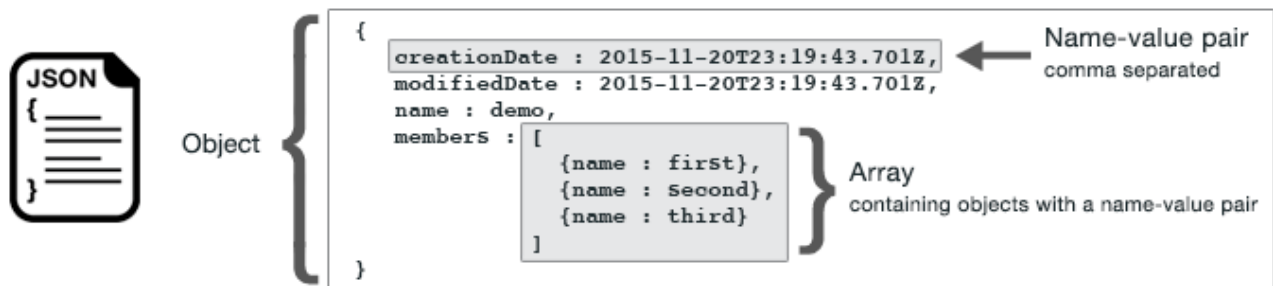- **Object**: a collection of name-value (or key-value) pairs. An object is defined within left ({) and right (}) braces. Each name-value pair begins with the name, followed by a colon, followed by the value. Name-value pairs are comma separated.

- **Array**: an ordered collection of values. An array is defined within left ([) and right (]) brackets. Items in the array are comma separated.

Below is a JSON example that contains an array of objects in which the objects represent different films in a streaming library. Each film is defined by two name-value pairs, one that specifies a unique value to identify that film and another that specifies a URL that points to the corresponding film's promotional image.
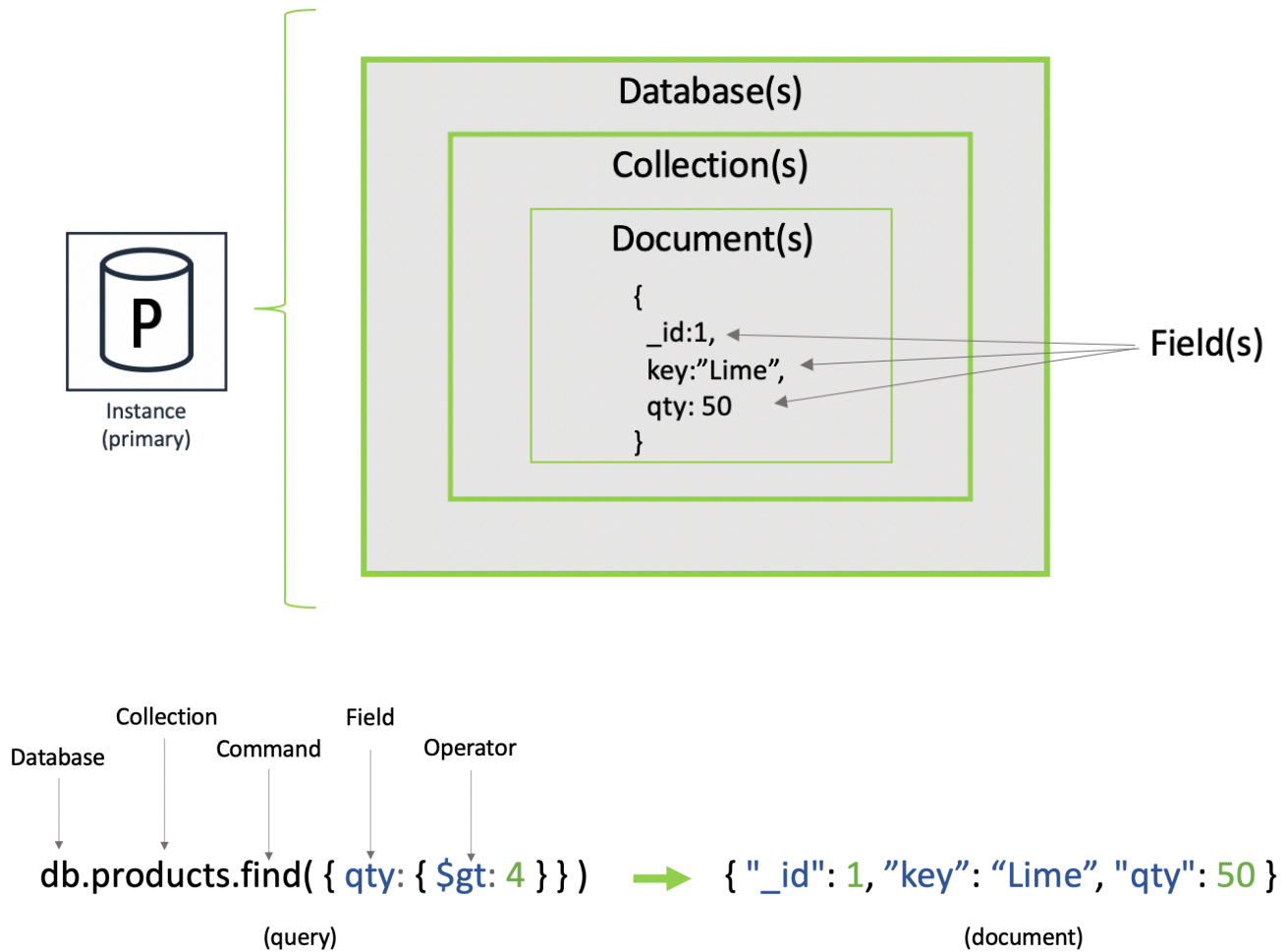
JSON

```
1   var films = [{"FilmID":"catmanbegins", "Image":"https://m.media-amaz
2     {"FilmID":"cabdriver", "Image":"https://m.media-amazon.com/images/
3     {"FilmID":"pulpnonfiction", "Image":"https://m.media-amazon.com/im
4     {"FilmID":"doctornormal", "Image":"https://m.media-amazon.com/imag
5     {"FilmID":"backtothepresent", "Image":"https://m.media-amazon.com/
```

