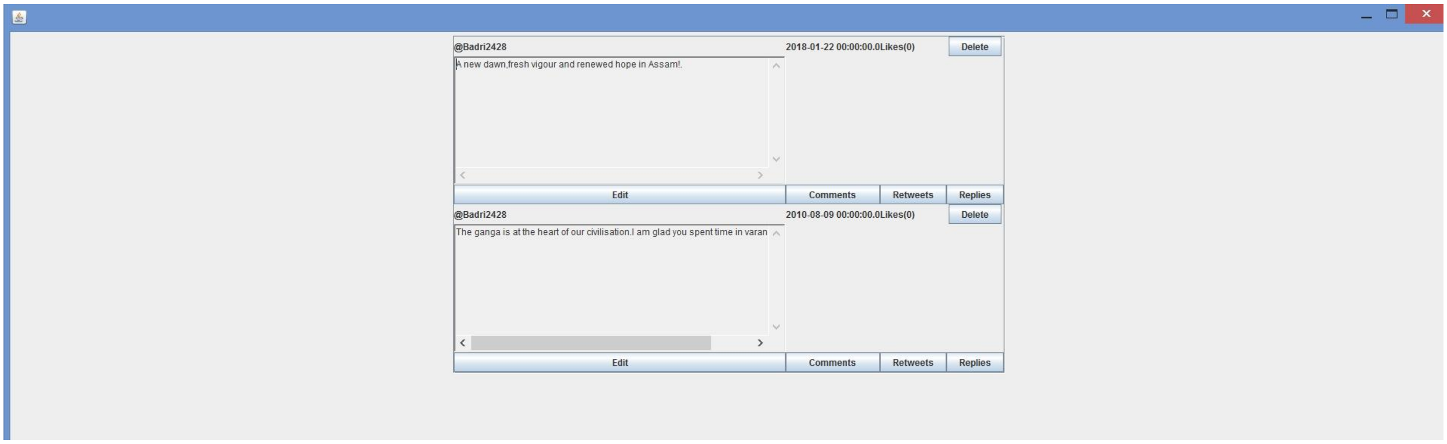
Hey!, you have updated a message



SQL Plus

```
SQL> select * from tweet;

MESSAGE
--------------------------------------------------------------------------------
P      LIKES TWEET_ID   CATEGORY
- ---------- ---------- ----------
Stay home,
stay safe,
stay Alert...!!!!
#conduct no exams
P          0 uiTb2XjusW Social

A new dawn,fresh vigour and renewed hope in Assam!.
P          0 TY1456278  Social
```

Look, the message has been updated and even you can delete a tweet just by clicking on delete button on a specific tweet you wanted to delete.

