Hands-on Lab: Kafka Python Client

Estimated time needed: **30** minutes

Objectives

After completing this lab you will be able to:

- Use `kafka-python` to interact with Kafka server in Python
- Send and receive messages through the Kafka-python client

About Skills Network Cloud IDE

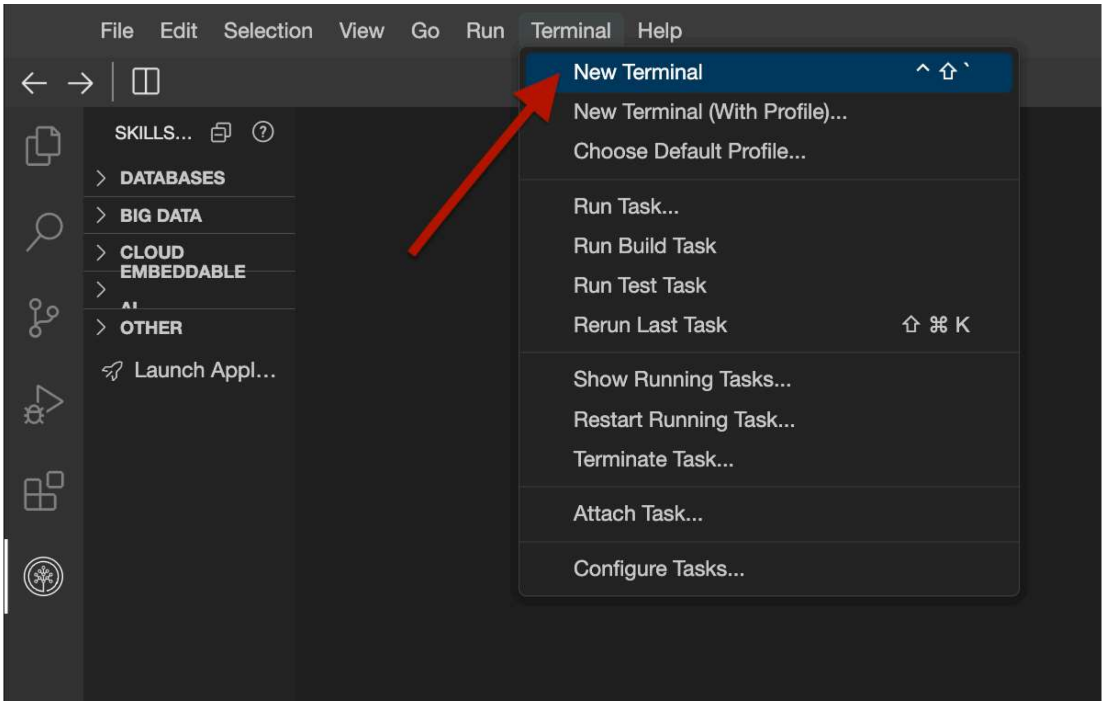
Skills Network Cloud IDE (based on Theia and Docker) provides an environment for hands on labs for course and project related labs. Theia is an open source IDE (Integrated Development Environment), that can be run on desktop or on the cloud. To complete this lab, we will be using the Cloud IDE based on Theia running in a Docker container.

