

# Data Engineer – Data Fellowship 12 IYKRA

## Assignment 2

Alya Mutiara Firdausyi

---

### Instruction

#### Task

Hey, data rangers! Our analytics team needs data related to our Supermarket sales for our **monthly reporting project**. Data is in CSV format which can be accessed here, [Supermarket sales](#). The business team also needs you to **generate any insights** we can get from those data. You'll collaborate with a data visualization expert who will create a dashboard for the business team. You need to **provide summary tables** for data visualization experts, such as monthly revenue, etc.

As per governance, you have to **ingest data to the data warehouse in a real-time manner**, then summary tables can be generated with either real-time or batch pipelines depending on the business needs.

#### Output

You'll need to provide screenshots as the output,

- Apache Kafka, topic list, and topic details.
  - Apache NiFi, connectors, relationships, and its configurations.
  - Informatica, mapping configs, map task configs, task flow configs, etc
- 

#### Goal:

Build a real-time data pipeline to ingest supermarket sales data, generate insights, and populate summary tables for data visualization.

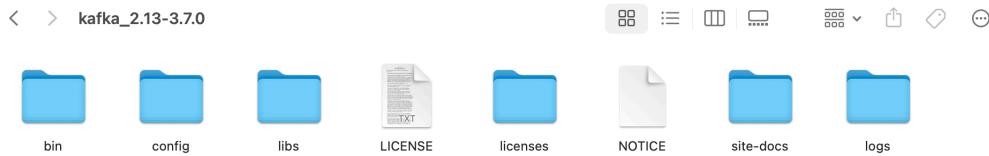
#### Tools:

- **Apache Kafka:** This will act as the central hub for ingesting the real-time sales data stream.
- **NiFi:** To create a data flow that reads data from a Kafka topic (a category for data streams), transforms the data (e.g., cleanses or filters it), and then routes it to the data warehouse.

- **Informatica:** This commercial platform offers a more comprehensive set of tools for data integration, including pre-built connectors for Kafka and data warehouses. We'd use Informatica's visual interface to build the data pipeline.
- **Data Warehouse:** This is the final destination for the processed data. It could be a cloud-based data warehouse like Google BigQuery.

**Step:**

1. Data Ingestion: setting up Kafka topic to receive the real-time sales data stream by continuously reading the CSV file at regular intervals.
  - a. Ensure that Kafka is already downloaded to the local computer. Then open a new blank terminal and locate the directory to the Kafka folder.



- b. Start the ZooKeeper Server in a new terminal.

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

```
[2024-04-21 14:56:36,772] INFO Using org.apache.zookeeper.server.NIOServerCnxnFactory
[2024-04-21 14:56:36,773] WARN maxCnxns is not configured, using default value 0. (org.apache.zookeeper.server.ServerCnxnFactory)
[2024-04-21 14:56:36,773] INFO Configuring NIO connection handler with 10s sessionless connection timeout, 2 selector thread(s), 16 worker threads, and 64 kB direct buffers. (org.apache.zookeeper.server.NIOServerCnxnFactory)
[2024-04-21 14:56:36,779] INFO binding to port 0.0.0.0/0.0.0.2181 (org.apache.zookeeper.server.NIOServerCnxnFactory)
[2024-04-21 14:56:36,785] INFO Using org.apache.zookeeper.server.watch.WatchManager as watch manager (org.apache.zookeeper.server.watch.WatchManagerFactory)
[2024-04-21 14:56:36,785] INFO Using org.apache.zookeeper.server.watch.WatchManager as watch manager (org.apache.zookeeper.server.watch.WatchManagerFactory)
[2024-04-21 14:56:36,785] INFO zookeeper.snapshotSizeFactor = 0.33 (org.apache.zookeeper.server.ZKDatabase)
[2024-04-21 14:56:36,786] INFO zookeeper.commitLogCount=500 (org.apache.zookeeper.server.ZKDatabase)
[2024-04-21 14:56:36,787] INFO zookeeper.snapshot.compression.method = CHECKED (org.apache.zookeeper.server.persistence.SnapStream)
[2024-04-21 14:56:36,789] INFO Reading snapshot /tmp/zookeeper/version-2/snapshot.c1 (org.apache.zookeeper.server.persistence.FileSnap)
[2024-04-21 14:56:36,793] INFO The digest in the snapshot has digest version of 2, with zxid as 0xcl, and digest value as 317007898581 (org.apache.zookeeper.server.DataTree)
[2024-04-21 14:56:36,806] INFO ZooKeeper audit is disabled. (org.apache.zookeeper.audit.ZAuditProvider)
[2024-04-21 14:56:36,887] INFO 28 txns loaded in 11 ms (org.apache.zookeeper.server.persistence.FileTxnSnapLog)
[2024-04-21 14:56:36,887] INFO Snapshot loaded in 22 ms, highest zxid is 0xd5, digest is 321733836589 (org.apache.zookeeper.server.ZKDatabase)
[2024-04-21 14:56:36,888] INFO Snapshotting: 0xd5 to /tmp/zookeeper/version-2/snapshot.d5 (org.apache.zookeeper.server.persistence.FileTxnSnapLog)
[2024-04-21 14:56:36,888] INFO Snapshot taken in 3 ms (org.apache.zookeeper.server.ZooKeeperServer)
[2024-04-21 14:56:36,816] INFO PrepRequestProcessor (sid:0) started, reconfigEnabled=false (org.apache.zookeeper.server.PrepRequestProcessor)
[2024-04-21 14:56:36,816] INFO zookeeper.request_throttler.shutdownTimeout = 10000 ms (org.apache.zookeeper.server.RequestThrottler)
[2024-04-21 14:56:36,829] INFO Using checkIntervalMs=60000 maxPerMinute=100000 maxNeverUsedIntervalMs=0 (org.apache.zookeeper.server.ContainerManager)
```

- c. Start the Kafka Server in another new terminal.

