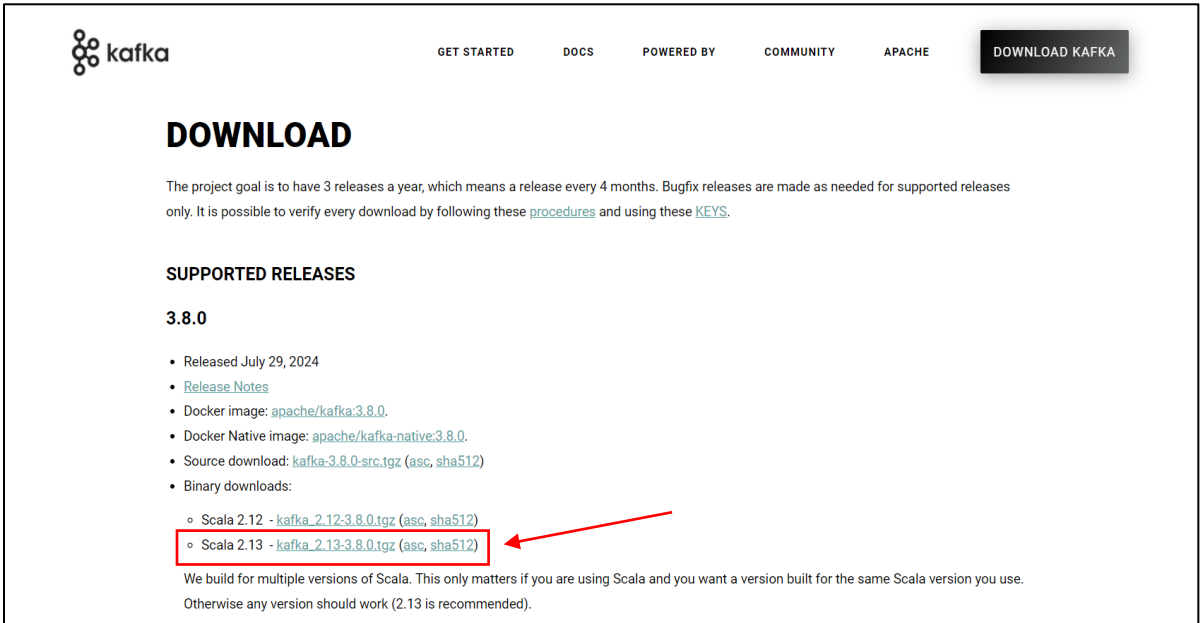


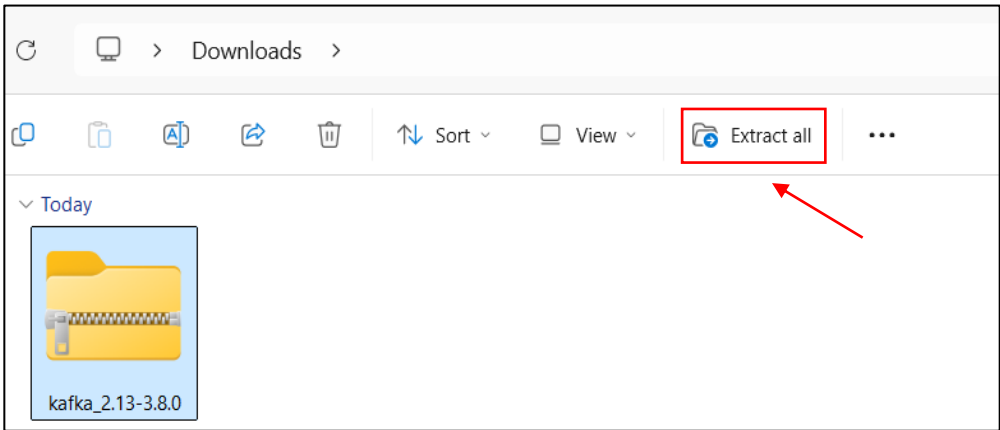
M1-FA2: Kafka

❖ First Step: Download and Install

The first step to do when installing Apache Kafka is to download the application through the official Apache website (<https://kafka.apache.org/downloads>). Scroll down and search for the latest version under the “Supported Releases” section and click the download link beside the Scala 2.13 as shown in the image below.



After you successfully downloaded the kafka_2.13-3.8.0 file, extract the folder and move the contents to another folder in the C drive. You can either hover over the file and press the right click to access the pop-up menu to extract it or simply click the “Extract all” option which is located above the downloaded file.