Important notice about this lab environment

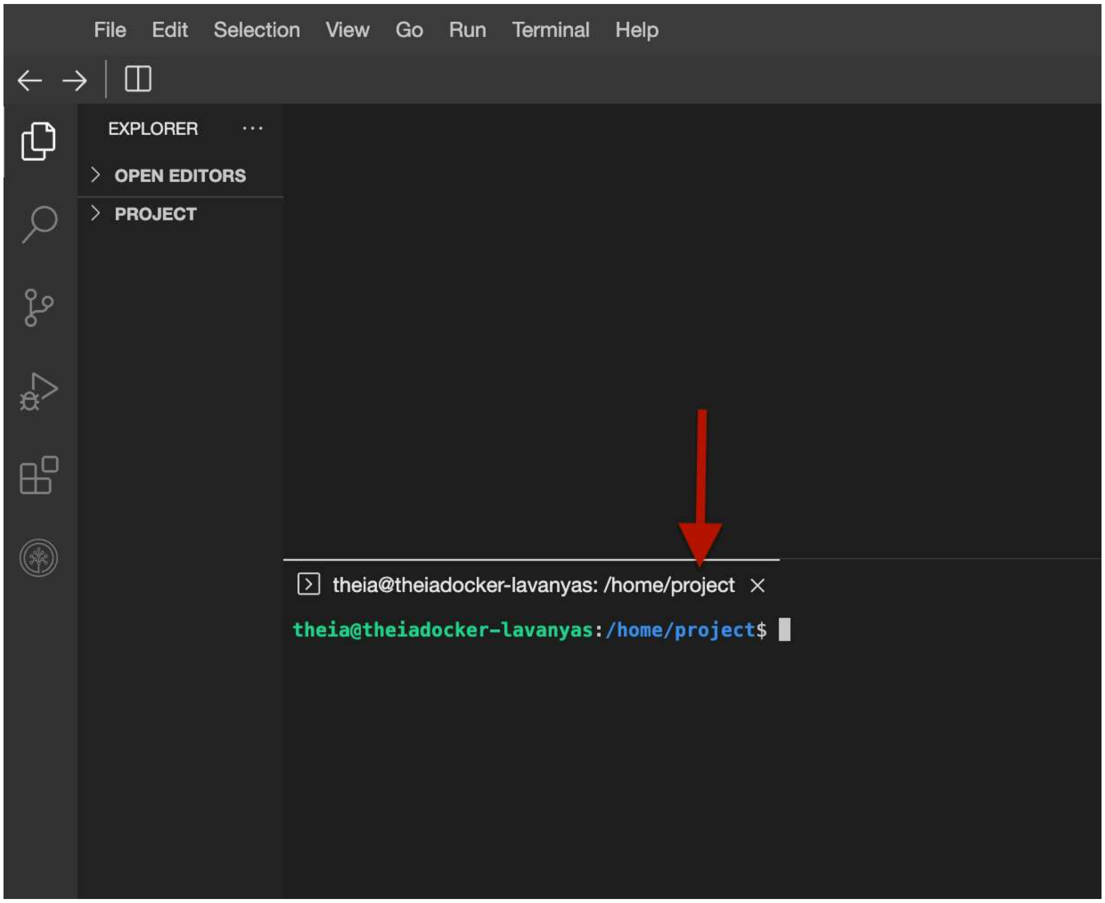
Please be aware that sessions for this lab environment are not persistent. A new environment is created for you every time you connect to this lab. Any data you may have saved in an earlier session will get lost. To avoid losing your data, please plan to complete these labs in a single session.

Exercise 1: Download and extract Kafka

1. Open a new terminal, by clicking on the menu bar and selecting **Terminal**->**New Terminal**, as shown in the image below.



This will open a new terminal at the bottom of the screen.



```
theia@theiadocker-lavanyas:/home/project$
```

Run the commands below on the newly opened terminal. (You can copy the code by clicking the little copy button on the bottom right of the codeblock below and then paste it, wherever you wish.)

2. Download Kafka, by running the command below:

```
wget https://archive.apache.org/dist/kafka/3.5.1/kafka_2.12-3.5.1.tgz
```

Copied! Executed!

3. Extract kafka from the zip file by running the command below.

```
tar -xzf kafka_2.12-3.5.1.tgz
```

Copied! Executed!

This creates a new directory `kafka_2.12-3.5.1` in the current directory.

