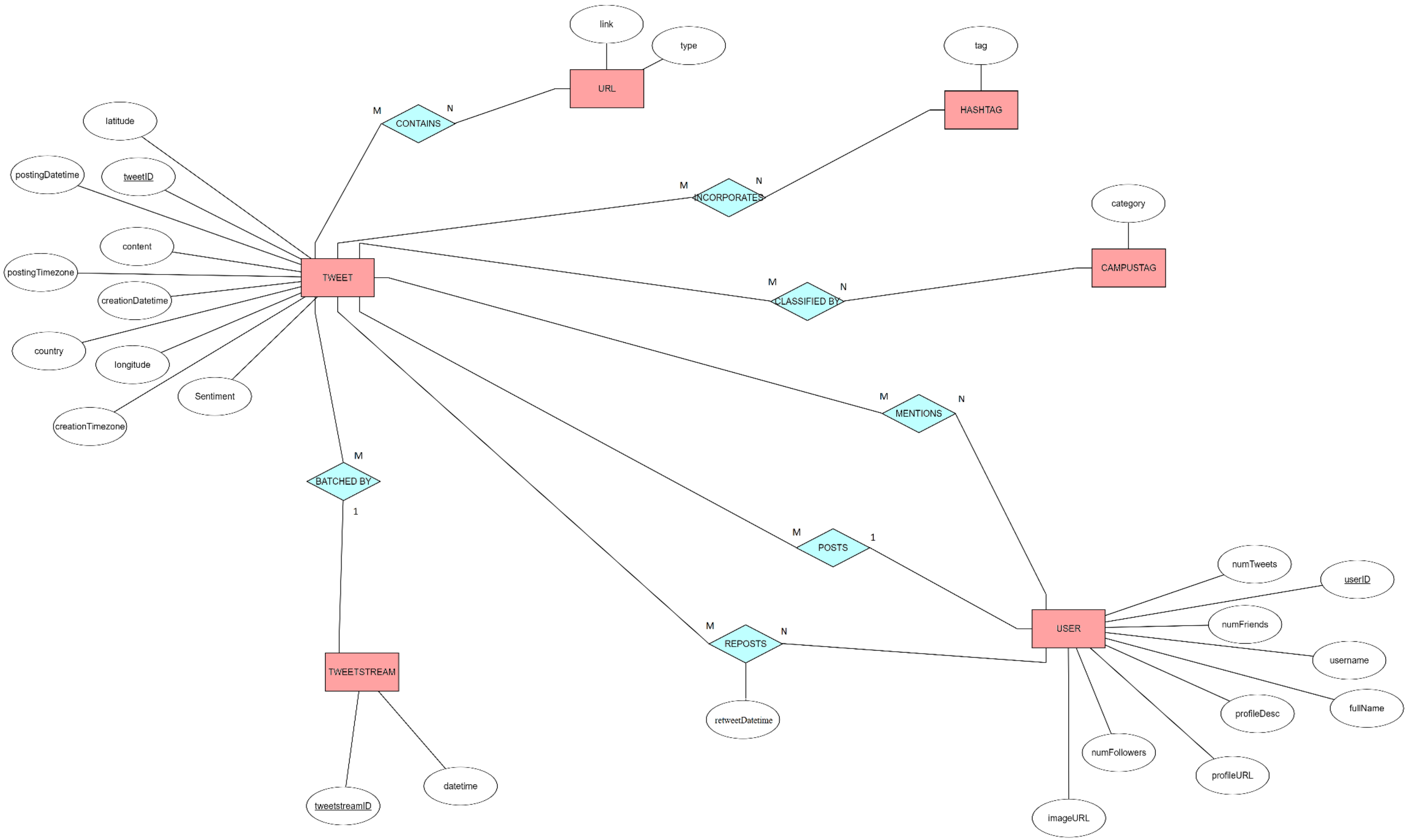
ASSIGNMENT 1

Kaustubh Jagtap – A0168820B

NATIONAL UNIVERSITY OF SINGAPORE

ER DIAGRAM


link
type
URL
tag
HASHTAG
M
CONTAINS
N
latitude
postingDatetime
tweetID
M
INCORPORATES
N
category
CAMPUSTAG
content
postingTimezone
TWEET
creationDatetime
M
CLASSIFIED BY
N
country
longitude
Sentiment
creationTimezone
M
MENTIONS
N
M
BATCHED BY
1
M
POSTS
1
numTweets
userID
numFriends
M
REPOSTS
N
USER
username
TWEETSTREAM
retweetDatetime
profileDesc
fullName
numFollowers
profileURL
datetime
tweetstreamID
imageURL