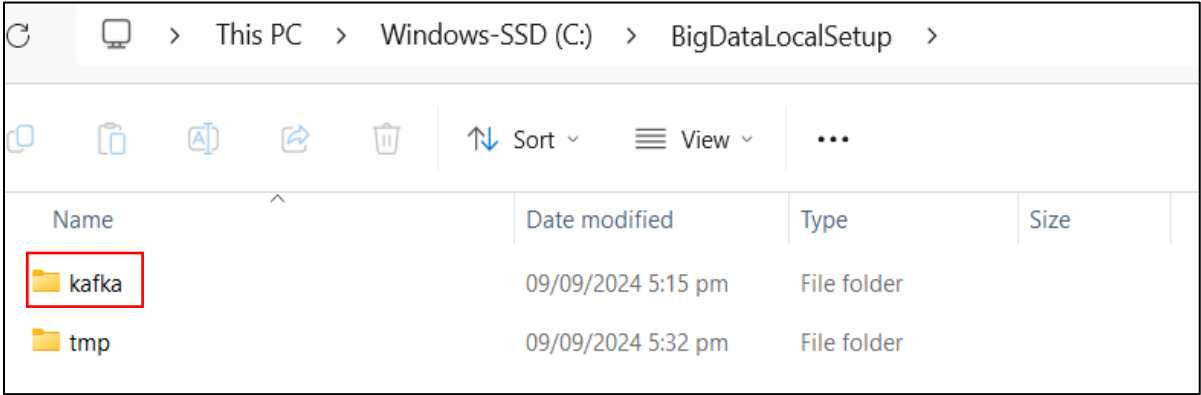
❖ Second Step: Create a New Folder for the Extracted File

The second step involves creating a new folder for the extracted Kafka file in your primary storage or C drive. You can also use any secondary drive like the D drive if your primary drive

is almost full or if you simply prefer to store the folder in backup drives. Rename the new folder as “BigDataLocalSetup” or anything you want as long as you have a new folder.

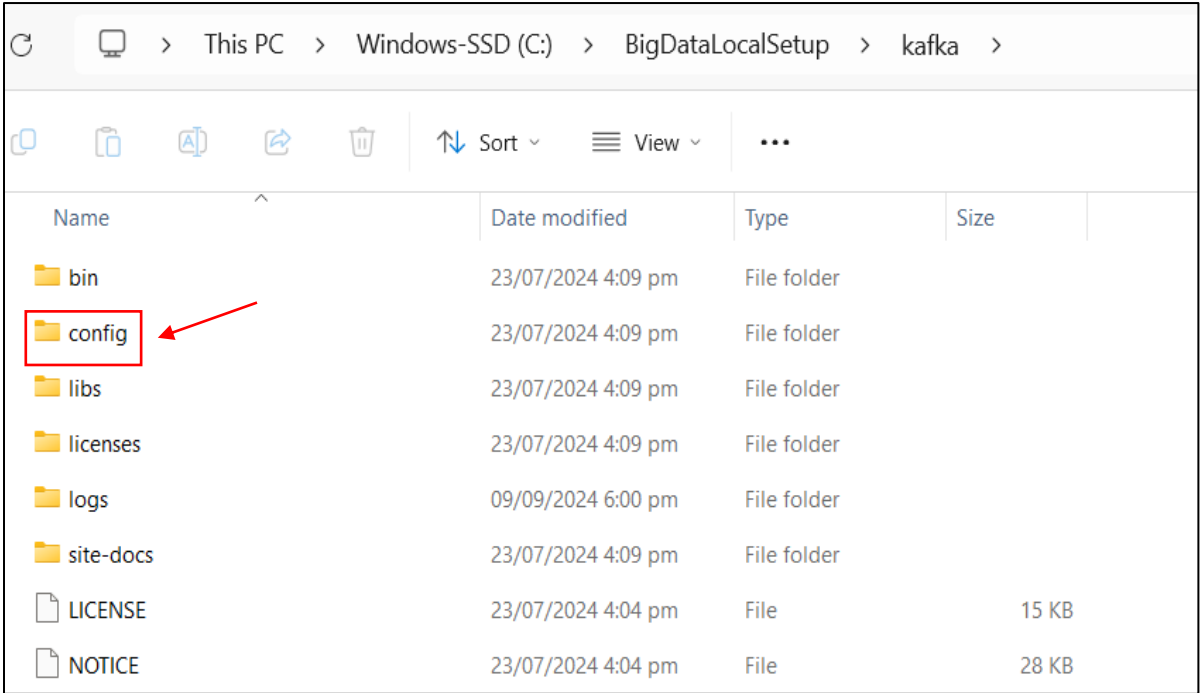


Click the “BigDataLocalSetup” folder to open it and copy the extracted kafka_2.13-3.8.0 file and paste it inside the new folder. You can keep the name of the file as it is but we prefer to rename it as “kafka” without the version number to make it more intuitive to set up.

