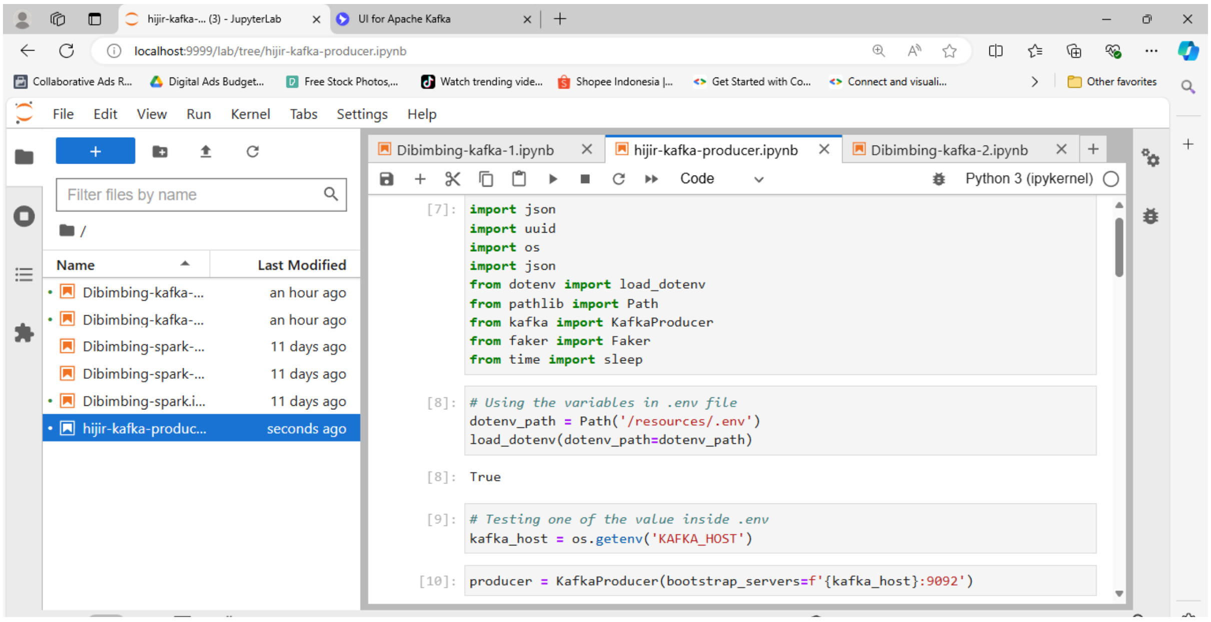
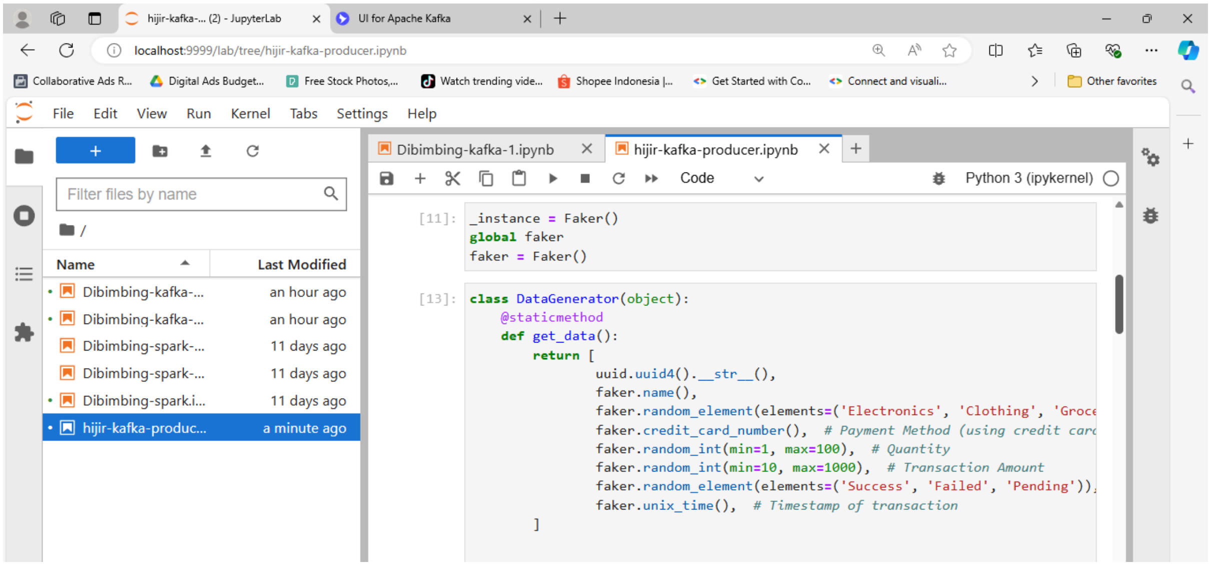
Homework - Kafka - Hijir