Exercise 2: Start ZooKeeper

1. Change to the `kafka_2.12-3.5.1` directory.

```
cd kafka_2.12-3.5.1
```

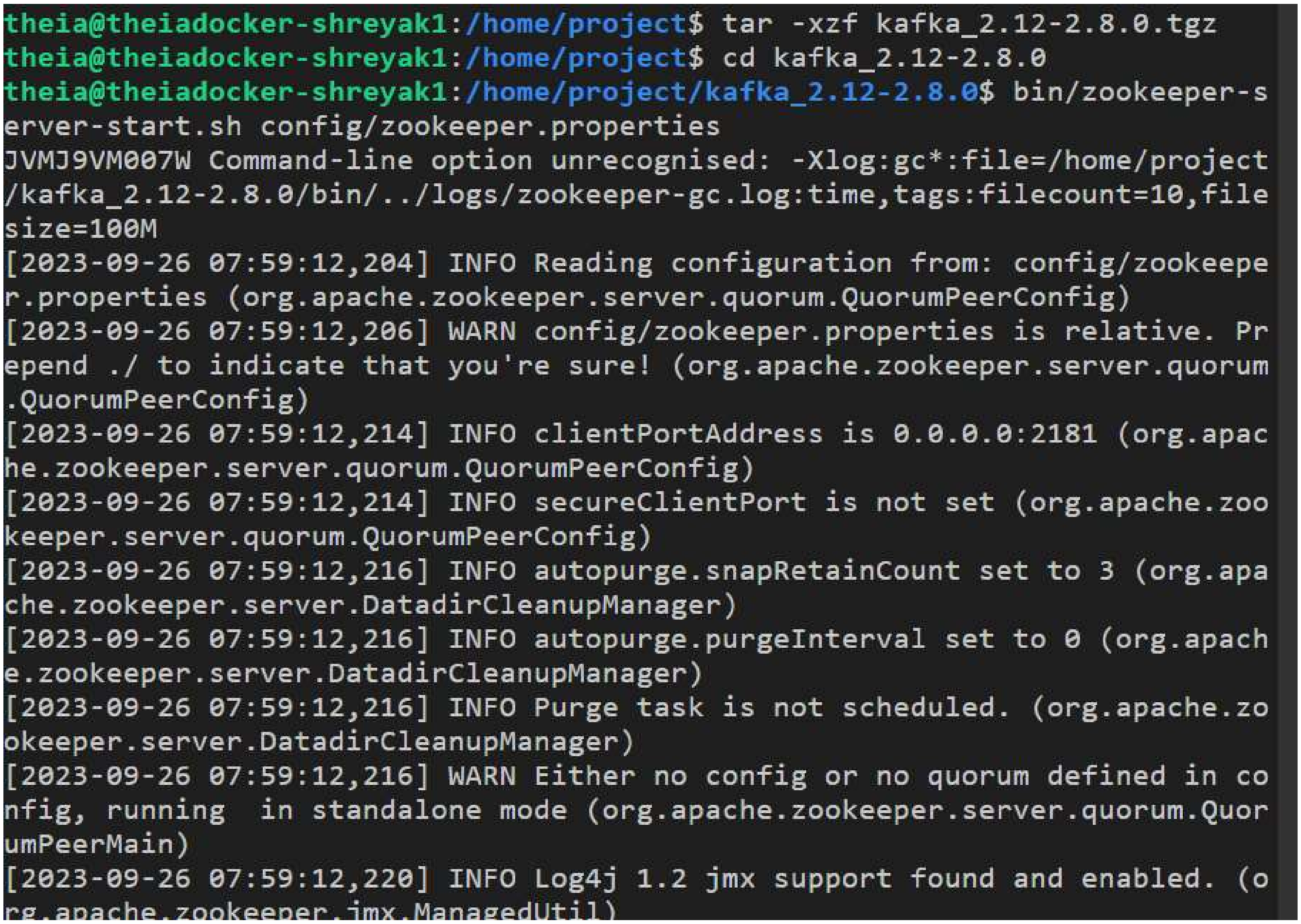
Copied! Executed!

2. ZooKeeper is required for Kafka to work. Start the ZooKeeper server.

```
bin/zookeeper-server-start.sh config/zookeeper.properties
```

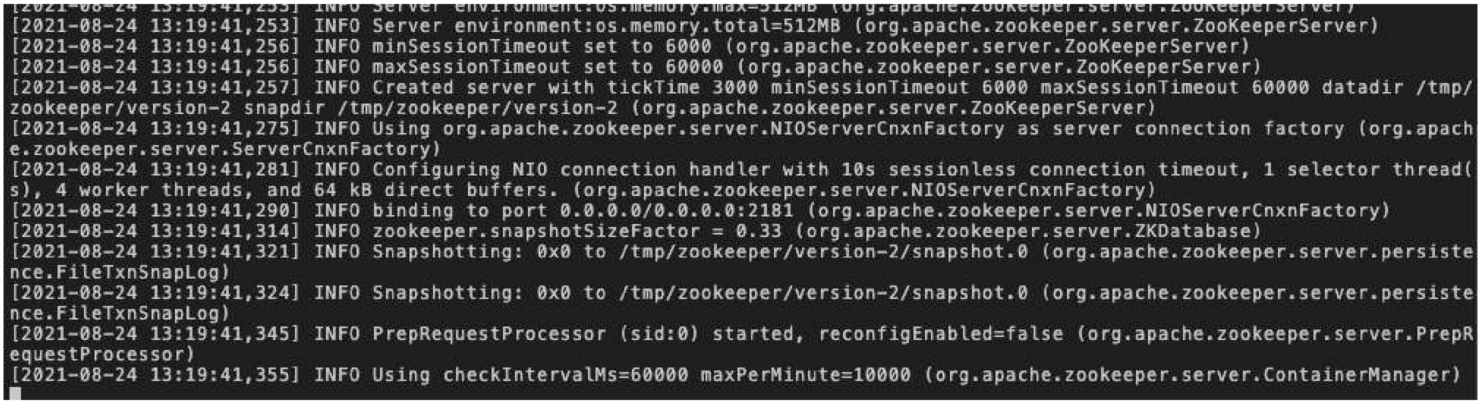
Copied! Executed!