Copy

What is a JSON document database?

A JSON document database is a type of nonrelational database that is designed to store and query data as JSON documents, rather than normalizing data across multiple tables, each with

document databases provide powerful and intuitive APIs for flexible and agile development.

[Amazon DocumentDB (with MongoDB compatibility)](https://aws.amazon.com/documentdb/what-is-json/) is a fast, scalable, highly available, and fully managed document database service that supports MongoDB workloads, that makes it easy to store, query, and index JSON data.
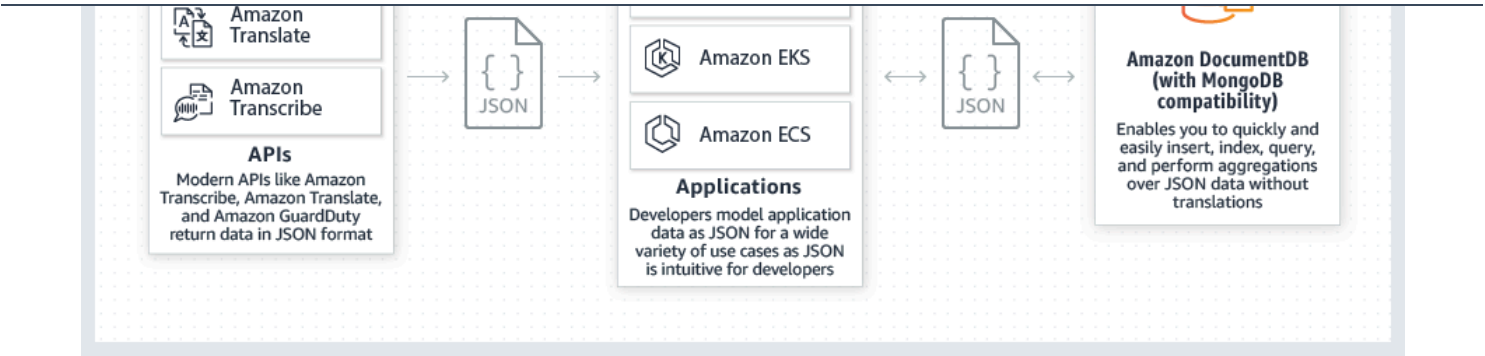
Amazon DocumentDB makes it easy to insert, query, index, and perform aggregations over JSON data.

