

Kafka 수동커밋

Kafka 각 Partition 에는 컨슈머가 처리한 메시지의 위치정보가 관리되는데 이를 OffSet 이라고 하며, 이 OffSet 의 Kafka 자동관리 여부에 따라 Auto Commit, Manual Commit 으로 구분된다.

10분 이내 완료

INTO THE LAB

Instruction

Kafka 수동 커밋

Kafka 커밋모드

- Kafka default 커밋모드는 autoCommit 이다.
- Kafka 커밋모드가 auto(default) 일 때 Partition 이 증가해 Rebalancing 이 발생하면 커밋되지 않은 Message 들은 자칫 컨슈머가 다시 Subscribe 하여 중복처리할 수도 있다.

Kafka 커밋모드 변경

- autoCommit 설정을 false 로 변경하여 수동커밋 모드로 변경한다.
- Product 마이크로서비스 application.yml 파일의 cloud.stream.kafka 하위의 설정을 주석해제하고 저장한다.

```
bindings:
  event-in:
    consumer:
      autoCommitOffset: false
```

- Order 와 Product 마이크로서비스를 기동한다.

```
cd order
mvn spring-boot:run
cd product
mvn spring-boot:run
```

현재, kafkatest 토픽의 파티션이 2 개이므로, 실행한 Product 서비스 Console 에 2 개의 파티션이 할당되었음을 볼 수 있다.

partitions assigned: [kafkatest-0, kafkatest-1]

Lag 확인

- Order 서비스에 포스팅하여 Kafka Event 를 발행한다.

```
http POST :8081/orders message=1st-Order
http POST :8081/orders message=2nd-Order
```

- Product 마이크로서비스는 메시지를 소모(처리)했음에도 불구하고 Partition 에서의 OffSet 이 증가하지 않아 Lagging 이 발생하고 있다.
- Partition Lagging 확인

```
$kafka_home/bin/kafka-consumer-groups.sh --bootstrap-server
localhost:9092 --group product --describe
```

Manual Commit

- Product 서비스에서 수동으로 ACK 를 날려 Manual Commit 을 해준다.
- Product 서비스의 PolicyHandler.java 에서 아래 메서드의 불럭주석을 해제하고 기존 메서드를 불럭주석 처리한다.

```
@StreamListener(KafkaProcessor.INPUT)
public void wheneverOrderPlaced_PrintMessage(@Payload
OrderPlaced orderPlaced, @Header(KafkaHeaders.ACKNOWLEDGMENT)
Acknowledgment acknowledgment) {

    System.out.println("Entering listener: " +
orderPlaced.getId());
    System.out.println("Entering listener: " +
orderPlaced.getMessage());

    acknowledgment.acknowledge();

}
```

- Product 마이크로서비스를 재시작한다.
- Console 로그를 조회하면 메시지가 재처리한 것이 확인된다.
- Order 서비스에 포스팅하여 Kafka Event 를 추가 발행한다.

```
http POST :8081/orders message=3rd-Order
http POST :8081/orders message=4th-Order
```

Lag 확인

- - Partition Lagging 을 재확인하면 이제는 Lagging 이 확인되지 않는다.

```
$kafka_home/bin/kafka-consumer-groups.sh --bootstrap-server
localhost:9092 --group product --describe
```

CheckPoints

1. 모든 요구사항을 만족하는가 ☐