

Back-End Development



mongoDB

Exploration

Database

- A **database** is *an organized collection of data*.
- The **main purpose** of database is to operate large amount of information by storing, retrieving and managing.
- There are many dynamic websites on the world wide web nowadays which are handled through databases. For example, a model to checks the availability of rooms in a hotel. It is an example of dynamic website that uses database.



Non-RDBMS / NoSQL

- **NoSQL** or **Non-RDBMS** (*Non Relational Database Management Systems*) is database that is not modeled like relational model based.
- The data structure used by NoSQL databases are different from relational databases. It's faster and more flexible than relational db. That's why it's increasingly used in *big data* & *real time web application*.
- Here are the example of NoSQL database: MongoDB, Cassandra, CouchDB, OrientDB, IBM Domino, ArangoDB & Apache Ignite.



MongoDB

MongoDB is free & open source cross-platform document-oriented database. It's classified as NoSQL database & uses JSON-like documents with schemas.

MongoDB has been developed by MongoDB Inc. since 11th February 2009, and is published under GNU Affero General Public License & the Apache License.

MongoDB Ranking

5th All DB-engines

341 systems in ranking, March 2018


















Rank			DBMS	Database Model	Score		
Mar 2018	Feb 2018	Mar 2017			Mar 2018	Feb 2018	Mar 2017
1.	1.	1.	Oracle +	Relational DBMS	1289.61	-13.67	-109.89
2.	2.	2.	MySQL +	Relational DBMS	1228.87	-23.60	-147.21
3.	3.	3.	Microsoft SQL Server +	Relational DBMS	1104.79	-17.25	-102.70
4.	4.	4.	PostgreSQL +	Relational DBMS	399.35	+10.97	+41.71
5.	5.	5.	MongoDB +	Document store	340.52	+4.10	+13.59
6.	6.	6.	DB2 +	Relational DBMS	186.66	-3.31	+1.75
7.	7.	7.	Microsoft Access	Relational DBMS	131.95	+1.88	-0.99
8.	8.	↑ 10.	Redis +	Key-value store	131.22	+4.21	+18.22
9.	9.	↑ 11.	Elasticsearch +	Search engine	128.54	+3.23	+22.32
10.	10.	↓ 8.	Cassandra +	Wide column store	123.49	+0.71	-5.70

