

Hexo

2016-10-24 • CLOUD COMPUTING

Twitt Map

To build a twittmap, we need to utilize the twitter API. Here, I chose twitter4j, which is an unofficial Java library for the Twitter API. With Twitter4J, you can easily integrate your Java application with the Twitter service. There are some tutorials here: [How to “Sign in with Twitter” using twitter4j](#).

Register twitter app

Click this [link](#) and fill out the following form, it should be fine. Just remember to grant your app access for read and write.

Include twitter4j

For those who are not familiar with java, it would be a painful process to learn how use new tools at first(I like VS better, lol). Don't worry, just follow the steps below.

1. Download twitter4j from [here](#).
2. Unzip it.
3. After you create a java project, Right-click on your Java project and Select Properties.
4. Choose Java Build Path and select the third tab, Libraries.
5. Click add external library and add twitter4j-core-4.0.4.jar.

Begin programming

A lot of examples are available. Here is the sample code used in tutorial mentioned above (Using eclipse).

```
1 package com.example.test;
2
3 import twitter4j.Status;
4 import twitter4j.Twitter;
5 import twitter4j.TwitterException;
6 import twitter4j.TwitterFactory;
7 import twitter4j.auth.AccessToken;
8 import twitter4j.auth.RequestToken;
9 import twitter4j.conf.ConfigurationBuilder;
10 import twitter4j.conf.Configuration;
11
12 import java.io.BufferedReader;
13 import java.io.IOException;
14 import java.io.InputStreamReader;
15
16 public class FirstClass {
17
18     public static void main(String[] args) {
19         // TODO Auto-generated method stub
20         String testStatus="Hello from twitter4j";
21
22         ConfigurationBuilder cb = new ConfigurationBuilder();
23
24         //the following is set without accesstoken- desktop client
25         cb.setDebugEnabled(true)
26             .setOAuthConsumerKey("CONSUMER-KEY")
27             .setOAuthConsumerSecret("CONSUMER-SECRET");
28         try{
29             TwitterFactory tf = new TwitterFactory(cb.build());
30             Twitter twitter = tf.getInstance();
31
32             try{
33                 RequestToken requestToken = twitter.getRequestToken();
34                 System.out.println("Got request token.");
35                 System.out.println("Request token: " + requestToken.getAuthorizationURL());
36                 System.out.println("Request token secret: " + requestToken.getSecret());
37                 AccessToken accessToken = null;
38
39                 BufferedReader br = new BufferedReader(new InputStreamReader(System.in));
40                 while(accessToken == null){
41                     System.out.println("Open the following URL and go to the page to get the request token.");
42                     System.out.println(requestToken.getAuthorizationURL());
43                     System.out.print("Enter the PIN(if available) and press the enter key: ");
44                     String pin = br.readLine();
45                     try {
46                         if (pin.length() > 0) {
47                             accessToken = twitter.getOAuthAccessToken(requestToken.getSecret(), pin);
```

```

48         } else {
49             accessToken = twitter.getOAuthAccessToken
50         }
51     } catch (TwitterException te) {
52         if (401 == te.getStatusCode()) {
53             System.out.println("Unable to get the acc
54         } else {
55             te.printStackTrace();
56         }
57     }
58 }
59 System.out.println("Got access token.");
60 System.out.println("Access token: " + accessToken.get
61 System.out.println("Access token secret: " + accessTol
62     } catch (IllegalStateException ie){
63         if (!twitter.getAuthorization().isEnabled
64         System.out.println("OAuth consumer key/secret is
65         System.exit(-1);
66     }
67 }
68 Status status = twitter.updateStatus(testStatu
69 System.out.println("Successfully updated the status to
70 System.exit(0);
71 } catch (TwitterException te) {
72     te.printStackTrace();
73     System.out.println("Failed to get timeline: " + te.getMes
74     System.exit(-1);
75 } catch (IOException ioe) {
76     ioe.printStackTrace();
77     System.out.println("Failed to read the system input.");
78     System.exit(-1);
79 }
80 }
81
82 }

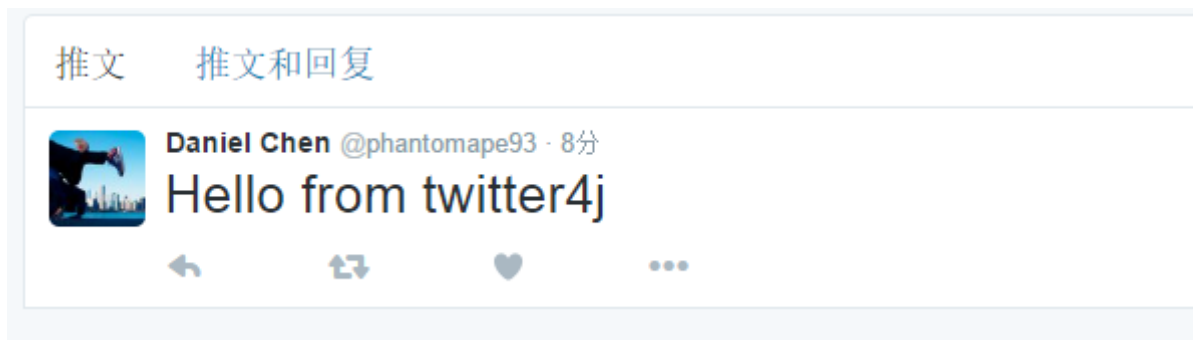
```

