

Apache Kafka Documentation

Introduction:-

Apache Kafka is a distributed publish-subscribe messaging system and a robust queue that can handle a high volume of data and enables you to pass messages from one end-point to another. Kafka is suitable for both offline and online message consumption. Kafka messages are persisted on the disk and replicated within the cluster to prevent data loss. Kafka is built on top of the ZooKeeper synchronization service. It integrates very well with Apache Storm and Spark for real-time streaming data analysis.

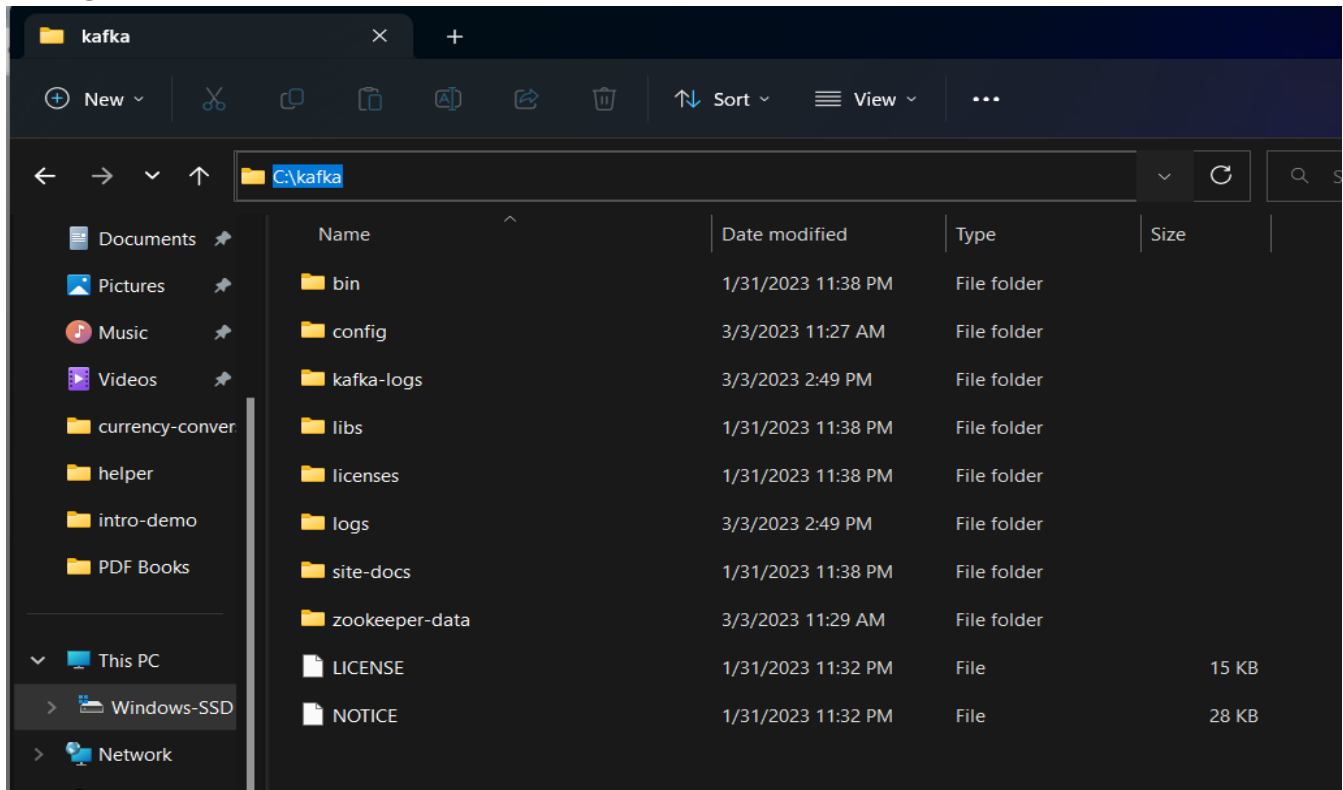
Benefits

Following are a few benefits of Kafka –

1. **Reliability** – Kafka is distributed, partitioned, replicated and fault tolerance.
2. **Scalability** – Kafka messaging system scales easily without down time..
3. **Durability** – Kafka uses Distributed commit log which means messages persists on disk as fast as possible, hence it is durable..
4. **Performance** – Kafka has high throughput for both publishing and subscribing messages. It maintains stable performance even many TB of messages are stored.