<https://db-engines.com/en/ranking>

MongoDB Ranking

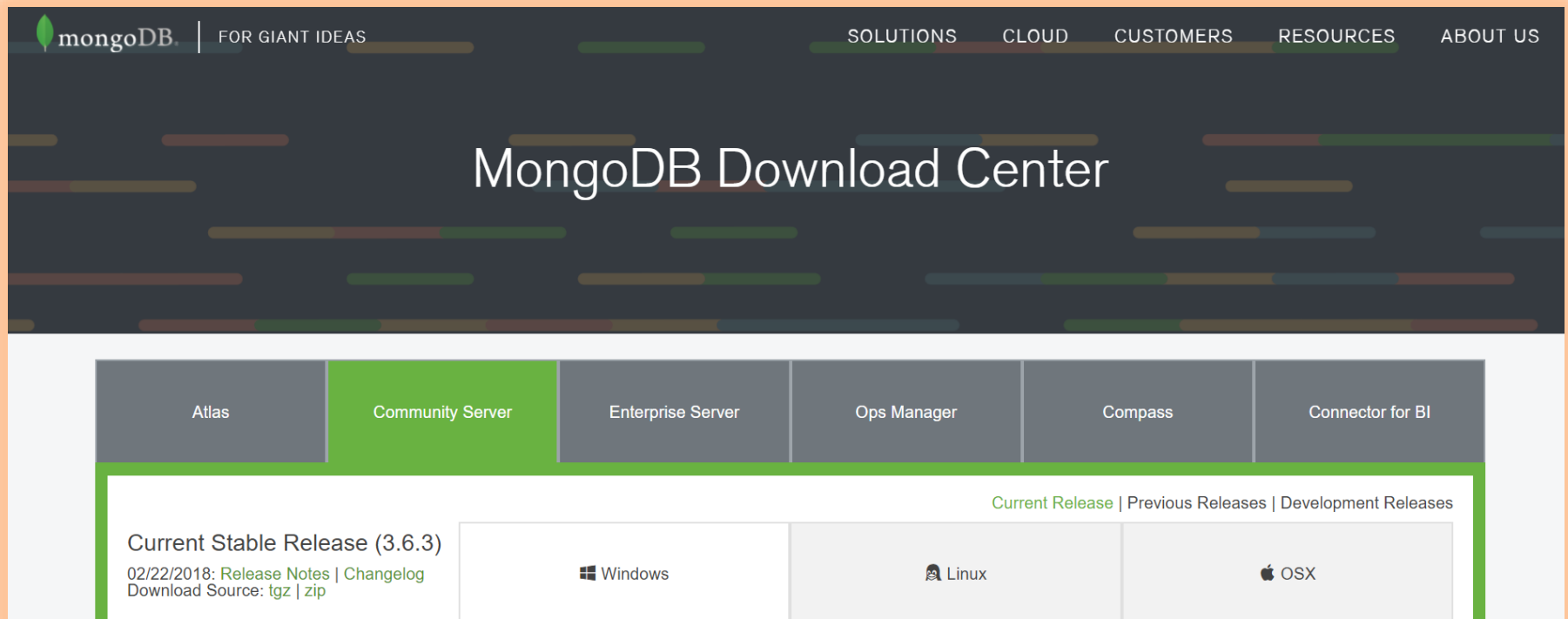
1st Doc Store DB-engines

44 systems in ranking, March 2018

Rank			DBMS	Database Model	Score		
Mar 2018	Feb 2018	Mar 2017			Mar 2018	Feb 2018	Mar 2017
1.	1.	1.	MongoDB 	Document store	340.52	+4.10	+13.59
2.	2.	2.	Amazon DynamoDB 	Multi-model 	42.46	+2.58	+11.33
3.	3.	3.	Couchbase 	Document store	32.90	+1.15	+2.86
4.	4.	4.	CouchDB	Document store	20.20	-0.09	-2.73
5.	5.	 9.	Microsoft Azure Cosmos DB 	Multi-model 	16.76	+0.57	+12.81
6.	6.	 5.	MarkLogic	Multi-model 	10.97	-0.05	-0.13
7.	 8.	 13.	Firebase Realtime Database	Document store	6.59	+0.75	+3.80
8.	 7.	 7.	OrientDB 	Multi-model 	6.46	+0.52	+1.12
9.	9.	 6.	RethinkDB	Document store	4.81	+0.12	-0.73
10.	10.	 8.	Cloudbant	Document store	4.30	+0.60	-0.69

<https://db-engines.com/en/ranking>

- Create ***data*** & ***db*** folder at ***C:/data/db***
- Download & install MongoDB



The screenshot shows the MongoDB Download Center website. The header includes the MongoDB logo and navigation links: FOR GIANT IDEAS, SOLUTIONS, CLOUD, CUSTOMERS, RESOURCES, and ABOUT US. The main heading is "MongoDB Download Center". Below this is a horizontal menu with tabs: Atlas, Community Server (highlighted in green), Enterprise Server, Ops Manager, Compass, and Connector for BI. Under the "Community Server" tab, there are links for "Current Release", "Previous Releases", and "Development Releases". The "Current Release" section displays "Current Stable Release (3.6.3)" with a date of "02/22/2018" and links to "Release Notes" and "Changelog". It also shows the "Download Source" as "tgz" and "zip". Below this, there are three columns for operating systems: Windows (with a Windows logo), Linux (with a Linux logo), and OSX (with an Apple logo).

Atlas	Community Server	Enterprise Server	Ops Manager	Compass	Connector for BI
	Current Release Previous Releases Development Releases				
	Current Stable Release (3.6.3) 02/22/2018: Release Notes Changelog Download Source: tgz zip	Windows	Linux	OSX	

- Open terminal:

```
$ cd C:\Program Files\MongoDB\Server\3.6\bin
$ mongod --dbpath C:\data\db
$ mongod
```

- If OK, it will show "Waiting for connections on port 27017". Then open new terminal:

```
$ cd C:\Program Files\MongoDB\Server\3.6\bin
$ mongo
```


How to Work With MongoDB Database

- Menampilkan daftar database:

```
$ show dbs
```

- Menampilkan db aktif (default: test):

```
$ db
```

- Membuat sekaligus menggunakan db “toko”:

```
$ use toko
```

- Hapus db “toko”:

```
$ use toko  
$ db.dropDatabase()
```

- Membuat db user:

```
$ db.createUser({user:"lintang",  
pwd:"1234", roles:["readWrite",  
"dbAdmin"]});
```

- Membuat collection “karyawan”:

```
$ db.createCollection('karyawan')
```

- Menampilkan daftar collection dalam db:

```
$ show collections
```

- Hapus collection “karyawan”:

```
$ db.karyawan.drop();
```

- Add 1 data JSON ke collection "karyawan":

```
$ db.karyawan.insert({nama:"Adi",usia:24});
```

- Add multiple data ke collection "karyawan":

```
$ db.karyawan.insert([  
  {nama:"Budi",usia:23},  
  {nama:"Caca",usia:25}]);
```

- Menampilkan data pada collection "karyawan":

```
$ db.karyawan.find();  
$ db.karyawan.find().pretty();  
$ db.karyawan.find({nama: "Adi"}).pretty();
```