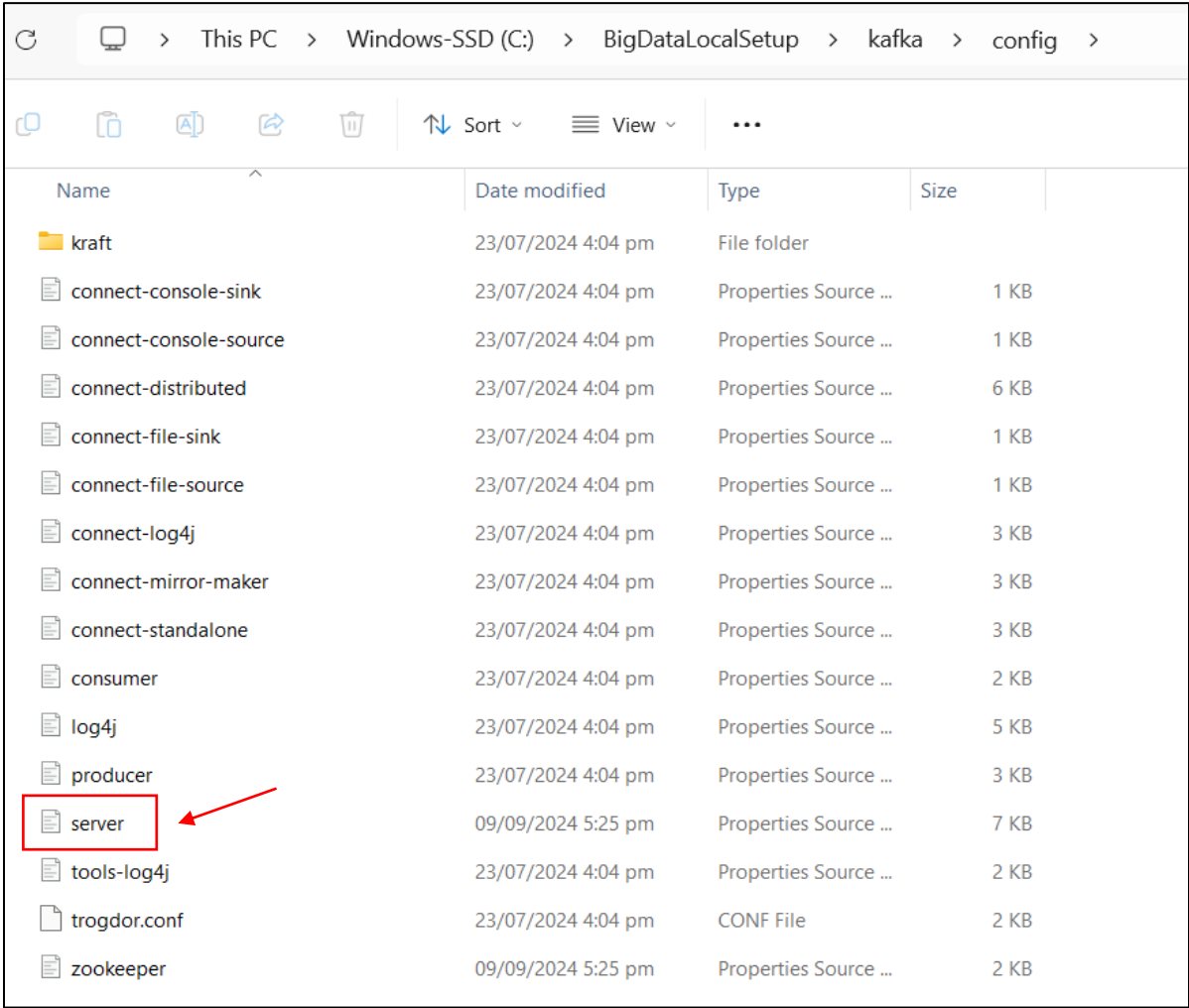


❖ **Third Step: Make Changes in the Configuration File**

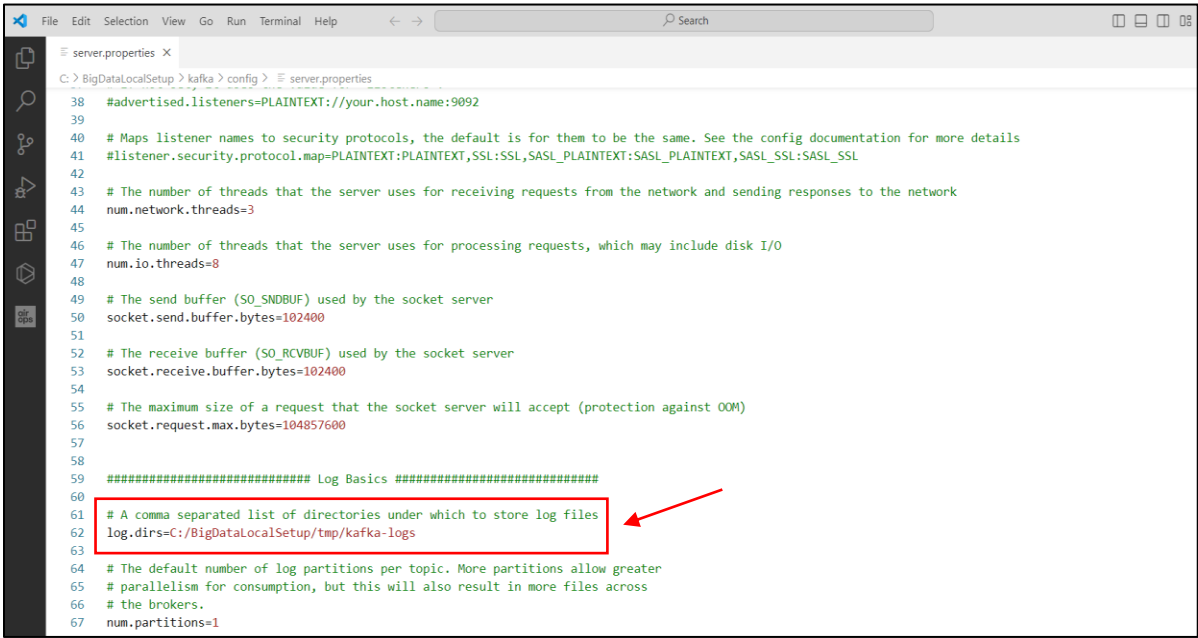
Open the “kafka” folder to access its contents as shown in the following image. Click the configuration “config” folder and search for the server.properties file.