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



A terminal window titled "kafka\_2.13-3.7.0 — zsh" showing the output of the command "bin/kafka-server-start.sh config/server.properties". The log includes several INFO messages from the Kafka coordinator and group metadata manager, indicating the loading of offsets and group metadata for consumer groups like "console-consumer" and "supermarketSales". The log shows various epoch numbers and offset ranges being processed.

- d. In another new terminal window, create **supermarketSales** topics.

```
bin/kafka-topics.sh --bootstrap-server localhost:9092 --topic supermarketSales --create --partitions 3 --replication-factor 1
```



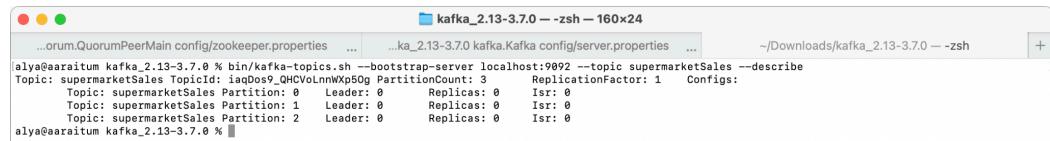
A terminal window titled "kafka\_2.13-3.7.0 — zsh" showing the output of the command "bin/kafka-topics.sh --bootstrap-server localhost:9092 --topic supermarketSales --create --partitions 3 --replication-factor 1". The log shows the creation of a new topic named "supermarketSales" with 3 partitions and a replication factor of 1. It also lists other topics in the cluster: "companies", "students", and "supersmarketSales".

Then check whether the topic exists or not.

```
bin/kafka-topics.sh --bootstrap-server localhost:9092 --list
```

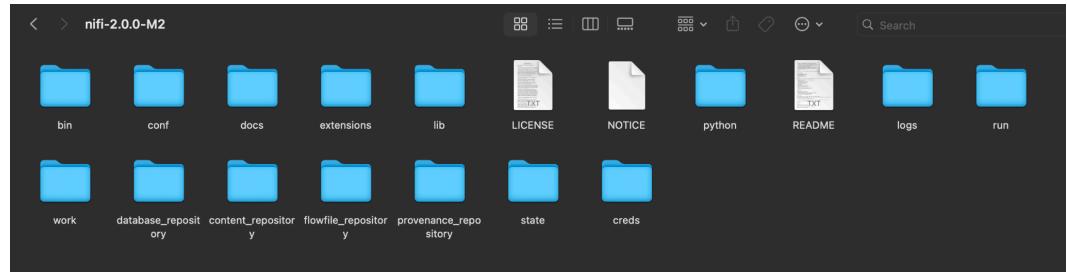
- e. Describe the supermarketSales topic.

```
bin/kafka-topics.sh --bootstrap-server localhost:9092 --topic supermarketSales --describe
```



A terminal window titled "kafka\_2.13-3.7.0 — zsh" showing the output of the command "bin/kafka-topics.sh --bootstrap-server localhost:9092 --topic supermarketSales --describe". The log provides detailed information about the topic, including its ID (1aqD9s9\_QHQV0LnnWkp5Qg), partition count (3), replication factor (1), and leader and replica details for each partition.

- f. Download the Supermarket Sales data from the link provided and change the format into a JSON.
2. Data Processing: use NiFi and Informatica to build a data pipeline that reads data from the Kafka topic, then cleanses, transforms, and aggregates the data to generate summary tables for monthly revenue.
- Prepare the NiFi server by ensuring to download the binary from the official NiFi website, then open a new blank terminal and locate the NiFi directory.



- b. Run the NiFi.

```
bin/nifi.sh start --wait-for-init
```

```
nifi-2.0.0-M2 — zsh — 82x24
[alya@aaraitum nifi-2.0.0-M2 % bin/nifi.sh start --wait-for-init
nifi.sh: JAVA_HOME not set; results may vary

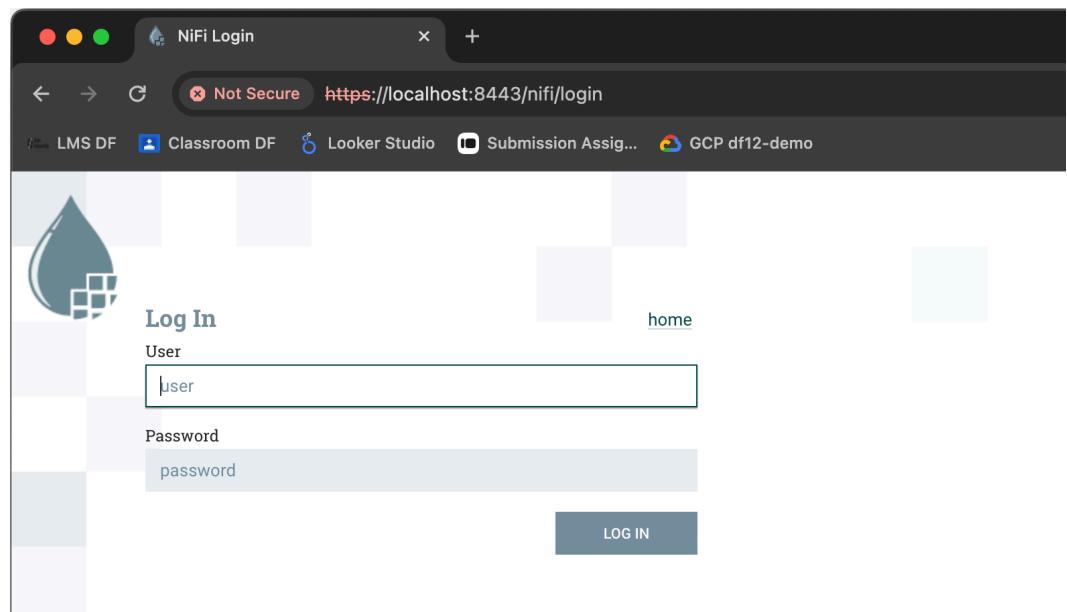
Java home:
NiFi home: /Users/alya/Downloads/nifi-2.0.0-M2

Bootstrap Config File: /Users/alya/Downloads/nifi-2.0.0-M2/conf/bootstrap.conf

NiFi has not fully initialized yet...
NiFi initialized.
Exiting startup script...

[alya@aaraitum nifi-2.0.0-M2 %]
```

- c. Open <https://localhost:8443/nifi>. Ensure to use HTTPS as NiFi is using a secure connection.