- Update **semua** property data:

```
$ db.karyawan.update({nama: "Adi"},  
  {nama:"Adi",usia:21});
```

- Update property data tertentu:

```
$ db.karyawan.update({nama: "Budi"},  
  {$set:{usia:26}});
```

- Update nama property:

```
$ db.karyawan.update({nama: "Caca"},  
  {$rename:{"usia":"umur"}});
```

- Update **semua** data isinya hanya usia = 21:

```
$ db.karyawan.updateMany({}, {usia:21});
```

- Update semua data property usia = 21:

```
$ db.karyawan.updateMany({},  
  {$set:{usia:21}});
```

- Update nama property di semua data:

```
$ db.karyawan.updateMany({},  
  {$rename:{ "usia" : "umur" }});
```

- Hapus 1 property dari sebuah data:

```
$ db.karyawan.update({nama: "Adi"},  
{$unset:{usia:true}});
```

```
$ db.karyawan.update({nama: "Budi"},  
{$unset:{usia:1}});
```

- Hapus 1 data:

```
$ db.karyawan.remove({nama: "Caca"});
```

- Hapus semua data:

```
$ db.karyawan.remove({});
```

- Tampilkan data yang memiliki property value nama = Budi **dan** usia = 21:

```
$ db.karyawan.find({$and:  
[ {nama:"Budi"}, {usia:21} ] }).pretty();
```

- Tampilkan data yang memiliki property value nama = Adi **atau** nama = Budi:

```
$ db.karyawan.find({$or:  
[ {nama:"Adi"}, {nama:"Budi"} ] }).pretty();
```


Lower and Greater Than

- Tampilkan data yang property usianya < 25:

```
$ db.karyawan.find({usia:{$lt:25}})
.pretty();
```

- Tampilkan data yang property usianya > 25:

```
$ db.karyawan.find({usia:{$gt:25}})
.pretty();
```

Lower and Greater Than Equal

- Tampilkan data yang property usianya ≤ 25 :

```
$ db.karyawan.find({usia:{$lte:25}})
.pretty();
```

- Tampilkan data yang property usianya ≥ 25 :

```
$ db.karyawan.find({usia:{$gte:25}})
.pretty();
```

- Tampilkan 2 data pertama:

```
$ db.karyawan.find().limit(2);
```

- Tampilkan 1 data setelah 2 data pertama:

```
$ db.karyawan.find().limit(1).skip(2);
```

- Tampilkan 3 data setelah 3 data pertama:

```
$ db.karyawan.find().limit(3).skip(3);
```

- Urutkan data ***ascending*** berdasarkan ***nama***:

```
$ db.karyawan.find().sort({nama:1});
```

- Urutkan data ***descending*** berdasarkan ***nama***:

```
$ db.karyawan.find().sort({nama:-1});
```

- Urutkan data ***ascending*** berdasarkan ***usia***:

```
$ db.karyawan.find().sort({usia:1});
```

- Urutkan data ***descending*** berdasarkan ***usia***:

```
$ db.karyawan.find().sort({usia:-1});
```

- Hitung jumlah data di collection “karyawan”:

```
$ db.karyawan.find().count();
```

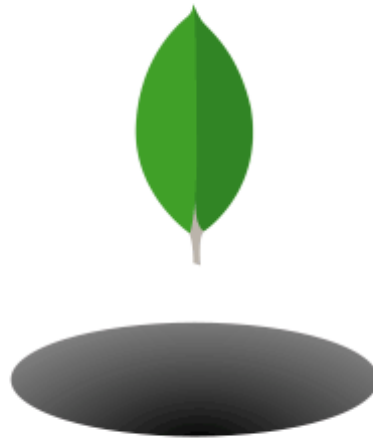
- Hitung jumlah data dg prop nama = Adi:

```
$ db.karyawan.find({nama:"Adi"}).count();
```

How to Work With MongoDB GUI Tools

Working with GUI

#1 Installing MongoDB Compass



MongoDB Compass is being installed.
It will launch once it is done.