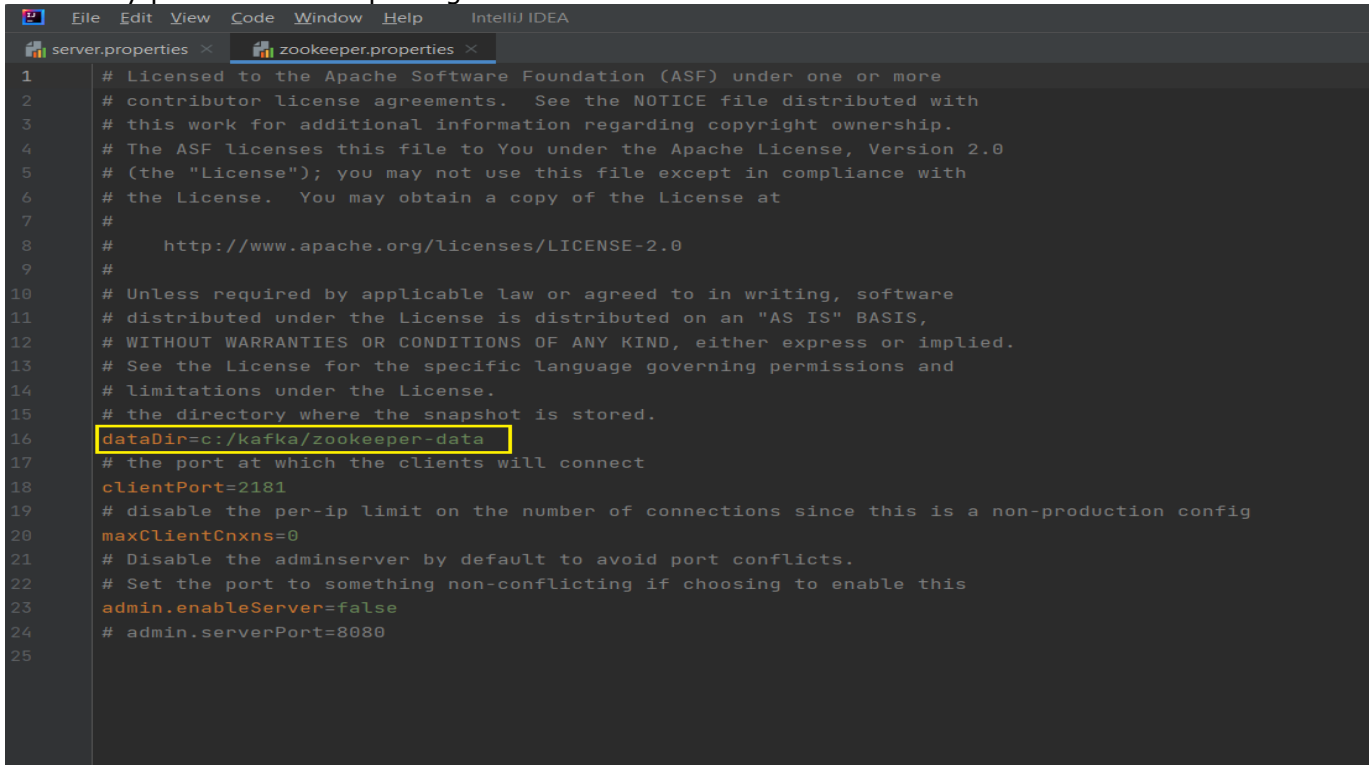
Installation of Windows:-

1. Download Apache Kafka latest release **kafka_x_xx_x.x.x.tgz** file from official site <https://kafka.apache.org/downloads>.
2. After download extract Tar file using 7zip and save extracted folder using name **kafka** and move to **C Drive**.