Without setup new docker - just in case the new docker kafka setup fails

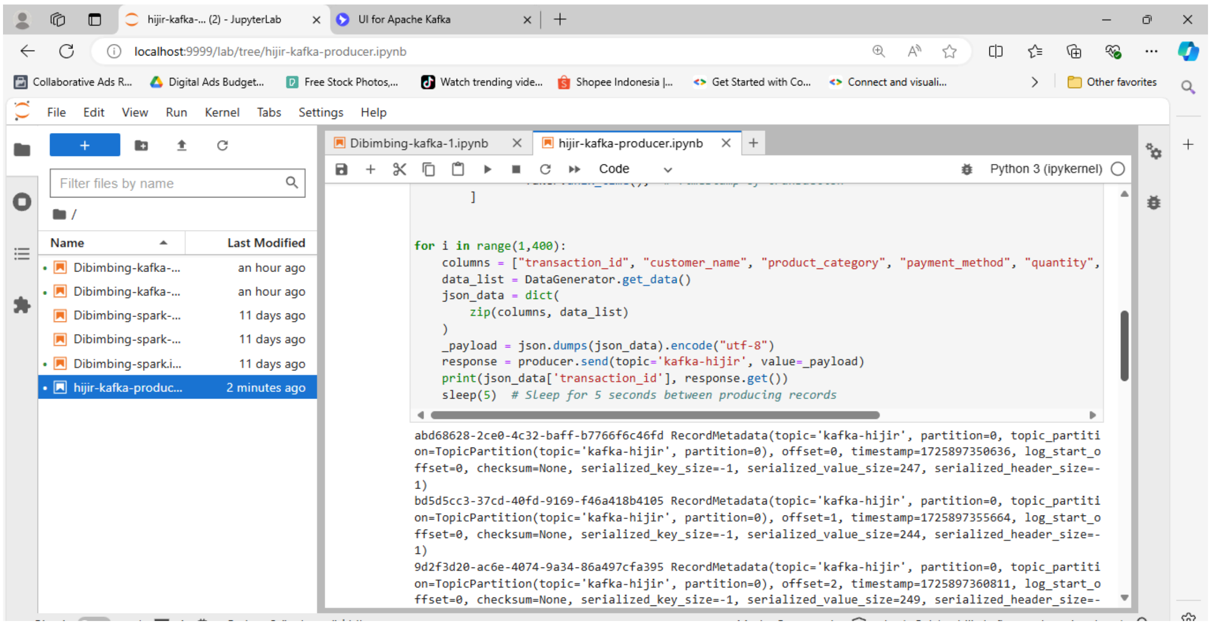


```
[7]: import json
     import uuid
     import os
     import json
     from dotenv import load_dotenv
     from pathlib import Path
     from kafka import KafkaProducer
     from faker import Faker
     from time import sleep

[8]: # Using the variables in .env file
     dotenv_path = Path('/resources/.env')
     load_dotenv(dotenv_path=dotenv_path)

[8]: True

[9]: # Testing one of the value inside .env
     kafka_host = os.getenv('KAFKA_HOST')

[10]: producer = KafkaProducer(bootstrap_servers=f'{kafka_host}:9092')
```

Jupyter ‘hijir-kafka-producer”



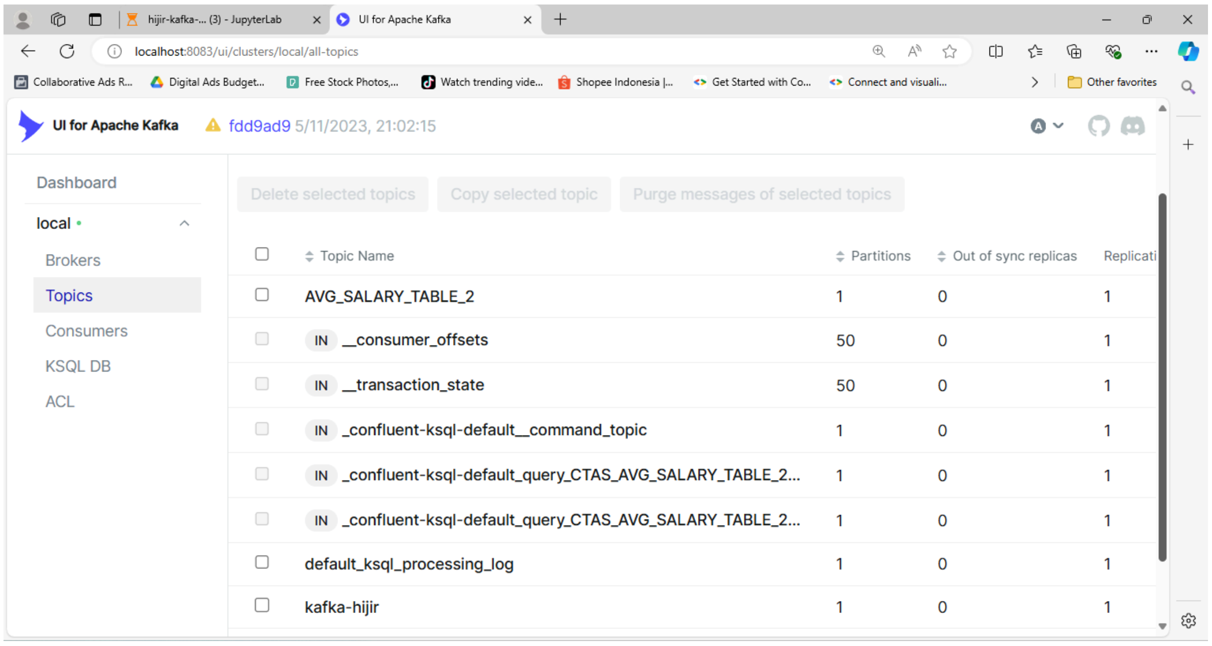
```
[11]: _instance = Faker()
      global faker
      faker = Faker()

[13]: class DataGenerator(object):
          @staticmethod
          def get_data():
              return [
                      uuid.uuid4().__str__(),
                      faker.name(),
                      faker.random_element(elements=('Electronics', 'Clothing', 'Groce
                      faker.credit_card_number(),  # Payment Method (using credit carc
                      faker.random_int(min=1, max=100),  # Quantity
                      faker.random_int(min=10, max=1000),  # Transaction Amount
                      faker.random_element(elements=('Success', 'Failed', 'Pending')),
                      faker.unix_time(),  # Timestamp of transaction
              ]
```


hijir-kafka-... (2) - JupyterLab
UI for Apache Kafka
localhost:9999/lab/tree/hijir-kafka-producer.ipynb
File Edit View Run Kernel Tabs Settings Help
Dibimbing-kafka-1.ipynb
hijir-kafka-producer.ipynb
Python 3 (ipykernel)

