KAFKA – Streamıng data

TEB ASSIGNMENT

SERAP AYDOGDU

28.05.2019

Contents

[KAFKA STREAMING TASK : 2](#_Toc9947955)

[Architecture 2](#_Toc9947956)

[Setting up environment 2](#_Toc9947957)

[1. Start Kafka Server 2](#_Toc9947958)

[2. Test Kafka Producer and Consumer 3](#_Toc9947959)

[About the Progress 6](#_Toc9947960)

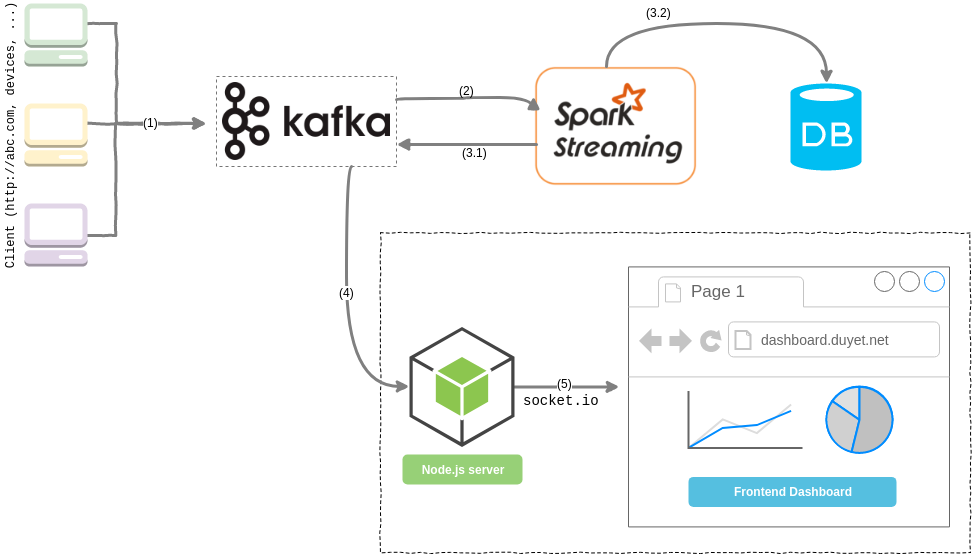
[1. Apache Spark Streaming 6](#_Toc9947961)

[2. Write to Casandra 6](#_Toc9947962)

# KAFKA STREAMING TASK :

# Architecture

* **[1] Producer :** Console
* **[2] Consumer :** Spark Streaming
* **[3] DB :** Casandra
* **[4] [5] Dashboard :** Nodejs / Kibana (3)



**(3)**

# Setting up environment

***\*\*Note: Zookeeper, Kafka, Spark, and Jupyter Notebook are needed to have already been installed.***

## Start Kafka Server

Run below code scripts separately in a command prompt on Windows (opening 3 different cmd windows and writing down each script separately)

# Start zookeeper

$ zkServer

# Start Kafka

$ kafka-server-start.bat %KAFKA\_HOME%\config\server.properties

# Create Topic Named “deneme2”

$ kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic deneme2

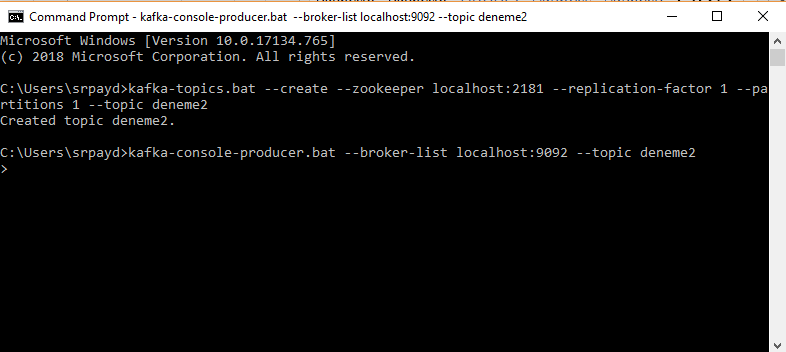
## Test Kafka Producer and Consumer

While Kafka server and Zookeeper are running properly on different terminals open another terminal and write down below scripts to prepare producer and to send messages to ***deneme2*** topic on Kafka.