RELATIONS

TWEETSTREAM (tweetstreamID, date_time)

USER (userID, username, fullName, numFriends, numFollowers, profileDesc, profileURL, imageURL, numTweets)

TWEET (tweetID, content, creationDatetime, creationTimezone, postingDatetime, postingTimezone, country, longitude, latitude, sentiment, **tweetStreamID**, **userID**)

MENTION (**userID**, **tweetID**)

REPOST (**tweetID**, **userID**, retweetDatetime)

CAMPUSTAG (**tweetID**, category)

URL (**tweetID**, link, type)

HASHTAG (**tweetID**, tag)

BOLD – Foreign Key

Underline – Primary Key

DATA TYPES

Relation	Attribute	Comment	Data Type
TWEETSTREAM	tweetstreamID	Primary	BIGINT
	date_time		DATETIME
TWEET	tweetID	Primary	BIGINT
	content		TEXT
	creationDatetime		DATETIME
	creationTimezone		TINYINT
	postingDatetime		DATETIME
	postingTimezone		TINYINT
	country		VARCHAR (74)
	longitude		DECIMAL (9,6)
	latitude		DECIMAL (9,6)
	sentiment		TINYINT
	tweetStreamID	Foreign	BIGINT
	userID	Foreign	BIGINT
USER	userID	Primary	BIGINT
	username		VARCHAR (50)
	fullName		VARCHAR (100)
	profileURL		VARCHAR (100)
	profileDesc		TEXT
	numFollowers		INT
	numFriends		INT
	imageURL		VARCHAR (100)
	numTweets		INT
MENTION	userID	Primary, Foreign	BIGINT
	tweetID	Primary, Foreign	BIGINT
REPOST	tweetID	Primary, Foreign	BIGINT
	userID	Primary, Foreign	BIGINT
	retweetDateTime		DATETIME
CAMPUSTAG	tweetID	Primary, Foreign	BIGINT
	category	Primary	CHAR (15)
URL	tweetID	Primary, Foreign	BIGINT
	link	Primary	VARCHAR (100)
	type		TINYINT
HASHTAG	tweetID	Primary, Foreign	BIGINT
	tag	Primary	VARCHAR (50)

Assumptions and Clarifications

- TWEET.country will be stored as VARCHAR (74) since the longest country name is 74 characters.

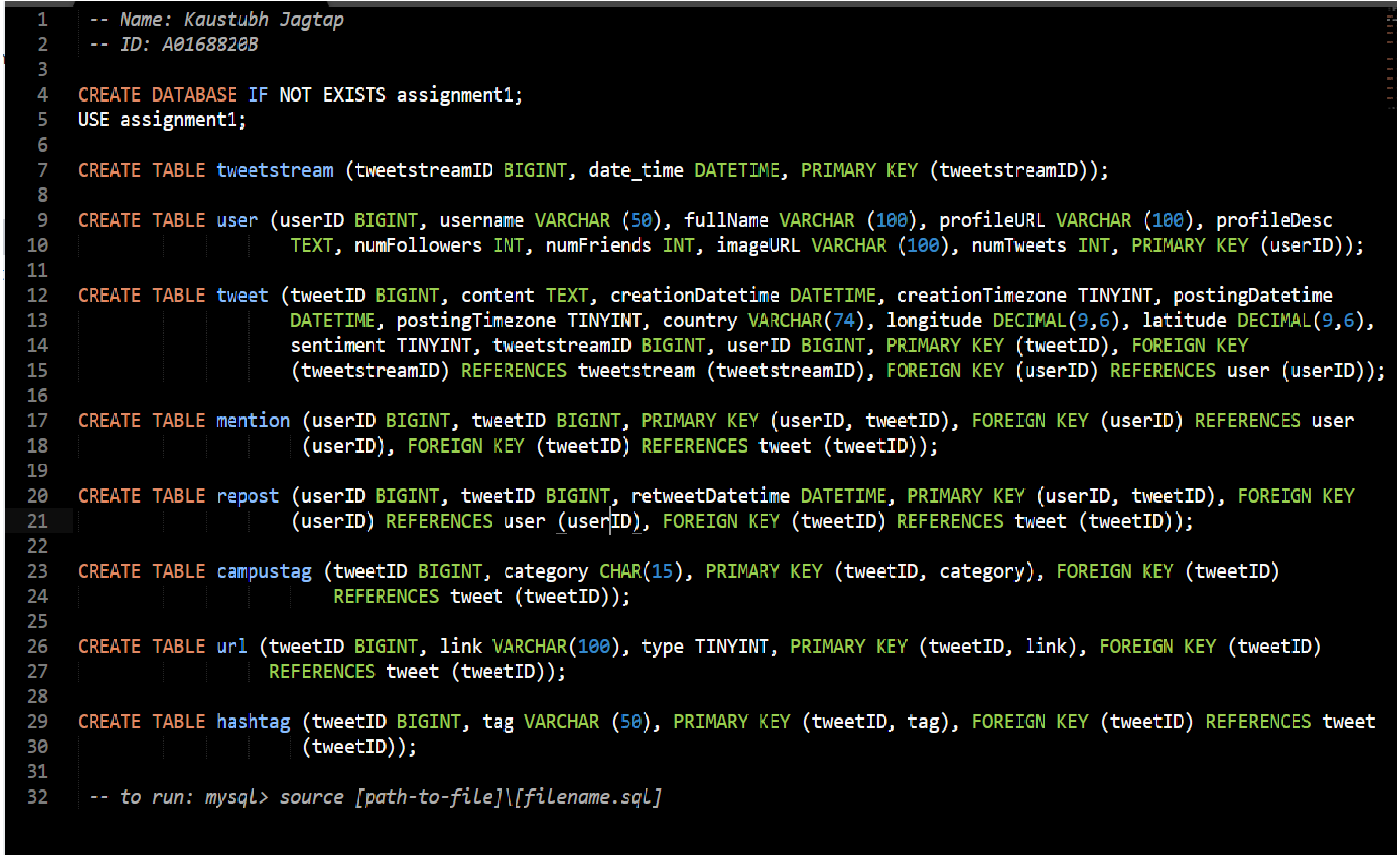
- TWEET.content of the tweet has to be stored as TEXT since it can go up to 280 characters.