- d. Login using the generated credential in the logs/nifi-app.log, then search for the "Generated Username" keyword.

```

org.apache.nifi.web.security.csrf.CsrfCookieFilter@66b545a5,
org.apache.nifi.web.security.x509.X509AuthenticationFilter@a139470,
org.springframework.security.oauth2.server.resource.web.authentication.BearerTokenAuthenticationFilter@7de2e3a1,
org.springframework.security.web.authentication.AnonymousAuthenticationFilter@1a8f0b1b,
org.apache.nifi.web.security.log.AuthenticationUserFilter@565bed82,
org.springframework.security.web.access.ExceptionTranslationFilter@13bd7c56,
org.springframework.security.web.access.intercept.AuthorizationFilter@187609cb]
2024-04-19 12:53:19,788 INFO [main] o.a.n.a.s.u.SingleUserLoginIdentityProvider

Generated Username [REDACTED]
Generated Password [REDACTED]

2024-04-19 12:53:19,789 INFO [main] o.a.n.a.s.u.SingleUserLoginIdentityProvider Run the
following command to change credentials: nifi.sh set-single-user-credentials USERNAME PASSWORD
2024-04-19 12:53:20,091 INFO [main] o.a.n.a.s.u.SingleUserLoginIdentityProvider Updating Login
Identity Providers Configuration [/conf/login-identity-providers.xml]
2024-04-19 12:53:20,262 INFO [main] o.a.n.w.c.ApplicationStartupContextListener Starting Flow
Controller...

```

- e. After successfully logging in, create a new processor at the top section of the page by dragging and dropping the Processor logo into the canvas. Then choose the ConsumeKafka.

Source	Type	Version	Tags
all groups	Type ▲		consume
<b>amazon attributes</b>	ConsumeGCPubSub	2.0.0-M2	gcp, google-cloud, google, cons...
<b>avro aws azure</b>	ConsumeGCPubSubLite	2.0.0-M2	gcp, google-cloud, google, cons...
<b>cloud csv</b>	ConsumeIMAP	2.0.0-M2	Imap, Email, Consume, Ingest, ...
<b>database fetch</b>	ConsumeJMS	2.0.0-M2	jms, receive, restricted, get, con...
<b>get google ingest</b>	ConsumeKafkaRecord_2_6	2.0.0-M2	PubSub, Consume, Ingest, Get, ...
<b>json listen logs</b>	ConsumeKafka_2_6	2.0.0-M2	PubSub, Consume, Ingest, Get, ...
<b>message</b>	ConsumeKinesisStream	2.0.0-M2	amazon, stream, consume, aws...
<b>microsoft put</b>	ConsumeMQTT	2.0.0-M2	MQTT, subscribe, consume, list...
<b>query record</b>	ConsumePOP3	2.0.0-M2	Email, Consume, Ingest, Messa...
<b>restricted source</b>	ConsumeSlack	2.0.0-M2	conversation.history, slack, uns...
<b>storage text</b>	ConsumeTwitter	2.0.0-M2	twitter, json, tweets, social medi...
<b>update</b>	ConsumeWindowsEventLog	2.0.0-M2	event. windows. inaest

**ConsumeKafka\_2\_6 2.0.0-M2** org.apache.nifi - nifi-kafka-2-6-nar  
Consumes messages from Apache Kafka specifically built against the Kafka 2.6 Consumer API. The complementary NiFi processor for sending messages is PublishKafka\_2\_6.

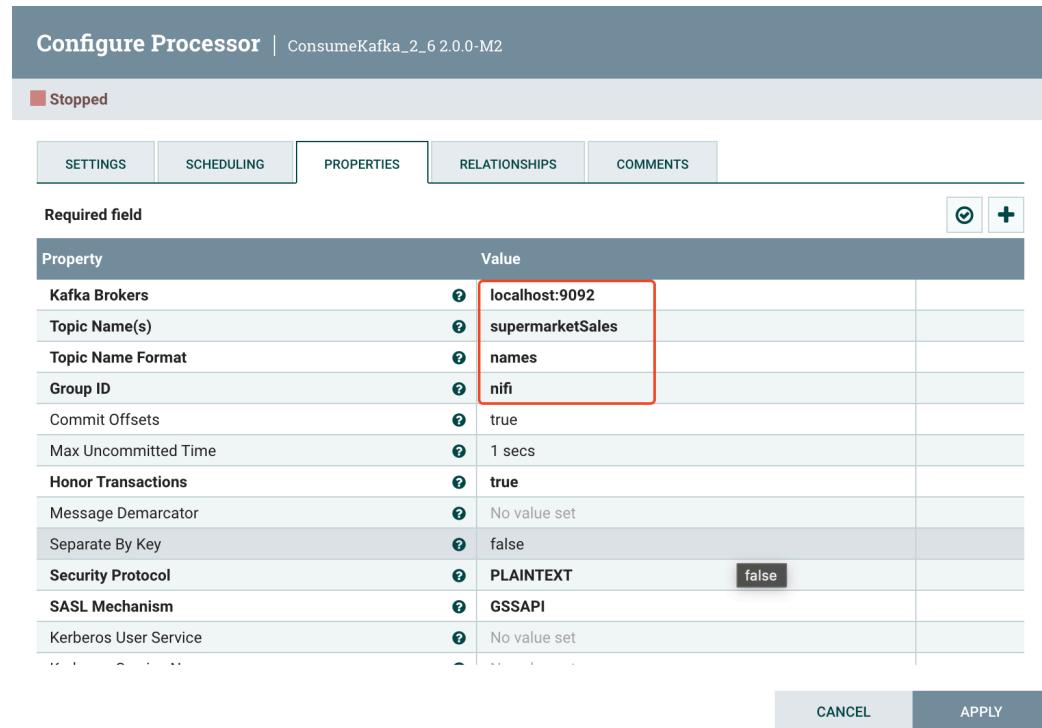
- f. Double-click on the ConsumeKafka processor, then go to the Properties tab. Change the value as below:

Kafka Brokers: localhost:9092

Topic Name(s) : supermarketSales

Topic Name Format: names

Group ID: nifi



The screenshot shows the NiFi 'Configure Processor' interface for a 'ConsumeKafka\_2\_6' processor. The top bar indicates the processor is 'Stopped'. Below are tabs for SETTINGS, SCHEDULING, PROPERTIES (selected), RELATIONSHIPS, and COMMENTS. The PROPERTIES tab displays the following configuration:

Property	Value
Kafka Brokers	localhost:9092
Topic Name(s)	supermarketSales
Topic Name Format	names
Group ID	nifi
Commit Offsets	true
Max Uncommitted Time	1 secs
Honor Transactions	true
Message Demarcator	No value set
Separate By Key	false
Security Protocol	PLAINTEXT
SASL Mechanism	GSSAPI
Kerberos User Service	No value set

At the bottom right are 'CANCEL' and 'APPLY' buttons.

- g. Add for another Processor, search for PutGCSObject which has a function to write data into a Cloud Storage bucket. Double-click the processor and go to the Properties tab. Change the value of Project ID to use the desired GCP Project ID, the Cloud Storage bucket that wants to be used in the Bucket property, and choose the GCPCredentialsControllerService in the GCP Credentials Provider Service property.

**Configure Processor | PutGCSObject 2.0.0-M2**

Stopped

SETTINGS	SCHEDULING	PROPERTIES	RELATIONSHIPS	COMMENTS																										
Required field																														
<table border="1"> <thead> <tr> <th>Property</th> <th>Value</th> </tr> </thead> <tbody> <tr> <td>Project ID</td> <td>df12-demo</td> </tr> <tr> <td>GCP Credentials Provider Service</td> <td>GCP Credentials Controller Service →</td> </tr> <tr> <td>Number of retries</td> <td>6</td> </tr> <tr> <td>Proxy host</td> <td>No value set</td> </tr> <tr> <td>Proxy port</td> <td>No value set</td> </tr> <tr> <td>HTTP Proxy Username</td> <td>No value set</td> </tr> <tr> <td>HTTP Proxy Password</td> <td>No value set</td> </tr> <tr> <td>Proxy Configuration Service</td> <td>No value set</td> </tr> <tr> <td>Storage API URL</td> <td>No value set</td> </tr> <tr> <td>Bucket</td> <td>df12-assignment2</td> </tr> <tr> <td>Key</td> <td> \${filename}</td> </tr> <tr> <td>Content Type</td> <td> \${mime.type}</td> </tr> </tbody> </table>					Property	Value	Project ID	df12-demo	GCP Credentials Provider Service	GCP Credentials Controller Service →	Number of retries	6	Proxy host	No value set	Proxy port	No value set	HTTP Proxy Username	No value set	HTTP Proxy Password	No value set	Proxy Configuration Service	No value set	Storage API URL	No value set	Bucket	df12-assignment2	Key	\${filename}	Content Type	\${mime.type}
Property	Value																													
Project ID	df12-demo																													
GCP Credentials Provider Service	GCP Credentials Controller Service →																													
Number of retries	6																													
Proxy host	No value set																													
Proxy port	No value set																													
HTTP Proxy Username	No value set																													
HTTP Proxy Password	No value set																													
Proxy Configuration Service	No value set																													
Storage API URL	No value set																													
Bucket	df12-assignment2																													
Key	\${filename}																													
Content Type	\${mime.type}																													
<span>CANCEL</span> <span>APPLY</span>																														

Configure the GCP Credentials Controller Service by clicking the arrow and going to the Controller Services. Then choose to configure the service and go to the Properties tab.

NiFi Flow Configuration

GENERAL	CONTROLLER SERVICES																									
<table border="1"> <thead> <tr> <th>Name</th> <th>Type</th> <th>Bundle</th> <th>State</th> <th>Scope</th> </tr> </thead> <tbody> <tr> <td>CSVReader</td> <td>CSVReader 2.0.0-M2</td> <td>org.apache.nifi - nifi-record-serialization...</td> <td>Enabled</td> <td>NiFi Flow</td> </tr> <tr> <td>CSVRecordSetWriter</td> <td>CSVRecordSetWriter 2.0.0-M2</td> <td>org.apache.nifi - nifi-record-serialization...</td> <td>Enabled</td> <td>NiFi Flow</td> </tr> <tr> <td>GCP Credentials Controller Service</td> <td>GCP Credentials Controller Service 2.0.0-...</td> <td>org.apache.nifi - nifi-gcp-nar</td> <td>Enabled</td> <td>NiFi Flow</td> </tr> <tr> <td>JsonTreeReader</td> <td>JsonTreeReader 2.0.0-M2</td> <td>org.apache.nifi - nifi-record-serialization...</td> <td>Enabled</td> <td>NiFi Flow</td> </tr> </tbody> </table>		Name	Type	Bundle	State	Scope	CSVReader	CSVReader 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow	CSVRecordSetWriter	CSVRecordSetWriter 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow	GCP Credentials Controller Service	GCP Credentials Controller Service 2.0.0-...	org.apache.nifi - nifi-gcp-nar	Enabled	NiFi Flow	JsonTreeReader	JsonTreeReader 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow
Name	Type	Bundle	State	Scope																						
CSVReader	CSVReader 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow																						
CSVRecordSetWriter	CSVRecordSetWriter 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow																						
GCP Credentials Controller Service	GCP Credentials Controller Service 2.0.0-...	org.apache.nifi - nifi-gcp-nar	Enabled	NiFi Flow																						
JsonTreeReader	JsonTreeReader 2.0.0-M2	org.apache.nifi - nifi-record-serialization...	Enabled	NiFi Flow																						

Fill the Service Account JSON File value with the GCP service account key in the JSON format inside the NiFi directory. Then click the lightning symbol to enable the service.

