# **Kafka local setup**

# Install Kafka in local

Follow this and download

<https://docs.confluent.io/platform/current/installation/installing_cp/zip-tar.html#configure-confluent-home>

1. Change the kafka server.properties to run on 9092 port, we just need to uncomment

Go to E:\study softwares\confluent kafka 7.5.2\confluent-7.5.2\etc\kafka\server.properties

**listeners=PLAINTEXT://localhost:9092**

1. Un comment **advertised.listeners=PLAINTEXT://localhost:9092**
2. Uncomment **listener.security.protocol.map=**

Jars

implementation 'org.apache.kafka:kafka-clients:2.8.0'

implementation 'org.apache.kafka:kafka-streams:2.8.0'

Create and use a free cluster

In confluent, u can create a free cluster and use it

Tools to create topic

Download a tool called Kafka offset explorer

. means current directory

--Double hyphen means predefined key

Windows means use backward slash

|  |  |
| --- | --- |
| Windows | Mac |
| %KAFKA\_HOME%  Whereas KAFKA\_HOME is an environment variable | $KAFKA\_HOME |
| Use BACK slash \ | Forward slash / |
| .bat | .sh |
| .. means current directory |  |
| .also sometimes current directory- I am also not clear |  |
| C:\kafka\_2.13-3.3.1>.\bin\windows\kafka-console-consumer.bat | If u just type this u will get all the options available with this batch file |
| --In the command means predefined option |  |

Download kafka from kafka quickstart

And make sure u have the kafka in c folder

1. Tips/ How to clean all previous data locally

If ur broker is starting very slowly or if logs size is too heavy or if u are unable to send the data to broker then delete or local logs

As we know all the data will be stored In files, if u want to delete all those kafka messages and if u want to make it clean as new then delete 2 directories

1) tmp

2) inside kafka folder we have a folder called logs we have to delete that as well, because of which all logs will be created again and again even if u created tmp folder

1. TIPS while running locally

Delete the kafka tmp folder and re run all the scripts once again create all the topics

1. See all options for that batch file

If u want to see all the options available with that command /with that batch file just type as below

.\bin\windows\kafka-topics.bat

.\bin\windows\kafka-consumer-groups.bat

Any batch file name

Not for every thing – it wont come for kafka-console-producer.bat

1. Start zookeeper

First always Start zookeeper with zookeeper properties file

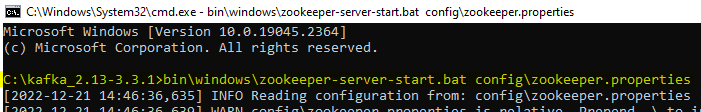
Always run the command outside bin folder because u need to provide the properties files which are in config folder which is outside of bin folder –so don’t run commands inside bin folder

**Way1 without using dot :-**

E:\study softwares\kafka-3.3.1-src>bin\windows\zookeeper-server-start.bat config\zookeeper.properties

Or

C:\kafka\_2.13-3.3.1\bin\windows\zookeeper-server-start.bat C:\kafka\_2.13-3.3.1\config\zookeeper.properties



Move the Kafka folder to c drive and start the zookeeper and its working initially when it was in e drive it was having class path issue now its solved after moving

.\bin\windows\zookeeper-server-start.bat .\etc\kafka\zookeeper.properties

**Way 2 with using dot .\ :-**

C:\kafka\_2.13-3.3.1>

.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties

Or way 3:- if u are using **confluent kafka**

In my machine I have installed kafka under path “C:\confluentkafka”

.\bin\windows\zookeeper-server-start.bat .\etc\kafka\zookeeper.properties

The above command says, there is a batch file called zookeeper-server-start.bat

In bin/windows folder run that batch file using zookeeper.properties which is present in config folder

If u got any error like below

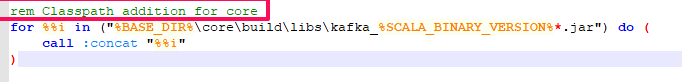
Classpath is empty. Please build the project first e.g. by running 'gradlew jarAll'

