**Lab 11: Working with Kafka Connect JDBC Sink Connector**

**Introduction**

Apache Kafka is one of the core technologies in Enterprise Architectures these days. More than 60% of Fortune 500 companies are using Kafka. The technology has evolved a lot over the last decade i.e. from Pub/Sub to a Complete Event Streaming Platform. The very first requirement to working in an event-driven system is to ingest data/events. For the same purpose, Kafka Connect was added to the Kafka Ecosystem in 2015. This enables us to integrate external systems without writing a single line of code (if the connector already exists).

**Let’s get Started**

Execute the below command to start confluent services:

|  |
| --- |
| confluent local services start |



Now create a topic named as **orders**:

|  |
| --- |
| kafka-topics --create --bootstrap-server localhost:9092 --replication-factor 1 --partitions 1 --topic orders |

After this, produce a test Avro data to the **orders** topic in Kafka:

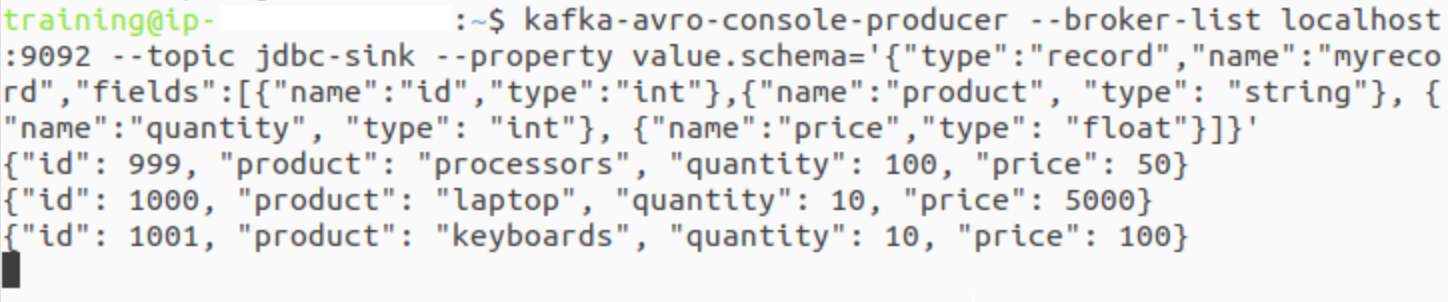
|  |
| --- |
| kafka-avro-console-producer --broker-list localhost:9092 --topic orders --property value.schema='{"type":"record","name":"myrecord","fields":[{"name":"id","type":"int"},{"name":"product", "type": "string"}, {"name":"quantity", "type": "int"}, {"name":"price","type": "float"}]}' |

Paste the following values:

