

# AWS 아키텍처 설계

## Chapter 04. Database

# 설계 시나리오

AWS 환경에서 서비스 환경을 구성 중인 조직에서 가용성과 확장성이 높은 데이터베이스가 필요한 상황이다.

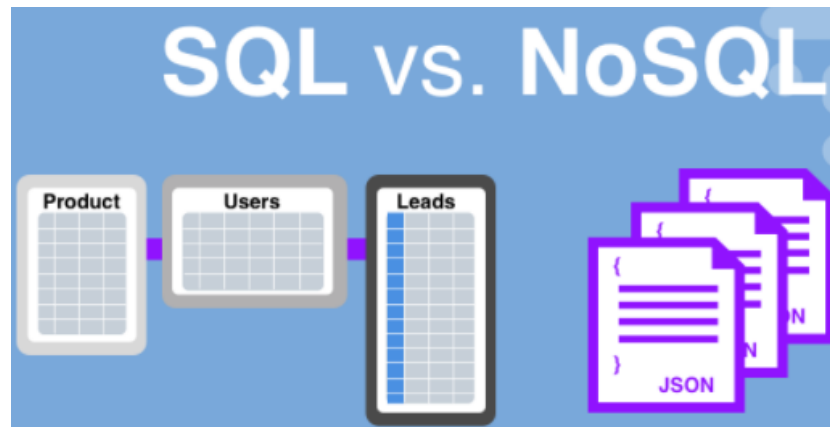
AWS 클라우드 환경에서 이러한 요구 사항에 맞게 데이터베이스를 구성하는 방법들은 무엇이 있을까?

01

# 데이터베이스 선택

# 데이터베이스 종류 선택

- 애플리케이션의 가용성과 성능은 데이터베이스 선택과 구성에 따라 다르다.
- 데이터베이스는 **관계형(SQL)**과 **비관계형(NoSQL)**으로 구분.
- 요청되는 **처리량**, **용량 확장 방식**, **데이터 형식**, **요구되는 내구성** 등에 따라 선택.