Name	Last Modified
Dibimbing-kafka-...	an hour ago
Dibimbing-kafka-...	an hour ago
Dibimbing-spark-...	11 days ago
Dibimbing-spark-...	11 days ago
Dibimbing-spark.i...	11 days ago
hijir-kafka-produc...	2 minutes ago

```python
    ]


for i in range(1,400):
    columns = ["transaction_id", "customer_name", "product_category", "payment_method", "quantity",
    data_list = DataGenerator.get_data()
    json_data = dict(
        zip(columns, data_list)
    )
    _payload = json.dumps(json_data).encode("utf-8")
    response = producer.send(topic='kafka-hijir', value=_payload)
    print(json_data['transaction_id'], response.get())
    sleep(5)  # Sleep for 5 seconds between producing records
```

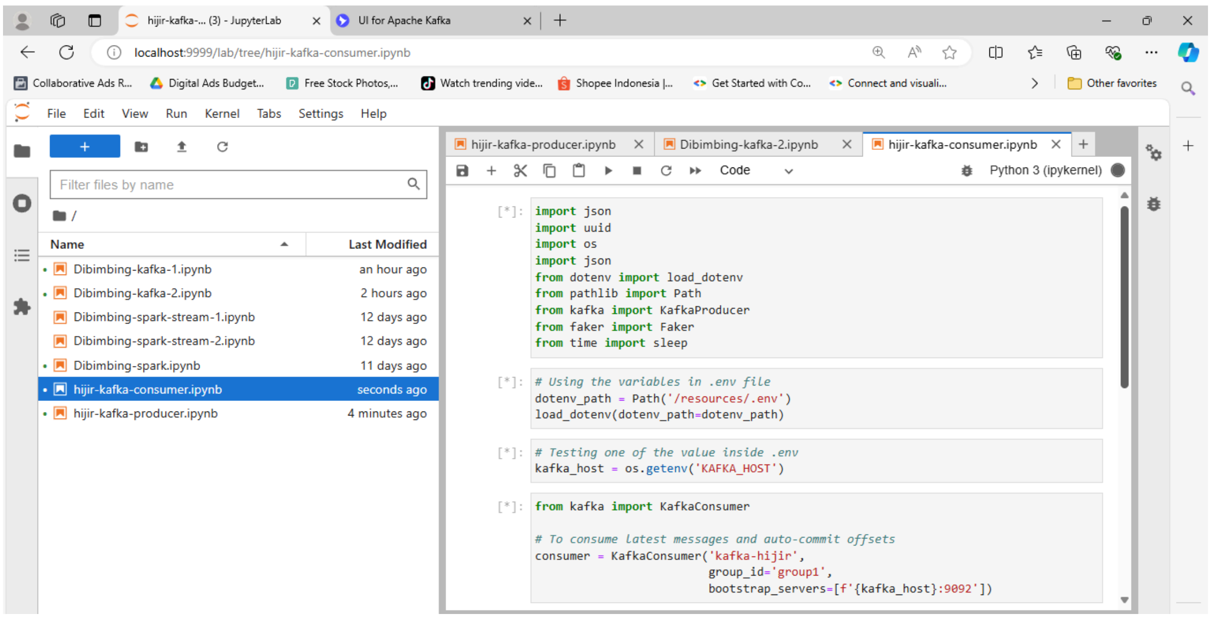
```
abd68628-2ce0-4c32-baff-b7766f6c46fd RecordMetadata(topic='kafka-hijir', partition=0, topic_partiti
on=TopicPartition(topic='kafka-hijir', partition=0), offset=0, timestamp=1725897350636, log_start_o
ffset=0, checksum=None, serialized_key_size=-1, serialized_value_size=247, serialized_header_size=-
1)
bd5d5cc3-37cd-40fd-9169-f46a418b4105 RecordMetadata(topic='kafka-hijir', partition=0, topic_partiti
on=TopicPartition(topic='kafka-hijir', partition=0), offset=1, timestamp=1725897355664, log_start_o
ffset=0, checksum=None, serialized_key_size=-1, serialized_value_size=244, serialized_header_size=-
1)
9d2f3d20-ac6e-4074-9a34-86a497cfa395 RecordMetadata(topic='kafka-hijir', partition=0, topic_partiti
on=TopicPartition(topic='kafka-hijir', partition=0), offset=2, timestamp=1725897360811, log_start_o
ffset=0, checksum=None, serialized_key_size=-1, serialized_value_size=249, serialized_header_size=-
```


hijir-kafka-... (3) - JupyterLab
UI for Apache Kafka
localhost:8083/ui/clusters/local/all-topics
UI for Apache Kafka
fdd9ad9 5/11/2023, 21:02:15
Dashboard
local
Brokers
Topics
Consumers
KSQL DB
ACL
Delete selected topics
Copy selected topic
Purge messages of selected topics

Topic Name	Partitions	Out of sync replicas	Replicati
AVG_SALARY_TABLE_2	1	0	1
IN __consumer_offsets	50	0	1
IN __transaction_state	50	0	1
IN _confluent-ksql-default__command_topic	1	0	1
IN _confluent-ksql-default_query_CTAS_AVG_SALARY_TABLE_2...	1	0	1
IN _confluent-ksql-default_query_CTAS_AVG_SALARY_TABLE_2...	1	0	1
default_ksql_processing_log	1	0	1
kafka-hijir	1	0	1

Event sudah terkirim ke kafka (**kafka-hijir**)