When ZooKeeper starts you should see an output like this:



```
theia@theiadocker-shreyak1:/home/project$ tar -xzf kafka_2.12-2.8.0.tgz
theia@theiadocker-shreyak1:/home/project$ cd kafka_2.12-2.8.0
theia@theiadocker-shreyak1:/home/project/kafka_2.12-2.8.0$ bin/zookeeper-s
erver-start.sh config/zookeeper.properties
JVMJ9VM007W Command-line option unrecognised: -Xlog:gc*:file=/home/project
/kafka_2.12-2.8.0/bin/../logs/zookeeper-gc.log:time,tags:filecount=10,file
size=100M
[2023-09-26 07:59:12,204] INFO Reading configuration from: config/zookeepe
r.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2023-09-26 07:59:12,206] WARN config/zookeeper.properties is relative. Pr
epend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum
.QuorumPeerConfig)
[2023-09-26 07:59:12,214] INFO clientPortAddress is 0.0.0.0:2181 (org.apac
he.zookeeper.server.quorum.QuorumPeerConfig)
[2023-09-26 07:59:12,214] INFO secureClientPort is not set (org.apache.zoo
keeper.server.quorum.QuorumPeerConfig)
[2023-09-26 07:59:12,216] INFO autopurge.snapRetainCount set to 3 (org.apa
che.zookeeper.server.DatadirCleanupManager)
[2023-09-26 07:59:12,216] INFO autopurge.purgeInterval set to 0 (org.apach
e.zookeeper.server.DatadirCleanupManager)
[2023-09-26 07:59:12,216] INFO Purge task is not scheduled. (org.apache.zo
okeeper.server.DatadirCleanupManager)
[2023-09-26 07:59:12,216] WARN Either no config or no quorum defined in co
nfig, running  in standalone mode (org.apache.zookeeper.server.quorum.Quor
umPeerMain)
[2023-09-26 07:59:12,220] INFO Log4j 1.2 jmx support found and enabled. (o
rg.apache.zookeeper.jmx.ManagedUtil)
```

You can be sure it has started when you see an output like this:



```
[2021-08-24 13:19:41,253] INFO Server environment:os.memory.max=512MB (org.apache.zookeeper.server.ZooKeeperServer)
[2021-08-24 13:19:41,253] INFO Server environment:os.memory.total=512MB (org.apache.zookeeper.server.ZooKeeperServer)
[2021-08-24 13:19:41,256] INFO minSessionTimeout set to 6000 (org.apache.zookeeper.server.ZooKeeperServer)
[2021-08-24 13:19:41,256] INFO maxSessionTimeout set to 60000 (org.apache.zookeeper.server.ZooKeeperServer)
[2021-08-24 13:19:41,257] INFO Created server with tickTime 3000 minSessionTimeout 6000 maxSessionTimeout 60000 datadir /tmp/
zookeeper/version-2 snapdir /tmp/zookeeper/version-2 (org.apache.zookeeper.server.ZooKeeperServer)
[2021-08-24 13:19:41,275] INFO Using org.apache.zookeeper.server.NIOServerCnxnFactory as server connection factory (org.apach
e.zookeeper.server.ServerCnxnFactory)
[2021-08-24 13:19:41,281] INFO Configuring NIO connection handler with 10s sessionless connection timeout, 1 selector thread(
s), 4 worker threads, and 64 kB direct buffers. (org.apache.zookeeper.server.NIOServerCnxnFactory)
[2021-08-24 13:19:41,290] INFO binding to port 0.0.0.0/0.0.0.0:2181 (org.apache.zookeeper.server.NIOServerCnxnFactory)
[2021-08-24 13:19:41,314] INFO zookeeper.snapshotSizeFactor = 0.33 (org.apache.zookeeper.server.ZKDatabase)
[2021-08-24 13:19:41,321] INFO Snapshotting: 0x0 to /tmp/zookeeper/version-2/snapshot.0 (org.apache.zookeeper.server.persiste
nce.FileTxnSnapLog)
[2021-08-24 13:19:41,324] INFO Snapshotting: 0x0 to /tmp/zookeeper/version-2/snapshot.0 (org.apache.zookeeper.server.persiste
nce.FileTxnSnapLog)
[2021-08-24 13:19:41,345] INFO PrepRequestProcessor (sid:0) started, reconfigEnabled=false (org.apache.zookeeper.server.PrepR
equestProcessor)
[2021-08-24 13:19:41,355] INFO Using checkIntervalMs=60000 maxPerMinute=10000 (org.apache.zookeeper.server.ContainerManager)
```