Then goto bin\windows\kafka-run-class

**Step 1 :**Navigate to \confluent-community-5.5.0–2.12\confluent-5.5.0\bin\windows folder.

**Step 2:**Open **kafka-run-class.bat** file.

**Step 3 :** Search **rem Classpath addition for core**in this bat file.The code would look like this :



**Step 4:**Now, just add the below code just above the **rem Classpath addition for core** line.

*rem classpath addition for LSB style path  
if exist %BASE\_DIR%\share\java\kafka\\* (  
call:concat %BASE\_DIR%\share\java\kafka\\*  
)*

1. Start kafka server with server.properties file

In Kafka folder server.properties

Un comment **advertised.listeners=PLAINTEXT://localhost:9092**

With .dot

C:\kafka\_2.13-3.3.1>

.\bin\windows\kafka-server-start.bat config\server.properties

Without dot

C:\kafka\_2.13-3.3.1>bin\windows\kafka-server-start.bat config\server.properties

Start the kafka with server.properties file

Once started u wil text with port number as “PLAINTEXT://DESKTOP-48ALSQ8:9092,”

Way 2- using confluent kafka

.\bin\windows\kafka-server-start.bat .\etc\kafka\server.properties

1. Start schema registry

As we don’t have the scripts in windows folder we cant start schema registry in windows machine

.\bin\windows\schema-registry-start.BAT .\etc\schema-registry\schema-registry.properties

1. Create the topic

Tip:- generally its best to create topic using tool called “**offsetExplorer” because it will exactly create topic with desired partitions**

Go to the kafka home location (not inside bin directory – come out side of bin directory)

.\bin\windows\kafka-topics.bat --create --topic stockso --partitions 4 --bootstrap-server localhost:9094

.\bin\windows\kafka-topics.bat --create --topic myTopic --partitions 1 –replication-factor 3 --bootstrap-server localhost:9092 –config min.insync.replicas=2

1. Create the console producer

.\bin\windows\kafka-console-producer.bat --topic myTopic --bootstrap-server localhost:9092

.\bin\windows\kafka-console-producer.bat --topic stocks --bootstrap-server localhost:9092

Create a producer and gave a sample csv file as below then the producer will put that file to the kafka server

SEND THE FILE DATA CSV data line by line

C:\kafka\_2.13-3.3.1>

.\bin\windows\kafka-console-producer.bat --topic stocks --bootstrap-server localhost:9094 <.\H1BLIST.csv

.\bin\windows\kafka-console-producer.bat --topics myTopic --bootstrap-server localhost:9092 < ..\data\sample2.csv

#### Produce with keys

Generally in a console producer we cant give keys with this u can send the keys

C:\kafka\_2.13-3.3.1>.\bin\windows\kafka-console-producer.bat --topic infosys --bootstrap-server localhost:9092 --property parse.key=true --property key.seperator=,

bin/kafka-console-producer.sh --topic demo\_testing3 --bootstrap-server 54.90.61.129:9092 --property parse.key=true --property key.separator=,

1. Produce with round robin fashion

This will produce the messages to all partitions in a round robin fashion,if key is not there by default all the messages will be sent to all partitions

# other terminal

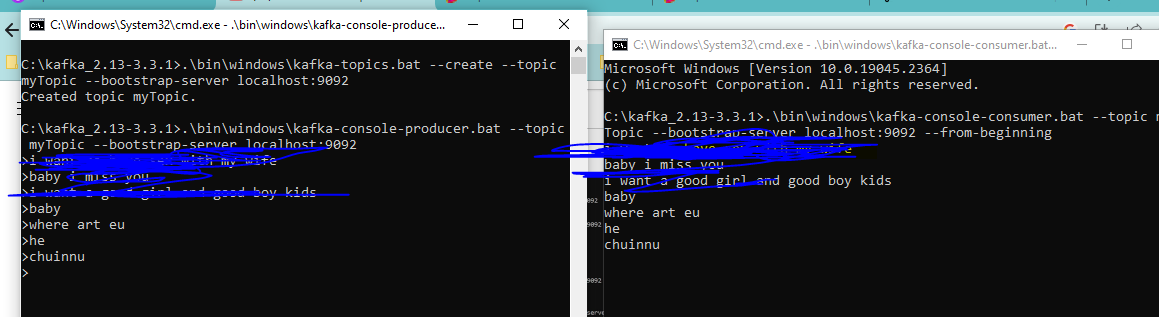
kafka-console-producer.sh --producer.config playground.