Jupyter “hijir-kafka-consumer”



localhost:9999/lab/tree/hijir-kafka-consumer.ipynb

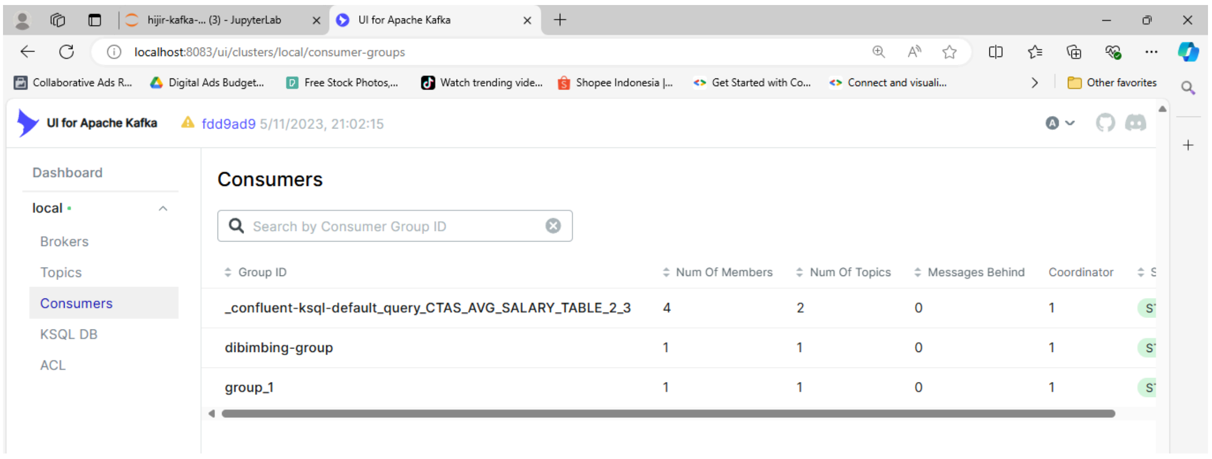
Name	Last Modified
Dibimbing-kafka-1.ipynb	an hour ago
Dibimbing-kafka-2.ipynb	2 hours ago
Dibimbing-spark-stream-1.ipynb	12 days ago
Dibimbing-spark-stream-2.ipynb	12 days ago
Dibimbing-spark.ipynb	11 days ago
hijir-kafka-consumer.ipynb	seconds ago
hijir-kafka-producer.ipynb	4 minutes ago

```
[*]: import json
     import uuid
     import os
     import json
     from dotenv import load_dotenv
     from pathlib import Path
     from kafka import KafkaProducer
     from faker import Faker
     from time import sleep
```

```
[*]: # Using the variables in .env file
     dotenv_path = Path('/resources/.env')
     load_dotenv(dotenv_path=dotenv_path)
```

```
[*]: # Testing one of the value inside .env
     kafka_host = os.getenv('KAFKA_HOST')
```

```
[*]: from kafka import KafkaConsumer

     # To consume latest messages and auto-commit offsets
     consumer = KafkaConsumer('kafka-hijir',
                              group_id='group1',
                              bootstrap_servers=[f'{kafka_host}:9092'])
```

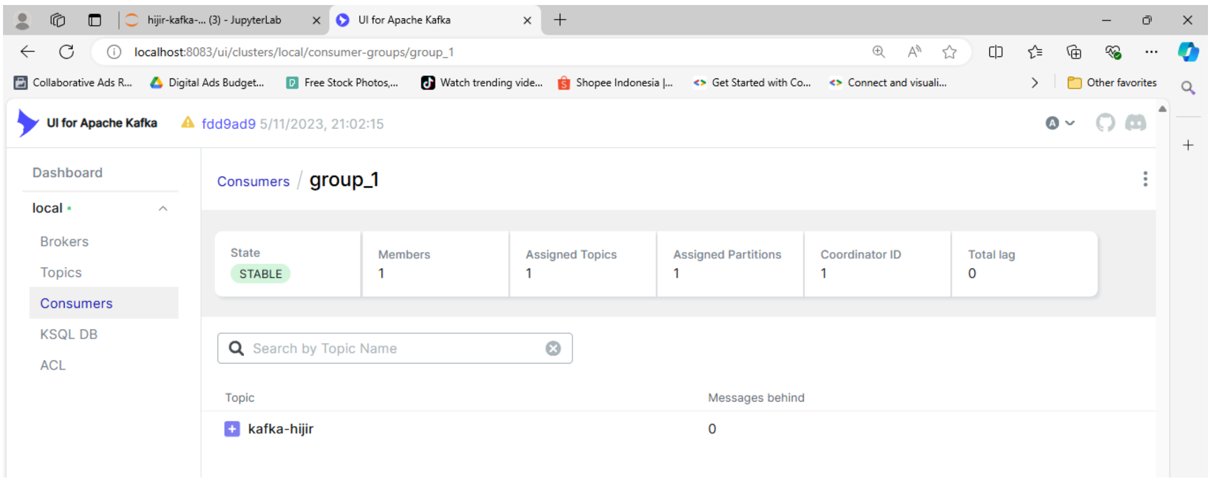


localhost:8083/ui/clusters/local/consumer-groups

UI for Apache Kafka fdd9ad9 5/11/2023, 21:02:15

Consumers

Group ID	Num Of Members	Num Of Topics	Messages Behind	Coordinator	S
_confluent-ksql-default_query_CTAS_AVG_SALARY_TABLE_2_3	4	2	0	1	S
dibimbing-group	1	1	0	1	S
group_1	1	1	0	1	S