< > nifi-2.0.0-M2

Name
> bin
> conf
> content_repository
creds
df12-demo-263a0648a7f1.json ←
> database_repository
> docs
> extensions
> flowfile_repository

**Controller Service Details** | GCP Credentials Controller Service 2.0.0-M2

► ENABLED DISABLE & CONFIGURE

SETTINGS PROPERTIES COMMENTS

**Required field**

Property	Value
Use Application Default Credentials	false
Use Compute Engine Credentials	false
Service Account JSON File	./creds/df12-demo-263a0648a7f1.json
Service Account JSON	No value set
Proxy Configuration Service	No value set

- h. Create a Connection from ConsumeKafka to the PutGCSObject.

**Create Connection**

DETAILS SETTINGS

From Processor  
**ConsumeKafka\_2\_6**  
 ConsumeKafka\_2\_6

To Processor  
**PutGCSObject**  
 PutGCSObject

Within Group  
 NiFi Flow

Within Group  
 NiFi Flow

For Relationships  
 success

- i. Create a new dataset and table in the BigQuery for storing the supermarket sales dataset.

**Create table**

**Source**  
 Create table from  
 Empty table

**Destination**  
 Project \* df12-demo  
 Dataset \* lab  
 Table \* supermarketSales

Table type  
 Native table

**Schema**  
 Edit as text

```
Press Option+F1 for Accessibility Options.
1. InvoiceID:STRING,
2. Branch:STRING,
3. City:STRING,
4. CustomerType:STRING,
5. Gender:STRING,
6. ProductLine:STRING,
7. UnitPrice:FLOAT,
8. Quantity:NUMERIC,
9. TaxFivePercent:NUMERIC,
```

CREATE TABLE CANCEL

On the Schema section, change to Edit as text and put this into the box.

```

InvoiceID:STRING,
Branch:STRING,
City:STRING,
CustomerType:STRING,
Gender:STRING,
ProductLine:STRING,
UnitPrice:FLOAT,
Quantity:NUMERIC,
TaxFivePercent:NUMERIC,
Total:NUMERIC,
Date:STRING,
Time:STRING,
Payment:STRING,
cogs:NUMERIC,
GrossMarginPercentage:NUMERIC,
GrossIncome:NUMERIC,
Rating:NUMERIC

```

Ensure that the table is created.

The screenshot shows the Data Catalog interface with the following details:

- Search Bar:** Type to search
- Table Name:** supermarketSales
- Actions:** QUERY, SHARE, COPY, SNAPSHOT, DELETE
- Schema Tab:** Selected
- Details:** DETAILS, PREVIEW, LINEAGE, DATA PROFILE, DATA QUALITY
- Filter:** Enter property name or value
- Columns:**

Field name	Type	Mode	Key	Collation	Default Value	Pol
InvoiceID	STRING	NULLABLE	-	-	-	-
Branch	STRING	NULLABLE	-	-	-	-
City	STRING	NULLABLE	-	-	-	-
CustomerType	STRING	NULLABLE	-	-	-	-
Gender	STRING	NULLABLE	-	-	-	-
ProductLine	STRING	NULLABLE	-	-	-	-
UnitPrice	FLOAT	NULLABLE	-	-	-	-
Quantity	NUMERIC	NULLABLE	-	-	-	-
TaxFivePercent	NUMERIC	NULLABLE	-	-	-	-
Total	NUMERIC	NULLABLE	-	-	-	-
Date	STRING	NULLABLE	-	-	-	-
Time	STRING	NULLABLE	-	-	-	-
Payment	STRING	NULLABLE	-	-	-	-
cogs	NUMERIC	NULLABLE	-	-	-	-
- Summary:** supermarketSales, df12-demo.lab

- Back to NiFi, add a ConvertRecord processor, and create a connection from PutGCSObject for the relationship success.

On the Properties tab, change the Record Reader value to use JsonTreeReader and the Record Writer value to CSVRecordSetWriter.

**Configure Processor | ConvertRecord 2.0.0-M2**

■ Stopped

SETTINGS	SCHEDULING	PROPERTIES	RELATIONSHIPS	COMMENTS
----------	------------	------------	---------------	----------

Required field

Property	Value
Record Reader	JsonTreeReader
Record Writer	CSVRecordSetWriter
Include Zero Record FlowFiles	true

**CANCEL** **APPLY**

- k. Add a new Processor and select LogAttribute. Create a Connection from PutGCSObject to create a relationship failure and from ConvertRecord to create a relationship failure also. On the Relationships tab, tick the terminate for relationship success.

**Configure Processor | LogAttribute 2.0.0-M2**

■ Stopped

SETTINGS	SCHEDULING	PROPERTIES	RELATIONSHIPS	COMMENTS
----------	------------	------------	---------------	----------

Automatically Terminate / Retry Relationships [?](#)