# 데이터베이스 종류 선택

## 관계형 데이터베이스

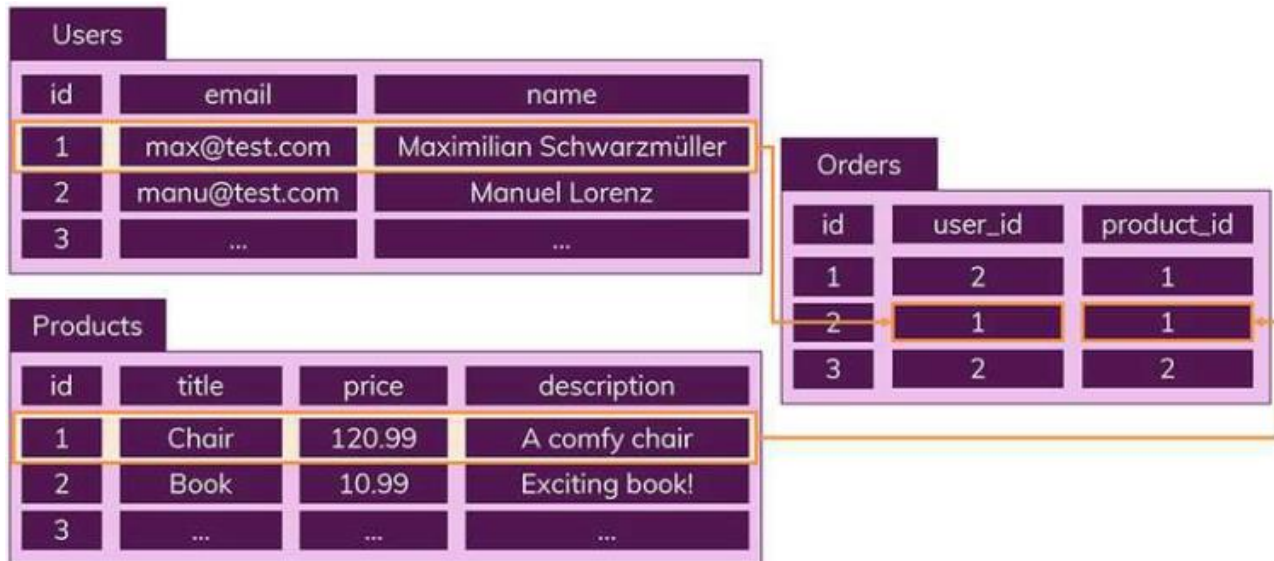


## 비관계형 데이터베이스



# 관계형 데이터베이스

- 관계형 데이터베이스는 하나 이상의 테이블을 포함하고 있으며 열과 행이 있는 스프레드시트로 시각화.
- 테이블에 데이터를 추가하기 전 **열의 이름**과 입력될 **데이터 형식**을 사전에 정의.
- 엄격한 스키마 규칙.



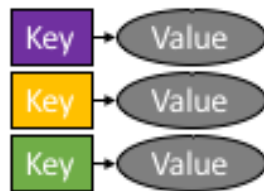
# 비관계형 데이터베이스



Column



Graph



Key-Value



Document

출처: <https://openclassrooms.com/>

# 관계형/비관계형 차이점

	관계형(SQL)	비관계형(NoSQL)
데이터 저장	테이블 – 행과 열	Key-value, Document 등
스키마	엄격한 스키마	동적
쿼리	SQL	문서 수집 집중
용량 확장	수직 확장	수평 확장





# Non-Managed Database(비관리형 데이터베이스)

고객 관리

App 최적화
규모 조정
고가용성 구성
데이터베이스 백업
DB S/W 패치
DB S/W 설치
OS 패치
OS 설치
서버 유지 관리
Rack / Stack
전원 / HVAC / Net

온프레미스 물리 서버에서  
직접 데이터베이스 구축

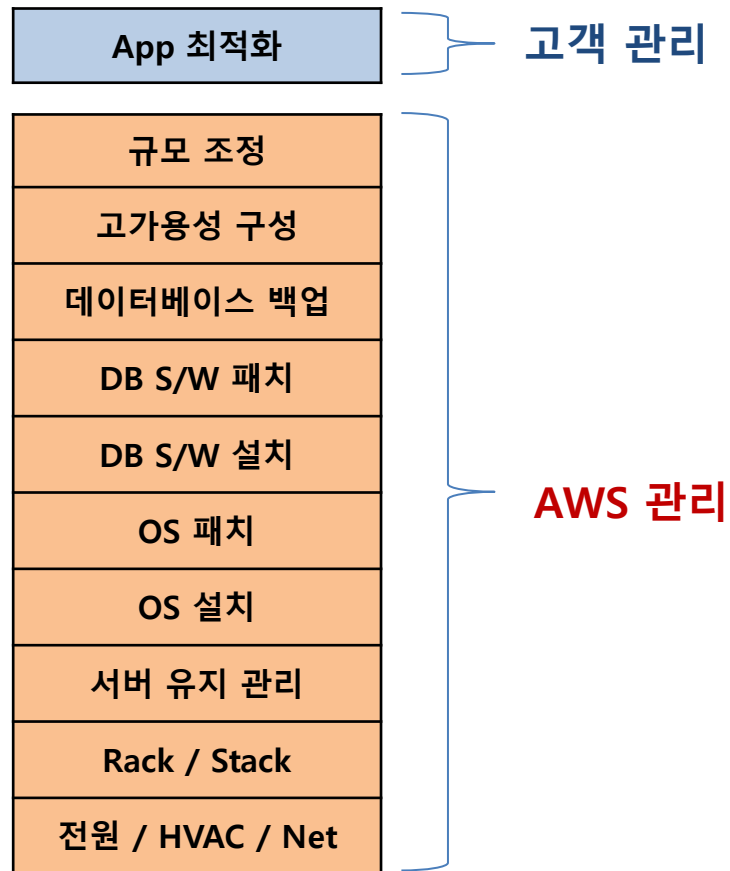
고객 관리

App 최적화
규모 조정
고가용성 구성
데이터베이스 백업
DB S/W 패치
DB S/W 설치
OS 패치
OS 설치
서버 유지 관리
Rack / Stack
전원 / HVAC / Net