This piece of code can post a twitter from your account, after running this u can see the following instructions.

Open the following URL and grant access to your account:
https://api.twitter.com/oauth/authorize?oauth_token=6NCYzAAAAAaxad
 Enter the PIN(if available) and hit enter after you granted access.

Instructions

Just click into the url it mentioned and grant access to your app, and then type in the PIN code that it gives u. After that, u can check ur twitter now.



Result

Use streaming API

Here is the code.

```

1  package com.example.test;
2
3  import twitter4j.*;
4  import twitter4j.conf.ConfigurationBuilder;
5
6  public class PrintSampleStream {
7
8      public static void main(String[] args) throws TwitterException{
9          ConfigurationBuilder cb = new ConfigurationBuilder();
10
11          //the following is set without accesstoken- desktop client
12          cb.setDebugEnabled(true)
13              .setOAuthConsumerKey("I won't tell u")
14              .setOAuthConsumerSecret("what my key and secret is")
15              .setOAuthAccessToken("LOL")
16              .setOAuthAccessTokenSecret(":)");
17
18          TwitterStream twitterStream = new TwitterStreamFactory(cb).getTwitterStream();
19          StatusListener listener = new StatusListener() {
20              public void onStatus(Status status) {
21                  System.out.println(status.getUser().getName() + " : " + status.getText());
22              }
23              public void onDeleteNotice(StatusDeletionNotice statusD) {}
24              public void onTrackLimitationNotice(int numberOfLimitedStatuses) {}
25              public void onException(Exception ex){
26                  ex.printStackTrace();
27              }
28              @Override
29              public void onScrubGeo(long arg0, long arg1) {
30                  // TODO Auto-generated method stub
31              }
32              @Override

```

```
33         public void onStallWarning(StallWarning arg0) {
34             // TODO Auto-generated method stub
35         }
36     };
37     twitterStream.addListener(listener);
38     twitterStream.sample();
39
40 }
41 }
```

This piece of sample code enables you to fetch tweets from twitter in real-time.

Fetch Filtered Stream

Sometimes, we would like to search information that we are interested in, so we may need to filter the stream so that we can get relevant twitters. The code is shown below.

```
1  package com.example.test;
2
3  import twitter4j.*;
4  import twitter4j.conf.ConfigurationBuilder;
5
6  import java.util.ArrayList;
7  import java.util.Arrays;
8
9  public class PrintFilteredStream {
10
11      public static void main(String[] args) throws TwitterException {
12          String[] keywordsArray = {"iphone", "samsung"};
13
14          ConfigurationBuilder cb = new ConfigurationBuilder();
15
16          //the following is set without accesstoken- desktop client
17          cb.setDebugEnabled(true)
18              .setOAuthConsumerKey("CONSUMER-KEY")
19              .setOAuthConsumerSecret("CONSUMER-SECRET")
20              .setOAuthAccessToken("ACCESS-TOKEN")
21              .setOAuthAccessTokenSecret("ACCESS-TOKEN-SECRET");
22
23          //      Implementing StatusListner
24          StatusListener listener = new StatusListener() {
25              @Override
26              public void onStatus(Status status) {
27                  System.out.println(status.getUser().getName() + " : "
```

```
28         }
29
30         @Override
31         public void onDeleteNotice(StatusDeletionNotice
32             System.out.println("Got a status deletion
33     }
34
35     @Override
36     public void onTrackLimitationNotice(int numberOfLimitedSta
37         System.out.println("Got track limitation n
38     }
39
40     @Override
41     public void onException(Exception ex){
42         ex.printStackTrace();
43     }
44
45     @Override
46     public void onScrubGeo(long userId, long upToStatu
47         System.out.println("Got scrub_geo event us
48     }
49     @Override
50     public void onStallWarning(StallWarning warning) {
51         System.out.println("Got stall warning:" +
52     }
53 };
54
55 TwitterStream twitterStream = new TwitterStreamFactory(cb.
56 twitterStream.addListener(listener);
57 ArrayList<Long> follow = new ArrayList<Long>();
58 ArrayList<String> track = new ArrayList<String>();
59 for(String arg : keywordsArray){
60     if(isNumericalArgument(arg)){
61         for(String id : arg.split(",")){
62             follow.add(Long.parseLong(id));
63         }
64     }else{
65         track.addAll(Arrays.asList(arg.split(","))
66     }
67 }
68 long[] followArray = new long[follow.size()];
69 for(int i = 0; i < follow.size(); i++){
70     followArray[i] = follow.get(i);
71 }
72 String[] trackArray = track.toArray(new String[track.size(
73 twitterStream.filter(new FilterQuery(0, followArray, track
74 }
```

```

75
76     private static boolean isNumericalArgument(String argument){
77         String args[] = argument.split(",");
78         boolean isNumericalArgument = true;
79         for(String arg : args){
80             try{
81                 Integer.parseInt(arg);
82             } catch(NumberFormatException nfe){
83                 isNumericalArgument = false;
84                 break;
85             }
86         }
87         return isNumericalArgument;
88     }
89 }

