

## KAFKA MESSAGING

### KAFKA CONSOLE

---

#### DOWNLOAD & EXTRACT

cd C:\Users\kwagn\kafka\bin\windows

---

#### START ZOOKEEPER

C:\Users\kwagn\kafka\bin\windows\zookeeper-server-start.bat C:\Users\kwagn\kafka\config\zookeeper.properties

```
C:\Users\kwagn\kafka>bin\windows\zookeeper-server-start.bat config\zookeeper.properties
[2019-02-02 11:29:30,118] INFO Reading configuration from: config\zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2019-02-02 11:29:30,131] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server...
```

---

#### START KAFKA

C:\Users\kwagn\kafka\bin\windows\kafka-server-start.bat C:\Users\kwagn\kafka\config\server.properties

```
C:\Users\kwagn\kafka>C:\Users\kwagn\kafka\bin\windows\kafka-server-start.bat config\server.properties
[2019-02-02 11:32:28,218] INFO Registered kafka:type=kafka.Log4jController...
```

---

#### CREATE TOPIC

C:\Users\kwagn\kafka\bin\windows\kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic test

---

#### SHOW AVAILABLE TOPICS

C:\Users\kwagn\kafka\bin\windows\kafka-topics.bat --list --zookeeper localhost:2181

```
C:\Users\kwagn>C:\Users\kwagn\kafka\bin\windows\kafka-topics.bat --list --zookeeper localhost:2181
__consumer_offsets
numtest
numtest_0
test
test2
C:\Users\kwagn>
```

---

#### INITIATE A PRODUCER

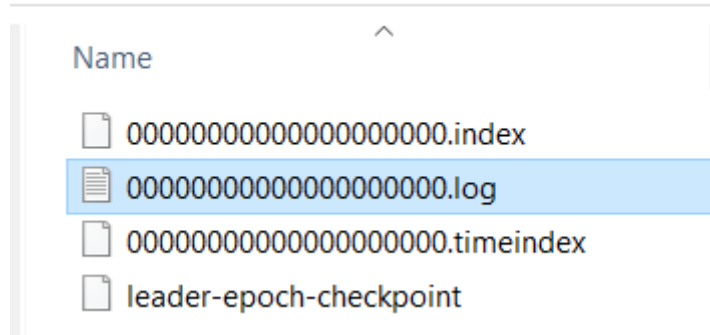
C:\Users\kwagn\kafka\bin\windows\kafka-console-producer.bat --broker-list localhost:9092 --topic test

>input

---

## STORED IN LOGS

Windows (C:) > tmp > kafka-logs > test-0



---

## SHOW THE MESSAGE LOG

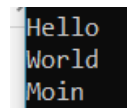
kafka-run-class.bat kafka.tools.DumpLogSegments --deep-iteration --print-data-log --files \\tmp\\kafka-logs\\test-0\\00000000000000000000.log

```
Starting offset: 0
offset: 0 position: 0 CreateTime: 1547465113478 invalid: true keysize: -1 valuesize: 5 magic: 2 compresscodec:
  NONE producerId: -1 producerEpoch: -1 sequence: -1 isTransactional: false headerKeys: [] payload: Hello
offset: 1 position: 73 CreateTime: 1547465115027 invalid: true keysize: -1 valuesize: 5 magic: 2 compresscodec
: NONE producerId: -1 producerEpoch: -1 sequence: -1 isTransactional: false headerKeys: [] payload: World
offset: 2 position: 146 CreateTime: 1547465456240 invalid: true keysize: -1 valuesize: 4 magic: 2 compresscode
c: NONE producerId: -1 producerEpoch: -1 sequence: -1 isTransactional: false headerKeys: [] payload: Moin
```