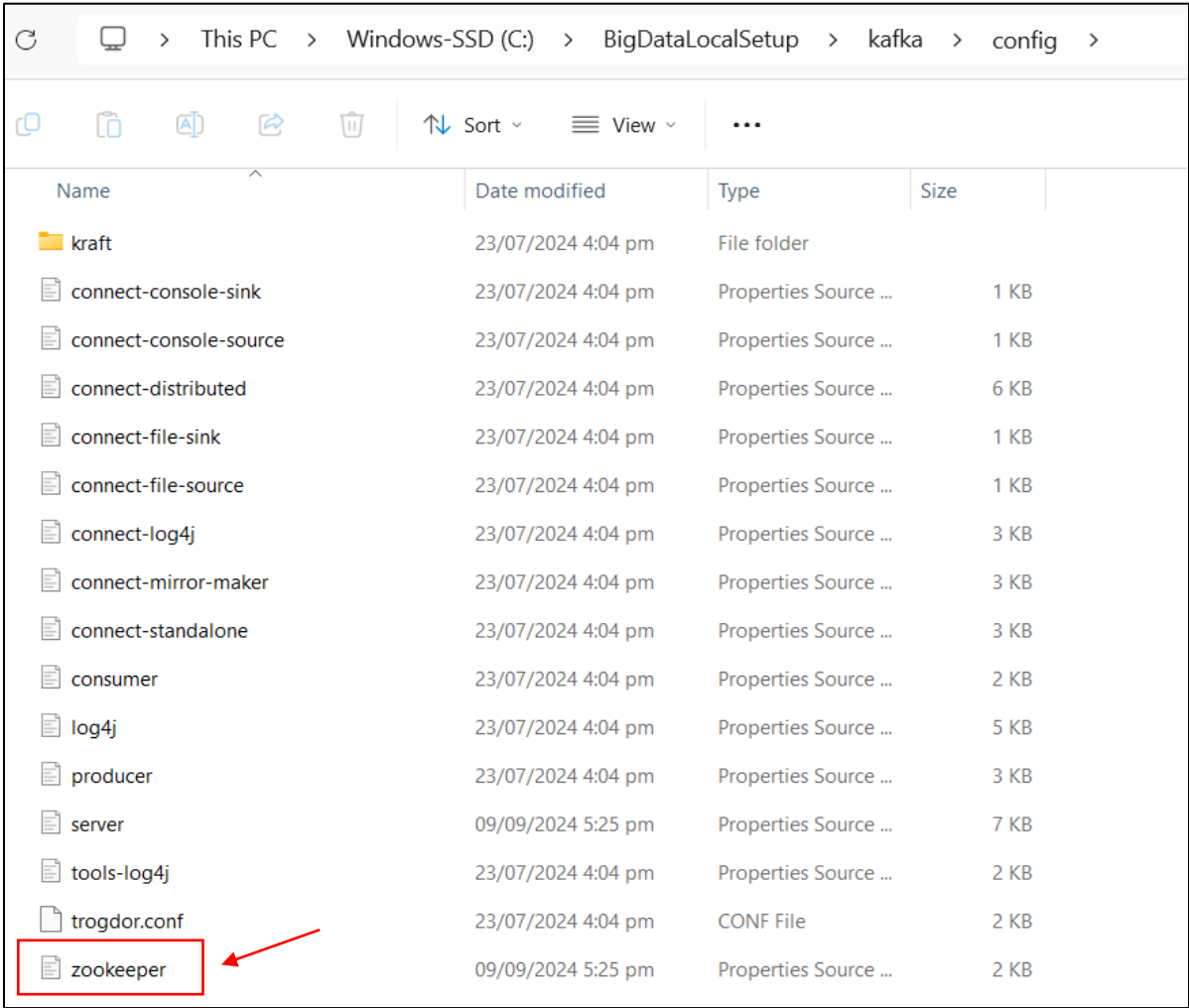
Open the server.properties file by clicking it and then you will be redirected to any integrated development environment (IDE) that you are using. For this particular case, we opened the file using Visual Studio Code (VS Code).



After the server.properties file is opened in VS Code, scroll down and look for the log directory (`log.dirs = /tmp/kafka-logs`) or you can simply use the search function by typing “log” so you can easily go to that specific line of code. In this case, the name of the newly created folder earlier is “BigDataLocalSetup” and it is located in the C drive. Adding it to the existing code will make it `C:/BigDataLocalSetup/tmp/kafka-logs`. Don’t forget to save the file before you exit the IDE.