> ZooKeeper, as of this version, is required for Kafka to work. ZooKeeper is responsible for the overall management of Kafka cluster. It monitors the Kafka brokers and notifies Kafka if any broker or partition goes down, or if a new broker or partition goes up.

Exercise 3: Start the Kafka broker service

1. Start a new terminal.

2. Change to the `kafka_2.12-3.5.1` directory.

```
cd kafka_2.12-3.5.1
```

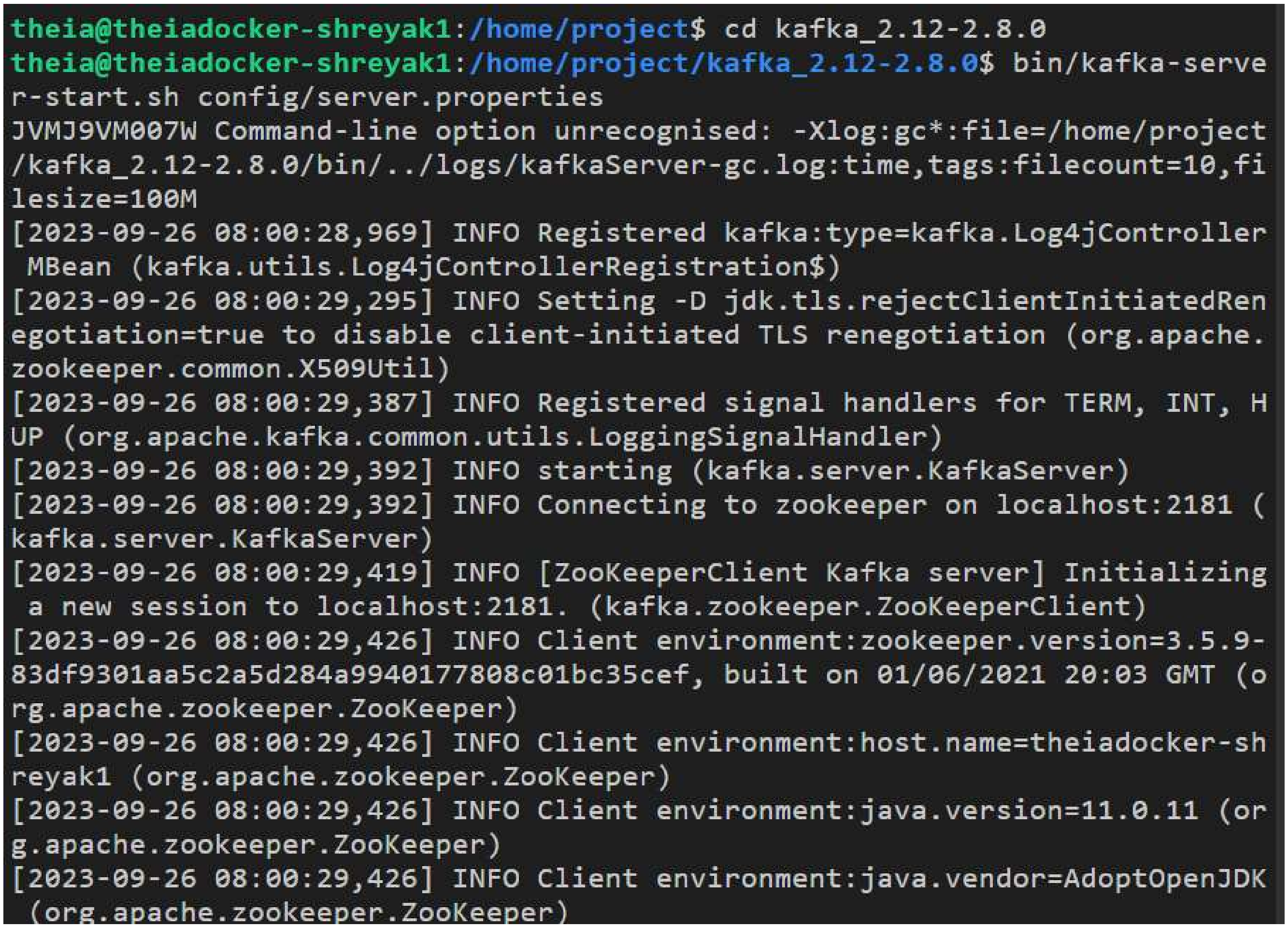
Copied! Executed!