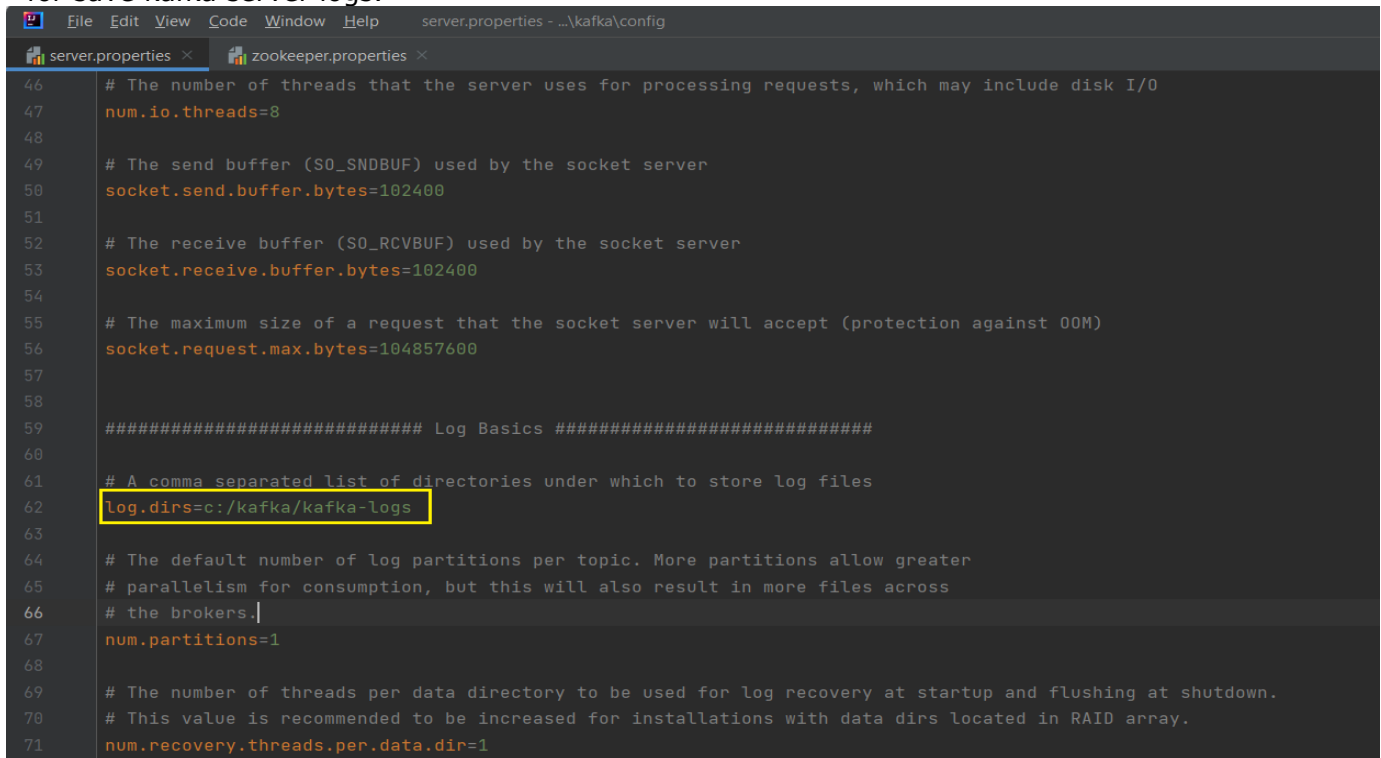
Configuration Kafka Server and ZooKeeper:-

1. In kafka folder go to **config** folder **zookeeper.properties** and edit it, go to **dataDir** section for set directory path for zookeeper log data.



```
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:/kafka/zookeeper-data
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=8080
25
```

2. Same in config folder go to **server.properties** and edit it, go to **log.dirs** section and set directory path for save kafka server logs.

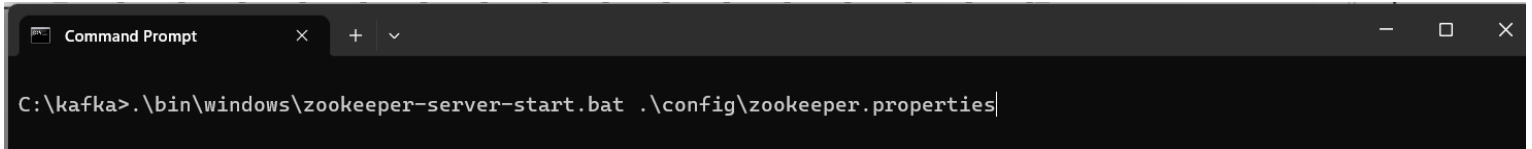


```
46 # The number of threads that the server uses for processing requests, which may include disk I/O
47 num.io.threads=8
48
49 # The send buffer (SO_SNDBUF) used by the socket server
50 socket.send.buffer.bytes=102400
51
52 # The receive buffer (SO_RCVBUF) used by the socket server
53 socket.receive.buffer.bytes=102400
54
55 # The maximum size of a request that the socket server will accept (protection against OOM)
56 socket.request.max.bytes=104857600
57
58 ##### Log Basics #####
59
60 # A comma separated list of directories under which to store log files
61 log.dirs=c:/kafka/kafka-logs
62
63 # The default number of log partitions per topic. More partitions allow greater
64 # parallelism for consumption, but this will also result in more files across
65 # the brokers.
66 num.partitions=1
67
68 # The number of threads per data directory to be used for log recovery at startup and flushing at shutdown.
69 # This value is recommended to be increased for installations with data dirs located in RAID array.
70 num.recovery.threads.per.data.dir=1
71
```

Run Kafka Server and ZooKeeper:-

1. Open CMD and go to directory path of kafka stored, **select path cd C:\kafka**
2. For running ZooKeeper run following command,