- URL.type is for now just 0 and 1, to signify whether it is external or twitter media. This leaves space to introduce a higher level of classification (say within external), and this can be done by using more integers to signify the types.

- Another table to store followers and friends between users could have been created, but I decided against this since this would make the database unnecessarily large and clunky. (Imagine having to store 500 followers for every person within Singapore, we would have to store the entire user profile of everyone in that network, even globally, and this would have a chain effect). For our purpose of sentiment analysis, simply knowing the number of friends and followers is enough to get a gauge of the person’s influence.

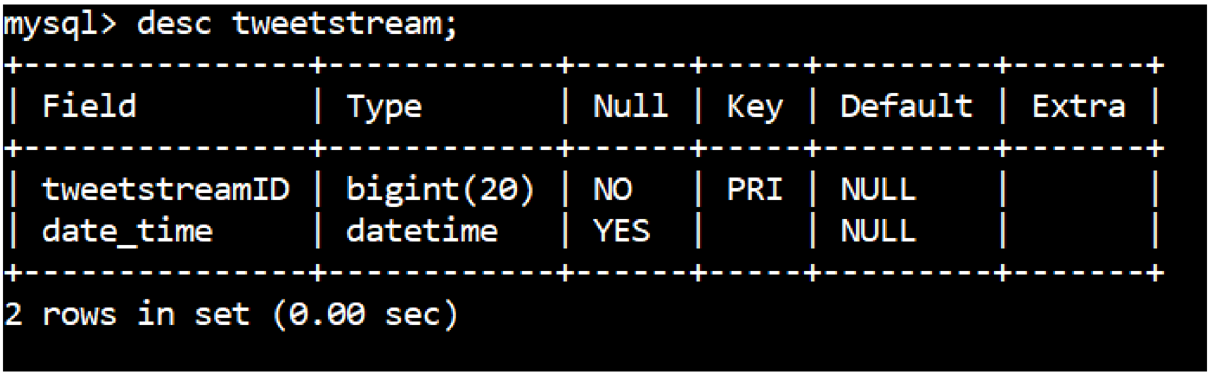
- A separate table has to be created for REPOST because we need a way for the original tweet to be referenced. Also, no additional content or data is created for a retweet, other than the date and time of the new tweet.

