

Table of Contents

MongoDB CRUD Introduction.....	2
Database Operations	3
Query	3
Data Modification	3

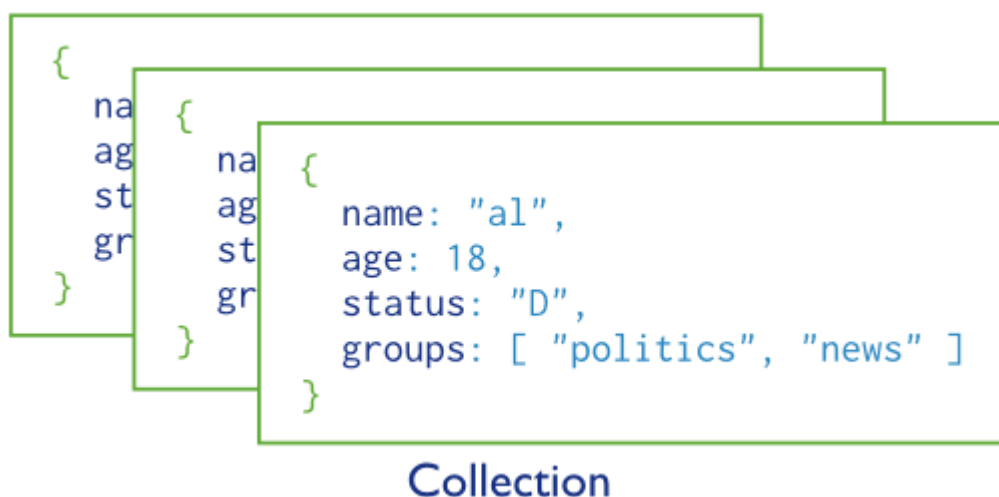
MongoDB CRUD Introduction

MongoDB stores data in the form of *documents*, which are JSON-like field and value pairs. Documents are analogous to structures in programming languages that associate keys with values (e.g. dictionaries, hashes, maps, and associative arrays). Formally, MongoDB documents are [BSON](#) documents. BSON is a binary representation of [JSON](#) with additional type information. In the documents, the value of a field can be any of the BSON data types, including other documents, arrays, and arrays of documents.

```
{  
  name: "sue",  
  age: 26,  
  status: "A",  
  groups: [ "news", "sports" ]  
}
```

← field: value
← field: value
← field: value
← field: value

MongoDB stores all documents in [collections](#). A collection is a group of related documents that have a set of shared common indexes. Collections are analogues to a table in relational databases.

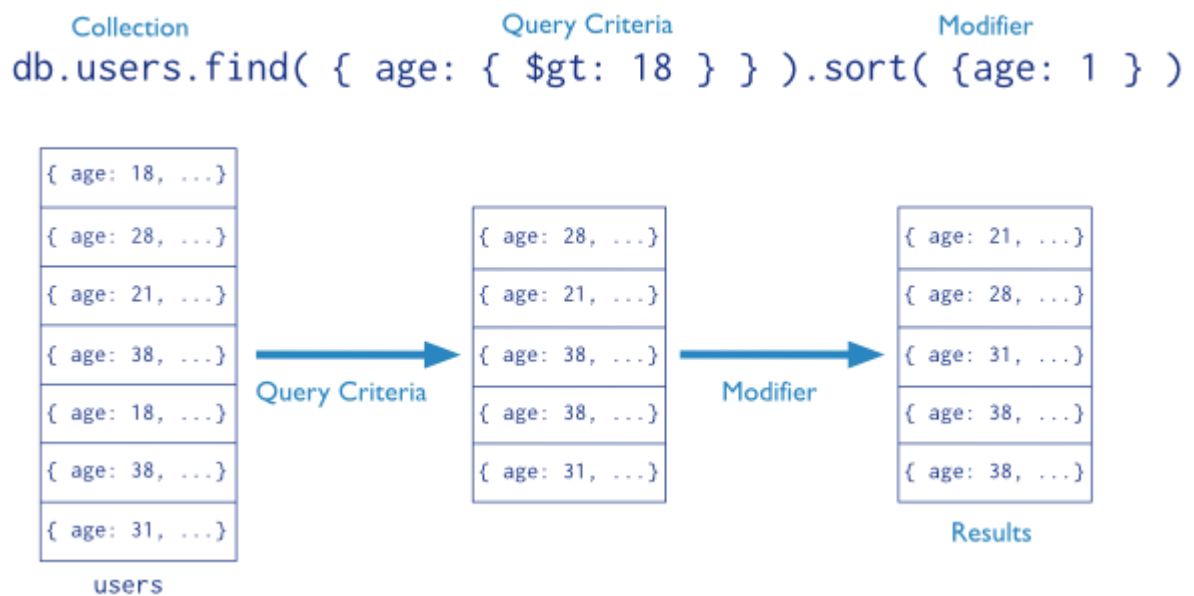


Database Operations

Query

In MongoDB a query targets a specific collection of documents. Queries specify criteria, or conditions, that identify the documents that MongoDB returns to the clients. A query may include a *projection* that specifies the fields from the matching documents to return. You can optionally modify queries to impose limits, skips, and sort orders.

In the following diagram, the query process specifies a query criteria and a sort modifier:



Data Modification

Data modification refers to operations that create, update, or delete data. In MongoDB, these operations modify the data of a single [collection](#). For the update and delete operations, you can specify the criteria to select the documents to update or remove.

In the following diagram, the insert operation adds a new document to the `users` collection.