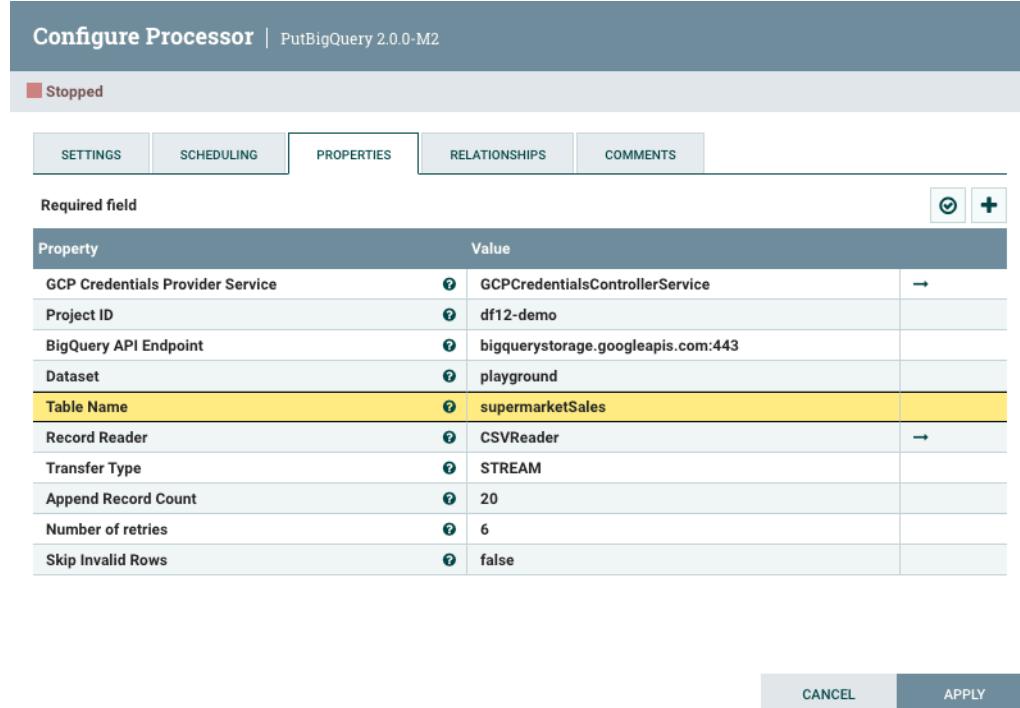
success  
 terminate  retry

All FlowFiles are routed to this relationship

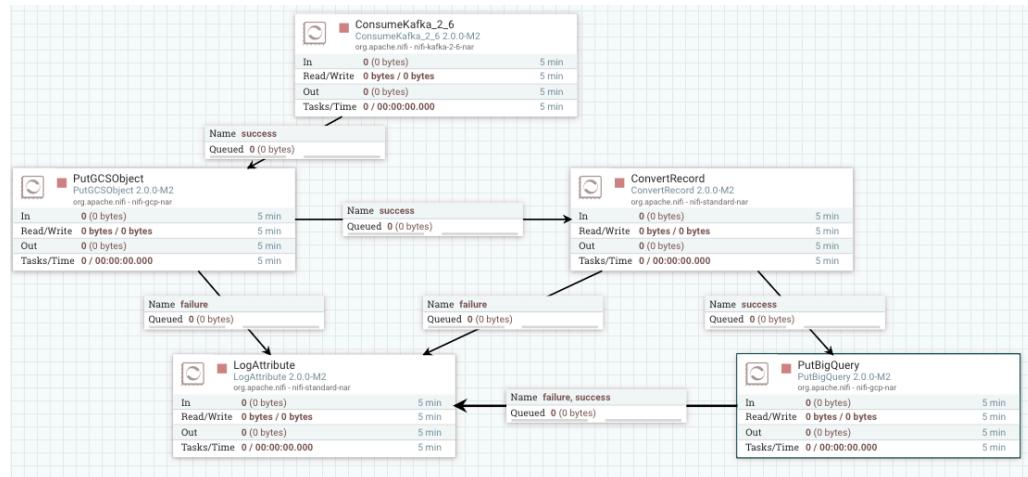
**CANCEL** **APPLY**

- l. Add a new PutBigQuery processor, then create a connection from ConvertRecord for a relationship success. Then create a connection to LogAttribute for both relationship success and failure.

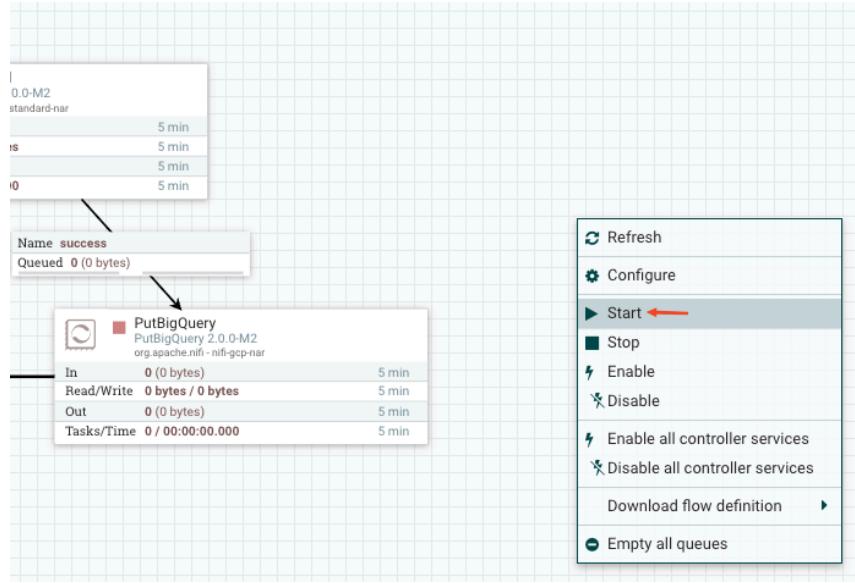
Then go to the Properties tab and change the properties of GCP Credentials Provider Service, Project ID, Dataset, Table Name, and the Record Reader.



- m. The NiFi canvas would look like this.



- n. Start the NiFi by right-clicking on the canvas.

