

## Кроки виконання завдання:

- Створення topics. Було створено три топіка: oie\_building\_sensors, oie\_humidity\_alerts, oie\_temperature\_alerts на локальному сервері. Для цього створено контейнер kafka через файл docker-compose.yml

```

my_name = "oie"
[print(topic) for topic in admin_client.list_topics() if my_name in topic]
print("-----")
# Закриваємо з'єднання з клієном
admin_client.close()

(spark_env) ole@AD-KOP9YGLWT kafka % docker-compose ps
● spark-env
(spark_env) ole@AD-KOP9YGLWT kafka % docker-compose up
(spark_env) ole@AD-KOP9YGLWT kafka % Goit_HW_topic5_create_topic.py
An error occurred: [Error 36] TopicAlreadyExistsError: Request 'CreateTopicsRequest_v3#createTopicRequest' failed with response 'CreateTopicsResponse_v3#topicResponses': topic_requests=[{'topic': 'oie_building_sensors', 'num_partitions': 3, 'replication_factor': 1, 'partition_assignment': [[1, 1], [1, 2], [1, 3]]}, {'topic': 'oie_humidity_alerts', 'num_partitions': 3, 'replication_factor': 1, 'partition_assignment': [[1, 1], [1, 2], [1, 3]]}, {'topic': 'oie_temperature_alerts', 'num_partitions': 3, 'replication_factor': 1, 'partition_assignment': [[1, 1], [1, 2], [1, 3]]]}, validation='validate_onlyFalse'] failed with response 'CreateTopicsResponse_v3#topicResponses': topic_error=[{'topic': 'oie_building_sensors', 'error_code': 36, 'error_message': "Topic 'oie_building_sensors' already exists."}, {'topic': 'oie_humidity_alerts', 'error_code': 36, 'error_message': "Topic 'oie_humidity_alerts' already exists."}, {'topic': 'oie_temperature_alerts', 'error_code': 36, 'error_message': "Topic 'oie_temperature_alerts' already exists."}]
(spark_env) ole@AD-KOP9YGLWT kafka %

```

- Створено та запущено в роботу consumer (підписаний на oie\_building\_sensors). Що приймає сповіщення від provider з даними, отриманими від датчиків (у даному випадку два датчика sensor\_building\_1, sensor\_building\_2) та обробляє отримані дані.

```

from kafka import KafkaProducer
from kafka.errors import NoBrokersAvailable
from Goit_HW_topic5_configs import kafka_config, building_sensors
import json
import uuid
import time
import random

# Створення Kafka Producer
try:
    producer = KafkaProducer(
        bootstrap_servers=kafka_config['bootstrap_servers'],
        security_protocol=kafka_config['security_protocol'],
        sasl_mechanism=kafka_config['sasl_mechanism'],
        sasl_plain_username=kafka_config['username'],
        sasl_plain_password=kafka_config['password'],
        value_serializer=lambda v: json.dumps(v).encode('utf-8'),
    )
    print("Kafka Producer created successfully.")
except NoBrokersAvailable:
    print("No Kafka brokers are available. Please check the connection settings.")
    exit()
except Exception as e:
    print(f"An unexpected error occurred: {e}")

(spark_env) ole@AD-KOP9YGLWT kafka % python Goit_HW_topic5_consumer.py
Received message: {'timestamp': 1762727866.214122, 'sensor_building_id': 1, 'temperature': 25.51, 'humidity': 40.45} with key: 1, partition 0
Received message: {'timestamp': 1762727866.223891, 'sensor_building_id': 1, 'temperature': 42.7, 'humidity': 83.05} with key: 1, partition 0
ALERT: High humidity detected from sensor 1!
Message sent to topic 'oie_temperature_alerts' successfully.
ALERT: High humidity detected from sensor 1!
Received message: {'timestamp': 1762727870.233643, 'sensor_building_id': 1, 'temperature': 39.97, 'humidity': 68.25} with key: 1, partition 0
Received message: {'timestamp': 1762727897.2410219, 'sensor_building_id': 2, 'temperature': 29.72, 'humidity': 68.65} with key: 2, partition 2
Received message: {'timestamp': 1762727897.359246, 'sensor_building_id': 2, 'temperature': 48.7, 'humidity': 18.86} with key: 2, partition 2
ALERT: High temperature detected from sensor 2!
Message sent to topic 'oie_temperature_alerts' successfully.
ALERT: High temperature detected from sensor 2!
Received message: {'timestamp': 1762727901.3592541, 'sensor_building_id': 2, 'temperature': 31.81, 'humidity': 79.91} with key: 2, partition 2
Received message: {'timestamp': 1762727903.366525, 'sensor_building_id': 2, 'temperature': 36.26, 'humidity': 32.69} with key: 2, partition 2
Received message: {'timestamp': 1762727907.365648, 'sensor_building_id': 2, 'temperature': 37.47, 'humidity': 76.8} with key: 2, partition 2
Received message: {'timestamp': 1762727907.365976, 'sensor_building_id': 2, 'temperature': 32.99, 'humidity': 84.19} with key: 2, partition 2
ALERT: High humidity detected from sensor 2!