# Terminal for producer

$ kafka-console-producer.bat --broker-list localhost:9092 --topic deneme2

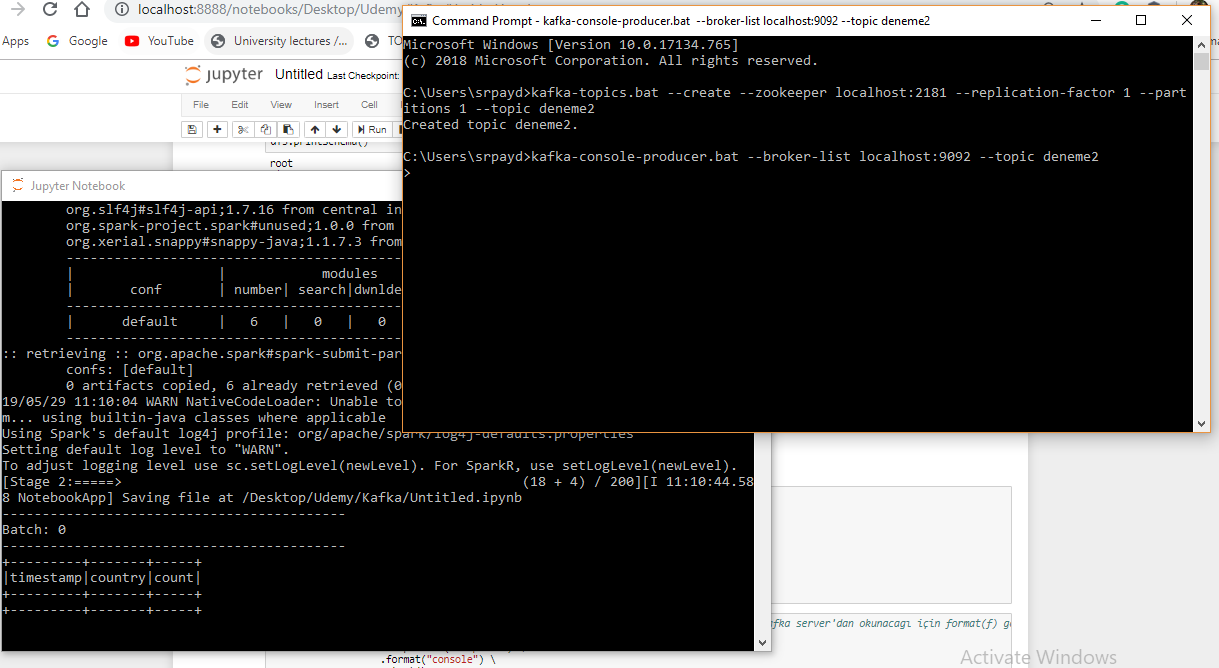
>



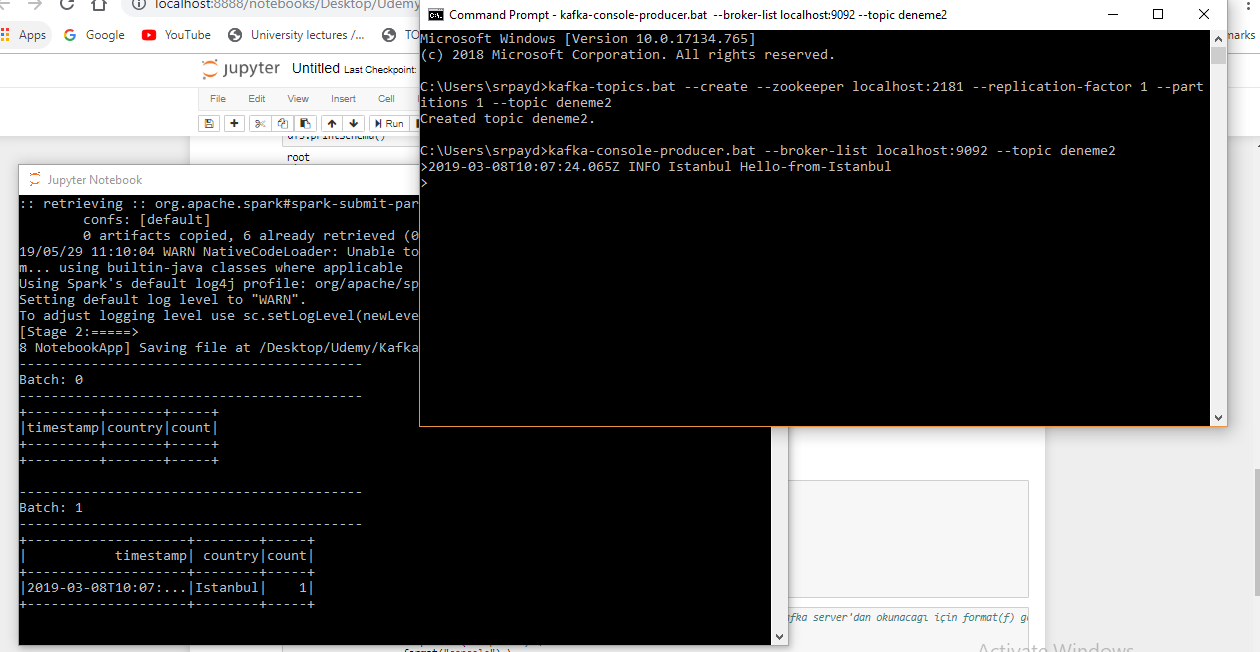
*Figure 1: Producer initialize and set up for message*

Then run the ***KafkaWordCount.py*** script, at that time we will be seeing word count operation’s output displays lively on Jupyter console.

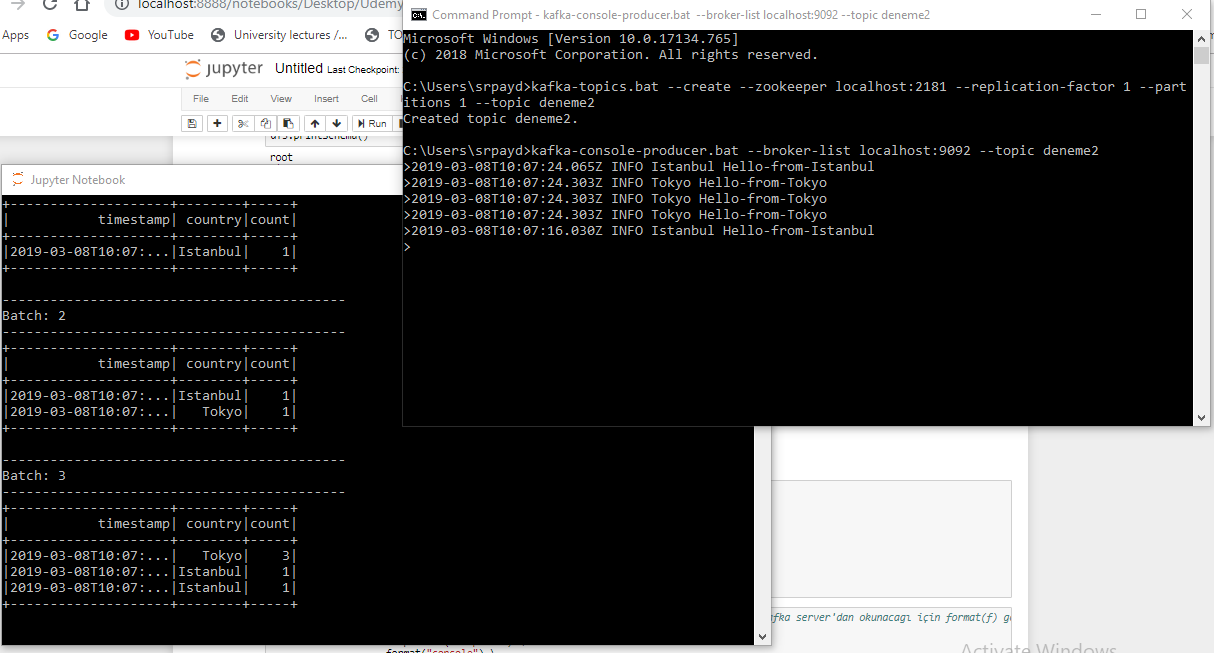
* **In Figure 2**, while we do not have any message on Kafka producer, so word count gets none computation.
* **In Figure 3**, input some message on producer console, then the computation is on the progress.
* **In Figure 4**, some other input messages come from the producer, and then the computation is again on the progress.
* **In Figure 5**, some other input messages come from the producer, and then the computation is again on the progress.