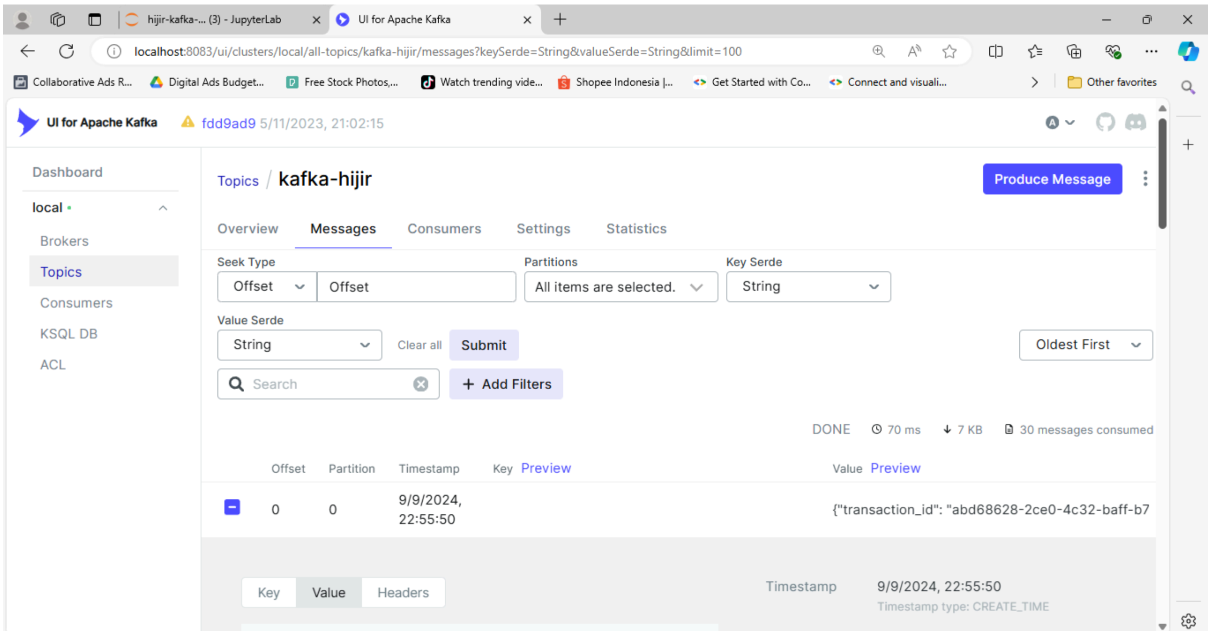


localhost:8083/ui/clusters/local/consumer-groups/group_1

Consumers / group_1

State	Members	Assigned Topics	Assigned Partitions	Coordinator ID	Total lag
STABLE	1	1	1	1	0

Topic	Messages behind
kafka-hijir	0


localhost:8083/ui/clusters/local/all-topics/kafka-hijir/messages?keySerde=String&valueSerde=String&limit=100

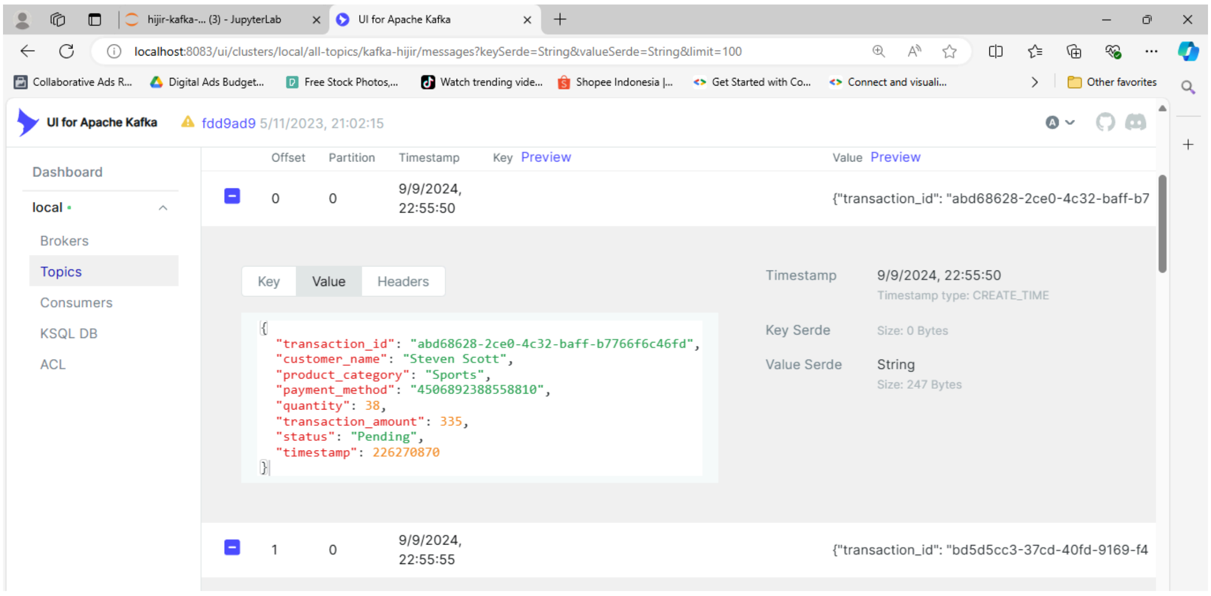
Topics / kafka-hijir

Seek Type: Offset | Partitions: All items are selected. | Key Serde: String | Value Serde: String | Oldest First

DONE 70 ms 7 KB 30 messages consumed

```
Offset  Partition  Timestamp            Key   Value
0       0          9/9/2024, 22:55:50         {"transaction_id": "abd68628-2ce0-4c32-baff-b7
```

Timestamp 9/9/2024, 22:55:50
Timestamp type: CREATE_TIME


```
Offset  Partition  Timestamp            Key   Value
0       0          9/9/2024, 22:55:50         {"transaction_id": "abd68628-2ce0-4c32-baff-b7
```

```
{
  "transaction_id": "abd68628-2ce0-4c32-baff-b7766f6c46fd",
  "customer_name": "Steven Scott",
  "product_category": "Sports",
  "payment_method": "4506892388558810",
  "quantity": 38,
  "transaction_amount": 335,
  "status": "Pending",
  "timestamp": 226270870
}
```