Connect to Host

Hostname

Port

Authentication

Replica Set Name

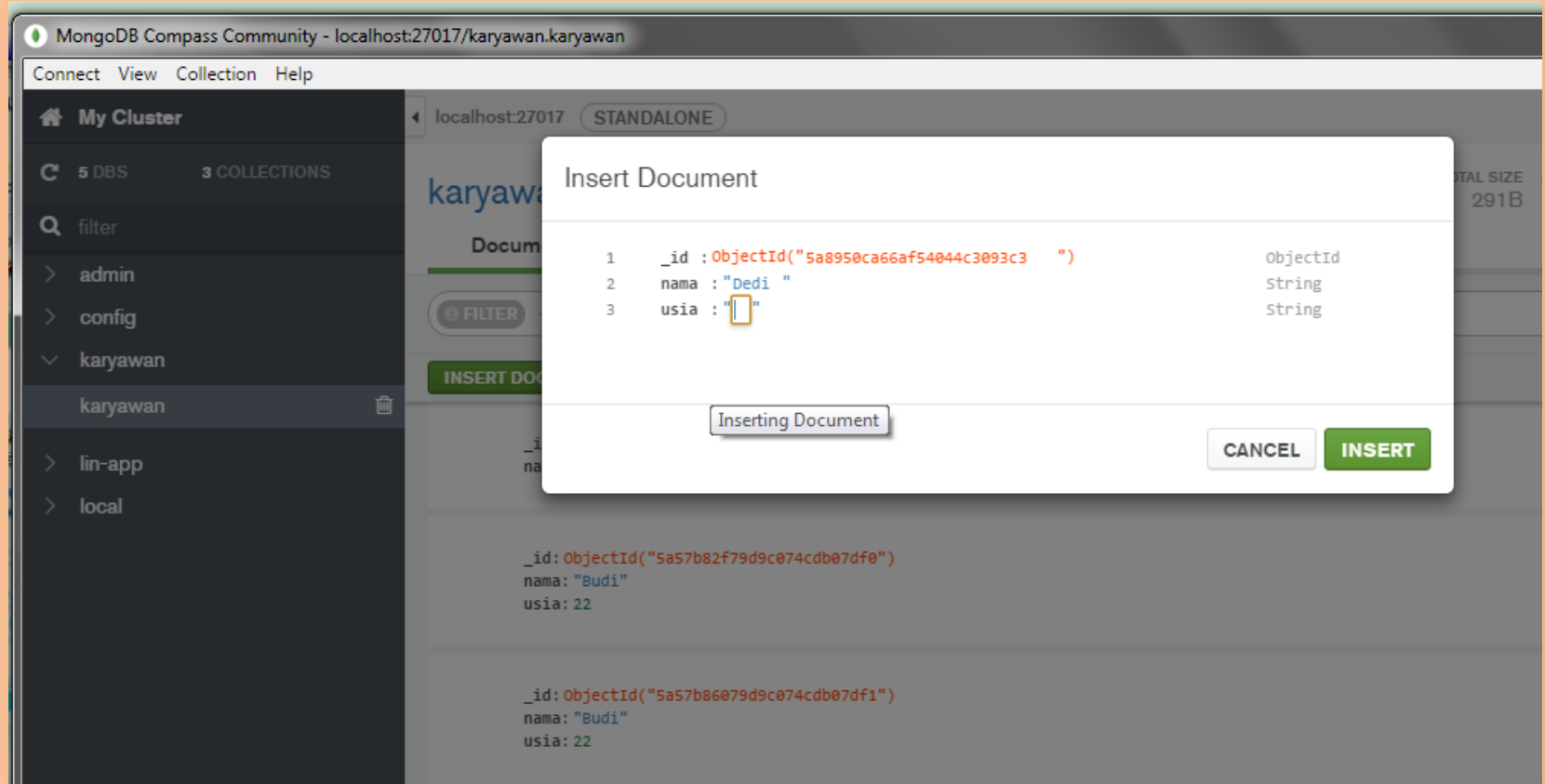
Read Preference

SSL

SSH Tunnel

Favorite Name ⓘ

CONNECT



How to Work With MongoDB Hosting

[PLANS & PRICING](#)[DOCUMENTATION](#)[SIGN UP](#)[LOG IN](#)


Trusted. Loved. Most widely deployed.

mLab is the leading Database-as-a-Service for MongoDB, powering over half a million deployments worldwide.

GET STARTED INSTANTLY with 500 MB FREE!

mlab.com/

[Home](#): { db: "tes1" }

Collection: koleksi_1 



Documents

Indexes

Stats


Tools

Documents

 Delete all documents in collection
  Add document



--- Start new search --- ▾

All Documents



Display mode: ☒ list ☐ table ([edit table view](#)) 

records / page 10 ▾ [1 - 2 of 2]

```
{
  "_id": {
    "$oid": "5aaf1c77734d1d1b828920a4"
  },
  "nama": "Andi",
  "usia": 24
}
```

```
{
  "_id": {
    "$oid": "5aaf1cf1734d1d1b828920b8"
  },
  "nama": "Budi",
  "usia": 26
}
```

records / page 10 ▾ [1 - 2 of 2]

Create document

```
1 {  
2   "nama": "Caca",  
3   "usia": 23  
4 }
```

[cancel and go back](#)

Create and go back

Create and continue editing

Back-End Development



mongoDB

Exploration