SQL CODE

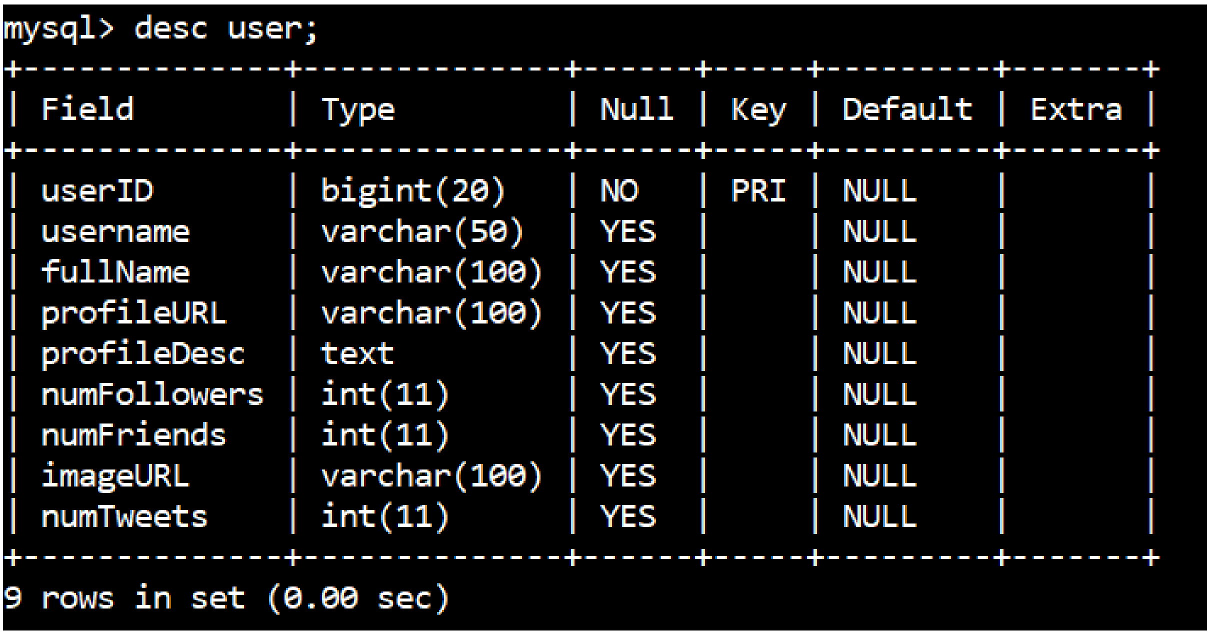


```sql
 -- Name: Kaustubh Jagtap
 -- ID: A0168820B

CREATE DATABASE IF NOT EXISTS assignment1;
USE assignment1;

CREATE TABLE tweetstream (tweetstreamID BIGINT, date_time DATETIME, PRIMARY KEY (tweetstreamID));

CREATE TABLE user (userID BIGINT, username VARCHAR (50), fullName VARCHAR (100), profileURL VARCHAR (100), profileDesc
                    TEXT, numFollowers INT, numFriends INT, imageURL VARCHAR (100), numTweets INT, PRIMARY KEY (userID));

CREATE TABLE tweet (tweetID BIGINT, content TEXT, creationDatetime DATETIME, creationTimezone TINYINT, postingDatetime
                    DATETIME, postingTimezone TINYINT, country VARCHAR(74), longitude DECIMAL(9,6), latitude DECIMAL(9,6),
                    sentiment TINYINT, tweetstreamID BIGINT, userID BIGINT, PRIMARY KEY (tweetID), FOREIGN KEY
                    (tweetstreamID) REFERENCES tweetstream (tweetstreamID), FOREIGN KEY (userID) REFERENCES user (userID));

CREATE TABLE mention (userID BIGINT, tweetID BIGINT, PRIMARY KEY (userID, tweetID), FOREIGN KEY (userID) REFERENCES user
                     (userID), FOREIGN KEY (tweetID) REFERENCES tweet (tweetID));

CREATE TABLE repost (userID BIGINT, tweetID BIGINT, retweetDatetime DATETIME, PRIMARY KEY (userID, tweetID), FOREIGN KEY
                    (userID) REFERENCES user (userID), FOREIGN KEY (tweetID) REFERENCES tweet (tweetID));

CREATE TABLE campustag (tweetID BIGINT, category CHAR(15), PRIMARY KEY (tweetID, category), FOREIGN KEY (tweetID)
                        REFERENCES tweet (tweetID));

CREATE TABLE url (tweetID BIGINT, link VARCHAR(100), type TINYINT, PRIMARY KEY (tweetID, link), FOREIGN KEY (tweetID)
                  REFERENCES tweet (tweetID));

CREATE TABLE hashtag (tweetID BIGINT, tag VARCHAR (50), PRIMARY KEY (tweetID, tag), FOREIGN KEY (tweetID) REFERENCES tweet
                     (tweetID));

 -- to run: mysql> source [path-to-file]\[filename.sql]
```