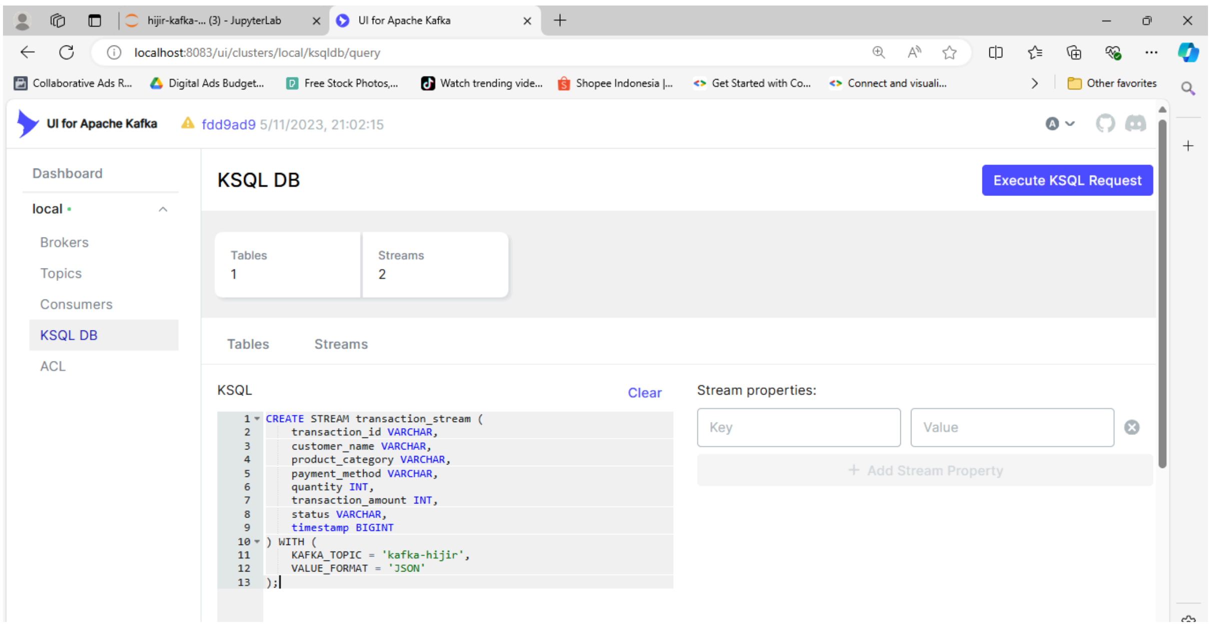
Timestamp 9/9/2024, 22:55:50
Timestamp type: CREATE_TIME
Key Serde Size: 0 Bytes
Value Serde String Size: 247 Bytes

```
1       0          9/9/2024, 22:55:55         {"transaction_id": "bd5d5cc3-37cd-40fd-9169-f4
```

Ksqldb

Melakukan agregrasi di kafka menggunakan sql

```
CREATE STREAM transaction_stream (
    transaction_id VARCHAR,
    customer_name VARCHAR,
    product_category VARCHAR,
    payment_method VARCHAR,
    quantity INT,
    transaction_amount INT,
    status VARCHAR,
    timestamp BIGINT
) WITH (
    KAFKA_TOPIC = 'kafka-hijir',
    VALUE_FORMAT = 'JSON'
);
```



localhost:8083/ui/clusters/local/ksqldb/query

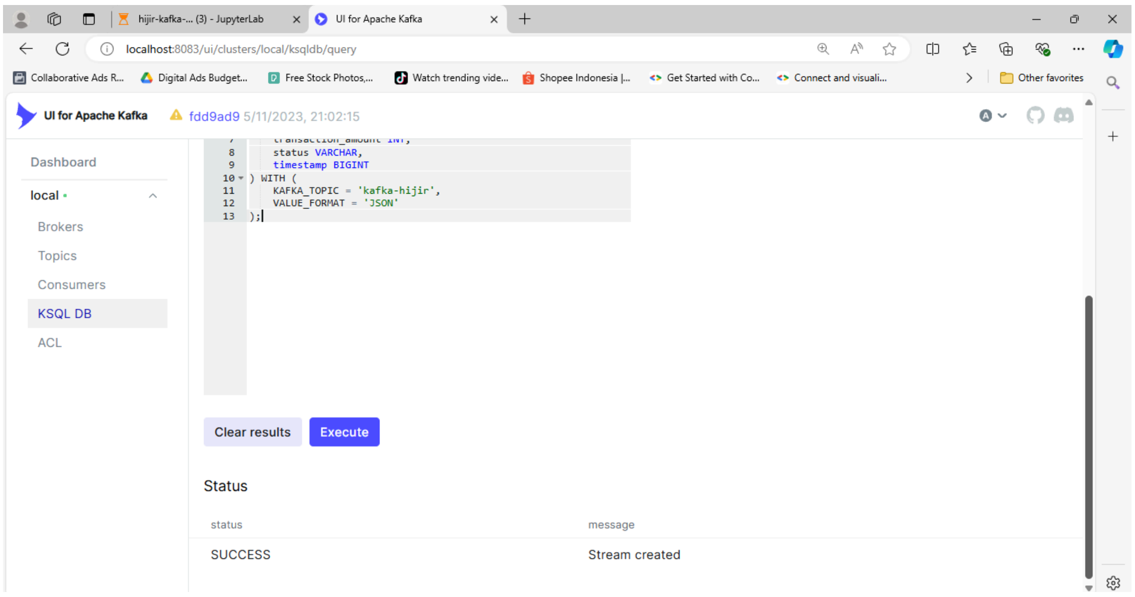
UI for Apache Kafka fdd9ad9 5/11/2023, 21:02:15

KSQL DB

Tables 1 | Streams 2

KSQL

```
CREATE STREAM transaction_stream (
    transaction_id VARCHAR,
    customer_name VARCHAR,
    product_category VARCHAR,
    payment_method VARCHAR,
    quantity INT,
    transaction_amount INT,
    status VARCHAR,
    timestamp BIGINT
) WITH (
    KAFKA_TOPIC = 'kafka-hijir',
    VALUE_FORMAT = 'JSON'
);
```



