Step 1: start Zookeeper, Kafka-server, Elasticsearch and Kibana severs (output1.png). Check on localhost:9200 if Elasticsearch is running. (output2.png)

Step 2: Open PyCharm and open project TwitterSentiment. Run StreamProducer.py and StreamConsumer.py for #trump. StreamConsumer.py stores the consumed tweets in an index tweet\_sentiment\_trump with index fields containing tweet’s author name, message, date, positive score, negative score, neutral score and compound score and sentiment class, which is computed from a vaderSentiment, a third-party Sentiment Analyzer. The sentiment classes are Positive, Negative and Neutral. These indices are stored on Elasticsearch. (output3.png and output4.png)

Step 3: Open Kibana on browser at localhost:5601 (output5.png), create a new index pattern with name tweet\_sentiment\_trump\*, which will match one index from Elasticsearch. (output6.png, output7.png, output8.png).

Step 4: Create a new visualization with the new created index, tweet\_sentiment\_trump\*, and create a Pie-chart consisting of 3 slices, Positive, Negative and Neutral sentiment class, showing the sentiments of tweets collected in Step 2. (output9.png, output10.png, output11.png, output12.png). **Another visualization is created showing the top ten twitter users with their count of Positive, Negative and Neutral tweets on #trump. (output13.png)**

Step 5: Repeat the steps again from step 2, for #coronavirus. Note the, new index that is created for #coronavirus is tweet\_sentimnet\_coronavirus\*. We work on this index pattern for #coronavirus in Kibana for visualization. (output14.png, output15.png, output16.png, output17.png, output18.png, output19.png, output20.png, output21.png, output22.png). **Another visualization is created showing the top ten twitter users with their count of Positive, Negative and Neutral tweets on #coronavirus. (output23.png)**. We check whether the topic twitter\_sentiment was created in zookeeper. (output24.png).