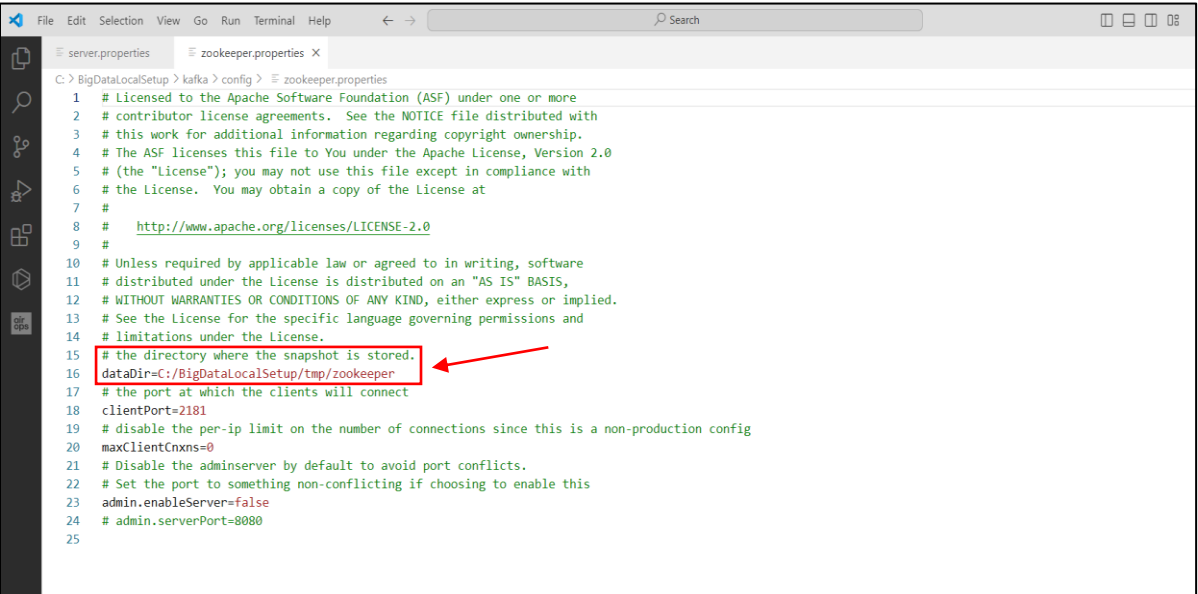


Go back to the configuration “config” folder and find the zookeeper.properties file and click it to view the codes inside just like what you did with the server.properties file.



Name	Date modified	Type	Size
kraft	23/07/2024 4:04 pm	File folder	
connect-console-sink	23/07/2024 4:04 pm	Properties Source ...	1 KB
connect-console-source	23/07/2024 4:04 pm	Properties Source ...	1 KB
connect-distributed	23/07/2024 4:04 pm	Properties Source ...	6 KB
connect-file-sink	23/07/2024 4:04 pm	Properties Source ...	1 KB
connect-file-source	23/07/2024 4:04 pm	Properties Source ...	1 KB
connect-log4j	23/07/2024 4:04 pm	Properties Source ...	3 KB
connect-mirror-maker	23/07/2024 4:04 pm	Properties Source ...	3 KB
connect-standalone	23/07/2024 4:04 pm	Properties Source ...	3 KB
consumer	23/07/2024 4:04 pm	Properties Source ...	2 KB
log4j	23/07/2024 4:04 pm	Properties Source ...	5 KB
producer	23/07/2024 4:04 pm	Properties Source ...	3 KB
server	09/09/2024 5:25 pm	Properties Source ...	7 KB
tools-log4j	23/07/2024 4:04 pm	Properties Source ...	2 KB
trogdor.conf	23/07/2024 4:04 pm	CONF File	2 KB
zookeeper	09/09/2024 5:25 pm	Properties Source ...	2 KB

