KAFKA-CONNECTORS

Add Plugins paths :

Add the plugin paths for FileStreamSource and Sink Connectors

and JdbcSourceConnector and SinkConnectors classes

```
nano etc/kafka/connect-distributed.properties
```

Add the following lines

```
listeners=http://0.0.0.0:8083
rest.advertised.host.name=localhost
rest.advertised.port=8083
plugin.path=/usr/share/java,/usr/share/filestream-connectors,/
usr/local/share/kafka/plugins,/usr/local/share/kafka/plugins/con
fluentinc-kafka-connect-jdbc-10.8.0
```

I. FileStreamSourceConnector :

<u>Create topic for stream messages from external source file to the topic</u>

```
        Kafka-topics --create --bootstrap-server
localhost:9092 --topic filesource –replication-factor 1
--partiotions 1
```

Create input.txt file for as source file and write some messages to input.txt

Create a source connector to import data into Kafka from a file.

```
curl -X POST http://localhost:8083/connectors \
-H 'Accept: */*' \
-H 'Content-Type: application/json' \
-d '{
"name": "file_source_connector",
"config": {
"connector.class":
"org.apache.kafka.connect.file.FileStreamSourceConnector",
"topic": "filesource",
"file": "/home/arunmg/input.txt",
"value.converter":
"org.apache.kafka.connect.storage.StringConverter"
}
}'
```

To check the connector is created or not

```
curl -X GET
http://localhost:8083/connectors/file_source_connector
```

To check the status of connector created

```
curl -X GET
http://localhost:8083/connectors/file_source_connector/status
```

To check the Topic consume messages from topic

```
kafka-console-consumer --bootstrap-server
localhost:9092 --topic filesource --from-beginning
```

II. FileStreamSinkConnector :

Create topic for stream messages To external sink file from the topic

```
        Kafka-topics --create --bootstrap-server
localhost:9092 --topic filesink --replication-factor 1
--partiotions 1
```

Create output.txt file for as sink file and stream some messages to output.txt using sink connector

Create a sink connector to export data from Kafka to a file.

```
curl -X POST http://localhost:8083/connectors \
-H 'Accept: */*' \
-H 'Content-Type: application/json' \
-d '{
"name": "file_sink_connector",
"config": {
"connector.class":
"org.apache.kafka.connect.file.FileStreamSinkConnect
or",
"topics": "filesink",
"file":
"/home/mpshriveena/Desktop/Platformatory/trials/output.txt",
"value.converter":
"org.apache.kafka.connect.storage.StringConverter"
}
}'
```

Check the contents of the output file.
cat /home/arunmg/output.txt

```
vi output.txt
```

To check the connector is created or not

```
curl -X GET
http://localhost:8083/connectors/file_source_connector
```

To check the status of connector created

```
curl -X GET
http://localhost:8083/connectors/file_source_connector/status
```

III. JdbcSourceConnector :

Source database : postgresql
Topic : postgres-jdbc
Table : employees

Create json application to connectors

```
curl -X POST http://localhost:8083/connectors -H
"Content-Type: application/json" -d '{
  "name": "jdbc_postgresql_source2",
  "config": {
    "connector.class":
"io.confluent.connect.jdbc.JdbcSourceConnector",
    "tasks.max": "1",
    "connection.url":
"jdbc:postgresql://127.0.0.1:5432/testdb",
    "connection.user": "Arun",
    "connection.password": "Arun123",
    "table.whitelist": "employees",
    "topic.prefix": "postgres-jdbc",
    "mode":"incrementing",
    "incrementing.column.name": "id",
    "poll.interval.ms":  "10000",
  }
}'
```

Login into the database psql :

```
        sudo -i -u  postgres
        Psql
        create user kiran with password 'Arun123';
        grant all privileges on database testdb to Arun;
```

Create table employees and insert some values into employees

```
Create table employees(id serial PRIVATE KEY, name
varchar(255) ,department varchar(55), salary int );
```

```
INSERT INTO employees  (name, department, salary)
VALUES (‘Arun’, ’IT’ ,40000);
```

Consume messages from the topic :

```
kafka-console-consumer --bootstrap-server
localhost:9092 --topic postgres-jdbc --from-beginning
```

IV. JdbcSinkConnector :

sink database : mydatabse
Topic : mysql-jdbc
Table : employees

Create json application to connectors

```
curl -X POST http://localhost:8083/connectors -H
"Content-Type: application/json" -d '{
  "name": "jdbc_mysqlserver_sink",
  "config": {
    "connector.class":
"io.confluent.connect.jdbc.JdbcSinkConnector",
   "tasks.max": "1",
   "topics": "mysql-jdbc",
   "connection.url":
"jdbc:mysql://127.0.0.1:3306/mydatabse",
   "connection.user": "root",
   "connection.password": "root_passwd",
   "auto.create": "true",
   "insert.mode": "insert",
   "pk.mode": "none",
   "fields.whitelist": "name,department,salary",
   "batch.size": "10",
   "value.converter":
"org.apache.kafka.connect.json.JsonConverter",
   "key.converter":
"org.apache.kafka.connect.json.JsonConverter",
   "value.convertor.schemas.enable": "false",
   "key.convertor.schemas.enable": "false",
   "table.name.format": "employees"
  }
}'
```

Login into the database psql :

```
sudo -i -u  mysql
```

During installing the mysql server in linux root user already created with passwd

Create table employees and insert some values into employees

Create table employees(id serial PRIVATE KEY, name varchar(255) ,department varchar(55), salary int);ck the

Query the table employees :

SELECT * FROM EMPLOYEES ;