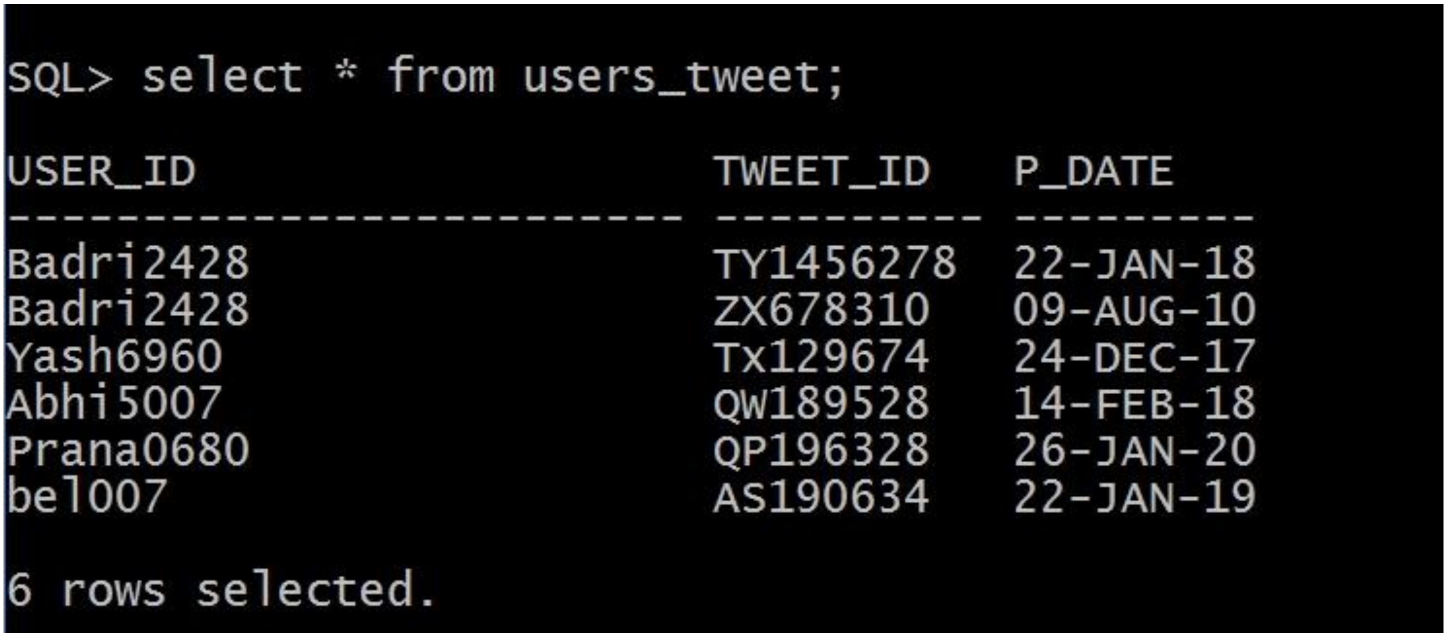


@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 2010-08-09 00:00:00.0Likes(0)

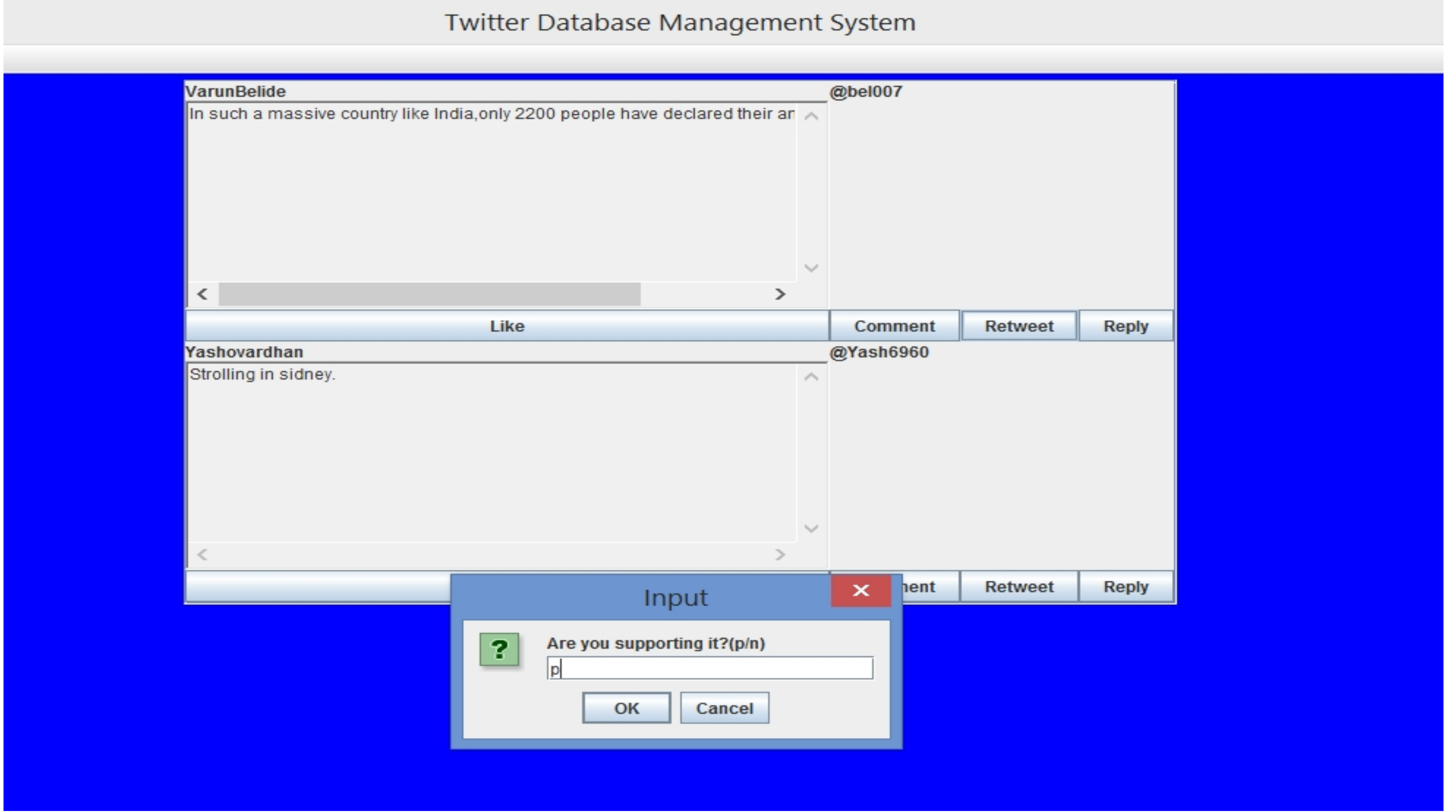
The ganga is at the heart of our civilisation.I am glad you spent time in varan



```
SQL> select * from users_tweet;

USER_ID                   TWEET_ID   P_DATE
------------------------- ---------- ---------
Badri2428                 TY1456278  22-JAN-18
Badri2428                 ZX678310   09-AUG-10
Yash6960                  Tx129674   24-DEC-17
Abhi5007                  QW189528   14-FEB-18
Prana0680                 QP196328   26-JAN-20
bel007                    AS190634   22-JAN-19

6 rows selected.
```