3. Run the commands below. This will start the Kafka message broker service.

```
bin/kafka-server-start.sh config/server.properties
```

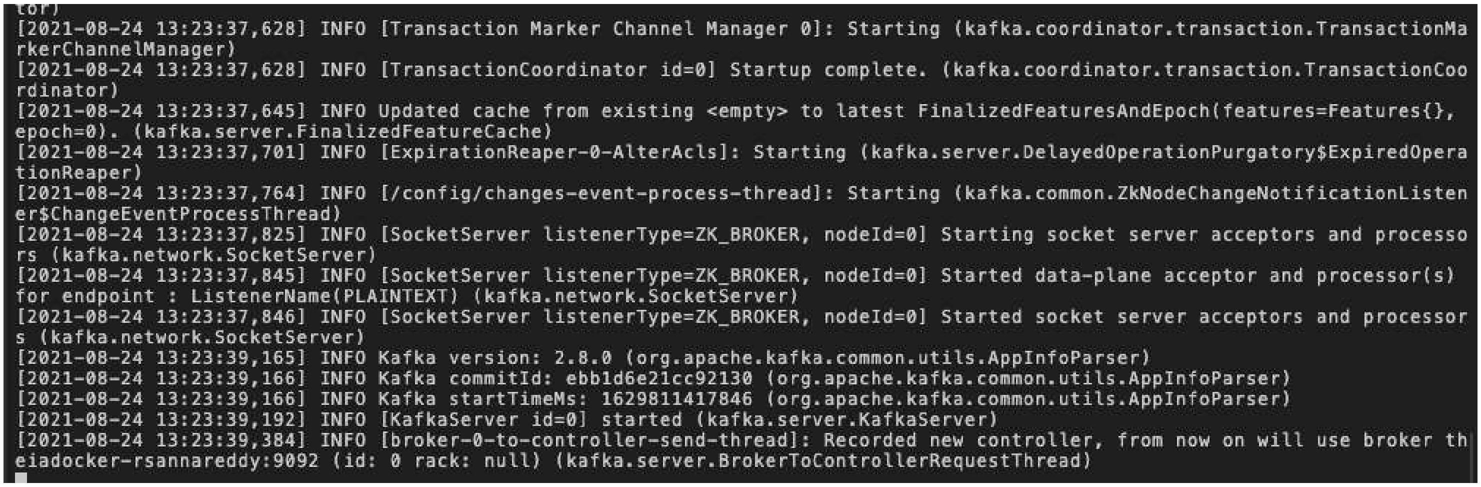
Copied! Executed!

