**Bab 1 : Pengenalan MongoDB**

Didalam bab 1 : pengenalan MongoDB anda akan mempelajari :

* -pengertian dari mongodb
* -perbedaan dari non sql dan sql

**Apa itu mongoDB ?**

MongoDB adalah database **noSQL** yang menganut sistem **Document Database** dancara penulisan Seperti **JSON.** MongoDB didirikan pada tahun 2007 yang dirilis ke public pada tahun 2009 dikembangkan oleh perusahaan bernama 10gen yang sekarang berganti nama menjadi MongoDB inc.

MongoDB tidak mengunakan bahasa sql ,namun mengunakan **javascript** sebagai bahasa utama untuk manipulasi document , tetapi jangan khawatir MongoDB bisa dipakai hampir di semua bahasa pemograman karena ada library masing masing yang bisa mentranslate ke bahasa pemogramman masing masing.

Struktur dari MongoDB mirip Seperti JSON , contoh dari stuktur MongoDB :

{

  "\_id": "5cf0029caff5056591b0ce7d",

  "firstname": "Jane",

  "lastname": "Wu",

  "address": {

    "street": "1 Circle Rd",

    "city": "Los Angeles",

    "state": "CA",

    "zip": "90404"

  },

  "hobbies": ["surfing", "coding"]

}

Sebenarnya MongoDb mengunakan struktur data BSON, Apa itu BSON ? **BSON** kependekan dari binary JSON adalah binary encoded serialization document seperti json , sama seperti JSON , DI BSON juga kita bisa mengunakan embedded object , array ,dan lain lain . Keunggulan dari BSON Ialah type data yang didukung lebih banyak , jika kalian masih pusing tentang json tidak apa apa karena kedepanya penulisanya hampir tidak ada bedanya dengan JSON.

**Perbedaan noSQL dan sql [belum lengkap]**

**Apa itu sql**

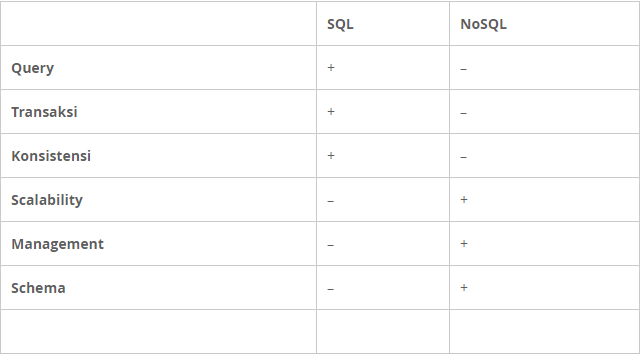
Sql adalah Structured Query Language yaitu bahasa khusus yang digunuakan untuk membuat dan mengolah database,syntax dari sql ini cukup mudah dipahami oleh manusia , contohnya SELECT FROM,WHERE,dll

**Apa itu noSQL**

NoSQL bukan merupahakan bahasa.NoSQL adalah database yang tidak memiliki perintah SQL dan konsep penyimpananya semistruktural atau tidak stuktural dan tidak harus memiliki relasi layaknya table table Mysql

Empat mekanisme dalam NoSQL

* Table-oriented, contoh: Google dengan Big Table dan juga Facebook dengan Cassandra,
* Graph-oriented,
* Document-oriented database, contoh: MongoDB dan juga CouchDB,
* Key-value store, contoh: Memcache dan Redis

**Perbedaan besar dari sql dan non sql**

**Bab 2 : Pengenalan & crud mongoDB**

Didalam bab 2: pengenalan MongoDB anda akan mempelajari :