```

Set up elasticsearch environment

1. Go to AWS console and select elasticsearch service(you can also choose to use elasticsearch by downloading and unzipping it from [here](#))
2. Set access police to open access or any way you like, and then use curl to create index using [index](#) API.

```

1  $ curl -XPUT 'http://search-twitter-1-kf5qeriqw5iu6uasbyv6dmwfbq.us-we
2      "settings" : {
3          "index" : {
4              "number_of_shards" : 3,
5              "number_of_replicas" : 2
6          }
7      }
8  }'

```

this piece of code will create an index named “user” in your AWS domain.

1. Next, you are going to create mapping. Here, the mapping called “profile” is created, which has three properties: text, latitude and longitude.

```

1  curl -XPOST search-twitter-1-kf5qeriqw5iu6uasbyv6dmwfbq.us-west-2.es.:
2  '
3  {
4      "profile" : {
5          "properties" : {
6              "text" : { "type" : "string"},
7              "latitude" : { "type" : "string"},

```

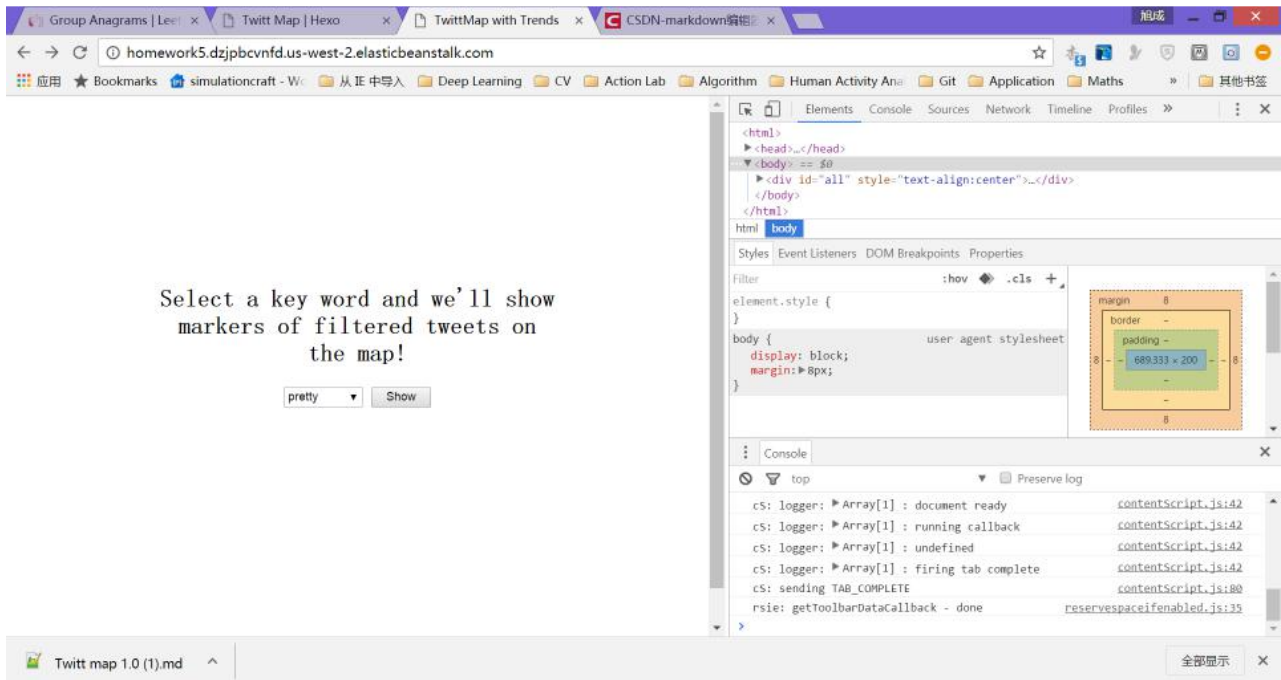
```
8         "longitude" : { "type" : "string"}
9     }
10 }
11 }
12 '}
```

Upload data to elasticsearch

To upload data to elasticsearch, you can use the curl command in cygwin, since I am using a windows machine, or put this piece of code into the java program that are fetching twitter. From twitter stream, we can obtain a JSONObject, which could by utilize into uploading to elasticsearch.