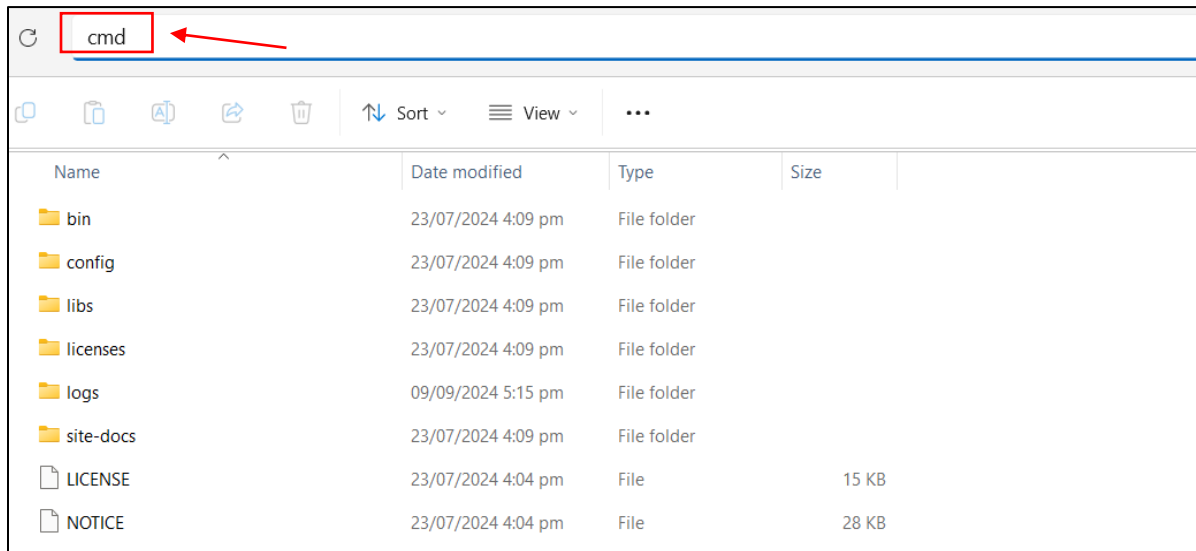
Repeat the same configuration process in the zookeeper.properties file. Using any IDE, scroll down and look for the data directory (`dataDir = /tmp/zookeeper`). Since the name of the newly created folder earlier is “BigDataLocalSetup” and it is located in the C drive, simply add it to the existing code to make it `C:/BigDataLocalSetup/tmp/zookeeper`. Again, don’t forget to save the file before you exit the IDE.



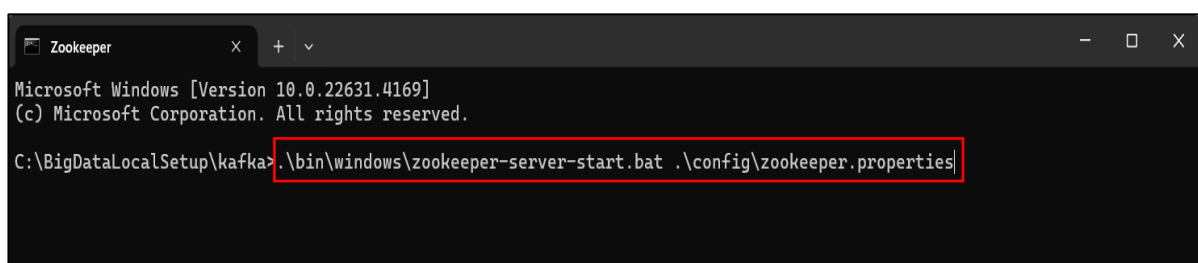
```
C:\BigDataLocalSetup\kafka>config>zookeeper.properties
1 # Licensed to the Apache Software Foundation (ASF) under one or more
2 # contributor license agreements. See the NOTICE file distributed with
3 # this work for additional information regarding copyright ownership.
4 # The ASF licenses this file to You under the Apache License, Version 2.0
5 # (the "License"); you may not use this file except in compliance with
6 # the License. You may obtain a copy of the License at
7 #
8 # http://www.apache.org/licenses/LICENSE-2.0
9 #
10 # Unless required by applicable law or agreed to in writing, software
11 # distributed under the License is distributed on an "AS IS" BASIS,
12 # WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
13 # See the License for the specific language governing permissions and
14 # limitations under the License.
15 # the directory where the snapshot is stored.
16 dataDir=C:/BigDataLocalSetup/tmp/zookeeper
17 # the port at which the clients will connect
18 clientPort=2181
19 # disable the per-ip limit on the number of connections since this is a non-production config
20 maxClientCnxns=0
21 # Disable the adminserver by default to avoid port conflicts.
22 # Set the port to something non-conflicting if choosing to enable this
23 admin.enableServer=false
24 # admin.serverPort=0000
25
```

❖ Fourth Step: Launch Zookeeper and Kafka Using the Command Prompt

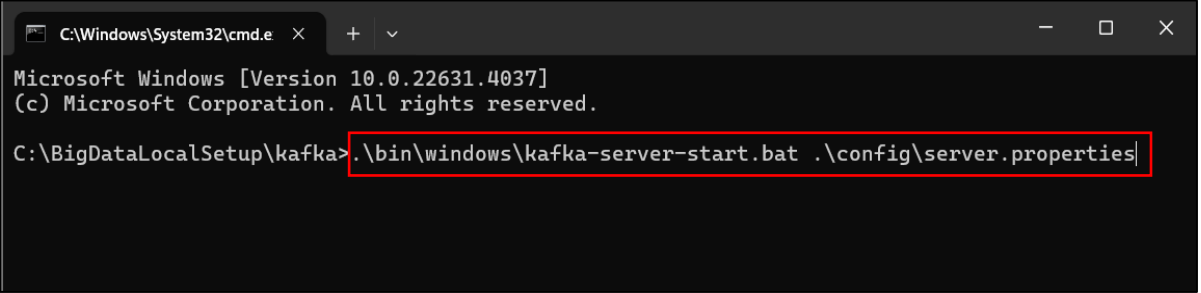
Go back to the “kafka” folder and in the command sequence, you can type `cmd` to open the command prompt directly from this folder. You can also do this by manually opening your command prompt and use the `cd` command to access the `C:\BigDataLocalSetup\kafka` file path. However, it is always easier to do the first option as shown below.



