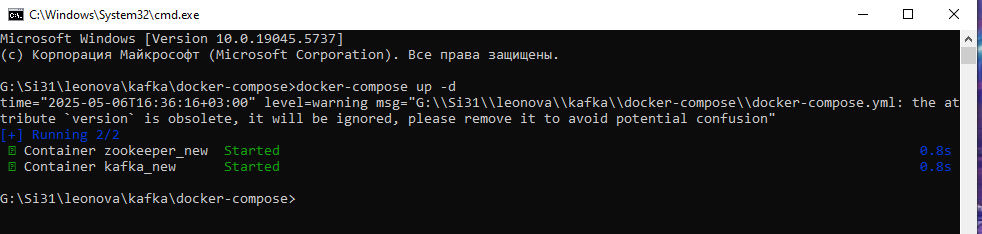
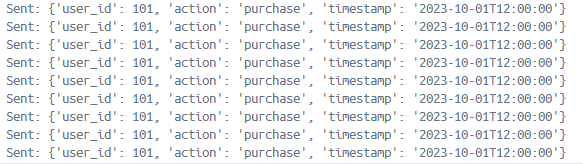
**ЛАБОРАТОРАНЯ РАБОТА**

РАЗРАБОТКА СИСТЕМЫ ОБРАБОТКИ СООБЩЕНИЙ С ИСПОЛЬЗОВАНИЕМ APACHE KAFKA

Запуск контейнеров:



Отправка сообщений. Команда python kafka-producer.py



\* kafka-producer.py-название созданного файла

Листинг программы:

from kafka import KafkaProducer

import json

import time

producer = KafkaProducer(

    bootstrap\_servers='localhost:9092',

    value\_serializer=lambda v: json.dumps(v).encode('utf-8')

)

while True:

    user\_action = {

        "user\_id": 101,

        "action": "purchase",

        "timestamp": "2023-10-01T12:00:00"

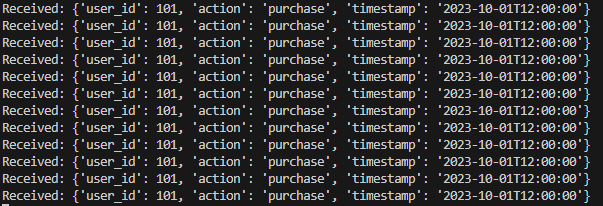
    }

    producer.send('user\_actions', user\_action)

    print(f"Sent: {user\_action}")

    time.sleep(1)  # Отправляем сообщение каждую секунду

Отображение полученных сообщений. Команда python kafka- consumer.py



\* kafka- consumer.py- название созданного файла

Листинг программы:

from kafka import KafkaConsumer

import json

consumer = KafkaConsumer(

    'user\_actions',

    bootstrap\_servers='localhost:9092',

    group\_id='user\_actions\_group',

    value\_deserializer=lambda x: json.loads(x.decode('utf-8'))

)

action\_count = {}

total\_messages = 0

for message in consumer:

    action = message.value['action']

    total\_messages += 1

    if action == "purchase":

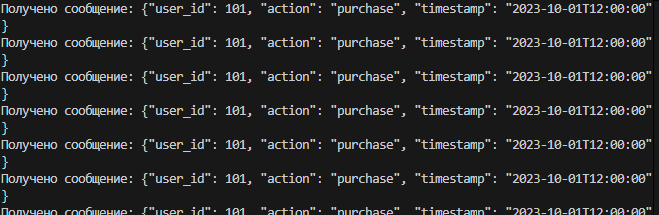
        action\_count[action] = action\_count.get(action, 0) + 1

    print(f"Received: {message.value}")

print(f"Total messages processed: {total\_messages}")

print(f"Action counts: {action\_count}")

Отображение ошибок. Команда python consumer2.py



\* consumer2.py- название созданного файла

Листинг программы:

from kafka import KafkaConsumer, KafkaProducer

producer = KafkaProducer(

    bootstrap\_servers='localhost:9092',

    value\_serializer=lambda x: x.encode('utf-8')

)

def process\_message(message):

    try:

        print(f"Получено сообщение: {message.value.decode('utf-8')}")

    except Exception as e:

        print(f"Ошибка обработки сообщения: {e}")

        producer.send('user\_actions\_dlt', value=message.value.decode('utf-8'))  # Отправка в DLT

consumer = KafkaConsumer(

    'user\_actions',

    bootstrap\_servers='localhost:9092',

    group\_id='user\_actions\_group',

    auto\_offset\_reset='earliest'

)

for message in consumer:

    process\_message(message)

