

Amaliy ish № 9

Oqimli ma'lumotlarni qayta ishlash

Ishning maqsadi:

Real vaqtda ma'lumotlarni qayta ishlash uchun Apache Kafka yordamida oqimli ma'lumotlar bilan ishlashni o'rganish.

Nazariy qism:

1. Oqimli ma'lumotlar nima?

- Oqimli ma'lumotlar real vaqt rejimida hosil bo'ladigan doimiy ma'lumotlar oqimidir. Misollar:
 - Veb-server jurnallari.
 - IoT sensorlaridan olingan ma'lumotlar.
 - Tvitlar yoki ijtimoiy media xabarlar.

2. Apache Kafka nima?

- Apache Kafka - oqimli ma'lumotlarni qayta ishlash uchun taqsimlangan platforma. Bu imkon beradi:
 - Turli manbalardan ma'lumotlarni to'plansh.
 - Ularni navbatlarda saqlash (topics).
 - Ma'lumotlarni real vaqtda qayta ishlash.
- Kafka quyidagilar uchun ishlatiladi:
 - Ma'lumotlar konveyerini qurish.
 - Real vaqtda ma'lumotlarni tahlil qilish.
 - Turli xil tizimlar o'rtasidagi integratsiya.

3. Kafkaning asosiy tarkibiy qismlari:

- *Producer*: Kafkaga ma'lumotlarni yuboradi.
- *Consumer*: Kafkadan ma'lumotlarini o'qiydi.
- *Topic*: Ma'lumotlar saqlanadigan navbat.
- *Broker*: Ma'lumotlarni boshqaradigan server.

4. Kafkani qanday o'rnatish kerak?

- Kafkani o'rnatish uchun uni rasmiy veb-saytdan yuklab oling: [Kafka Downloads](#).

- Shu bilan bir qatorda, Docker-dan foydalanish mumkin:

1. `docker run -d --name zookeeper -p 2181:2181 wurstmeister/zookeeper`
2. `docker run -d --name kafka -p 9092:9092 --link zookeeper:zookeeper -e KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181 -e KAFKA_ADVERTISED_HOST_NAME=localhost wurstmeister/kafka`

5. Python orqali Kafka bilan qanday ishlash mumkin?

Pythonda Kafka bilan ishlash uchun `kafka-python` yoki `confluent-kafka` kutubxonalardan foydalaniladi. Ularni buyruq bilan o'rnatish:

```
pip install kafka-python
```

Amaliy qism:

1. Kafka o'rnatilishi:

- Agar siz Docker-dan foydalansangiz, Zookeeper va Kafka-ni ishga tushiring:

1. `docker run -d --name zookeeper -p 2181:2181 wurstmeister/zookeeper`
2. `docker run -d --name kafka -p 9092:9092 --link zookeeper:zookeeper -e KAFKA_ZOOKEEPER_CONNECT=zookeeper:2181 -e KAFKA_ADVERTISED_HOST_NAME=localhost wurstmeister/kafka`

- Yoki rasmiy veb-saytdan o'rnatish.

2. Ma'lumotlarni tayyorlash:

- Biz Kafkaga oddiy matnli xabarlar yuborish orqali oqim ma'lumotlarini simulyatsiya qilamiz.

3. Ma'lumotlar bilan ishlash:

- Ikki Python faylini yarating: `producer.py` (ma'lumotlarni yuborish uchun) va `consumer.py` (ma'lumotlarni o'qish uchun).

Fayl `producer.py`:

```
from kafka import KafkaProducer
import time
```

```
# Kafkaga ulanish
```

```
producer = KafkaProducer(bootstrap_servers='localhost:9092')
```

```
# Xabarlarini yuborish
```

```
for i in range(10):
```

```
    message = f"Message {i}"
```

```
producer.send('test_topic', value=message.encode('utf-8'))
print(f"Sent: {message}")
time.sleep(1)

producer.flush()
```

Fayl `consumer.py`:

```
from kafka import KafkaConsumer

# Kafkaga ulanish
consumer = KafkaConsumer(
    'test_topic',
    bootstrap_servers='localhost:9092',
    auto_offset_reset='earliest',
    enable_auto_commit=True,
    group_id='my-group'
)

# Xabarlarni o'qish
print("Waiting for messages...")
for message in consumer:
    print(f"Received: {message.value.decode('utf-8')}")
```

4. Dasturni ishga tushirish:

- Kafka Producer-ni ishga tushirish:
`python producer.py`
- Alohida terminalda Kafka Consumer-ni ishga tushiring:
`python consumer.py`
- Real vaqtda yuborilgan va o'qilayotgan xabarlarni ko'rasiz.

Qo'shimcha materiallar:

- Rasmiy Kafka hujjatlari: [Kafka hujjatlari](#).
- Kafka-Python hujjatlari: [Kafka-Python hujjatlari](#).
- Kafka qo'llanmasi: [Kafka darslik](#).

Amaliy vazifa:

Vazifa: № 8

Apache Kafka yordamida quyidagilarni bajaring:

1. Yangi topic yarating `sensor_data`.
2. Har soniyada mavzuga tasodifiy raqamlar (masalan, harorat) yuboradigan *Producer* yozing.
3. Mavzudan ma'lumotlarni o'qiydigan va uni ekranda ko'rsatadigan *Consumer* yozing.