Like, comment and retweet works like this...,,,

Coming to Retweet: When a user retweet's corresponding likes and comments will be reflected onto your tweet. Once you retweet , the text on the button changes to retweeted and clicking on it again, user can undo his retweet.



Twitter Database Management System

VarunBelide @bel007

In such a massive country like India,only 2200 people have declared their ar

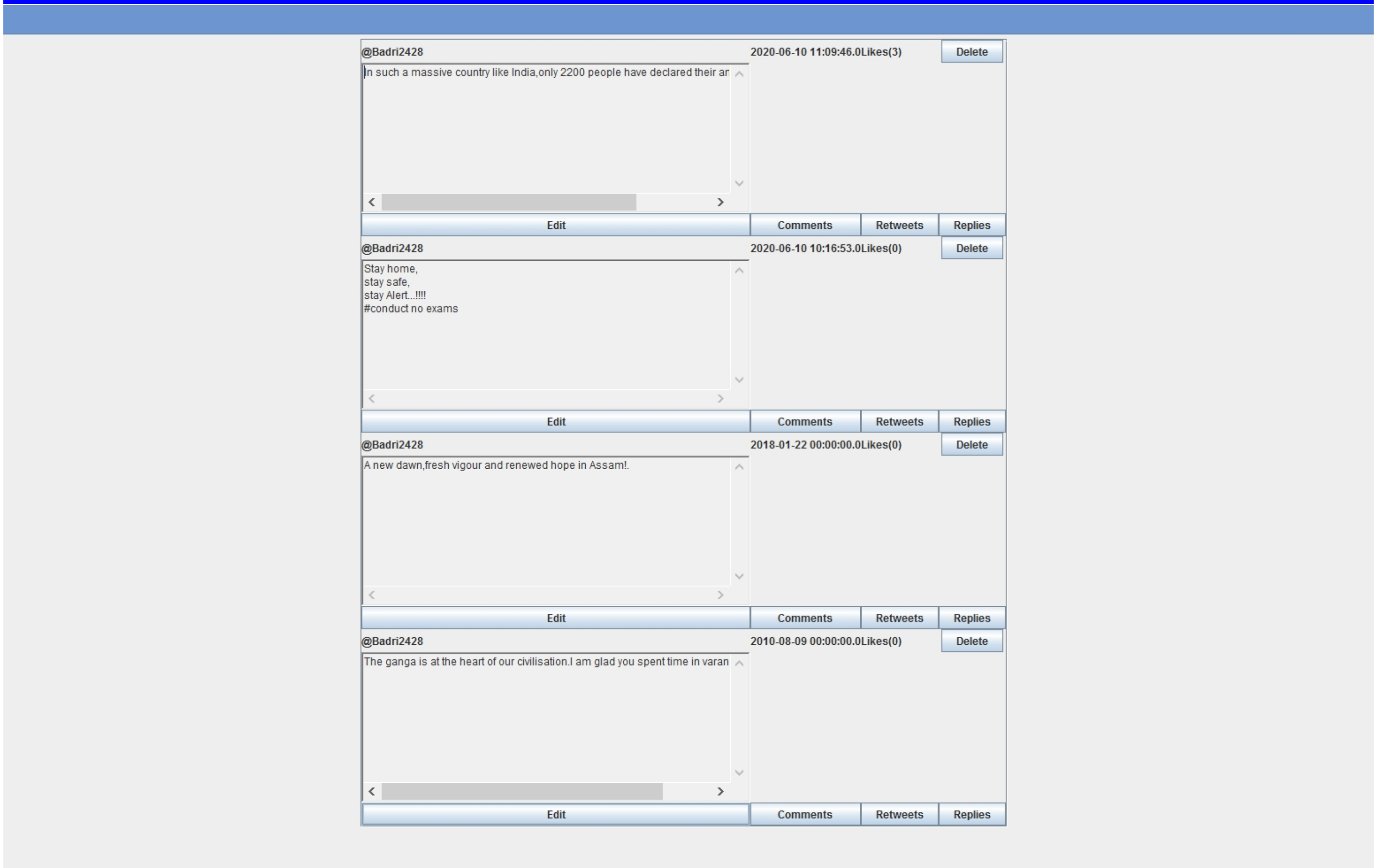
Yashovardhan @Yash6960

Strolling in sidney.

Input

Are you supporting it?(p/n)

p



@Badri2428 2020-06-10 11:09:46.0Likes(3)

In such a massive country like India,only 2200 people have declared their ar

@Badri2428 2020-06-10 10:16:53.0Likes(0)

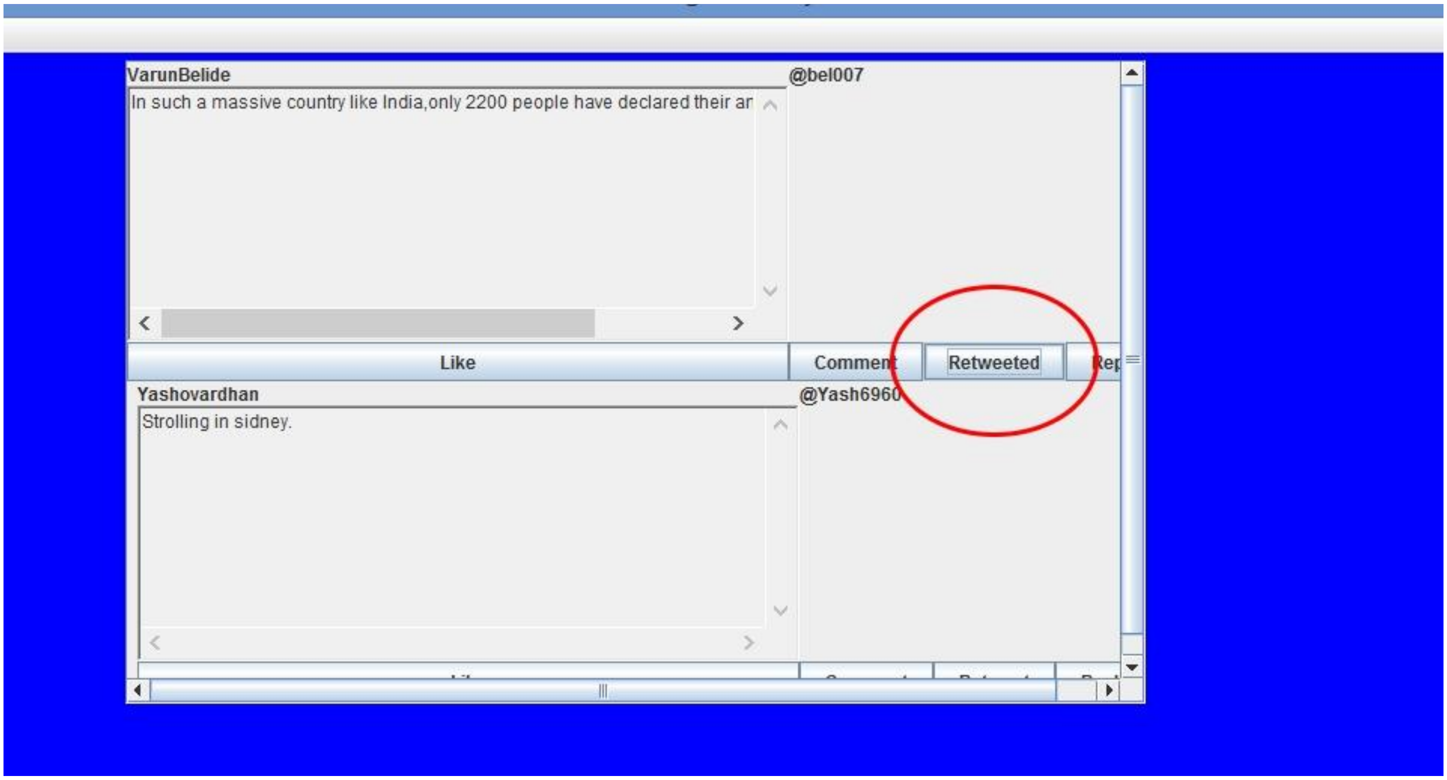
Stay home,
stay safe,
stay Alert...!!!!
#conduct no exams