When Kafka starts, you should see an output like this:



```
theia@theiadocker-shreyak1:/home/project$ cd kafka_2.12-2.8.0
theia@theiadocker-shreyak1:/home/project/kafka_2.12-2.8.0$ bin/kafka-serve
r-start.sh config/server.properties
JVMJ9VM007W Command-line option unrecognised: -Xlog:gc*:file=/home/project
/kafka_2.12-2.8.0/bin/../logs/kafkaServer-gc.log:time,tags:filecount=10,fi
lesize=100M
[2023-09-26 08:00:28,969] INFO Registered kafka:type=kafka.Log4jController
 MBean (kafka.utils.Log4jControllerRegistration$)
[2023-09-26 08:00:29,295] INFO Setting -D jdk.tls.rejectClientInitiatedRen
egotiation=true to disable client-initiated TLS renegotiation (org.apache.
zookeeper.common.X509Util)
[2023-09-26 08:00:29,387] INFO Registered signal handlers for TERM, INT, H
UP (org.apache.kafka.common.utils.LoggingSignalHandler)
[2023-09-26 08:00:29,392] INFO starting (kafka.server.KafkaServer)
[2023-09-26 08:00:29,392] INFO Connecting to zookeeper on localhost:2181 (
kafka.server.KafkaServer)
[2023-09-26 08:00:29,419] INFO [ZooKeeperClient Kafka server] Initializing
 a new session to localhost:2181. (kafka.zookeeper.ZooKeeperClient)
[2023-09-26 08:00:29,426] INFO Client environment:zookeeper.version=3.5.9-
83df9301aa5c2a5d284a9940177808c01bc35cef, built on 01/06/2021 20:03 GMT (o
rg.apache.zookeeper.ZooKeeper)
[2023-09-26 08:00:29,426] INFO Client environment:host.name=theiadocker-sh
reyak1 (org.apache.zookeeper.ZooKeeper)
[2023-09-26 08:00:29,426] INFO Client environment:java.version=11.0.11 (or
g.apache.zookeeper.ZooKeeper)
[2023-09-26 08:00:29,426] INFO Client environment:java.vendor=AdoptOpenJDK
 (org.apache.zookeeper.ZooKeeper)
```

You can be sure it has started when you see an output like this:



```
tor)
[2021-08-24 13:23:37,628] INFO [Transaction Marker Channel Manager 0]: Starting (kafka.coordinator.transaction.TransactionMa
rkerChannelManager)
[2021-08-24 13:23:37,628] INFO [TransactionCoordinator id=0] Startup complete. (kafka.coordinator.transaction.TransactionCoo
rdinator)
[2021-08-24 13:23:37,645] INFO Updated cache from existing <empty> to latest FinalizedFeaturesAndEpoch(features=Features{},
epoch=0). (kafka.server.FinalizedFeatureCache)
[2021-08-24 13:23:37,701] INFO [ExpirationReaper-0-AlterAcls]: Starting (kafka.server.DelayedOperationPurgatory$ExpiredOpera
tionReaper)
[2021-08-24 13:23:37,764] INFO [/config/changes-event-process-thread]: Starting (kafka.common.ZkNodeChangeNotificationListen
er$ChangeEventProcessThread)
[2021-08-24 13:23:37,825] INFO [SocketServer listenerType=ZK_BROKER, nodeId=0] Starting socket server acceptors and processo
rs (kafka.network.SocketServer)
[2021-08-24 13:23:37,845] INFO [SocketServer listenerType=ZK_BROKER, nodeId=0] Started data-plane acceptor and processor(s)
for endpoint : ListenerName(PLAINTEXT) (kafka.network.SocketServer)
[2021-08-24 13:23:37,846] INFO [SocketServer listenerType=ZK_BROKER, nodeId=0] Started socket server acceptors and processor
s (kafka.network.SocketServer)
[2021-08-24 13:23:39,165] INFO Kafka version: 2.8.0 (org.apache.kafka.common.utils.AppInfoParser)
[2021-08-24 13:23:39,166] INFO Kafka commitId: ebb1d6e21cc92130 (org.apache.kafka.common.utils.AppInfoParser)
[2021-08-24 13:23:39,166] INFO Kafka startTimeMs: 1629811417846 (org.apache.kafka.common.utils.AppInfoParser)
[2021-08-24 13:23:39,192] INFO [KafkaServer id=0] started (kafka.server.KafkaServer)
[2021-08-24 13:23:39,384] INFO [broker-0-to-controller-send-thread]: Recorded new controller, from now on will use broker th
eiadocker-rsannareddy:9092 (id: 0 rack: null) (kafka.server.BrokerToControllerRequestThread)
```

Exercise 4: Create a topic in the Admin.py file

Install kafka-python

1. Open a new terminal and change to the `kafka_2.12-3.5.1` directory.

```
cd kafka_2.12-3.5.1
```

Copied! Executed!