AWS 관리

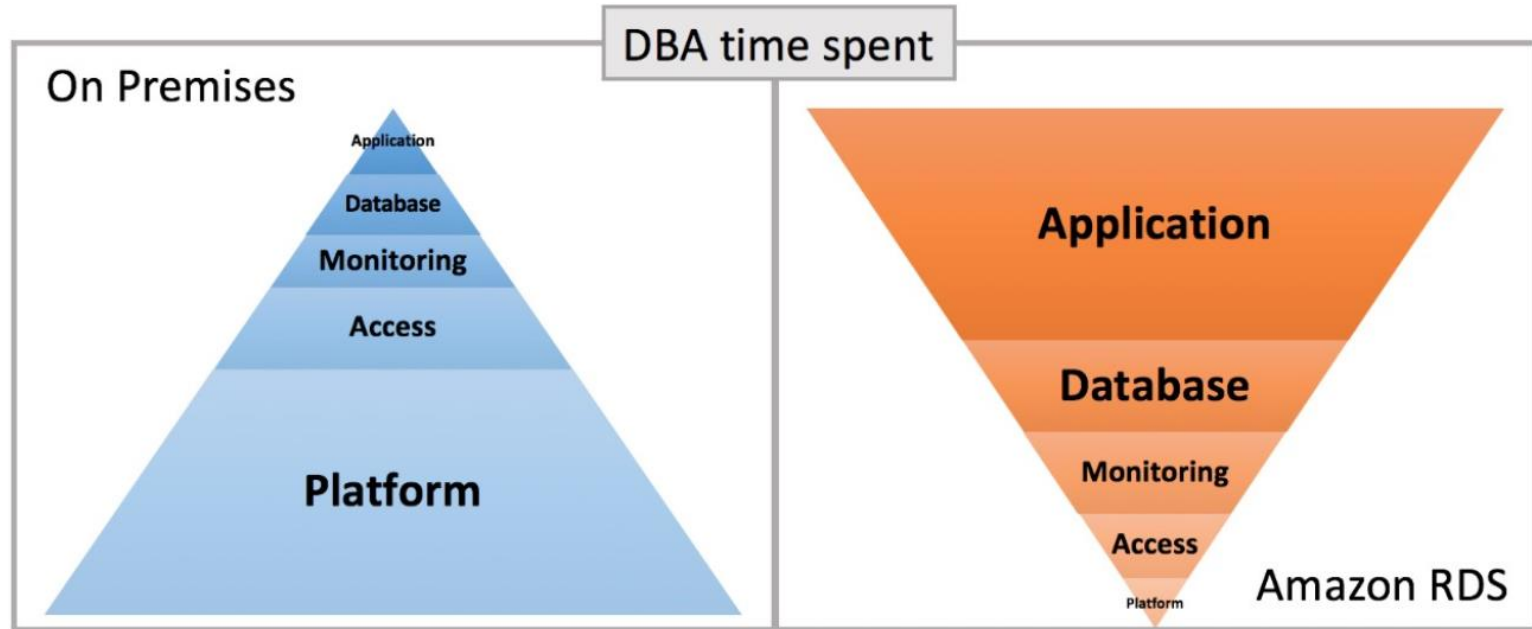
Amazon EC2에서 직접  
데이터베이스 구축

# Managed Database(관리형 데이터베이스)



**AWS 관리형 서비스로**  
데이터베이스 구축

# Managed Database



출처: <https://aws.amazon.com/ko/blogs/korea/part-1-role-of-the-dba-when-moving-to-amazon-rds-responsibilities/>

# AWS 데이터베이스 서비스

- AWS는 다음과 같은 다양한 관리형 데이터베이스 서비스를 제공.

## ☰ 데이터베이스

RDS

DynamoDB

ElastiCache

Neptune

Amazon QLDB

Amazon DocumentDB

Amazon Keyspaces

Amazon Timestream

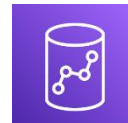
## <관계형 데이터베이스>



Amazon  
RDS

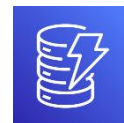


Amazon  
Aurora



Amazon  
Redshift

## <비관계형 데이터베이스>



Amazon  
DynamoDB



Amazon  
ElastiCache



Amazon  
DocumentDB

02

## Amazon RDS / Amazon DynamoDB

# Amazon RDS(Relational Database Service)

- AWS 클라우드에서 관계형 데이터베이스를 더 쉽게 설치, 운영 및 확장할 수 있는 서비스.
- 시간 소모적인 관리 작업을 자동화.
- 비용 효율적이고 크기 조정 가능한 용량 제공.
- 다음과 같은 데이터베이스 엔진을 지원.