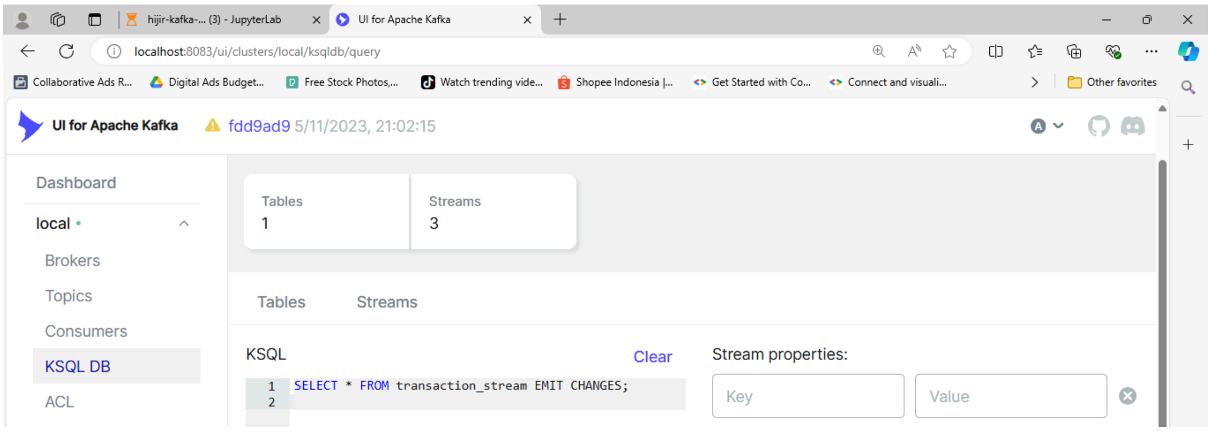
```
    status VARCHAR,
    timestamp BIGINT
) WITH (
    KAFKA_TOPIC = 'kafka-hijir',
    VALUE_FORMAT = 'JSON'
);
```

Status

```
status      message
SUCCESS     Stream created
```

Berhasil membuat stream

```
SELECT * FROM transaction_stream EMIT CHANGES;
```

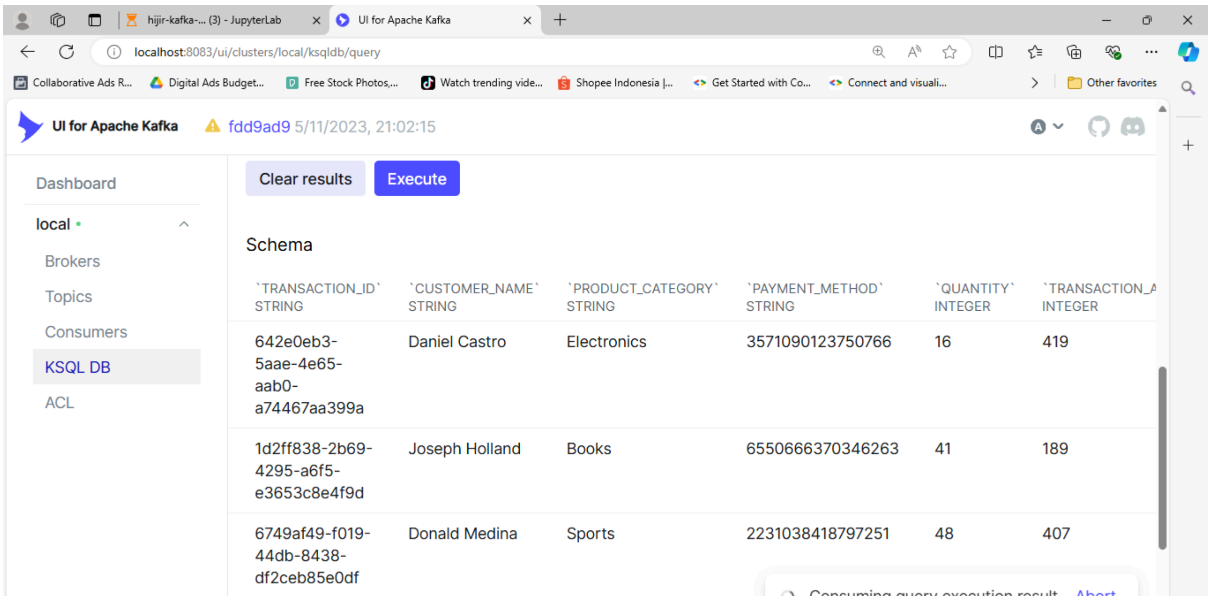


Tables	Streams
1	3

KSQL

```
SELECT * FROM transaction_stream EMIT CHANGES;
```

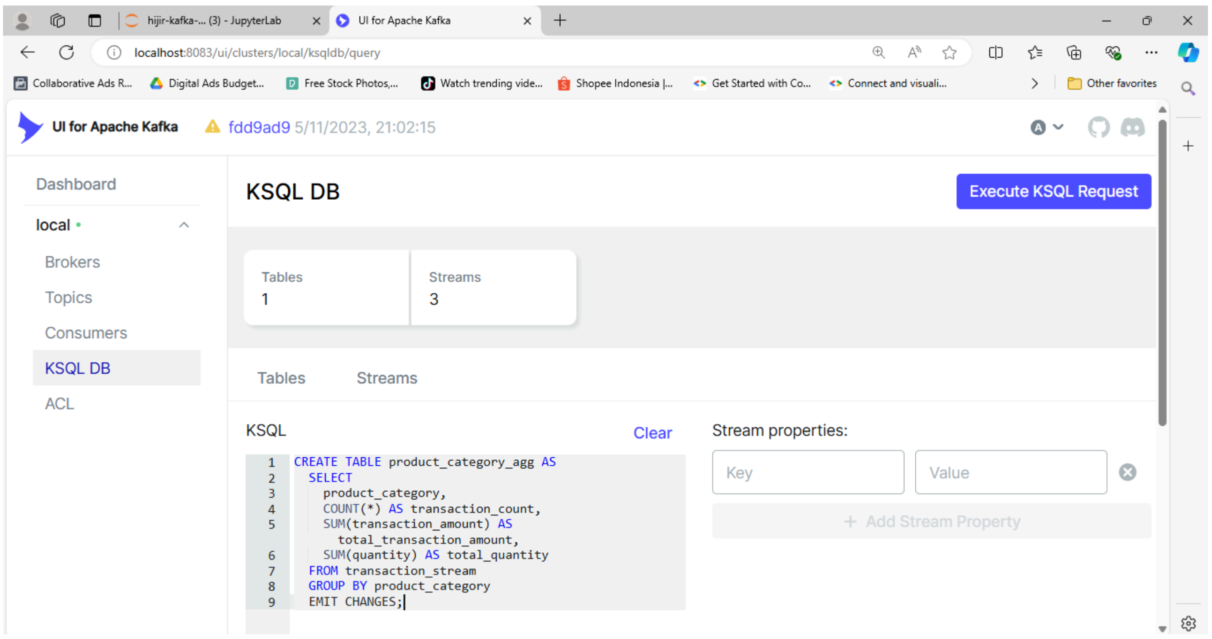
Stream properties:



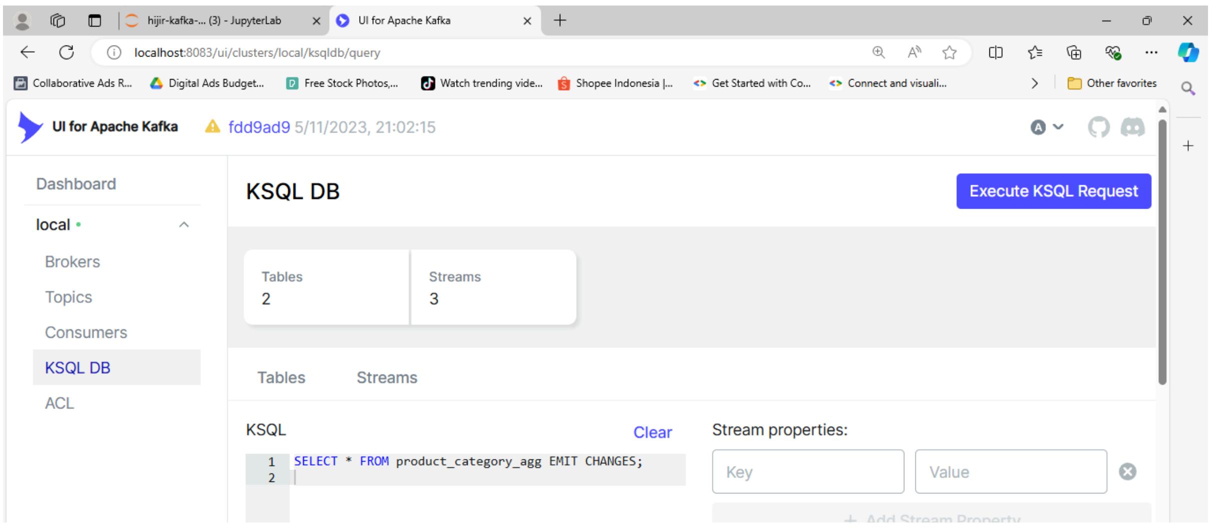
Schema

\`TRANSACTION_ID\` STRING	\`CUSTOMER_NAME\` STRING	\`PRODUCT_CATEGORY\` STRING	\`PAYMENT_METHOD\` STRING	\`QUANTITY\` INTEGER	\`TRANSACTION_A INTEGER
642e0eb3-5aae-4e65-aab0-a74467aa399a	Daniel Castro	Electronics	3571090123750766	16	419
1d2ff838-2b69-4295-a6f5-e3653c8e4f9d	Joseph Holland	Books	6550666370346263	41	189
6749af49-f019-44db-8438-df2ceb85e0df	Donald Medina	Sports	2231038418797251	48	407

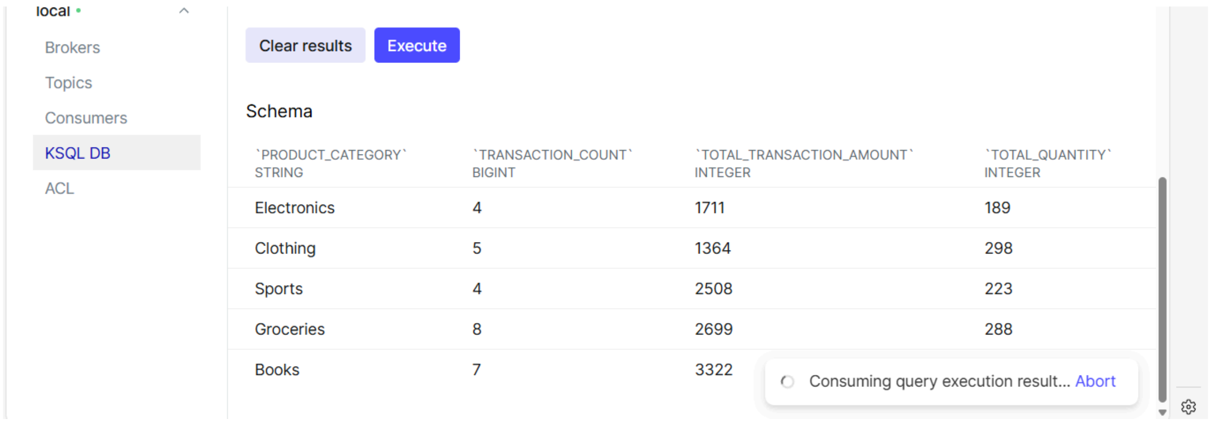
Agregasi



```
CREATE TABLE product_category_agg AS
  SELECT
    product_category,
    COUNT(*) AS transaction_count,
    SUM(transaction_amount) AS
      total_transaction_amount,
    SUM(quantity) AS total_quantity
  FROM transaction_stream
  GROUP BY product_category
  EMIT CHANGES;
```



```
SELECT * FROM product_category_agg EMIT CHANGES;
```



Schema

`PRODUCT_CATEGORY` STRING	`TRANSACTION_COUNT` BIGINT	`TOTAL_TRANSACTION_AMOUNT` INTEGER	`TOTAL_QUANTITY` INTEGER
Electronics	4	1711	189
Clothing	5	1364	298
Sports	4	2508	223
Groceries	8	2699	288
Books	7	3322	

Berikut ini adalah hasil agregasi dari tabel product_category