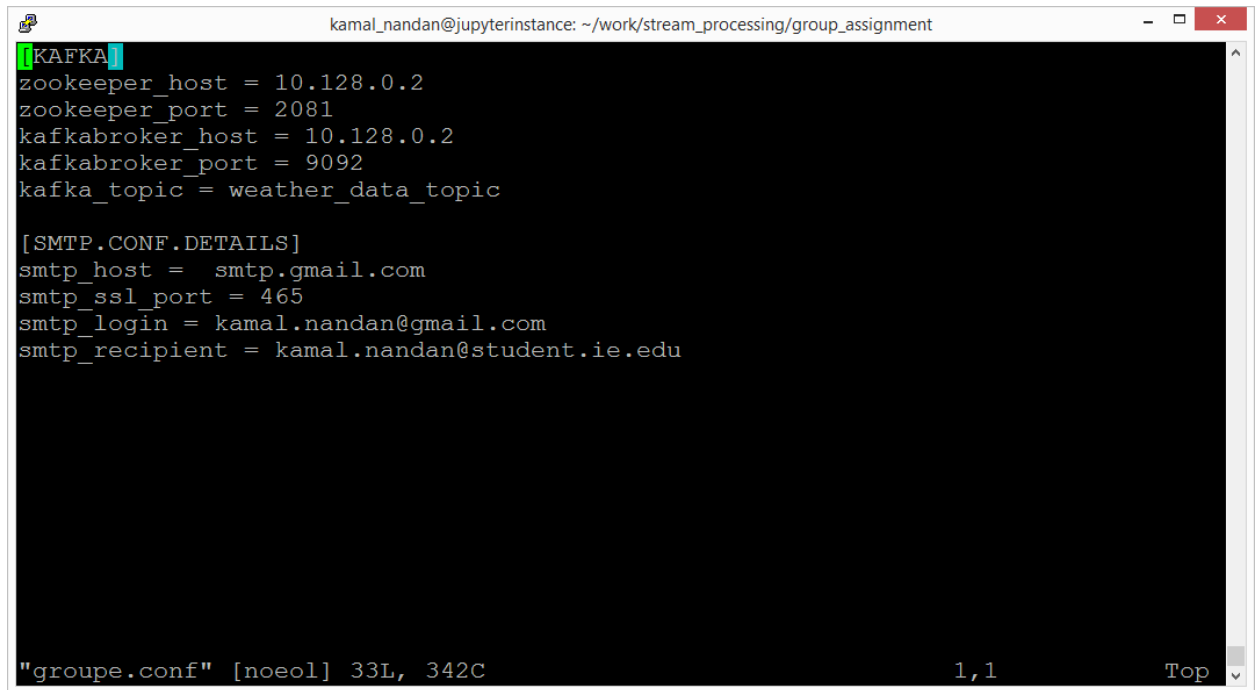## Screenshots and steps to run the program:

1. Open "groupe.conf" configuration file and set the required parameters such as kafka broker host, port and topics etc.

   If you want email notification for simulation of notification purposes, provide SMTP details too.

2. Launch the script "KafkaWeatherDataProducer.py" by typing the following command and pressing enter:

   - python KafkaWeatherDataProducer.py

```
kamal_nandan@jupyterinstance: ~/work/stream_processing/group_assignment        _  □  ×
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$
kamal_nandan@jupyterinstance:~/work/stream_processing/group_assignment$ python K
afkaWeatherDataProducer.py
10.128.0.2:9092
Random no. is  2
40.5549 -105.7160 0 68687.4 267.75262 238.24579 240.33876 9.903259 12.378311

Random no. is  4
40.7549 -105.9160 0 68049.26 266.3579 238.24579 232.4723 5.310219 6.076233

Random no. is  1
40.4549 -105.6160 0 64520.707 264.79843 249.7873 253.35245 9.927673 9.903259

Random no. is  4
40.7549 -105.9160 1 68043.766 266.29993 233.06009 233.39519 7.3335953 8.087402

Random no. is  3
40.6549 -105.8160 0 67496.266 265.3569 201.10538 204.59914 5.218666 5.059967
```

This script is a simulator for IoT devices that will be sending the weather data to the Kafka broker. Once this script starts, it starts reading the weather data records randomly from different files and starts sending the data into the Kafta topic.

3. Open another terminal, browse into the same folder as before and run the shell script "./run_kafka_consumer.sh" by typing the following command:

- **./run_kafka_consumer.sh**

Doing the above will launch the python script called **"ConsumeWeatherDataFromKafka.py"** which will start pulling the weather data from kafka topic in a streaming fashion and run the required algorithm, determine the most favorable direction in which the boat should move.

Important note: In real life, the stream-processor would be sending the data to a database from where it would be picked up by the REST engine and published on to the on-board devices, on the boat, that would have subscribed to the notifications. For now we are simulating the notifications by sending emails.

In the following screenshot, we see the "KafkaCosnumer" in action:

```
                     kamal_nandan@jupyterinstance: ~/work/stream_processing/group_assignment        –  □  ×
2018-06-28 20:54:12 INFO  ContextHandler:781 - Started o.s.j.s.ServletContextHan
dler@2a129f5f{/metrics/json,null,AVAILABLE,@Spark}
['40.6549', '-105.8160', '2', '67484.36', '264.9846', '206.23615', '208.74112',
'8.130135', '8.145393\n']
latlongkey: 40.6549-105.8160
Length of dict 1
Keep moving in the same direction as before
['40.5549', '-105.7160', '1', '68678.24', '267.73737', '236.2627', '238.18536',
'10.742516', '13.4891815\n']
latlongkey: 40.5549-105.7160
Length of dict 2
>>>>>>>>>>>>>>>>>>Go to coordinates>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
40.5549,-105.716
successfully sent the mail
['40.7549', '-105.9160', '1', '68043.766', '266.29993', '233.06009', '233.39519'
, '7.3335953', '8.087402\n']
latlongkey: 40.7549-105.9160
Length of dict 3
>>>>>>>>>>>>>>>>>>Go to coordinates>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>
40.7549,-105.916
successfully sent the mail
['40.5549', '-105.7160', '2', '68667.26', '267.6855', '234.4499', '235.88916', '
11.310158', '14.17585\n']
latlongkey: 40.5549-105.7160
```

We see in the above screenshots that the Consumer script would run its algorithm and determine the best co-ordinate to which the boat must move to and send notification to the subscriber.

To simulate the sending of notification, as a make-shift arrangement, we are sending notifications over email.

4. To stop the above scripts, press control-c at both the terminals

Future scope for improvement:

1. REST interface to which the on-board devices would have subscribed.
2. More granularity in notification