Amazon RDS

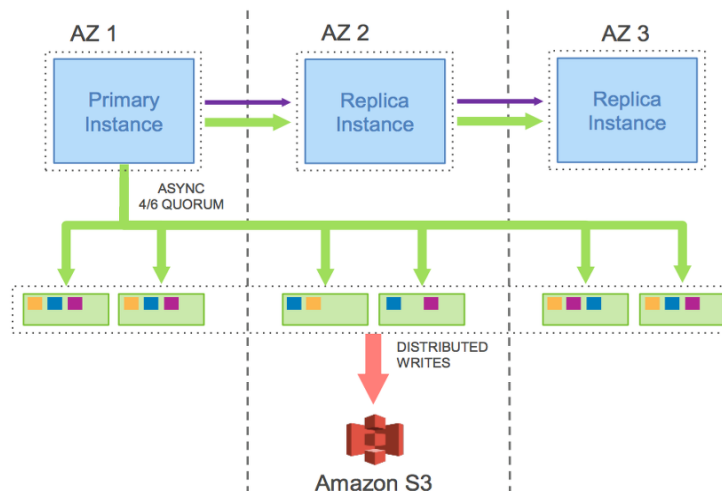


# Amazon Aurora

- 클라우드 용으로 구축된 MySQL 및 PostgreSQL 호환 관계형 데이터베이스.
  - 표준 MySQL 보다 최대 5배 / 표준 PostgreSQL보다 최대 3배 많은 처리량 제공.
- MySQL 및 PostgreSQL과 완벽 호환, 다른 변경 작업 없이 애플리케이션을 새로운 Aurora 인스턴스에 쉽게 연결 가능.
- 높은 가용성 및 안정성.
  - 3개 가용 영역에 6개의 데이터 사본을 복제, Amazon S3에 지속적으로 데이터를 백업.



Amazon Aurora



# Amazon RDS 보안 관리

- AWS IAM을 통해 Amazon RDS 리소스를 관리할 수 있는 관리자 제어.
- DB 엔진 보안 기능을 통해 데이터베이스 로그인 사용자 제어.
- 저장 데이터 및 전송 데이터 암호화.
- 보안 그룹을 통해 데이터베이스 접근 제어.

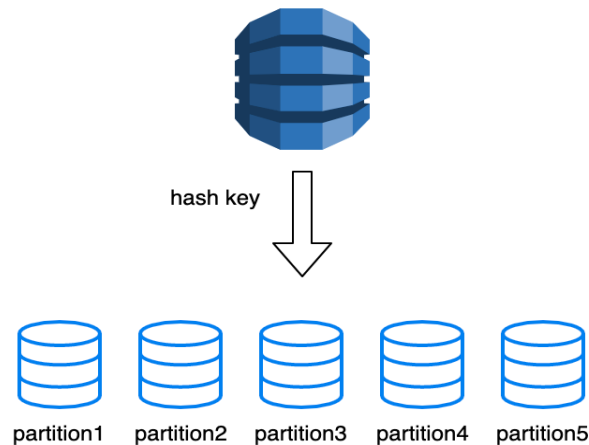


# Amazon DynamoDB

- 빠르고 일관된 응답 시간. (10밀리초 미만)
- 완전 관리형 비관계형 데이터베이스.
- 리전 별로 생성, 성능과 가용성을 위해 데이터를 여러 가용 영역에 자동 복제하여 저장.
- 서버리스 컴퓨팅.
- 데이터 용량과 요청이 증가하면 자동으로 테이블 파티셔닝을 통해 수평 확장.