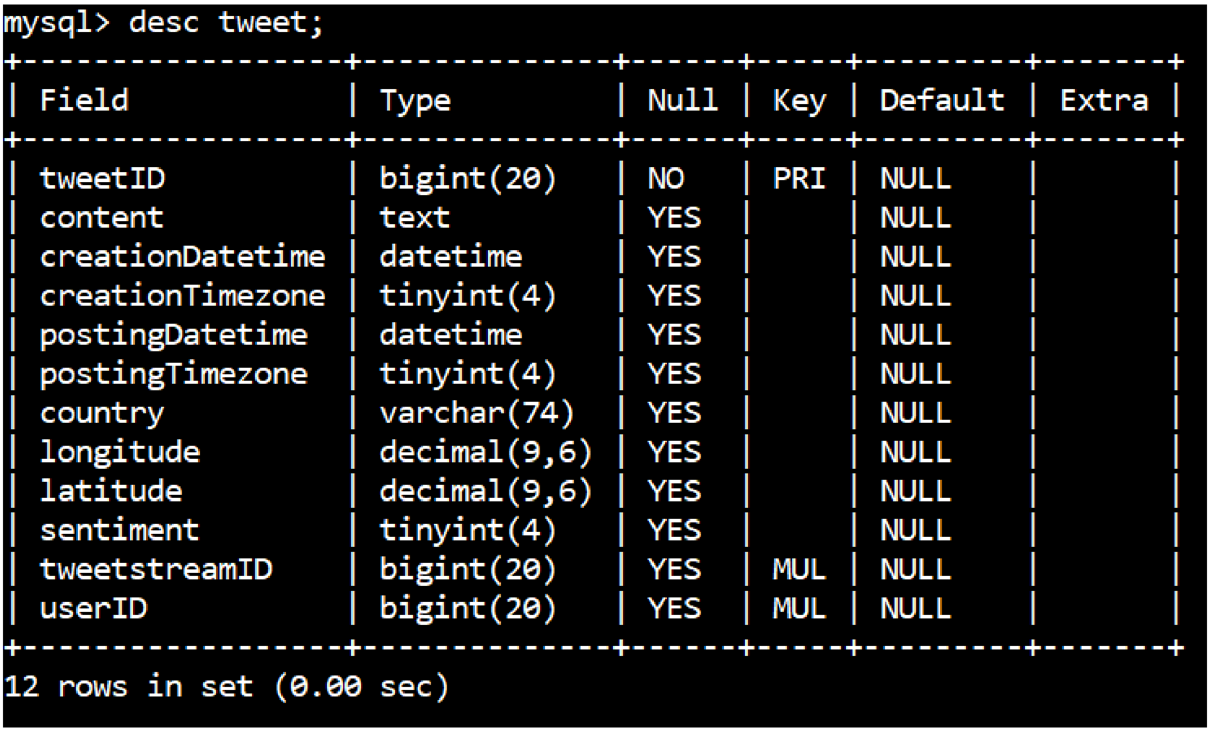
Tables in Database



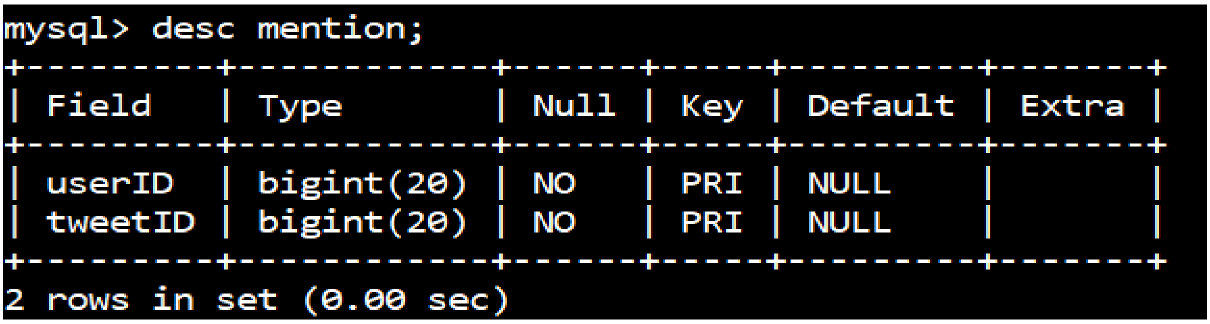
```
mysql> desc tweetstream;
+---------------+------------+------+-----+---------+-------+
| Field         | Type       | Null | Key | Default | Extra |
+---------------+------------+------+-----+---------+-------+
| tweetstreamID | bigint(20) | NO   | PRI | NULL    |       |
| date_time     | datetime   | YES  |     | NULL    |       |
+---------------+------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```



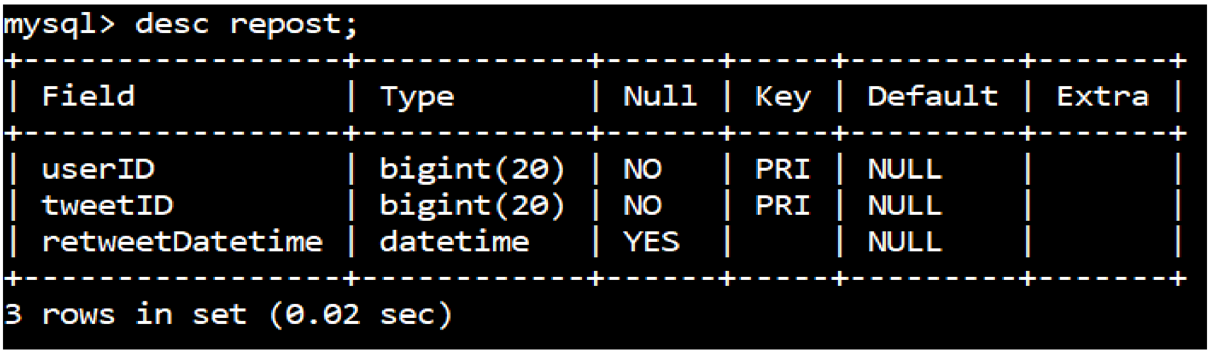
```
mysql> desc user;
+--------------+--------------+------+-----+---------+-------+
| Field        | Type         | Null | Key | Default | Extra |
+--------------+--------------+------+-----+---------+-------+
| userID       | bigint(20)   | NO   | PRI | NULL    |       |
| username     | varchar(50)  | YES  |     | NULL    |       |
| fullName     | varchar(100) | YES  |     | NULL    |       |
| profileURL   | varchar(100) | YES  |     | NULL    |       |
| profileDesc  | text         | YES  |     | NULL    |       |
| numFollowers | int(11)      | YES  |     | NULL    |       |
| numFriends   | int(11)      | YES  |     | NULL    |       |
| imageURL     | varchar(100) | YES  |     | NULL    |       |
| numTweets    | int(11)      | YES  |     | NULL    |       |
+--------------+--------------+------+-----+---------+-------+
9 rows in set (0.00 sec)
```



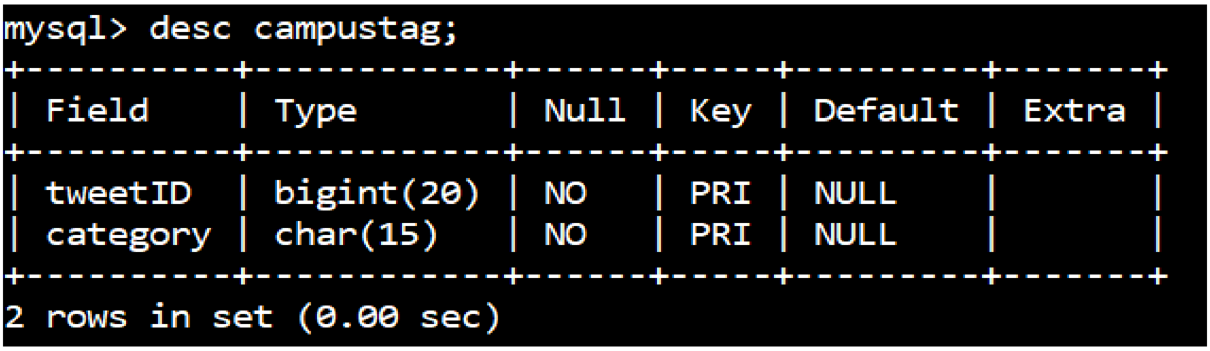
```
mysql> desc tweet;
