**Installation**

**A. JDK Setup**

1. Start the JRE installation and hit the “Change destination folder” checkbox, then click 'Install.'



2. Change the installation directory to any path without spaces in the folder name. E.g. C:\Java\jre1.8.0\_xx\. (By default it will be C:\Program Files\Java\jre1.8.0\_xx), then click 'Next.'

3. Now open the system environment variables dialogue by opening Control Panel -> System -> Advanced system settings -> Environment Variables.

4. Hit the New User Variable button in the User variables section, then type JAVA\_HOME in *Variable name*and give your jre path in the *Variable value.* It should look like the below image:

*(Java path and version may change according to the version of Kafka you are using)*

5. Now click OK.

6. Search for a Path variable in the “System Variable” section in the “Environment Variables” dialogue box you just opened.

7. Edit the path and type “;%JAVA\_HOME%\bin” at the end of the text already written there, just like the image below:



8. To confirm the Java installation, just open cmd and type “*java –version.”*You should be able to see the version of Java you just installed.



If your command prompt somewhat looks like the image above, you are good to go. Otherwise, you need to recheck whether your setup version matches the correct OS architecture (x86, x64), or if the environment variables path is correct.

**B. ZooKeeper Installation**

1. Go to

E:\software\A08\file\apache-zookeeper-3.5.6-bin.tar\apache-zookeeper-3.5.6-bin\apache-zookeeper-3.5.6-bin\conf

2. Rename file “zoo\_sample.cfg” to “*zoo.cfg”*

3. Open zoo.cfg in any text editor, like Notepad; I prefer Notepad++.

4. Find and edit dataDir=/tmp/zookeeper to :\zookeeper-3.4.7\data

5. Add an entry in the System Environment Variables as we did for Java.

a. Add ZOOKEEPER\_HOME = C:\zookeeper-3.4.7 to the System Variables.

b. Edit the System Variable named “Path” and add ;%ZOOKEEPER\_HOME%\bin;

6. You can change the default Zookeeper port in zoo.cfg file (Default port 2181).

6a) To change the admin server port number, go to zoo.cfg file, add

admin.serverPort=7070

7. Run ZooKeeper by opening a new cmd and type zkserver.

8. You will see the command prompt with some details, like the image below:



Congratulations, your ZooKeeper is up and running on port 2181!

**C. Setting Up Kafka**

1. Go to your Kafka config directory. For me its *C:\kafka\_2.11-0.9.0.0\config*

2. Edit the file “server.properties.”

3. Find and edit the linelog.dirs=/tmp/kafka-logs” to “log.dir= C:\kafka\_2.11-0.9.0.0\kafka-logs*.*

4. If your ZooKeeper is running on some other machine or cluster you can edit *“zookeeper.connect:2181”* to your custom IP and port. For this demo, we are using the same machine so there's no need to change. Also the Kafka port and broker.id are configurable in this file. Leave other settings as is.

5. Your Kafka will run on default port 9092 and connect to ZooKeeper’s default port, 2181.

**D. Running a Kafka Server**

*Important: Please ensure that your ZooKeeper instance is up and running before starting a Kafka server.*

1. Go to your Kafka installation directory: *C:\kafka\_2.11-0.9.0.0\*

Rename dir to kafka

1. Open a command prompt here by pressing *Shift + right click*and choose the “Open command window here” option).

2a. In the command prompt set java memory

Change java heap options in kafka-sever-start.bat

set JAVA\_OPTS=-Xms512m -Xmx512m -XX:MaxPermSize=256m

3. Now type .\bin\windows\kafka-server-start.bat .\config\server.properties and press Enter.

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

Image title

4. If everything went fine, your command prompt will look like this:

  
5. Now your Kafka Server is up and running, you can create topics to store messages. Also, we can produce or consume data from Java or Scala code or directly from the command prompt.

**E. Creating Topics**

1. Now create a topic with the name *“test”* and a replication factor of 1, as we have only one Kafka server running. If you have a cluster with more than one Kafka server running, you can increase the replication-factor accordingly, which will increase the data availability and act like a fault-tolerant system.

2. Open a new command prompt in the location *C:\kafka\_2.11-0.9.0.0\bin\windows.*

3. Type the following command and hit Enter:

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



**F. Creating a Producer and Consumer to Test Server**

1. Open a new command prompt in the location *C:\kafka\_2.11-0.9.0.0\bin\windows*

2. To start a producer type the following command:

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

3. Again open a new command prompt in the same location as *C:\kafka\_2.11-0.9.0.0\bin\windows*

4. Now start a consumer by typing the following command:

kafka-console-consumer.bat --bootstrap-server localhost:9092 --topic test --from-beginning

5. Now you will have two command prompts, like the image below:



6. Now type anything in the producer command prompt and press Enter, and you should be able to see the message in the other consumer command prompt.



7. If you are able to push and see your messages on the consumer side, you are done with Kafka setup.

**Some Other Useful Commands**

1. List Topics: kafka-topics.bat --list --zookeeper localhost:2181
2. Describe Topic: kafka-topics.bat --describe --zookeeper localhost:2181 --topic [Topic Name]
3. Read messages from the beginning: kafka-console-consumer.bat --zookeeper localhost:2181 --topic [Topic Name] --from-beginning
4. Delete Topic: kafka-run-class.bat kafka.admin.TopicCommand --delete --topic [topic\_to\_delete] --zookeeper localhost:2181