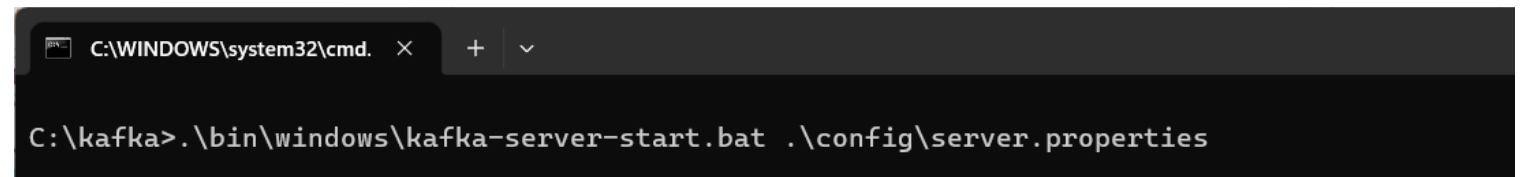
.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties



```
Command Prompt
C:\kafka>.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
```

Zookeeper will be start running on default port **2181**.

3. Open another CMD for run kafka server go same above path **cd C:\kafka**
4. For running Kafka Server run following command,



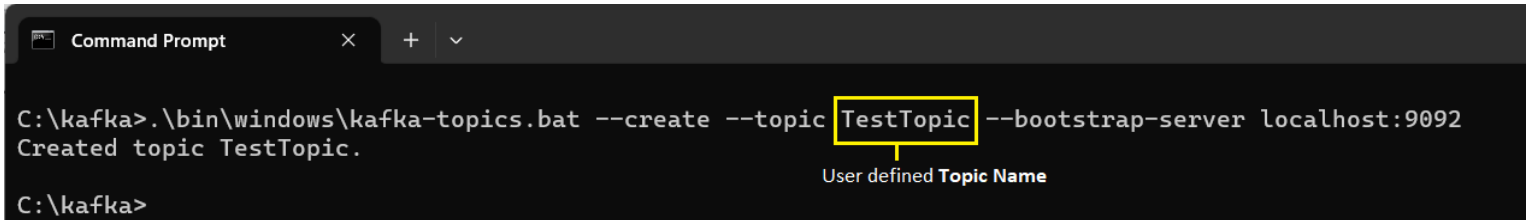
```
C:\WINDOWS\system32\cmd.
C:\kafka>.\bin\windows\kafka-server-start.bat .\config\server.properties
```

Kafka server will be start on default port **9092**.

Create Topics:-

For creating topic in CMD run following command,

.\bin\windows\kafka-topics.bat --create --topic <topic_name> --bootstrap-server localhost:9092



```
Command Prompt
C:\kafka>.\bin\windows\kafka-topics.bat --create --topic TestTopic --bootstrap-server localhost:9092
Created topic TestTopic.
C:\kafka>
```

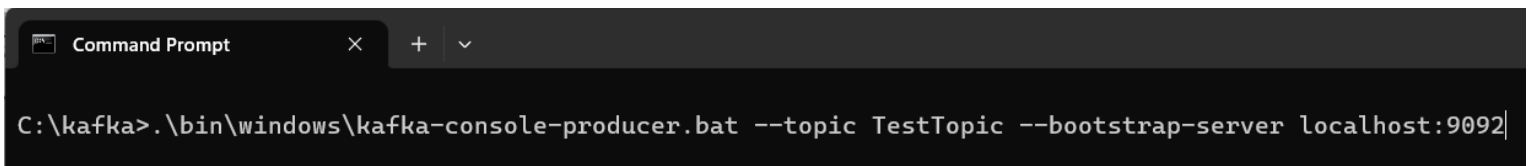
User defined Topic Name

Create Kafka Events using Kafka Producer:-

For creating events use following command,

.\bin\windows\kafka-console-producer.bat --topic <topic_name> --bootstrap-server localhost:9092


NOTE: Topic name should be same as created earlier.



```
Command Prompt
C:\kafka>.\bin\windows\kafka-console-producer.bat --topic TestTopic --bootstrap-server localhost:9092
```

And hit enter and start typing some messages for sent to the consumer.

```
Command Prompt - .\bin\win x + v
C:\kafka>.\bin\windows\kafka-console-producer.bat --topic TestTopic --bootstrap-server localhost:9092
>This is first event
>Welcome to kafka server events
>
```



Listen Kafka Events using Kafka Consumer:-

For listen message coming from producer use following command,

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

```
Command Prompt x + v
C:\kafka>.\bin\windows\kafka-console-consumer.bat --topic TestTopic --from-beginning --bootstrap-server localhost:9092|
```

Messages will be showing which is entered in producer screen,

```
Command Prompt - .\bin\win x + v
C:\kafka>.\bin\windows\kafka-console-consumer.bat --topic TestTopic --from-beginning --bootstrap-server localhost:9092
This is first event
Welcome to kafka server events
```