+------------------+--------------+------+-----+---------+-------+
| Field            | Type         | Null | Key | Default | Extra |
+------------------+--------------+------+-----+---------+-------+
| tweetID          | bigint(20)   | NO   | PRI | NULL    |       |
| content          | text         | YES  |     | NULL    |       |
| creationDatetime | datetime     | YES  |     | NULL    |       |
| creationTimezone | tinyint(4)   | YES  |     | NULL    |       |
| postingDatetime  | datetime     | YES  |     | NULL    |       |
| postingTimezone  | tinyint(4)   | YES  |     | NULL    |       |
| country          | varchar(74)  | YES  |     | NULL    |       |
| longitude        | decimal(9,6) | YES  |     | NULL    |       |
| latitude         | decimal(9,6) | YES  |     | NULL    |       |
| sentiment        | tinyint(4)   | YES  |     | NULL    |       |
| tweetstreamID    | bigint(20)   | YES  | MUL | NULL    |       |
| userID           | bigint(20)   | YES  | MUL | NULL    |       |
+------------------+--------------+------+-----+---------+-------+
12 rows in set (0.00 sec)
```



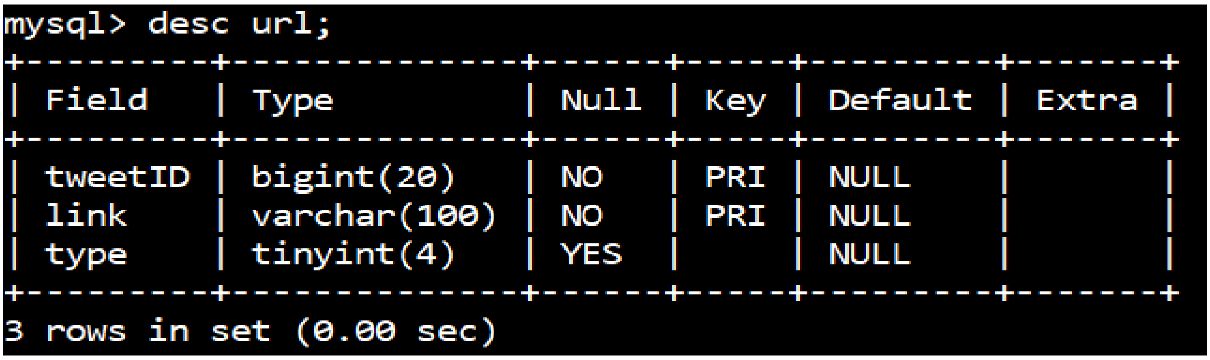
```
mysql> desc mention;
+---------+------------+------+-----+---------+-------+
| Field   | Type       | Null | Key | Default | Extra |
+---------+------------+------+-----+---------+-------+
| userID  | bigint(20) | NO   | PRI | NULL    |       |
| tweetID | bigint(20) | NO   | PRI | NULL    |       |
+---------+------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```