@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 2010-08-09 00:00:00.0Likes(0)

The ganga is at the heart of our civilisation.I am glad you spent time in varan

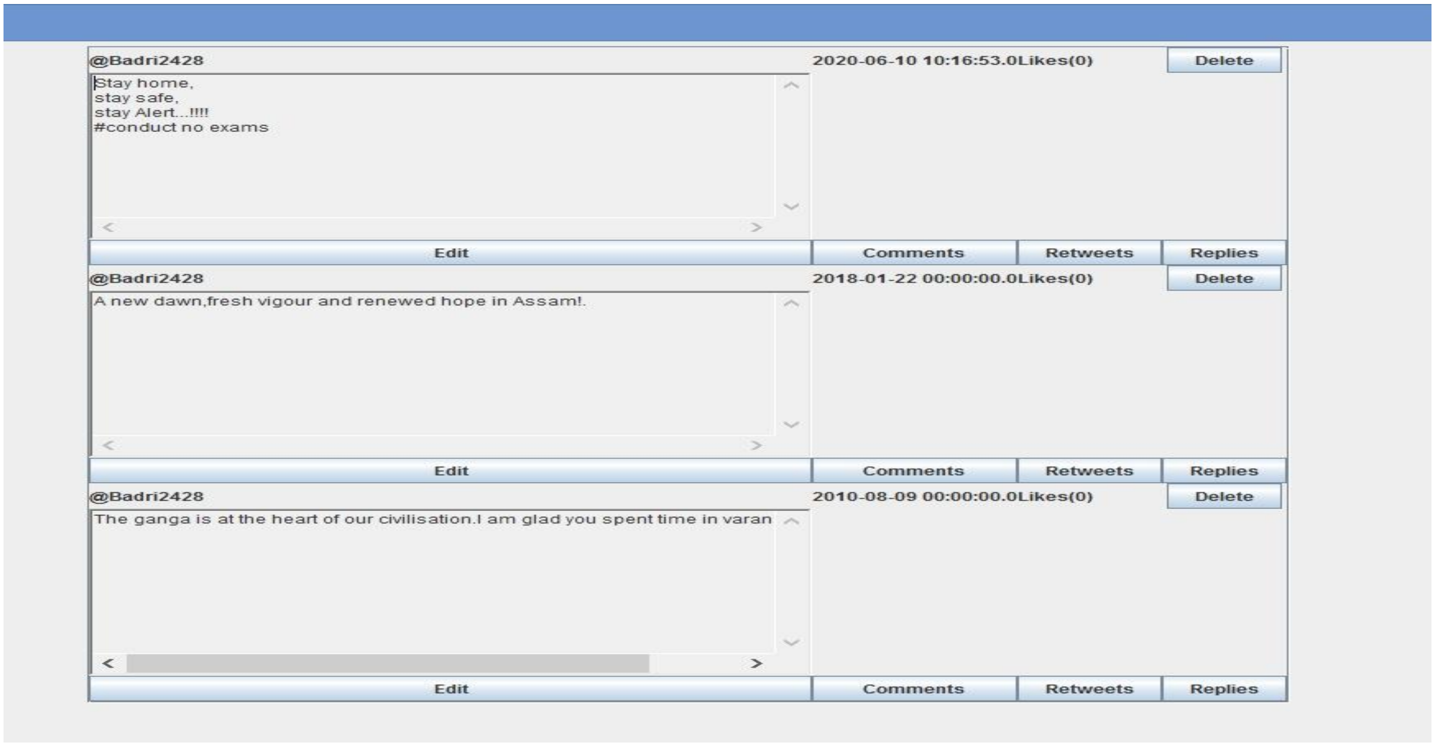


VarunBelide @bel007

In such a massive country like India,only 2200 people have declared their ar