And then in the command prompt, rename the tab and type the command `.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties`. Click the “enter” button and the following results will show. It means that you have successfully launched the Zookeeper. Based on the Kafka documentation, ZooKeeper is held responsible to maintain the data configuration, manage synchronization, and can even allow group services which makes it important to Apache Kafka’s proper functioning.

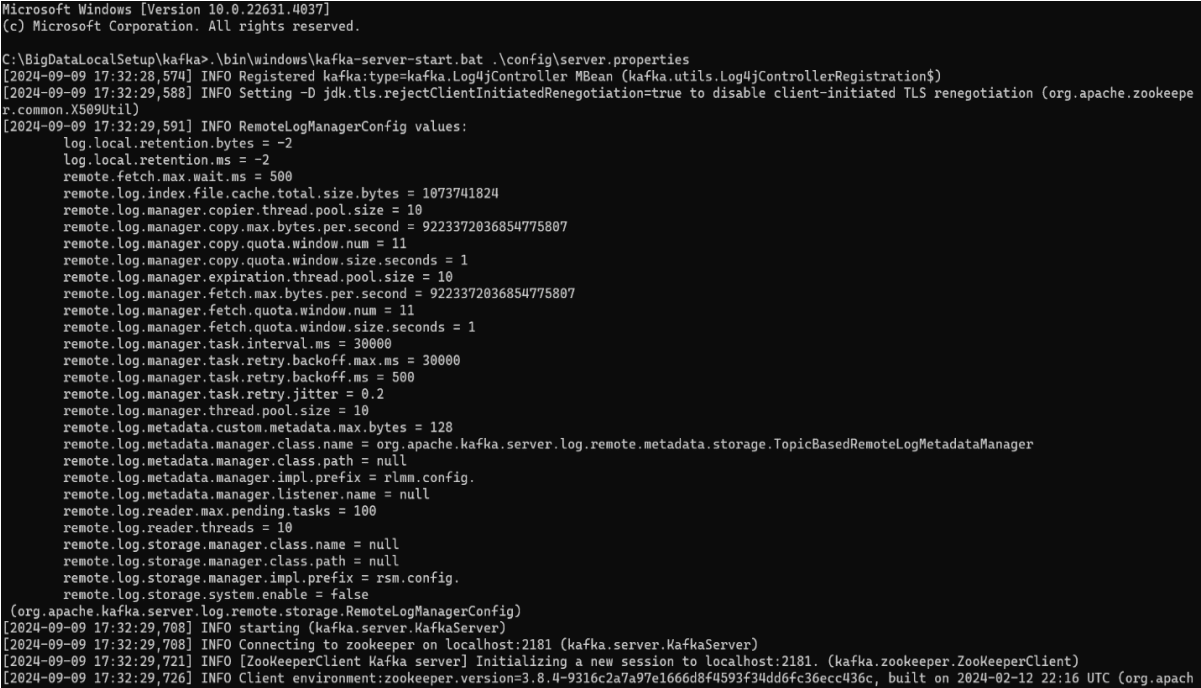


After launching the Zookeeper, go back to the kafka folder again and type the cmd in the command sequence to open another command prompt. This time, type the command `.\bin\windows\kafka-server-start.bat .\config\server.properties` in the prompt and click enter and the following results will show. Kafka will output a series of log messages to indicate the progress of the server starting up. Since we are using a version of Kafka that relies on ZooKeeper, the broker needs to communicate with ZooKeeper. After Kafka is successfully started, you can proceed with Kafka tasks like creating topics, producing messages, and consuming messages.



```
C:\Windows\System32\cmd.e  X  +  v
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-server-start.bat .\config\server.properties
```