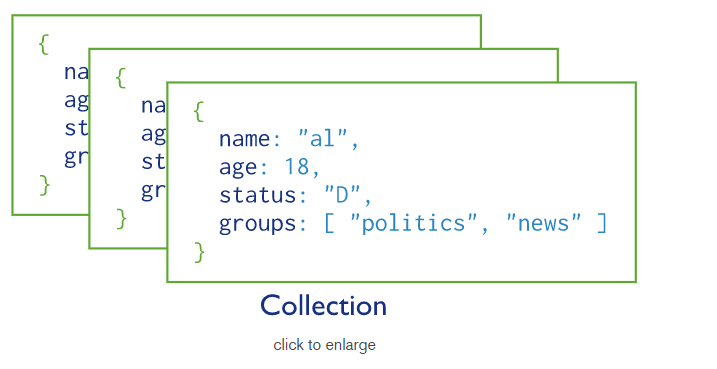
**Memahami Database ,collections & documents ?**

**Database**

Database adalah container / wadah untuk menyimpan collection

**Collection**

Collection adalah tempat penyimpanan document jika dianalogikan ke database sql , collection itu seperti table. Coba perhatikan gambar berikut



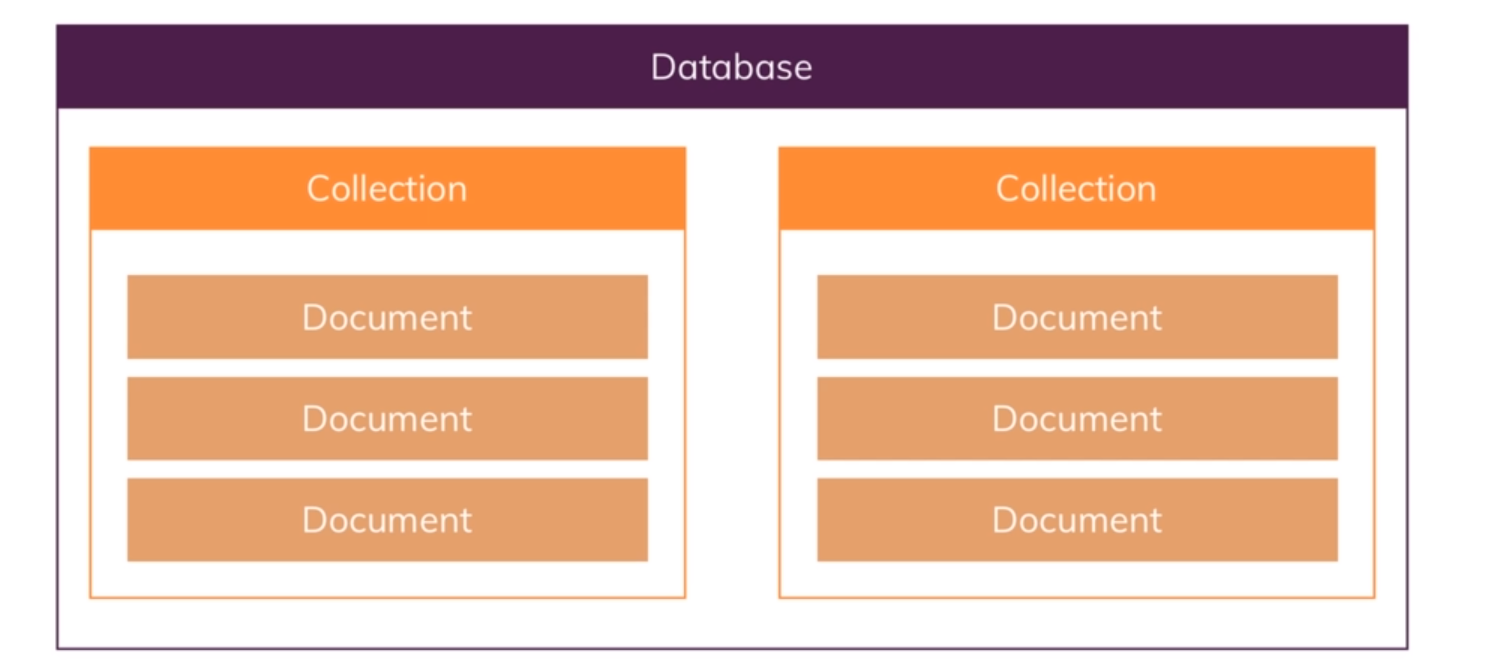
Gambar itu menjelaskan , Collection itu seperti container dari document /data/

**Document**

MongoDB Menyimpan data sebagai bson document ,jika dianologikan ke database sql , document itu seperti record record dari tabel.