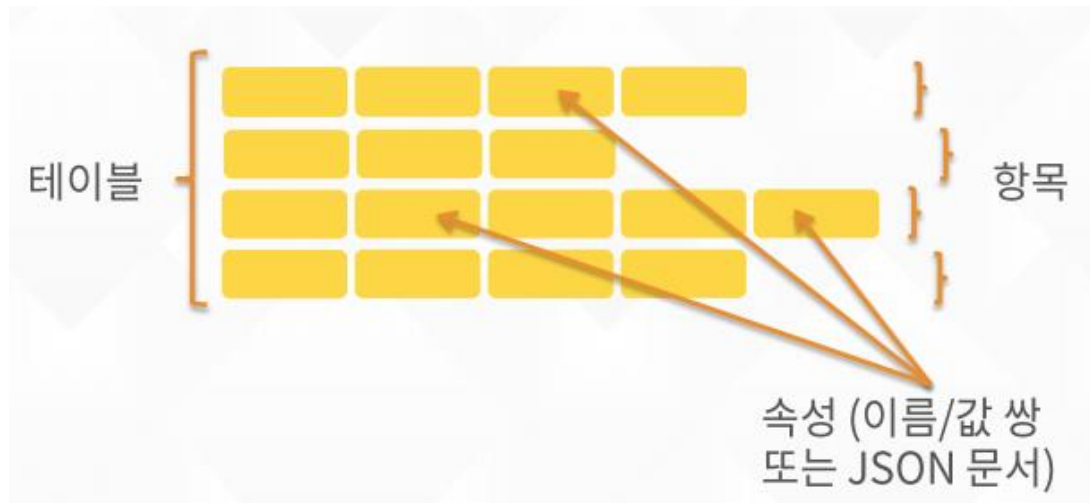


Amazon  
DynamoDB



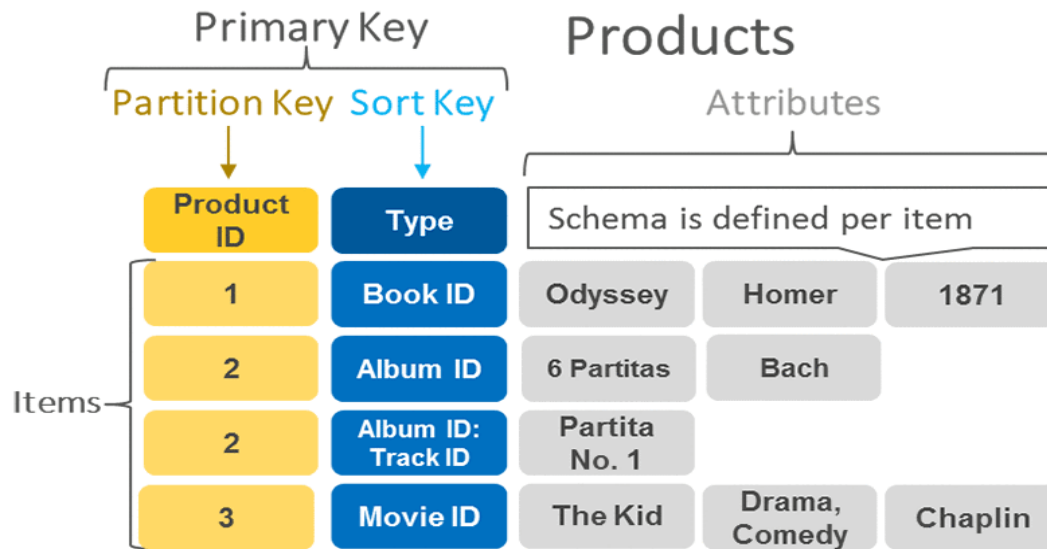
# Amazon DynamoDB

- Table은 Item(항목)들의 모음. Item(항목) 개수 제한은 없다.
- Item(항목)은 Attribute(속성)들의 모음. Attribute(속성) 개수 제한은 없다.