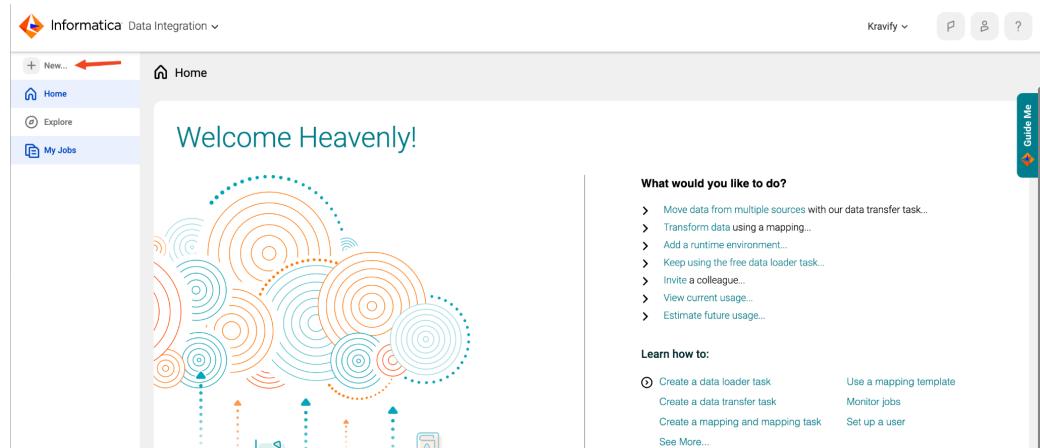


- Open a new terminal and run this command to send the message to the Kafka Producer.

```
cat file/supermarketSales.json | bin/kafka-console-producer.sh --broker-list localhost:9092 --topic supermarketSales
```



- Log in to the Informatica Intelligent Cloud Services page. On the left sidebar, select New.



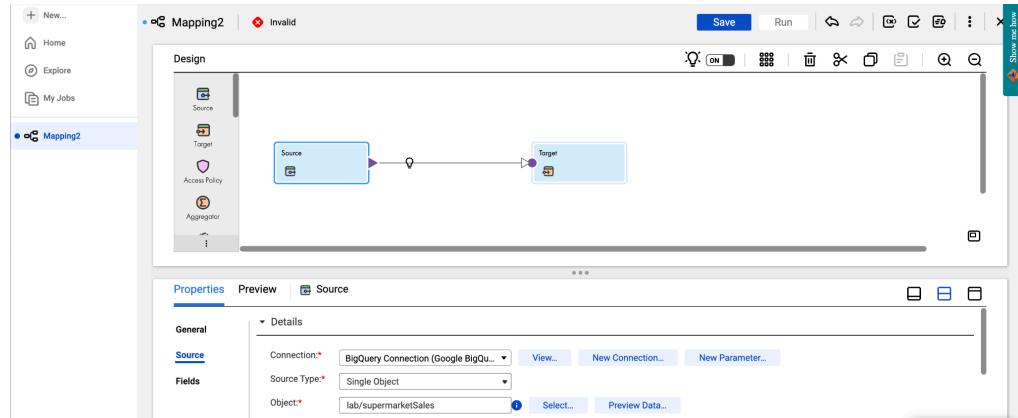
Select Mappings, then create a Mapping.

## New Asset

Select the type of asset you want. Some asset types include templates for common integration patterns.

The screenshot shows the 'New Asset' interface with a sidebar on the left containing links for Tasks, Mappings, Maplets, Taskflows, and Components. The 'Mappings' link is underlined, indicating it is selected. The main area is titled 'Mapping' with a red arrow pointing to it. Below the title is the sub-instruction: 'Create a mapping. This mapping can then be used in one or more mapping tasks.' There are four categories listed: Integration, Cleansing, Warehousing, and Industry Solutions.

- q. Select the source and configure the connection to use BigQuery.



### Select Source Object

Select a source object, then click OK. You can also search for a source object.

The screenshot shows the 'Select Source Object' dialog. On the left, there's a tree view under 'BigQuery Connection (Google BigQuery V2)' showing 'Packages' and 'Lab'. The 'supermarketSales(lab)' package is selected. On the right, a table lists objects:

Select	Name	Label	Description	Type
<input type="radio"/>	students	students		TABLE
<input checked="" type="radio"/>	supermarketSales	supermarketSales		TABLE

At the bottom, it says 'Displaying 2 matching object(s.)' and has 'OK' and 'Cancel' buttons.

- r. We need to change the date format from string to date format by adding the Aggregation. Then in the Incoming Field, add a Field rule to change the current date column into \_date so we can add the correct type column with the date name into the table.

## Configure Field Rules

Rule Details      Rename Fields

Rename selected fields:  Bulk  Individual

Field Name	Rename	Origin
Date	_date	lab/supermarketSales

?

OK      Cancel

Then in the Expression tab, add an expression TO\_DATE using \_date column and save it as the Date field name.

Properties   Preview      ChangeColumnType

General      Incoming Fields      Expression      Window      Advanced

Create simple expressions. You can also use expression macros to create complex expressions.  
 Allow additional fields and expressions during task creation

Expressions

Field Name	Expression	Default Value	Type	Precision	Scale
Date	TO_DATE(_date, 'MM/DD/YYYY')		date/time	29	9

- s. Create two Aggregations to summarize data based on the date (monthly) and Product Line. For the monthly summary, we need to use the Group By Date column and for the Product Line use the Product Line column.

## Edit Field

X

Create new output field, variable field, input macro field or output macro field.

Field Type:

Name:<sup>\*</sup>

Type:<sup>\*</sup>

Precision:<sup>\*</sup>

Scale:

Description:



OK

Cancel

## Field Expression: total\_sales( double, 15, 0 )

X

Configure the expression by adding fields, functions, and variables.

Expression:

The expression is valid.

Fields

Fields	Expression	Validate
Date InvoiceID Branch City CustomerType Gender ProductLine	SUM(Total)	<input type="button" value="Validate"/>

Select a field to see detailed information about it.



OK

Cancel

## Edit Field

X

Create new output field, variable field, input macro field or output macro field.

Field Type:	Output Field
Name:*	TotalQuantity
Type:*	integer
Precision:*	10
Scale:	0
Description:	monthly total quantity product sold



OK

Cancel

### Field Expression: TotalQuantity( integer, 10, 0 )

Configure the expression by adding fields, functions, and variables.

Expression:	Not Parameterized
<p> The expression is valid.</p>	
Fields	Expression
Gender	SUM(Quantity)
ProductLine	
UnitPrice	

We can create several aggregation based on the data.