Yashovardhan @Yash6960

Strolling in sidney.



@Badri2428 2020-06-10 10:16:53.0Likes(0)

Stay home,
stay safe,
stay Alert...!!!!
#conduct no exams

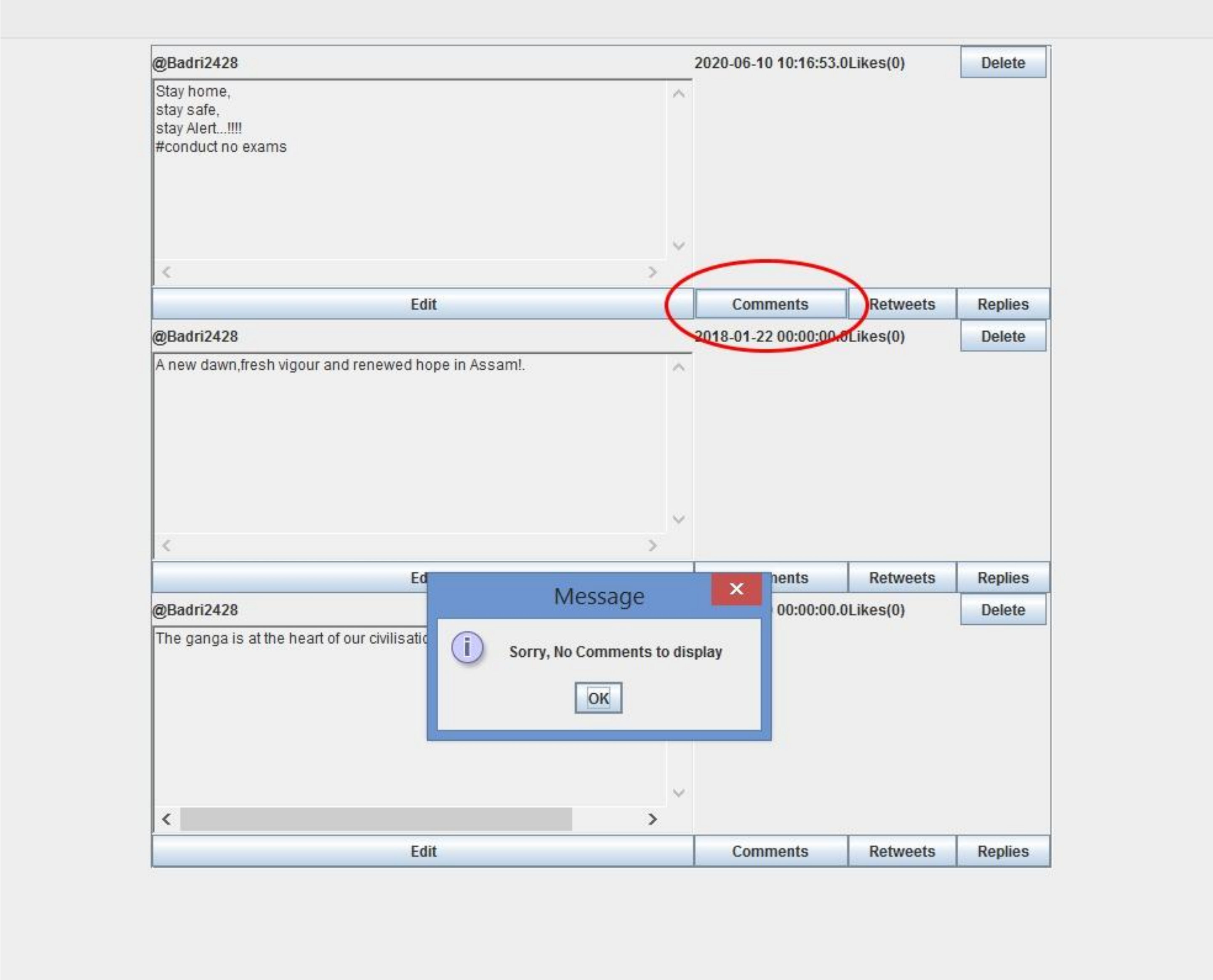
@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 2010-08-09 00:00:00.0Likes(0)