config-bootstrap-server cluster.playground.cdkt. io:9092 --producer-property partitioner.class=org.apache. kafka.clients.producer. Round RobinPartitioner --topic second\_topic

1. Create console consumer

.\bin\windows\kafka-console-consumer.bat --topic myTopic --bootstrap-server localhost:9092 --from-beginning

.\bin\windows\kafka-console-consumer.bat --topic infosys --bootstrap-server localhost:9092



1. Console consumer in a group

If all consumers are in same group they will share the work load

If all consumers are in different group they will act as separate consumers and they will not share the work load same like persons in different teams, if same team they will help each other

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic stocks-tips --from-beginning --group group1

Ex:-2

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9094 --topic stocks --from-beginning --group cg1

|  |  |
| --- | --- |
| C:\kafka\_2.13-3.3.1>**.\bin\windows\kafka-console-consumer.bat** | If u just type this **kafka-console-consumer.bat** in command prompt u will get all the options available with this batch file |

Ex:- 2

C:\kafka\_2.13-3.3.1\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --whitelist "hello-producer-1|hello-producer-2"

Ex:-3

See u have to come out of bin and run that

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic EmployeesInfo --from-beginning --group cg1

# consuming from beginning

Below Is .sh means linux commands change to equivalent windows bat files and run those bat files

kafka-console-consumer.sh --consumer.config playground. config-bootstrap-server cluster.playground.cdkt. io:9092 --topic second\_topic --from-beginning

1. 1 consumer consuming from 2 topics

C:\kafka\_2.13-3.3.1\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --whitelist "hello-producer-1|hello-producer-2"

.\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --whitelist "hello-producer-1|hello-producer-2"

1. Describe the topics

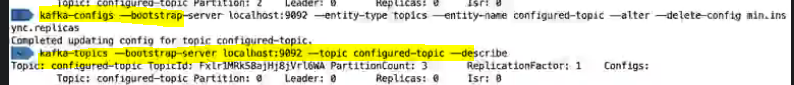
# describe the consumer group again – shocking so in the entire broker can’t we have 1 more consumer group with same name even for diff topic, so consumer group name should be unique in the entire broker?

C:\kafka\_2.13-3.3.1>

.\bin\windows\kafka-consumer-groups.bat --bootstrap-server localhost:9094 --describe --group cg1

.\bin\windows\kafka-consumer-groups.bat --describe --group cg1 -- bootstrap-server localhost:9094

kafka-consumer-groups.sh --command-config playground. config-bootstrap-server cluster.playground.cdkt. io:9092 --describe --group my-first-application



**See the log policy of a kafka topic**

Kafka-topic.bat –bootstrap-server localhost:9092 –describe –topic \_\_consumer\_offsets

If u describe this inbuilt topic u will see log compaction policy

1. Reset the offsets

# Dry Run: reset the offsets to the beginning of each partition , dry run means it will just give the demo to which point it will reset , before resetting if u want to know with the existing command till where u can reset then run ths dry run command

Caution :- consumer must be off before u reset the offset

kafka-consumer-groups.sh --command-config playground. config-bootstrap-server cluster.playground.cdkt. io:9092 group my-first-application --reset-offsets --to-earliest --topic third\_topic --dry-run

# execute flag is needed

.\bin\windows\kafka-consumer-groups.bat --bootstrap-server localhost:9092 --group cg1 --reset-offsets --to-earliest --topic stockso –execute

kafka-consumer-groups.sh --command-config playground. config-bootstrap-server cluster.playground.cdkt. io:9092-group my-first-application --reset-offsets --to-earliest --topic third\_topic –execute

### 7. analyze the dump using below



Installing multi node cluster

Go to kafka config folder and make 3 copies of server.property file

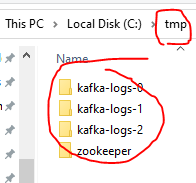
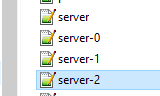
(in C:\kafka\_2.13-3.3.1\config)

So that every time u start kafka server we can provide different serer file

C:\kafka\_2.13-3.3.1>bin\windows\kafka-server-start.bat config\server-0.properties