2. Install the `kafka-python` package by running the following command.

```
pip3 install kafka-python
```

Copied!

3. Create a new file named `admin.py` by running the following command.

```
touch admin.py
```

Copied! Executed!

4. Click the button below to open the file in edit mode and paste the following content in the file and save it.

Open **admin.py** in IDE

```
from kafka.admin import KafkaAdminClient,NewTopic
admin_client = KafkaAdminClient(bootstrap_servers="localhost:9092", client_id='test')
topic_list = []
new_topic = NewTopic(name="bankbranch", num_partitions= 2, replication_factor=1)
topic_list.append(new_topic)
admin_client.create_topics(new_topics=topic_list)
```

Copied!

We are creating a topic "bankbranch" through this code.

Exercise 5: Create Producer.py file

You need a producer to send messages to Kafka. You will find the code for the producer in the producer.py file.

1. Create a new file named `producer.py` by running the following command.

```
touch producer.py
```

Copied! Executed!

2. Click the button below to open the file in edit mode and paste the following content in the file and save it.

Open **producer.py** in IDE

```
from kafka import KafkaProducer
import json
producer = KafkaProducer(value_serializer=lambda v: json.dumps(v).encode('utf-8'))
producer.send("bankbranch", {'atmid':1, 'transid':100})
producer.send("bankbranch", {'atmid':2, 'transid':101})

producer.flush()

producer.close()
```

Copied!

The producer is sending across two messages through this code. These messages will be received by the consumer.

Exercise 6: Create Consumer.py file

You need a consumer to read messages from Kafka. The code for consumer will be written in consumer.py file.

1. Create a new file named `consumer.py` by running the following command.

```
touch consumer.py
```

Copied! Executed!

2. Click the button below to open the file in edit mode and paste the following content in the file and save it.

Open **consumer.py** in IDE

```
from kafka import KafkaConsumer
consumer = KafkaConsumer('bankbranch',
                        group_id=None,
                         bootstrap_servers=['localhost:9092'],
                         auto_offset_reset = 'earliest')
print("Hello")
print(consumer)

for msg in consumer:
    print(msg.value.decode("utf-8"))
```

Copied!

Exercise 7: Execute the three Python files

1. Execute admin.py and producer.py by executing the following commands in terminal:

```
python3 admin.py
python3 producer.py
```

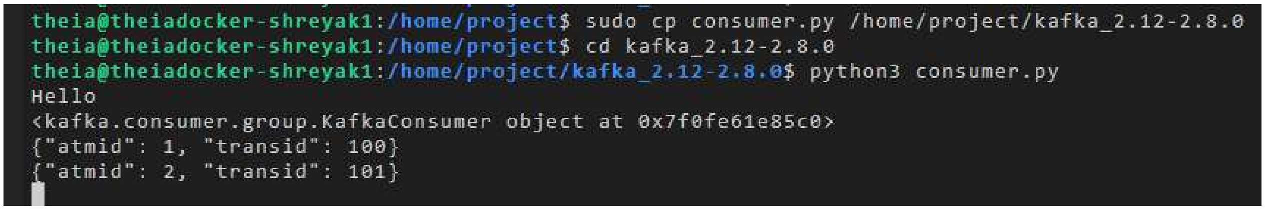
Copied! Executed!

2. Open a new terminal and execute the following commands to run consumer.py:

```
cd kafka_2.12-3.5.1
python3 consumer.py
```

Copied! Executed!

Your consumer should print the messages sent by the producer as follows:



```
theia@theiadocker-shreyak1:/home/project$ sudo cp consumer.py /home/project/kafka_2.12-2.8.0
theia@theiadocker-shreyak1:/home/project$ cd kafka_2.12-2.8.0
theia@theiadocker-shreyak1:/home/project/kafka_2.12-2.8.0$ python3 consumer.py
Hello
<kafka.consumer.group.KafkaConsumer object at 0x7f0fe61e85c0>
{"atmid": 1, "transid": 100}
{"atmid": 2, "transid": 101}
```

Congratulations, you have completed this lab!

Authors

Ramesh Sana Reddy
Shreya Khurana

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2023-10-10	0.2	Steve Hord	QA pass with edits
2023-09-28	0.1	Shreya Khurana	Created initial version of the lab