The ganga is at the heart of our civilisation.I am glad you spent time in varan

Also user can check for replies for his tweets from his/her followers, lets check for comments for a particular tweet.



@Badri2428 2020-06-10 10:16:53.0Likes(0)

Stay home,
stay safe,
stay Alert...!!!!
#conduct no exams

@Badri2428 2018-01-22 00:00:00.0Likes(0)

A new dawn,fresh vigour and renewed hope in Assam!.

@Badri2428 00:00:00.0Likes(0)

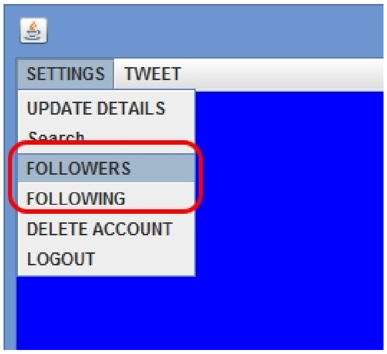
The ganga is at the heart of our civilisatio

Message

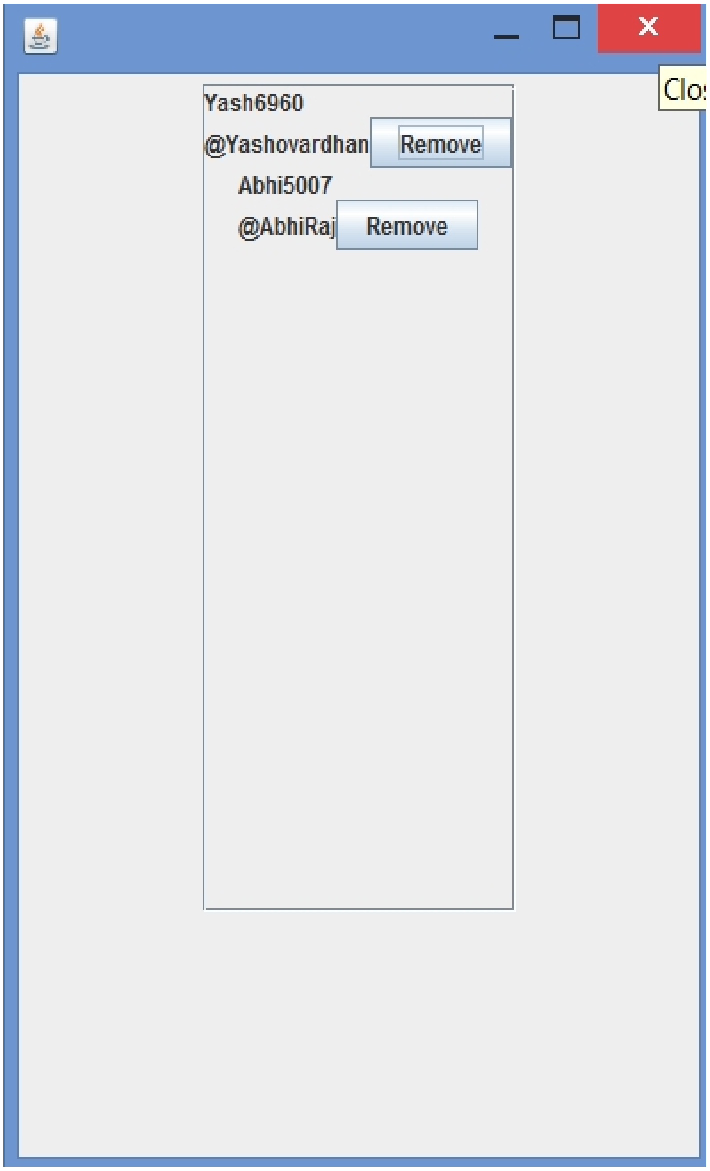
Sorry, No Comments to display

Now a user can check for his followers and following via..,,,

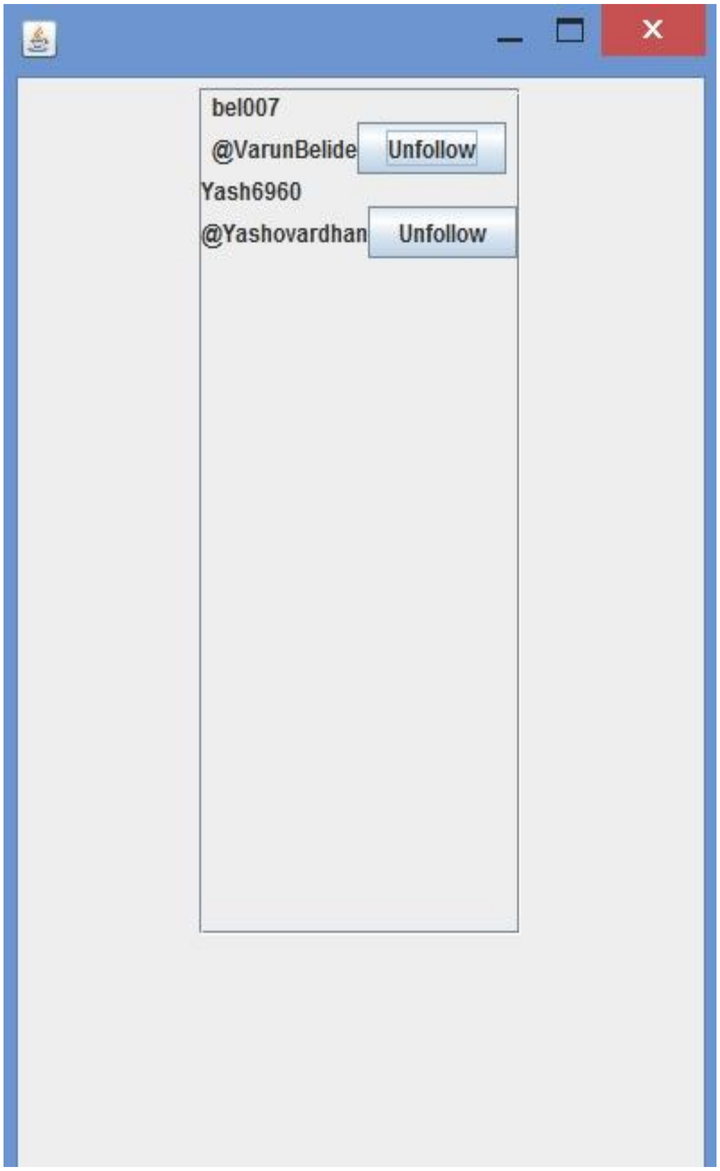
home>settings>followers/following



You can follow someone just by searching his/her username/userid in the search bar, Already shown.



Yash6960
@Yashovardhan
Abhi5007
@AbhiRaj



bel007
@VarunBelide
Yash6960
@Yashovardhan

User can remove users from following and followers just by clicking on unfollow and remove buttons.

CONCLUSION:

Thus, a Java AWT,SWING based twitter simulation is being created, which is connected to the Oracle 11g database. Therefore, all the operations performed are directly updated on the respective tables created in the database.

REFERENCES:

https://buffer.com/library/twitter-analytics/
https://twitter.com/explore
https://docs.oracle.com/javase/8/docs/api/
https://www.javatpoint.com/dbms-tutorial