---

## CONSUMING MESSAGES

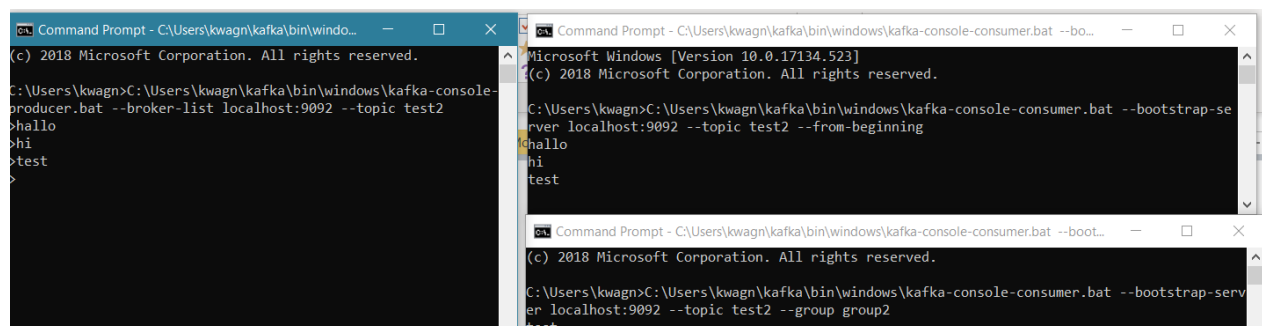
C:\Users\kwagn\kafka\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic test --from-beginning



---

## SECOND CONSUMER

C:\Users\kwagn\kafka\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic test2 --group group2



## KAFKA PYTHON

<https://towardsdatascience.com/kafka-python-explained-in-10-lines-of-code-800e3e07dad1>

## START ZOOKEEPER, KAFKA AND CREATE TOPIC

```
C:\Users\kwagn>C:\Users\kwagn\kafka\bin\windows\kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic numtest
Created topic "numtest".
```

## CODING A PRODUCER / CONSUMER DEMO

Files are public in github: <https://github.com/kwbln/CS4BD-Edl/tree/master/StreamingKafka>

## TESTING

Start zookeeper + start kafka + start demo

## BROKER PROBLEMS

kafka.errors.NoBrokersAvailable: NoBrokersAvailable

```
kafka.errors.NoBrokersAvailable: NoBrokersAvailable
```

<https://stackoverflow.com/questions/47168342/kafka-1-0-stops-with-fatal-shutdown-error-logs-directory-failed>

after restart, the topic need to be recreated

## LOG FROM A SUCCESSFUL RUN

```
C:\Users\kwagn\AppData\Local\Programs\Python\Python37-32\python.exe "C:/Users/kwagn/Github/CS4BD-Edl/3 - Streaming Kafka/KDemo.py"
Producer topic A: n 20 numbers, even True, sleep for 1s
Producer topic B: n 10 numbers, even False, sleep for 2s
timestamp | topic | offset | value
1549203597576 | A | 441 | b'4'
1549203599577 | A | 442 | b'6'
1549203599787 | B | 344 | b'3'
1549203601579 | A | 443 | b'8'
1549203603580 | A | 444 | b'10'
1549203603789 | B | 345 | b'5'
1549203605581 | A | 445 | b'12'
1549203607583 | A | 446 | b'14'
1549203607791 | B | 346 | b'7'
1549203609584 | A | 447 | b'16'
1549203611586 | A | 448 | b'18'
1549203611792 | B | 347 | b'9'

Process finished with exit code 0
```

## KAFKA STREAMS

<https://kafka.apache.org/21/documentation/streams/>

<https://www.confluent.io/blog/introducing-kafka-streams-stream-processing-made-simple/>

<https://jaxenter.de/kafka-0-11-60108>

## INNOVATION

Kafka offers the possibility to transmit messages from a producer to the broker idempotent. This means that sending operations, which may have to be carried out several times due to problems, result in exactly one written message in the addressed topic. Duplicates can thus be avoided within Topic Partition.

Processing "EXACTLY ONCE" neither leads to a sustained loss of messages nor to unwanted duplicates of messages in topics.

## PREVIOUS KAFKA VERSIONS

The following two semantics were available for processing:

"AT MOST ONCE": In the event that a producer receives no feedback due to a timeout or an error occurs, there is no resending. Affected messages therefore do not end up in the respective topic. In order to be able to exclude potential duplicates by repeated sending, a possible message loss is accepted.

"AT LEAST ONCE": Producers can be configured to automatically resend messages in case of timeouts or errors, which can lead to duplicate messages in the topic depending on the error. In order to avoid message loss, possible duplicates are accepted, which may have to be handled later and outside Kafka, depending on the application.

## ALPHAGO