*{"id": 999, "product": "processors", "quantity": 100, "price": 50}*

*{"id": 1000, "product": "laptop", "quantity": 10, "price": 5000}*

*{"id": 1001, "product": "keyboards", "quantity": 10, "price": 100}*

**

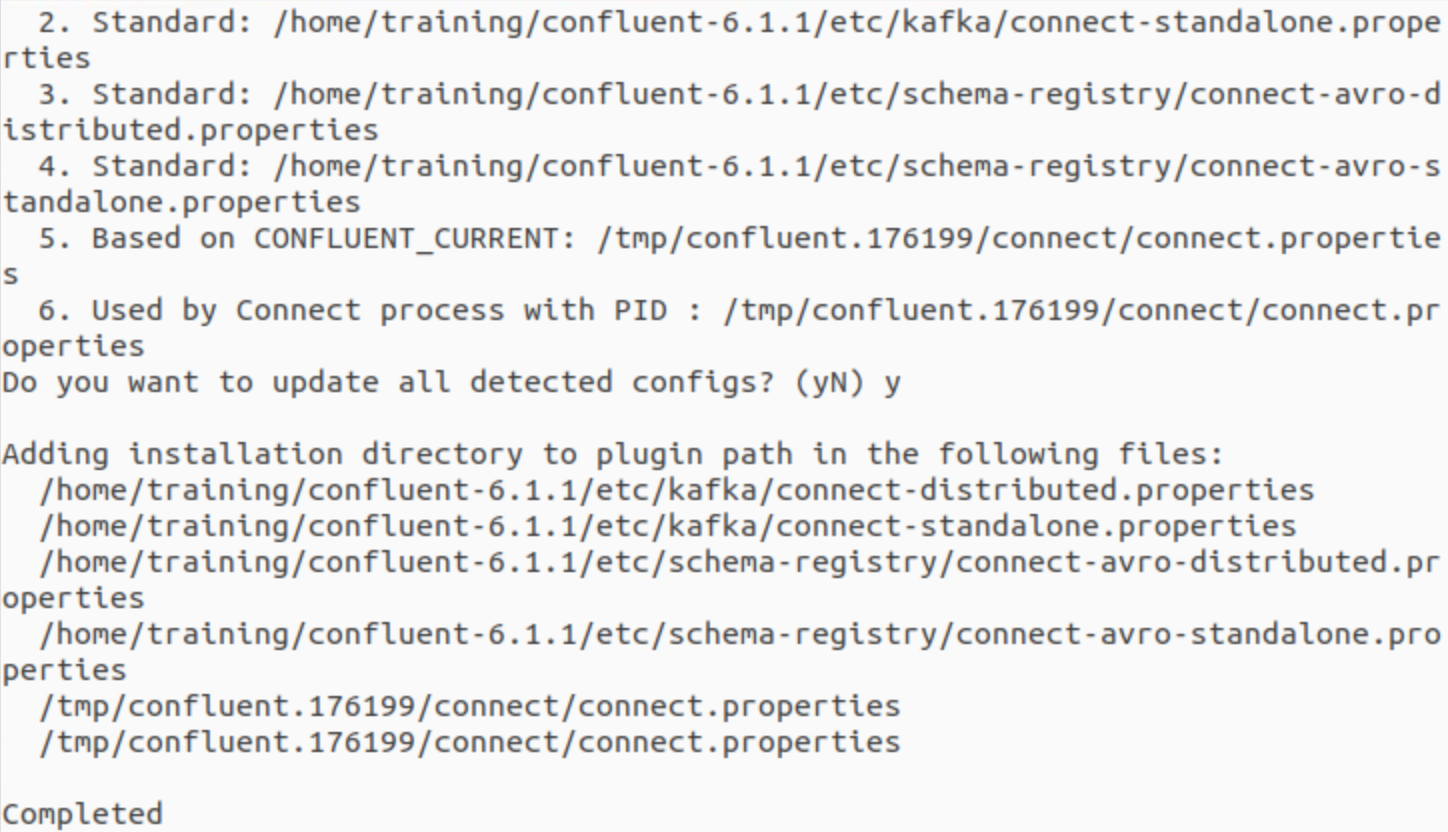
Open a new terminal and install the hdfs-sink connector:

|  |
| --- |
| confluent-hub install confluentinc/kafka-connect-jdbc:latest |

It is going to ask for a lot of options, do the following:

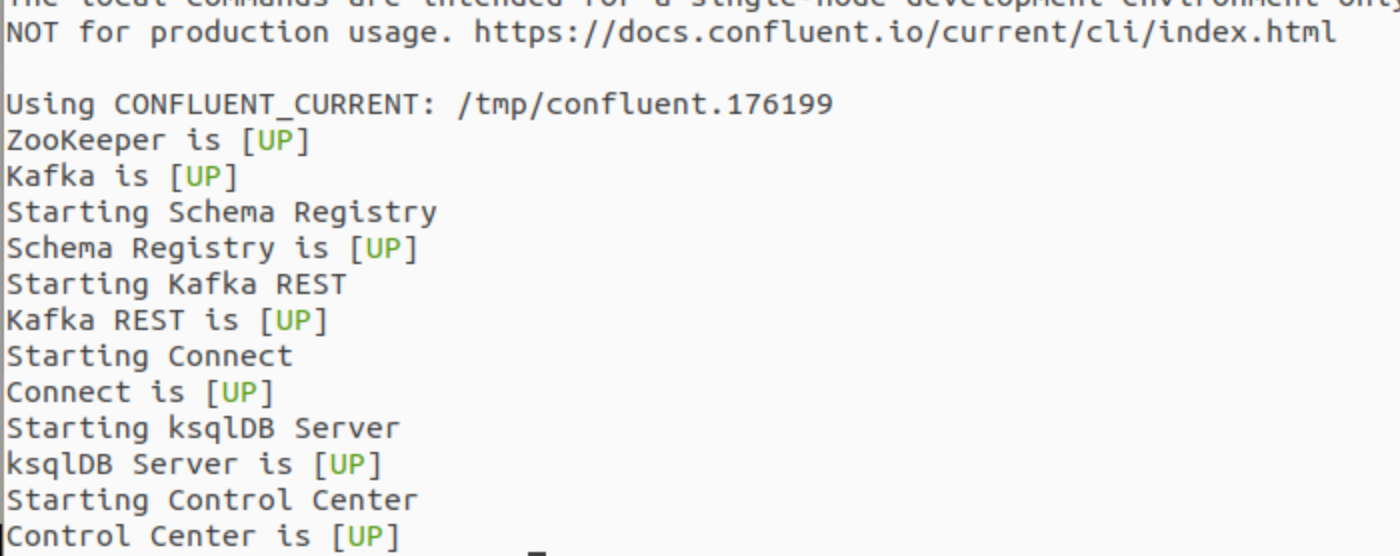
1. Press 1, then

2. Press Y for each other option



Now restart the Confluent platform using the following commands:

|  |
| --- |
| confluent local services stop  confluent local services start |

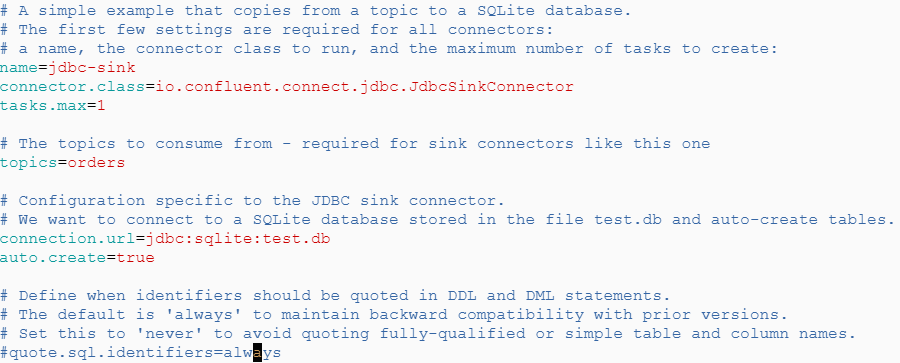


Open and run following:

***cd confluent-7.7.1/share/confluent-hub-components/confluentinc-kafka-connect-jdbc/etc/***

***sudo vi sink-quickstart-sqlite.properties***

and make sure you are changing the topic ***name*** to ***jdbc-sink*** and also having the same properties as follow:

******

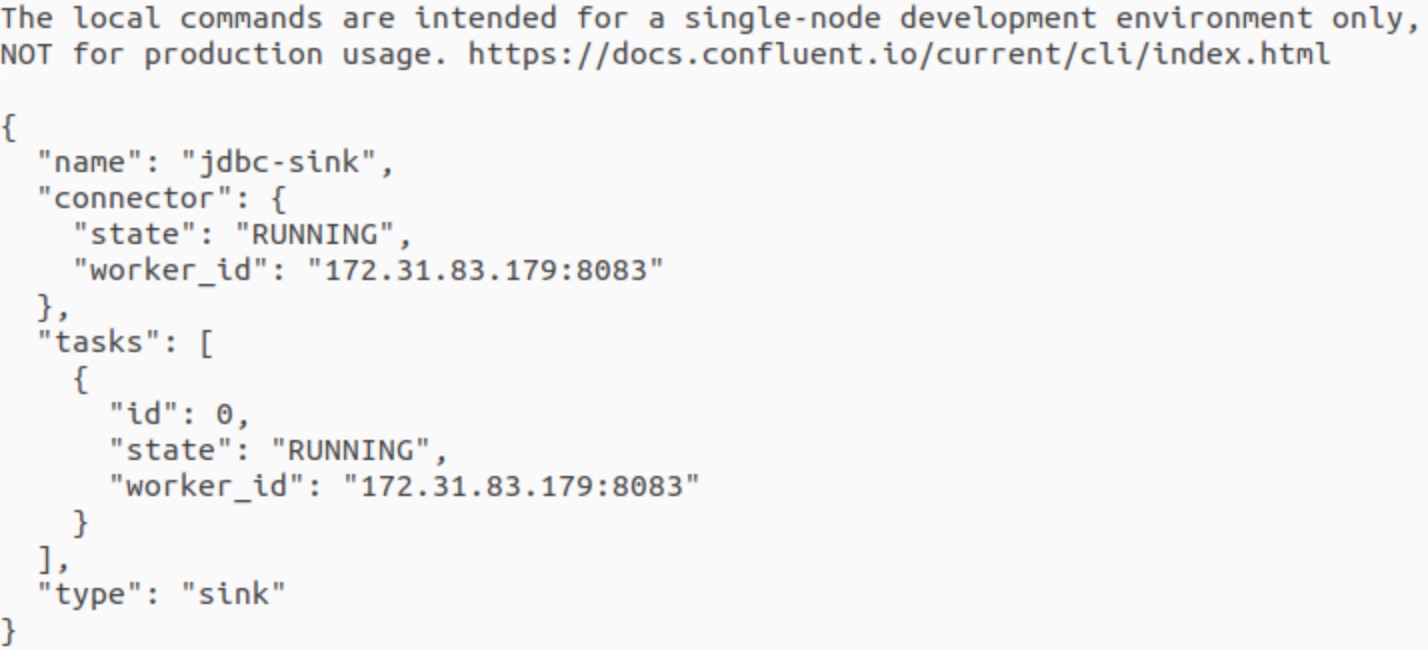
Let’s load the jdbc-sink connector:

|  |
| --- |
| confluent local services connect connector load jdbc-sink --config share/confluent-hub-components/confluentinc-kafka-connect-jdbc/etc/sink-quickstart-sqlite.properties |



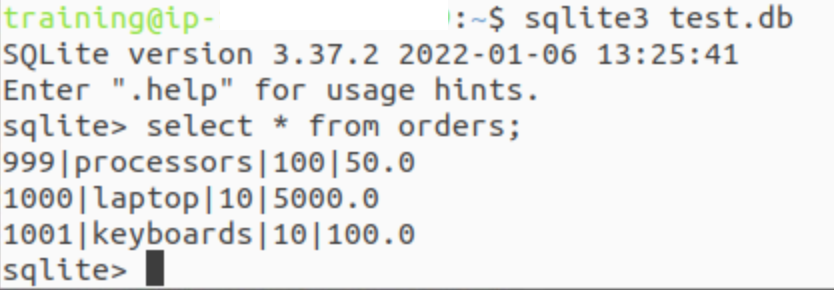
Open a new terminal, you can confirm that the connector is in a **RUNNING** state using following command:

|  |
| --- |
| confluent local services connect connector status jdbc-sink |



After this let’s validate that the avro data is in sqlite3 or not:

|  |
| --- |
| > sqlite3 test.db > select \* from orders; |



**Voila!!** We have successfully completed this Exercise.