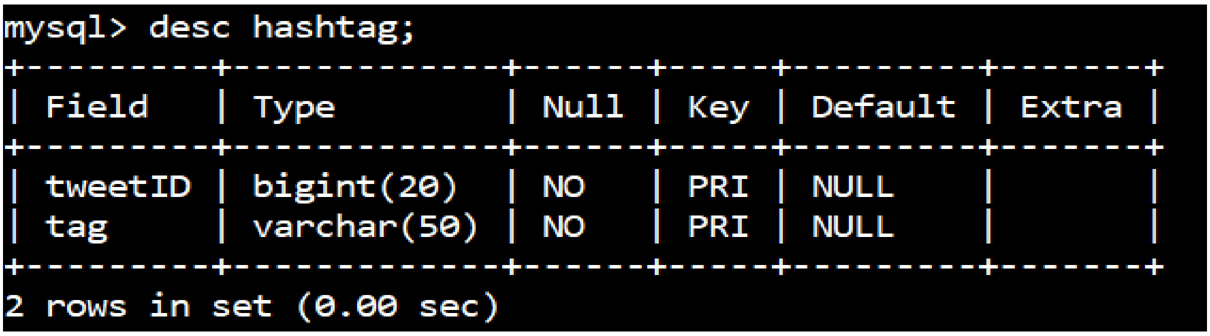
```
mysql> desc repost;
+-----------------+------------+------+-----+---------+-------+
| Field           | Type       | Null | Key | Default | Extra |
+-----------------+------------+------+-----+---------+-------+
| userID          | bigint(20) | NO   | PRI | NULL    |       |
| tweetID         | bigint(20) | NO   | PRI | NULL    |       |
| retweetDatetime | datetime   | YES  |     | NULL    |       |
+-----------------+------------+------+-----+---------+-------+
3 rows in set (0.02 sec)
```



```
mysql> desc campustag;
+----------+------------+------+-----+---------+-------+
| Field    | Type       | Null | Key | Default | Extra |
+----------+------------+------+-----+---------+-------+
| tweetID  | bigint(20) | NO   | PRI | NULL    |       |
| category | char(15)   | NO   | PRI | NULL    |       |
+----------+------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```



