



Firestore와 MongoDB 비교

MongoDB Atlas 무료 체험

Firestore와 MongoDB는 모두 애플리케이션 개발자를 위한 강력하고 유용하며 확장 가능한 최신 데이터베이스 플랫폼을 만들기 위해 노력하고 있습니다. 관리 형 서비스가 클라우드의 우위로 부상하면서 업계가 휩쓸면서 개발자의 기대치가 높아지고 있습니다. 모든 산업에서 새로운 세대의 소비자는 매우 유용하고 대폭 단순화 된 인터페이스와 완전 관리 형 셀프 서비스 경험을 요구하고 있으며 개발자도 마찬가지입니다. 새로운 수준의 개발자 생산성과 데이터 유틸리티를 열어주는 데이터베이스, 그리고 전체적으로 소프트웨어 개발 역사에서 흥미로운 시기입니다.

Firestore와 MongoDB의 차이점에 대해 이야기하기 전에 두 시스템 간의 유사점부터 시작하겠습니다. 둘 다 JSON과 유사한 문서 데이터 모델 및 스키마가 유사한 사후 관계형 데이터베이스이며, 둘 다 애플리케이션 개발 및 수평 확장 성을 위해 구축되었습니다. Firestore와 MongoDB 모두 개발자가 신속하게 시작하고 빌드 할 때 데이터 구조를 반복 할 수 있도록 하며, 둘 다 확장 가능한 데이터베이스 클러스터에 쉽게 배포 할 수 있도록 데이터를 '컬렉션'으로 분리합니다.

Firestore에는 실시간 데이터베이스와 Cloud Firestore라는 두 개의 데이터 저장소가 있으며, 각각은 애플리케이션 개발의 다른 부분에 최적화되어 있습니다.

실시간 데이터베이스는 데이터의 신속한 수집 및 처리를 위해 설계된 게시 / 구독 (pubsub) 시스템에 가깝지만 Cloud Firestore는 장기 데이터 저장 및 검색을 위해 설계되었습니다. Cloud Firestore는 비동기 처리 용으로 설계된 SQL과 유



Python 예 :

```
ref = db.reference('aliens')

snapshot = ref.order_by_child('number_of_eyes').equal_to(3).get()

aliens = [{key: value} for key, value in snapshot.items()]
```

MongoDB는 고성능 및 동급 최고의 보안으로 알려진 더욱 강력한 문서 데이터베이스이며 Firebase에 비해 몇 가지 장점이 있습니다. 예를 들어 MongoDB는 온 프레미스 또는 클라우드 (MongoDB Atlas 또는 자체 관리 클라우드 MongoDB 사용)에서 운영 할 수 있지만 Firebase는 순전히 클라우드 데이터베이스 서비스입니다.

MongoDB 쿼리 언어 (MQL)는 처음부터 이러한 데이터 저장소의 중심에있는 JSON과 유사한 문서 구조에 대한 고급 쿼리 및 업데이트를 수행하도록 설계되었습니다. MQL은 MongoDB 문서 및 API와 동일한 JSON 구문을 사용하며 매우 강력하고 복잡한 쿼리를 손으로 또는 프로그래밍 방식으로 쉽게 조합 할 수 있습니다.

Python 예 :

```
aliens = db['aliens'].find({'number_of_eyes': 3, 'number_of_toes': {'$gt': 10}})
```

Firebase와 MongoDB는 모두 유연성과 시장 출시 속도를 허용하는 최신 사후 관계형 데이터베이스이며 Firebase는 소규모 애플리케이션에 더 많이 사용되며 MongoDB는 빅 데이터 및 고성능 사용 사례에 더 많이 사용됩니다.



솔루션보다 관계형 선구자와 유사합니다.

예를 들어 MongoDB는 ACID 트랜잭션, 스키마 유효성 검사, 심지어 교차 컬렉션 조인까지 지원하며, 기존에 Oracle 또는 MySQL과 같은 관계형 데이터베이스의 도메인이었던 것과 유사한 워크로드를 처리하는 동시에 스케일 우선 아키텍처 및 구조를 제공합니다. NoSQL 솔루션을 대표하는 유연성.

