

* Messaging의 정의와 필요성

- 메시징은 애플리케이션(서비스)간의 통신을 목적으로 한 매개체인 메시지를 주고 받는 것을 의미함

서비스 제공을 위해 서로 통신을 해야함

- 주문을 담당하는 애플리케이션과 배송을 담당하는 애플리케이션이 각각 있다면 원활한 서비스 제공을 위해 서로 통신을 해야함
- 주문 애플리케이션과 배송 애플리케이션이 서로 주고 받는 데이터가 바로 메시지
- 주문 애플리케이션과 배송 애플리케이션이 직접 메시지를 주고 받는 것을 동기식 통신이라 부름
- 주문 애플리케이션과 배송 애플리케이션 사이에 매개체(Queue)를 두고 메시지를 주고 받는 것을 비동기식 통신이라 부름
- 주문 / 배송 애플리케이션의 데이터 처리 스펙이 다르다면 통신에 문제가 생길 수 있음, 한쪽이 메시지를 처리하는 동안 다른 한쪽은 그 메시지를 처리하지 못하고 유실할 것임
- 각각의 애플리케이션이 직접 통신하는 것을 방지하고(Decouple) 매개체(Queue)를 통해 메시지를 주고 받는 것이 이러한 문제에 대한 대안이 될 수 있음
- SQS, SNS, Kinesis가 이러한 분리 매개체에 해당함
 - SQS : Queue Model
 - SNS : Publish/Subscribe Model
 - Kinesis : Real-time Streaming Model

* Simple Queue Service(SQS)란?

- 애플리케이션간의 직접적 통신이 아닌 Queue(대기열, 큐)를 두어 비동기식 메시징 시스템을 제공하는 서비스
- 메시지 생산자가 생성한 메시지(서비스 요청)를 Queue에 저장하면 메시지 소비자가 이를 소비(수집, Polling)할 수 있음
- AWS 초창기 서비스 중의 하나, 2006년 런칭 시작
- 처리량에 제한이 없고 대기열에 있는 메시지 수에 제한이 없는 특징을 지님
- 각 메시지는 최대 256KB의 텍스트로 구성될 수 있음
- 메시지가 대기열에 남아 있는 기간은 기본 4일에서 최대 14일
- 처리되지 않은 서비스는 다시 Queue에 보존되며 다른 소비자가 열람할 수 있음
- 'Polling' 기반 서비스

* SQS의 동작 과정

1. 메시지 생성(송신) 과정 : 애플리케이션 1(생산자)이 애플리케이션 2(소비자)로 메시지를 보내는 경우, 애플리케이션 1이 SDK(SendMessage API)를 활용해 메시지를 큐에 저장함
2. 메시지 소비(수신) 과정 : 애플리케이션 2(소비자)가 Queue를 열어 자신에게 온 메시지를 확인하고 이를 수행함
3. 메시지 소비(수신) 과정 : 애플리케이션 2가 SDK(DeleteMessage API)를 활용해 수행한 메시지를 삭제함

* Queue(대기열) 유형

- Standard 대기열과 FIFO 대기열로 나뉨
- Standard 대기열 : 표준 서비스로 초당 무제한에 가까운 요청을 처리할 수 있으며 최소 한 번의 메시지를 송신하나 순서가 보장되지 않음
- FIFO 대기열 : 선입선출을 지키는 대기열로 표준 대기열보다 순서가 더 보장되며 송신된 메시지 순으로 수신할 수 있으며, 초당 300개까지 처리 가능하며 중복이 발생하지 않음

* 중복 전송과 메시지 가시성 시간 제한

- 메시지는 적어도 한 번 전송될 수 있음
 - 메시지를 열어볼 수 있는 소비자는 여러 서비스가 될 수 있기 때문에 한 소비자가 메시지 처리가 늦다면 삭제되지 않고 남게 됨
 - 삭제되지 않은 메시지를 다른 소비자가 열어보고 처리할 수 있음
 - 메시지가 소비자에게 중복 전송되는 결과를 낳음
- Message Visibility Timeout(메시지 가시성 시간 제한)
 - 소비자가 메시지를 받고 처리하는 동안 해당 메시지가 다른 소비자에게 보이지 않는 특성, 기본 30초
 - 만일 소비자가 메시지를 받고 가시성 시간 제한 내에 메시지를 처리하지 못 한다면 메시지가 Queue에 남고 다른 소비자가 볼 수 있게 되어 중복 전송됨
 - 이를 방지하기 위해 소비자는 ChangeMessageVisibility API를 활용하여 메시지 가시성 시간 제한을 늘릴 수 있음

* Long Polling(긴 폴링)

- Queue가 비어있을 때, 소비자가 메시지가 올 때까지 기다리는 것
- 최대 20초까지 설정 가능
- 설정 이유
 - Queue에서 대기하며 메시지를 즉각 수신하여 지연시간을 줄임

- API 호출 횟수를 줄여 효율성을 높임

* 메시지 암호화

- HTTPS API 를 이용해 전송 중인 메시지를 암호화할 수 있고, KMS Key 를 통해 대기열 메시지를 암호화할 수 있음
- IAM Policy 를 이용해 SQS API 로의 접근을 통제할 수 있음