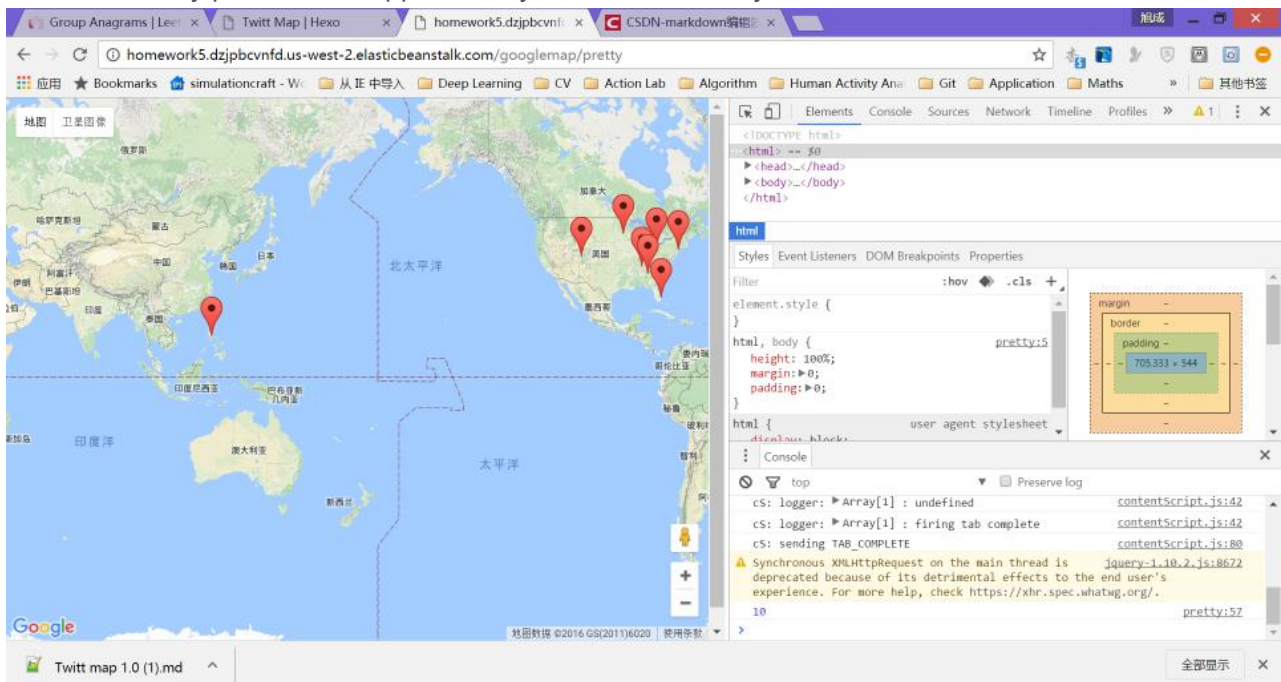
```
1  String url = "http://search-twitter-1-kf5qeriqw5iu6uasbyv6dmwfbq.us-west-2
2  HttpClient client = HttpClientBuilder.create().build();
3  HttpPost put = new HttpPost(url);
4  put.setHeader("Content-type", "application/json");
5  StringEntity params =new StringEntity(json.toString());
6  put.setEntity(params);
7  HttpResponse response = client.execute(put);
8  System.out.println("Response Code:"+response.getStatusLine().getStatusCode
9
10  BufferedReader rd = new BufferedReader(new InputStreamReader(response.getE
11  StringBuffer result = new StringBuffer();
12  String line = "";
13  while ((line = rd.readLine()) != null) {
14      result.append(line);
15  }
16
17  System.out.println(result.toString());
```

UI for our application



Main UI

This is the entry point for our application, you can choose keyword for twitter.



Google Map

After you choose the keyword and click the “show” button, you can see the corresponding twitter location marked on the map.

Every 5 seconds, the front end will refresh and the counter, which can be seen in the second picture as 10, will gradually increase as our backend program continuously fetching data from twitter.

Share

CATEGORIES

[Cloud Computing](#) (1)
[Computer Vision](#) (2)
[example](#) (1)

ARCHIVES

[十月 2016](#) (3)
[三月 2016](#) (4)

RECENTS

[Twitt Map](#)
[Hough Transform for Line Detection](#)
[Google Tango on ROS](#)
(no title)
(no title)