*Figure 2: Consumer computes a message*

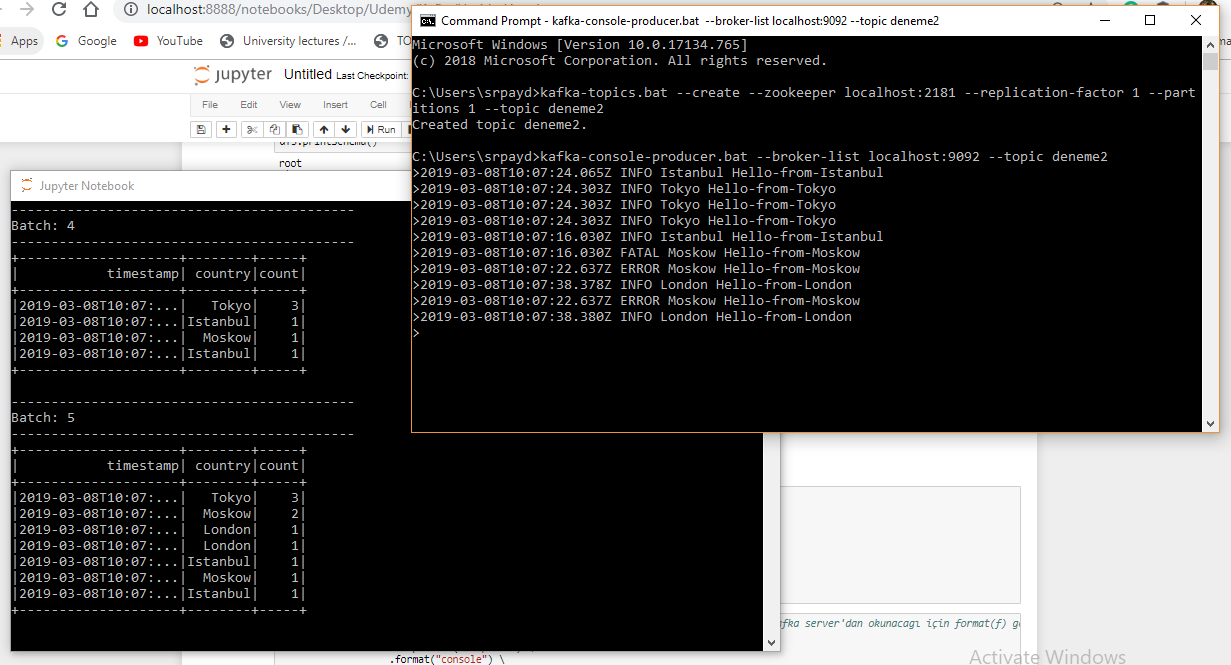


*Figure 3: Consumer starts computation again when a modifying happens on producer console*



*Figure 4: Consumer starts computation again when a modifying happens on producer console*

*Note \*\*: Word count computation occurred based on timestamp and country values.*



*Figure 5: Consumer starts computation again when a modifying happens on producer console*

*Note \*\*: The red highlighted circle holds all messages which were sent to* ***deneme2*** *topic.*

# About the Progress

## Apache Spark Streaming

All comments hold on Spark Streaming script (***KafkaWordCount.py***) step by step.

After setting up the environment and necessary consoles (Kafka start, topic create, producer initializing) ***KafkaWordCount.py*** is needed to run to see results.

Spark Streaming continuously reads every incoming data from the producer. The outputs of word count computing are sent to the Jupyter notebook console.

## Write to Casandra

Planned to write Structure Streaming Data into Cassandra using Pyspark API instead of console, however, could not able to set up the Casandra environment. Due to I showed my results lively on Jupyter console.

***\*\* Note:*** *By the way, I have put the related Casandra script in the* ***KafkaWordCount.py****, however, I could not be able to run it. Because of that, it is commented.*