Properties Preview MonthlyAggregator

General Create simple aggregate expressions. You can also use expression macros to create complex aggregate expressions.  
 Allow additional fields and expressions during task creation

Incoming Fields

Group By

Aggregate

Advanced

Field Name	Expression	Type	Precision	Scale	Field Description
MonthlySales	SUM(Total)	double	15	0	monthly sales from Total column
MonthlyQuantity	SUM(Quantity)	integer	10	0	monthly total quantity product sold
MonthlyPurchase	COUNT(InvoiceID)	integer	10	0	monthly purchase

Properties Preview | ProductLineAggregator

**RatingPerProductLine**

**General**

Create simple aggregate expressions. You can also use expression macros to create complex aggregate expressions.

Allow additional fields and expressions during task creation

**Incoming Fields**

**Group By**

**Aggregate**

Field Name	Expression	Type	Precision	Scale
TotalSalesPerProductLine	SUM(Total)	double	15	0
QuantityPerProductLine	SUM(Quantity)	integer	10	0
RatingPerProductLine	Avg(Rating)	double	15	0

**Advanced**

t. Set the Target in a new dataset in BigQuery.

Mapping2 | Invalid

**Design**

**Properties** **Preview** Target

**General**

**Incoming Fields**

**Target**

**Target Fields**

**Field Mapping**

**Details**

Connection: **BigQuery Connection (Google BigQu...)**

Target Type: **Single Object**

Object: **supermarketSalesSummary/monthlySum**

Operation: **Insert**

u. Run the query and wait until it is successful.

Mapping2 | Valid

**Design**

**Properties** **Preview** Target

**General**

**Incoming Fields**

**Target**

**Target Fields**

**Field Mapping**

**Details**

Connection: **BigQuery Connection (Google BigQu...)**

Target Type: **Single Object**

Object: **supermarketSalesSummary/monthlySum**

Operation: **Insert**

**My Jobs** | Data Integration

Jobs (1 of 7) Updated 6:57:02 AM PDT

Asset Name: Mapping2	Add Field
Instance Name: <b>Mapping2-1</b>	Location: Default
	Subtasks
	Start Time: <b>Apr 22, 2024, 6:56 AM</b>
	End Time
	Rows Processed
	Status: <b>Running</b>

The screenshot shows the Informatica Data Integration interface. At the top, there's a header with 'My Jobs' and 'Data Integration'. Below it, a sub-header says 'Jobs (1 of 7)' and 'Updated 6:57:52 AM PDT'. There are icons for refresh, search, and find. A dropdown menu shows 'Asset Name: Mapping2' and an 'Add Field' button. The main table has columns: Instance Name, Location, Subtasks, Start Time, End Time, Rows Processed, and Status. One row is shown: 'Mapping2-1' under 'Location', 'Default' under 'Subtasks', 'Apr 22, 2024, 6:56 AM' under 'Start Time', 'Apr 22, 2024, 6:57 ...' under 'End Time', '6' under 'Rows Processed', and 'Success' with a green checkmark under 'Status'.

v.

3. Data Storage: route the processed data to the data warehouse (BigQuery).
  - a. Ensure the data is available in the GCS.

The screenshot shows the Google Cloud Storage console. It displays a bucket named 'df12-assignment2'. Under the 'OBJECTS' tab, there is one object listed: 'supermarketSales'. The details show it's a CSV file, 350.3 KB in size, created on Apr 22, 2024, at 2:56:50 PM, with 'Standard' storage class and 'Not public' public access. The 'PERMISSIONS' tab is also visible.

- b. Ensure the table is created in the BigQuery.

The screenshot shows the BigQuery UI. It lists a table named 'supermarketSales' with 15 rows. The columns are: Row, InvoiceID, Branch, City, CustomerType, Gender, ProductLine, UnitPrice, Quantity, and TaxFreePc. The data includes various branches like A, C, and B across cities like Yangon and Naypyitaw, with gender and product details.

Row	InvoiceID	Branch	City	CustomerType	Gender	ProductLine	UnitPrice	Quantity	TaxFreePc
1	750-67-8428	A	Yangon	Member	Female	Health and beauty	74.69	7	26.1
2	226-31-3081	C	Naypyitaw	Normal	Female	Electronic accessories	15.28	5	:
3	631-41-3108	A	Yangon	Normal	Male	Home and lifestyle	46.33	7	16.2
4	123-19-1176	A	Yangon	Member	Male	Health and beauty	58.22	8	23.
5	373-73-7910	A	Yangon	Normal	Male	Sports and travel	86.31	7	30.2
6	699-14-3026	C	Naypyitaw	Normal	Male	Electronic accessories	85.39	7	29.8
7	355-53-5943	A	Yangon	Member	Female	Electronic accessories	68.84	6	20.
8	315-22-5665	C	Naypyitaw	Normal	Female	Home and lifestyle	73.56	10	3
9	665-32-9167	A	Yangon	Member	Female	Health and beauty	36.26	2	9.
10	692-92-5582	B	Mandalay	Member	Female	Food and beverages	54.84	3	8.
11	351-62-0822	B	Mandalay	Member	Female	Fashion accessories	14.48	4	2.
12	529-56-3974	B	Mandalay	Member	Male	Electronic accessories	25.51	4	5.
13	365-64-0515	A	Yangon	Normal	Female	Electronic accessories	46.95	5	11.7
14	252-56-2699	A	Yangon	Normal	Male	Food and beverages	43.19	10	21.
15	829-34-3910	A	Yangon	Normal	Female	Health and beauty	71.38	10	3:

## Additional Questions

1. I see lots of manipulation to the input data which I assume the condition in real life would be far from ideal, how a data engineer would handle that?  
In this case, the CSV file collected from Kaggle was converted into a single-line JSON format, then streamed into Kafka, and NiFi would convert it back to a CSV file. The question is, what for? Why should we convert to JSON just to convert it back to CSV?
2. Is there any best practice for handling errors that are caused by different column types in the table so it wouldn't cause a null value?
3. On a company scale, is it needed to route the data from source to Kafka, then ingest it into NiFi which would insert the data into BigQuery, then transform using Informatica

from BigQuery data and put the result in another BigQuery table? Can we actually use Kafka to ingest data straight to BigQuery? And also, what's the point of putting the data into GCS?