```

У разі відповідності параметрів заданим умовам (перевищення певного порогу значень) вбудований provider формує alert сповіщення та відправляє його в інші топіки (oie\_humidity\_alerts, oie\_temperature\_alerts ) в залежності від того, який з параметрів відповідає заданим умовам.

```

kafka > Goit_HW_topic5_producer.py ...
1 from kafka import KafkaProducer
2 from kafka.errors import NoBrokersAvailable
3 from Goit_HW_topic5_configs import kafka_config, building_sensors
4 import json
5 import uuid
6 import time
7 import random
8
9 # Створення Kafka Producer
10 try:
11     producer = KafkaProducer(
12         bootstrap_servers=kafka_config['bootstrap_servers'],
13         #security_protocol=kafka_config['security_protocol'],
14         #ssl_mechanism=kafka_config['ssl_mechanism'],
15         #ssl_plain_username=kafka_config['username'],
16         #ssl_plain_password=kafka_config['password'],
17         value_serializer=lambda v: json.dumps(v).encode('utf-8'),
18     )
19     print("Kafka Producer created successfully.")
20 except NoBrokersAvailable:
21     print("No Kafka brokers are available. Please check the connection settings.")
22     exit()
23 except Exception as e:
24     print(f"An unexpected error occurred: {e}")

```

(spark\_env) oleg@KQP9YG7LMT kafka % python Goit\_HW\_topic5\_consumer.py

Received message: {'timestamp': 176227866.24122, 'sensor\_building\_id': 1, 'temperature': 25.51, 'humidity': 48.45} with key: 1, partition 0

Received message: {'timestamp': 176227866.24122, 'sensor\_building\_id': 1, 'temperature': 42.7, 'humidity': 83.05} with key: 1, partition 0

ALERT: High temperature detected from sensor 1!

Message sent to topic 'oie\_temperature\_alerts' successfully.

ALERT: High humidity detected from sensor 1!

Message sent to topic 'oie\_humidity\_alerts' successfully.

Received message: {'timestamp': 176227870.23043, 'sensor\_building\_id': 1, 'temperature': 39.97, 'humidity': 68.25} with key: 1, partition 0

Received message: {'timestamp': 176227870.23043, 'sensor\_building\_id': 2, 'temperature': 29.72, 'humidity': 68.05} with key: 2, partition 0

Received message: {'timestamp': 176227870.35246, 'sensor\_building\_id': 2, 'temperature': 40.7, 'humidity': 18.06} with key: 2, partition 0

ALERT: High temperature detected from sensor 2!

Message sent to topic 'oie\_temperature\_alerts' successfully.

ALERT: High humidity detected from sensor 2!

Message sent to topic 'oie\_humidity\_alerts' successfully.

Received message: {'timestamp': 176227981.35941, 'sensor\_building\_id': 2, 'temperature': 31.01, 'humidity': 70.01} with key: 2, partition 0

Received message: {'timestamp': 176227981.35941, 'sensor\_building\_id': 2, 'temperature': 35.28, 'humidity': 33.58} with key: 2, partition 0

Received message: {'timestamp': 176227985.375648, 'sensor\_building\_id': 2, 'temperature': 37.47, 'humidity': 76.08} with key: 2, partition 0

Received message: {'timestamp': 176227987.383976, 'sensor\_building\_id': 2, 'temperature': 32.99, 'humidity': 84.15} with key: 2, partition 0

ALERT: High humidity detected from sensor 2!

- Створено та запущено в роботу provider. Що імітує роботу двох датчиків (у даному випадку два датчики sensor\_building\_1, sensor\_building\_2), генерує сповіщення з даними, отриманими від датчиків , та відправляє сповіщення в топік oie\_building\_sensors, який слухає consumer (підписаний на oie\_building\_sensors).

```

kafka > Goit_HW_topic5_producer.py ...
1 from kafka import KafkaProducer
2 from kafka.errors import NoBrokersAvailable
3 from Goit_HW_topic5_configs import kafka_config, building_sensors
4 import json
5 import uuid
6 import time
7 import random
8
9 # Створення Kafka Producer
10 try:
11     producer = KafkaProducer(
12         bootstrap_servers=kafka_config['bootstrap_servers'],
13         #security_protocol=kafka_config['security_protocol'],
14         #ssl_mechanism=kafka_config['ssl_mechanism'],
15         #ssl_plain_username=kafka_config['username'],
16         #ssl_plain_password=kafka_config['password'],
17         value_serializer=lambda v: json.dumps(v).encode('utf-8'),
18     )
19     print("Kafka Producer created successfully.")
20 except NoBrokersAvailable:
21     print("No Kafka brokers are available. Please check the connection settings.")
22     exit()
23 except Exception as e:
24     print(f"An unexpected error occurred: {e}")

```

(spark\_env) oleg@KQP9YG7LMT kafka % python Goit\_HW\_topic5\_consumer.py

Received message: {'timestamp': 176227866.24122, 'sensor\_building\_id': 1, 'temperature': 25.51, 'humidity': 48.45} with key: 1, partition 0

Received message: {'timestamp': 176227866.24122, 'sensor\_building\_id': 1, 'temperature': 42.7, 'humidity': 83.05} with key: 1, partition 0

ALERT: High temperature detected from sensor 1!