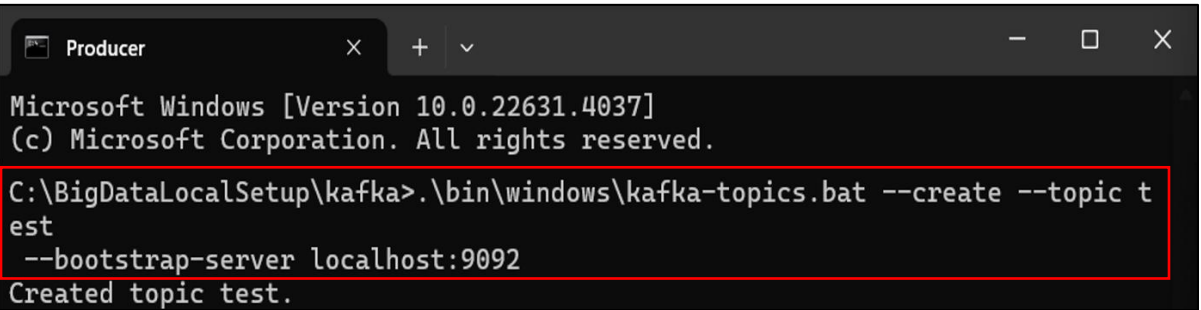


```
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-server-start.bat .\config\server.properties
[2024-09-09 17:32:28,574] INFO Registered kafka:type-kafka.Log4jController MBean (kafka.utils.Log4jControllerRegistration$)
[2024-09-09 17:32:29,588] INFO Setting -D jdk.tls.rejectClientInitiatedRenegotiation=true to disable client-initiated TLS renegotiation (org.apache.zookeeper.common.X509Util)
[2024-09-09 17:32:29,591] INFO RemoteLogManagerConfig values:
    log.local.retention.bytes = -2
    log.local.retention.ms = -2
    remote.fetch.max.wait.ms = 500
    remote.log.index.file.cache.total.size.bytes = 1073741824
    remote.log.manager.copier.thread.pool.size = 10
    remote.log.manager.copy.max.bytes.per.second = 9223372036854775807
    remote.log.manager.copy.quota.window.num = 11
    remote.log.manager.copy.quota.window.size.seconds = 1
    remote.log.manager.expiration.thread.pool.size = 10
    remote.log.manager.fetch.max.bytes.per.second = 9223372036854775807
    remote.log.manager.fetch.quota.window.num = 11
    remote.log.manager.fetch.quota.window.size.seconds = 1
    remote.log.manager.task.interval.ms = 30000
    remote.log.manager.task.retry.backoff.max.ms = 30000
    remote.log.manager.task.retry.backoff.ms = 500
    remote.log.manager.task.retry.jitter = 0.2
    remote.log.manager.thread.pool.size = 10
    remote.log.metadata.custom.metadata.max.bytes = 128
    remote.log.metadata.manager.class.name = org.apache.kafka.server.log.remote.metadata.storage.TopicBasedRemoteLogMetadataManager
    remote.log.metadata.manager.class.path = null
    remote.log.metadata.manager.impl.prefix = rlm.config.
    remote.log.metadata.manager.listener.name = null
    remote.log.reader.max.pending.tasks = 100
    remote.log.reader.threads = 10
    remote.log.storage.manager.class.name = null
    remote.log.storage.manager.class.path = null
    remote.log.storage.manager.impl.prefix = rsm.config.
    remote.log.storage.system.enable = false
(org.apache.kafka.server.log.remote.storage.RemoteLogManagerConfig)
[2024-09-09 17:32:29,708] INFO starting (kafka.server.KafkaServer)
[2024-09-09 17:32:29,708] INFO Connecting to zookeeper on localhost:2181 (kafka.server.KafkaServer)
[2024-09-09 17:32:29,721] INFO [ZooKeeperClient Kafka server] Initializing a new session to localhost:2181. (kafka.zookeeper.ZooKeeperClient)
[2024-09-09 17:32:29,726] INFO Client environment:zookeeper.version=3.8.4-9316c2a7a97e1666d8f4593f34dd6fc36ecc436c, built on 2024-02-12 22:16 UTC (org.apach
```

❖ **Fifth Step: Create a Topic in Kafka for Verification**

Now that both Kafka and Zookeeper are successfully launched, you can already create a topic to test if the Kafka server is functioning as expected. You need to open two more command prompts using the same process. In one of the command prompts, rename the tab as “Producer” and type in the `.\bin\windows\kafka-topics.bat --create --topic test --bootstrap-server localhost:9092` command and click enter.



```
Producer  X  +  v
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.

C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-topics.bat --create --topic test
--bootstrap-server localhost:9092
Created topic test.
```


In the other command prompt, rename the tab as "Consumer" and type the `.\bin\windows\kafka-console-consumer.bat --topic test --from beginning --bootstrap-server localhost:9092` command and click enter.

```
Consumer
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-console-consumer.bat --topic
test --from-beginning --bootstrap-server localhost:9092
```

You will notice that a topic test is created in the Producer tab. It is important to note that in a Kafka server, the Producer and Consumer are connected and can communicate through topics, so when the Producer sends a message to the specified topic named "test", the Consumer will be able to receive it. After testing, it also proves that Kafka can efficiently handle real-time data streams as demonstrated in the following codes and terminal output.

```
Producer
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-topics.bat --create --topic t
est
--bootstrap-server localhost:9092
Created topic test.
C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-console-producer.bat --topic
test
>.\BigDataLocalSetup\kafka>.\bin\windows\kafka-console-producer.bat --topic
test --bootstrap-server localhost:9092
>hi
>this is group5
>testing our first kafka activity
>testing setup
>testing console
>thank you
>
```

```
Consumer
Microsoft Windows [Version 10.0.22631.4037]
(c) Microsoft Corporation. All rights reserved.
C:\BigDataLocalSetup\kafka>.\bin\windows\kafka-console-consumer.bat --topic
test --from-beginning --bootstrap-server localhost:9092

hi
this is group5
testing our first kafka activity
testing setup
testing console
thank you
|
```

REFERENCES:

Apache Software Foundation (n.d.). *DOCUMENTATION: Kafka 3.8 Documentation*.

https://kafka.apache.org/documentation/#basic_ops

Big Tech Talk. (2023, August 28). *How to Install Apache Kafka on Windows* [Video]. YouTube.

<https://www.youtube.com/watch?v=heXd6JA2TQc&t=249s>