C:\kafka\_2.13-3.3.1>bin\windows\kafka-server-start.bat config\server-1.properties

C:\kafka\_2.13-3.3.1>bin\windows\kafka-server-start.bat config\server-2.properties



Every broker should have a different broker id, which makes broker unique

1. So open each broker and edit the below broker.id field and for 3 servers u will see 3 folders

**broker.id=1**

in realtime we will have multiple brokers /cluster of brokers, each will run on different node

1. so open each server.properties file and change the port , so that they will be running separately

listeners=PLAINTEXT://:9093

another property file with 9094

all 3 brokers will listen to 3 different ports and with these they will start on different nodes

1. when you are running multiple broker , each broker should have different log location

do change values as

open **server-o.properties** file and do the modification

log.dirs=**/tmp/kafka-logs-0**

# initially it would be log.dirs=**/tmp/kafka-logs**

open server-1.properties file and do the modification

log.dirs=/tmp/kafka-logs-1

open server-2.properties file and do the modification

log.dirs=/tmp/kafka-logs-2

1. The only problem with above is the above will create these file in c:temp

So now change them as

1 ) make below changes in server.properties file

log.dirs=../tmp/kafka-log-\*

initially there is no double dot,whereas double dot means current directory

so from intellij u will run these commands so then and there itself in that folder these files will be created

1. open zookeeper.properties

dataDir=../tmp/zookeeper

after doing that first start zookeeper and then start kafka server with 1st property file and then start second kafka server with second property file

### Change everything to current directory

1. create an environment variable called KAFKA\_HOME because to run the below command given by udemy sir prasanth, we should have set the variable and set as below

C:\kafka\_2.13-3.3.1>setx KAFKA\_HOME C:\kafka\_2.13-3.3.1

SUCCESS: Specified value was saved.

this will set to user variables instead to a global variable

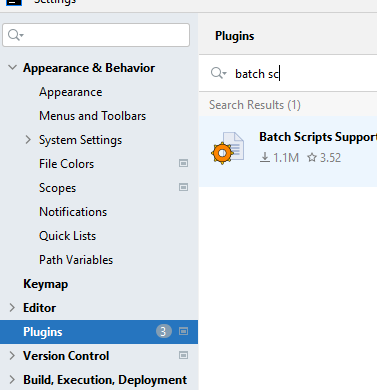
In this command only we will use the environment variable %KAFKA\_HOME%\bin\windows\kafka-server-start.bat %KAFKA\_HOME%\etc\kafka\server-0.properties

### **Cfg Batch scripts plugin in intellij**

You should be able to run the .cmd files from intellij idea

if u can’t run the command then install a plugin in intellij named “**Batch scripts support**”

The pro



To run 1st kafka node

C:\kafka\_2.13-3.3.1\bin\windows\kafka-server-start.bat C:\kafka\_2.13-3.3.1\config\server-0.properties

This will use server-0.properties file where this broker would be running on 9092

We have opened that prop file and changed the node

C:\kafka\_2.13-3.3.1\bin\windows\kafka-server-start.bat C:\kafka\_2.13-3.3.1\config\server-1.properties

This server-1.properties file will have node id as 9093 as this broker will run on node 9093

C:\kafka\_2.13-3.3.1\bin\windows\kafka-server-start.bat C:\kafka\_2.13-3.3.1\config\server-2.properties

This server-2.properties file will run the kafka broker on 9094

1. creating multiple topics

C:\kafka\_2.13-3.3.1\bin\windows\kafka-topics.bat --create --bootstrap-server localhost:9092 --topic hello-producer-1 --partitions 5 --replication-factor 3 --config min.insync.replicas=2

This will create topic with name **hello-producer-1**

Create 2nd topic

C:\kafka\_2.13-3.3.1\bin\windows\kafka-topics.bat --create --bootstrap-server localhost:9092 --topic hello-producer-2 --partitions 5 --replication-factor 3 --config min.insync.replicas=2

1. consume from 2 topics

C:\kafka\_2.13-3.3.1\bin\windows\kafka-console-consumer.bat --bootstrap-server localhost:9092 --from-beginning --whitelist "hello-producer-1|hello-producer-2"