Message sent to topic 'oie\_temperature\_alerts' successfully.

ALERT: High humidity detected from sensor 1!

Message sent to topic 'oie\_humidity\_alerts' successfully.

Received message: {'timestamp': 176227870.23043, 'sensor\_building\_id': 1, 'temperature': 39.97, 'humidity': 68.25} with key: 1, partition 0

Received message: {'timestamp': 176227870.23043, 'sensor\_building\_id': 2, 'temperature': 29.72, 'humidity': 68.05} with key: 2, partition 0

Received message: {'timestamp': 176227870.35246, 'sensor\_building\_id': 2, 'temperature': 40.7, 'humidity': 18.06} with key: 2, partition 0

ALERT: High temperature detected from sensor 2!

Message sent to topic 'oie\_temperature\_alerts' successfully.

ALERT: High humidity detected from sensor 2!

Message sent to topic 'oie\_humidity\_alerts' successfully.

Received message: {'timestamp': 176227981.35941, 'sensor\_building\_id': 2, 'temperature': 31.01, 'humidity': 70.01} with key: 2, partition 0

Received message: {'timestamp': 176227981.35941, 'sensor\_building\_id': 2, 'temperature': 35.28, 'humidity': 33.58} with key: 2, partition 0

Received message: {'timestamp': 176227985.375648, 'sensor\_building\_id': 2, 'temperature': 37.47, 'humidity': 76.08} with key: 2, partition 0

Received message: {'timestamp': 176227987.383976, 'sensor\_building\_id': 2, 'temperature': 32.99, 'humidity': 84.15} with key: 2, partition 0

ALERT: High humidity detected from sensor 2!

- Створено та запущено в роботу ще один consumer (підписаний на oie\_humidity\_alerts, oie\_temperature\_alerts), що приймає сповіщення від provider з alert сповіщеннями (містять оригінальні показники датчика), та виводять попередження у консоль, що містить показники датчика та причину сповіщення.

The screenshot shows a code editor interface with several tabs open. The active tab is 'Goit\_HW\_topic5\_producer.py'. The code implements a Kafka producer to send JSON-formatted data to a topic. It includes error handling for broker availability and unexpected exceptions.

```
1 # Создание Kafka Producer
2 try:
3     producer = KafkaProducer(
4         bootstrap_servers=kafka_config['bootstrap_servers'],
5         security_protocol=kafka_config['security_protocol'],
6         ssl_cafile=kafka_config['ssl_cafile'],
7         ssl_certfile=kafka_config['ssl_certfile'],
8         ssl_keyfile=kafka_config['ssl_keyfile'],
9         #ssl_plain_password=kafka_config['password'],
10        value_serializer=lambda v: json.dumps(v).encode('utf-8'))
11
12     print("Kafka Producer created successfully!")
13 except NoBrokersAvailable:
14     print("No brokers are available. Please check the connection settings.")
15     exit()
16
17 except Exception as e:
18     print(f"An unexpected error occurred: {e}")
19
20
21 (spark_env) ole@ole-K039GULM: ~ % python Goit_HW_topic5_consumer_print.py
22 Details:
23 {
24     'alert_timestamp': 1762277868.235016, 'alert_type': 'HIGH_TEMPERATURE', 'original_timestamp': 1762277868.223091, 'sensor_building_id': 1, 'temperature': 40.7, 'message': 'Temperature 40.7 exceeds 40°C threshold.'}
25 ALERT: Humidity 83.05% is outside the 20-80% normal range.
26 Details:
27 {
28     'alert_timestamp': 1762277868.235031, 'alert_type': 'HIGH_HUMIDITY', 'original_timestamp': 1762277868.223091, 'sensor_building_id': 1, 'humidity': 83.05, 'message': 'Humidity 83.05% is outside the 20-80% normal range.'}
29 ALERT: Temperature 40.7°C exceeds 40°C threshold.
30 Details:
31 {
32     'alert_timestamp': 1762277869.358429, 'alert_type': 'HIGH_TEMPERATURE', 'original_timestamp': 1762277869.352246, 'sensor_building_id': 2, 'temperature': 40.6, 'message': 'Temperature 40.6 exceeds 40°C threshold.'}
33 ALERT: Humidity 18.06% is outside the 20-80% normal range.
34 Details:
35 {
36     'alert_timestamp': 1762277869.363459, 'alert_type': 'HIGH_HUMIDITY', 'original_timestamp': 1762277869.352246, 'sensor_building_id': 2, 'humidity': 18.06, 'message': 'Humidity 18.06% is outside the 20-80% normal range.'}
37 ALERT: Humidity 84.15% is outside the 20-80% normal range.
38 Details:
39 {
40     'alert_timestamp': 1762277869.390534, 'alert_type': 'HIGH_HUMIDITY', 'original_timestamp': 1762277869.383076, 'sensor_building_id': 2, 'humidity': 84.15, 'message': 'Humidity 84.15% is outside the 20-80% normal range.'}
```