Relational Database vs. JSON Document Database Terminology

The following table compares terminology used by JSON document databases with terminology used by relational databases using SQL.

Relational database (SQL)	JSON document database
Table	Collection
Row	Document
Column	Field
Primary key	ObjectID
Index	Index
View	View
Nested table or object	Embedded document
Array	Array

Content management

A JSON document database is a great choice for content management applications, such as blogs and video platforms, because each entity can be stored as a single JSON document. Should the data model need to change, only the affected documents need to be updated, with no need for schema updates and no database downtime required.

Catalogs

JSON document databases are efficient and effective for storing catalog information. For example, in an e-commerce app, different products usually have different numbers of attributes. These attributes can be described in a single JSON document for easy management and faster reading speed than would be possible with a relational database.

User profiles

JSON document databases are a good solution for online profiles in which different users provide different types of information. Using a JSON document database, you can store each user's profile efficiently by storing only the attributes that are specific to each user. JSON document databases easily manage this level of individuality and fluidity.

Real-time big data

Being able to extract operational information in real time is critical in a highly competitive business environment. By using JSON document databases, a business can store and manage operational data from any source and concurrently feed the data to the BI engine of choice for analysis, with no need to have two separate environments.

Customers managing JSON data with Amazon DocumentDB

"We, the Deutsche Fußball Liga GmbH (DFL) use AWS to power our publishing platform, delivering the latest news from the league to millions of fans worldwide. As our content and user base grew, we had a focus on maintaining our relational database's performance. We migrated to Amazon DocumentDB (with MongoDB compatibility) because of its flexible schema, native JSON support that enables our developers to deploy faster, and its decoupled architecture can scale our read throughput in minutes. We prototyped the application locally using the MongoDB Community Edition, and quickly deployed to production on Amazon DocumentDB. Everything worked just as we expected and we have successfully scaled our database performance."

Andreas Heyden, CEO DFL Digital Sports, EVP Digital Innovations - DFL Group

"FINRA's Data Collection platform stores millions of regulatory filings from hundreds of thousands customers, such as broker dealers, investment advisors, and stock exchanges. Our old platform was built using a relational database that stored data in XML, which had a rigid query structure and required us to write custom code for data versioning and schema validation. We chose Amazon DocumentDB because it natively stores data in JSON, making it simpler to query and index regulatory documents. This reduces our development cycles, while extending the usability of our data by easily integrating with other systems that leverage JSON. Because DocumentDB is a fully managed service, our databases are scalable, highly available, backed up, and encrypted without any overhead from our engineering teams."

Ranga Rajagopal, Senior Director, Enterprise Data Platforms - FINRA

[Read the case study »](#)

"At Habby, we use Amazon DocumentDB to store, query, and analyze both game player data and game operational data as well as data for promotional activities. Amazon DocumentDB is a natural and convenient choice for Habby as all of our game data is JSON and Amazon DocumentDB makes it easy to read and write that JSON data to the database. We chose Amazon DocumentDB because of the ease-of-management and its compatibility with MongoDB. We had initially planned to self-manage MongoDB on EC2 but found through testing that it was not convenient and required more work than we were willing to invest. With Amazon DocumentDB, our development team can scale, iterate, and upgrade games quickly, the marketing team can carry out high pertinence promotion activities, and our customer service team can troubleshoot problems from game players efficiently."

Shuxiang Zhao, Chief Technology Officer - HABBY PTE.LTD.

[Check out more Amazon DocumentDB customer stories](#)