```
mysql> desc url;
+---------+--------------+------+-----+---------+-------+
| Field   | Type         | Null | Key | Default | Extra |
+---------+--------------+------+-----+---------+-------+
| tweetID | bigint(20)   | NO   | PRI | NULL    |       |
| link    | varchar(100) | NO   | PRI | NULL    |       |
| type    | tinyint(4)   | YES  |     | NULL    |       |
+---------+--------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```



```
mysql> desc hashtag;
+---------+-------------+------+-----+---------+-------+
| Field   | Type        | Null | Key | Default | Extra |
+---------+-------------+------+-----+---------+-------+
| tweetID | bigint(20)  | NO   | PRI | NULL    |       |
| tag     | varchar(50) | NO   | PRI | NULL    |       |
+---------+-------------+------+-----+---------+-------+
2 rows in set (0.00 sec)
```

QUERIES

A. For every country available in our data, how many tweets originated from there?

```
SELECT country, count(*) as numTweets
FROM tweet GROUP BY country;
```

B. What are the 10 most common user hashtags used in our collected tweets?

```
SELECT tag, count(*) as numTags
FROM hashtag GROUP BY tag
ORDER BY numTags
DESC LIMIT 10;
```

C. Who is the most influential twitter user in our stream? **(Use number of followers as proxy for influence)**

```
SELECT * FROM user
WHERE numFollowers =
(SELECT MAX(numFollowers) FROM user);
```

D. How many tweets are there for each sentiment?

```
SELECT sentiment, count(*) AS numTweets
FROM tweet GROUP BY sentiment;
```

E. (i) Which faculty has the most tweets with a positive sentiment?

```
SELECT category AS campus, count(*) AS numTweets
FROM campustag INNER JOIN tweet
ON tweet.tweetID = campustag.tweetID
WHERE sentimentID = 1
GROUP BY campus
ORDER BY numTweets
DESC LIMIT 1;
```

(ii) Which faculty has the least tweets with a positive sentiment?

```
SELECT category AS campus, count(*) AS numTweets
FROM campustag INNER JOIN tweet
ON tweet.tweetID = campustag.tweetID
WHERE sentimentID = 1
GROUP BY campus
ORDER BY numTweets
ASC LIMIT 1;
```

(iii) Which faculty has the most tweets with a negative sentiment? – **extension of question**

```
SELECT category AS campus, count(*) AS numTweets
FROM campustag INNER JOIN tweet
ON tweet.tweetID = campustag.tweetID
WHERE sentimentID = -1
GROUP BY campus
ORDER BY numTweets
DESC LIMIT 1;
```

F. (i) What were the most common open-day related hashtags (other than #NUSOpenDay18)?

```
CREATE TABLE temp AS
(SELECT tweet.tweetID FROM tweet INNER JOIN HashTag
ON Tweet.tweetID = HashTag.tweetID
WHERE cast('2018-03-09' AS DATE) <= postingDate <= cast('2018-03-11' AS DATE)
AND tag = '#NUSOpenDay18'
GROUP BY tweetID);


SELECT tag, count(*) AS numTags FROM hashtag
WHERE tweetID IN (SELECT tweeID from temp)
AND tag != '#NUSOpenDay18'
GROUP BY tag ORDER BY numTags
DESC LIMIT 3;
```

(ii) What were the top 3 retweeted tweets and who were their users?

```
SELECT username, tweetID, count(*) AS numTweets
FROM repost INNER JOIN user on repost.userID = user.userID
WHERE tweetID in (SELECT tweetID from temp)
GROUP BY username, tweetID
ORDER BY numTweets
DESC LIMIT 3;
```

(iii) What was the breakdown of sentiment across these tweets?

```
SELECT sentiment, count(*) from tweet
WHERE tweetID IN (SELECT tweetID FROM temp)
GROUP BY sentiment;

DROP TABLE temp;
```