# Amazon DynamoDB

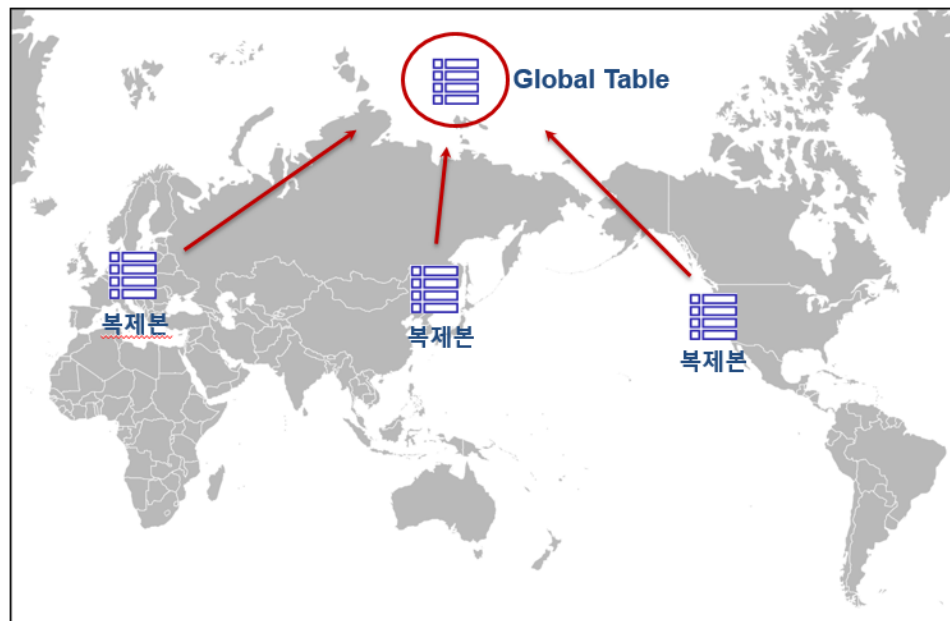
- NoSQL 방식으로 스키마는 필요 없지만, 각 데이터 식별을 위한 **Primary Key (기본키)**를 지정해야 한다.
  - **Partition Key** 방식 / **Partition Key + Sort Key** 방식.
  - **Partition** 안에서 **Sort Key**로 범위에 대해 빠른 질의 가능.



출처: <https://aws.amazon.com/ko/blogs/database/choosing-the-right-dynamodb-partition-key>

# Amazon DynamoDB 글로벌 테이블

- 대규모 확장이 필요한 글로벌 애플리케이션에 빠른 로컬 읽기 및 쓰기 성능을 지원.
- 애플리케이션이 특정 리전의 복제본 테이블에 데이터 작성시 다른 리전의 복제본 테이블로 해당 내용을 자동 복사.
- 리전 문제 발생시 애플리케이션의 가용성을 높게 유지.



# Amazon DynamoDB 용량

- DynamoDB는 선택한 옵션 기능과 함께 DynamoDB 테이블에서 데이터 읽기, 쓰기, 저장에 대해 요금을 부과.

DynamoDB > 테이블 > hello > 용량 편집

읽기/쓰기 용량 편집

용량 모드 정보

☐ 온디맨드  
애플리케이션이 수행하는 실제 읽기 및 쓰기에 대해 요금을 지불하는 방식으로 결제를 간소화합니다.

☒ 프로비저닝된  
읽기/쓰기 용량을 미리 할당하는 방식으로 요금을 관리하고 최적화합니다.

테이블 용량

읽기 용량

Auto Scaling [정보](#)  
실제 트래픽 패턴에 따라, 사용자를 대신하여 프로비저닝 처리 용량을 동적으로 조정합니다.

☒ 켜기  
☐ 끄기

최소 용량 단위  
1

최대 용량 단위  
10

목표 사용률(%)  
70

쓰기 용량

Auto Scaling [정보](#)  
실제 트래픽 패턴에 따라, 사용자를 대신하여 프로비저닝 처리 용량을 동적으로 조정합니다.