Firebase 대 MongoDB Atlas

MongoDB 팀의 서비스로서의 플래그십 데이터베이스인 [MongoDB Atlas](#)에 대해 이야기하지 않고 Google의 Firebase와 MongoDB를 비교하는 것은 공정하지 않습니다. Firebase는 데이터베이스이지만 특히 호스팅, 인증, 데이터 기반 트리거 및 분석을 포함하는 모바일 앱 개발자를 위해 설계된 애플리케이션 개발 플랫폼이기 때문입니다. MongoDB Atlas는 자동화 된 작업, 호스팅, 백업, 트리거 등을 통해 MongoDB를 완전 관리 형 데이터베이스 세계로 가져옵니다.

Firebase는 모바일 애플리케이션 개발을 위해 명시 적으로 설계되었으며 전체 사용자 인터페이스와 온 보딩 흐름이 이러한 사용 사례를 중심으로 구축되었습니다. 따라서 Firebase에는 하나의 Firebase 인스턴스를 하나의 모바일 앱에 연결하고 동일한 콘솔에서 애플리케이션 사용량을 모니터링하는 등 모바일 애플리케이션을위한 더 많은 기능이 있으며 MongoDB Atlas는 범용 데이터 개발을 위해 구축되었으며 더 많은 버튼과 노브가 있습니다. 성능 조정, 타사 비즈니스 인텔리전스 도구와의 통합, Atlas Data Lake와 같은 대규모 데이터 설치를위한 고급 기능.

2019년 [Realm](#)을 인수 하고 Atlas 플랫폼에 대한 지속적인 투자를 통해 MongoDB는 Firebase의 모바일 개발 기능 세트와 일치 (또는 초과)하는 동시에 애플리케이션 개발 공간 외부에서 데이터를 사용할 수있는 더 많은 도구를 제공하고 있습니다. .

Firebase와 MongoDB 성능은 어떻게 비교되니까?



MongoDB가 더 높은 성능이라는 것은 의미가 있습니다. 실시간 애플리케이션과 빠르게 움직이는 데이터 파이프 라인을 염두에두고 관계형 저장소와 동일한 종류의 온라인 트랜잭션 처리를 처리하도록 설계되었습니다.

앱 개발에 더 좋은 것은 무엇입니까?

MongoDB와 Firebase는 모두 모바일 애플리케이션 개발을위한 환상적인 도구입니다. 서비스로서의 완전한 백엔드 (BEaaS)와 최소한의 노력을 원하는 경우 Firebase가 이상적이지만 MongoDB (특히 MongoDB Atlas)는 애플리케이션 개발을위한 가장 강력하고 유연한 플랫폼과 기능을 제공합니다. 동일한 시스템에서 트랜잭션 및웨어 하우스 스타일 워크로드를 모두 지원합니다.

Firebase와 MongoDB를 사용하는 경우

경험상 작은 프로젝트 나 전용 모바일 애플리케이션을 구축하고 있고 하나의 클라우드 제공 업체에 종속 되어도 괜찮다면 Firebase가 시작하기에 좋은 곳입니다. 보다 범용적인 데이터 솔루션, 대규모 성능 및 고급 쿼리를 찾고 있다면 MongoDB가 귀하의 요구 사항을 가장 잘 충족 할 것입니다. 중간에 있다면 [MongoDB Atlas](#)를 확인하세요!

MongoDB Atlas 시작하기

시요 카드언이 MongoDB이 와져 과리 혀 서비스르서이 데이터베이스르 르르르 시요해버세오

<https://www.mongodb.com/firebase-vs-mongodb#:~:text=MongoDB is a more robust,purely a cloud database service.>



자원

[NoSQL 데이터베이스 설명](#)

[MongoDB 아키텍처 가이드](#)

[MongoDB 엔터프라이즈 고급](#)

[MongoDB 아틀라스](#)

[MongoDB 영역](#)

[MongoDB 엔지니어링 블로그](#)

교육 및 지원

[과정 카탈로그보기](#)

[인증](#)

[MongoDB 매뉴얼](#)

[설치](#)

[지원 센터](#)



시작하기

인기 주제

MongoDB Atlas로 AWS에서 MongoDB 실행

MongoDB Atlas로 마이그레이션

클라우드 데이터베이스 란?

MongoDB Realm으로 REST API 구축

약

MongoDB, Inc.

지도

보도 자료

채용

투자자

법적 고지

개인 정보 보호 정책

보안 정보

신뢰 센터

사무실 위치



우리를 따라라

[페이스 북](#)

[Github](#)

[유튜브](#)

[트위터](#)

[LinkedIn](#)

[스택 오버플로](#)

[경력](#)

[영어](#)



© 2021 MongoDB, Inc.

Mongo, MongoDB, and the MongoDB leaf logo are registered trademarks of MongoDB, Inc.