☒ 켜기  
☐ 끄기

최소 용량 단위  
1

최대 용량 단위  
10

목표 사용률(%)  
70

예상 읽기/쓰기 용량 비용

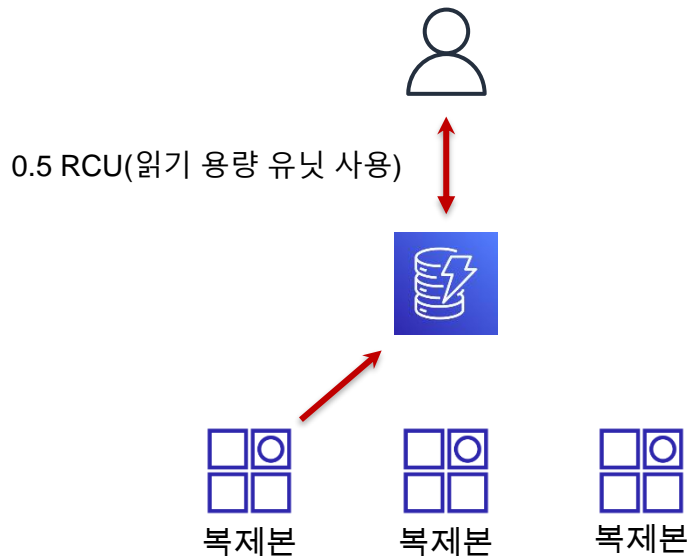
현재 설정에 따라, 테이블 및 인덱스에 대해 프로비저닝된 읽기 및 쓰기 용량의 총 예상 비용은 다음과 같습니다. 자세한 내용은 [프로비저닝된 용량에 대한 Amazon DynamoDB 요금](#) [문서](#)(를) 참조하세요.

총 읽기 용량 단위	총 쓰기 용량 단위	리전	추정 비용
5	5	us-east-1	US\$2.91/월

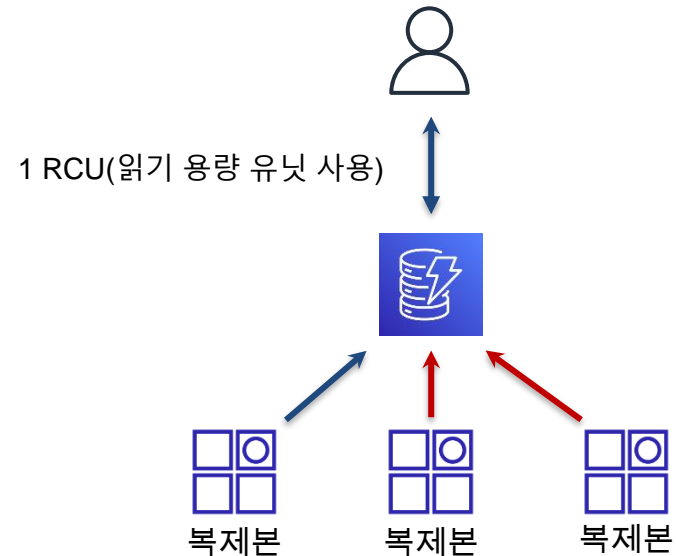
# Amazon DynamoDB 일관성

- DynamoDB는 여러 가용 영역에 데이터 복사본을 저장. 보통 1초 이내 모든 저장소 데이터가 일관성을 갖는다.
  - **최종 일관성 읽기**는 최근 완료된 쓰기 작업을 반영하지 않을 수도 있다. (기본 동작)
  - **강력한 일관성 읽기**는 가장 최신 데이터로 응답을 반환. (ConsistentRead 파라미터 true 설정)

## 최종 일관성 읽기 (Eventually Consistent Read)



## 강력한 일관성 읽기 (Strongly Consistent Read)



# Amazon DynamoDB 보안 관리

- AWS IAM을 통해 Amazon DynamoDB 리소스를 사용할 수 있는 사용자 제어.
- 저장 데이터 암호화.
- 전송 데이터 암호화.

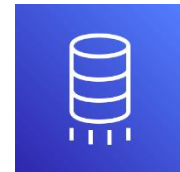
# 03

## 데이터베이스 마이그레이션



# Amazon DMS(Database Migration Service)

- 데이터베이스를 AWS로 빠르고 안전하게 마이그레이션할 수 있도록 지원.
- 동종 마이그레이션 뿐 아니라 이기종 데이터베이스 플랫폼 간의 마이그레이션도 지원.
  - AWS SCT(Schema Conversion Tool)를 통해 소스의 스키마와 코드를 대상 데이터베이스에 맞게 변환 후 마이그레이션.
- 일회성 마이그레이션 / 지속적 마이그레이션 지원.
- 데이터베이스 규모가 너무 크거나 네트워크가 느린 경우, 혹은 보안 규정 상 데이터 마이그레이션이 힘든 경우 발생.



AWS DMS



**AWS Snowball Edge**  
**사용**