Jika belum mengerti bisa lihat gambar ini



Digambar tersebut menjalaskan Database manampung collection dan collection menampung documents

**Membuat Database dan collection**

Membuat database dan collection di mongoDB Sangatlah mudah,kita langsung praktek saja.

Saya anggap kalian sudah menginstall Mongo dan setting path nya , jika belum kembali ke bab 1 bagian cara install mongoDB.

**Mengakses Server MongoDB**

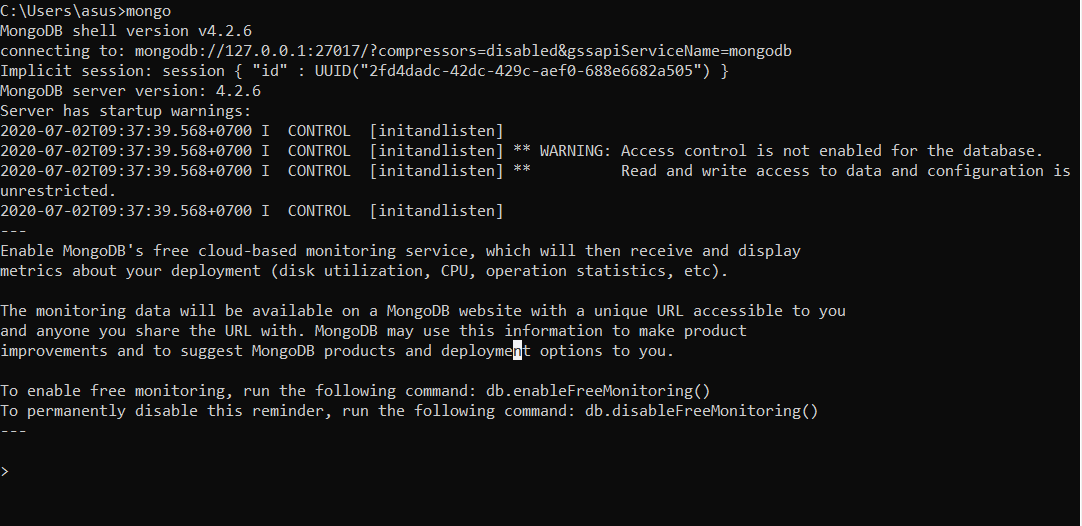
Silahkan kalian buka terminal atau shell , untuk mengakses srever mongoDB Sangat mudah Kalian hanya cukup ketik perintah Mongo

Perintah Mongo Berfungsi untuk membuka shell MongoDB.

Silahkan kalian ketik perintah ini di terminal kalian.

mongo

Kalau berhasil seharus muncul seperti ini, jika sudah begini kalian sukses masuk ke mongoDB Shell



Unuk cek nya coba kalian ketik perintah ini

show dbs

maka akan memunculkan list database , jika kalian belum pernah pakai mongoDB seharus secara default ada 3 database bawaan dari mongoDB

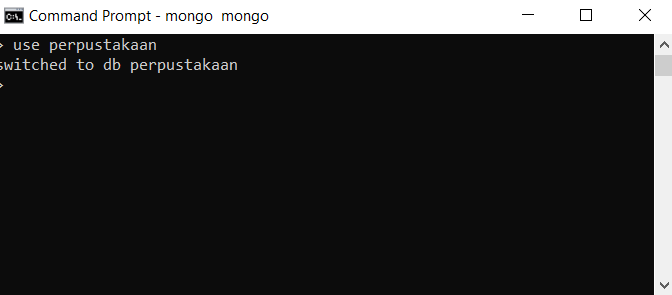
**Untuk keluar dari mongo Server tekan CTRL + C**

**Membuat Database**

Untuk membuat database sangatlah simple kalian hanya perlu mengetikan perintah

use namaDatabase

disini saya akan mencoba membuat database perpustakaan



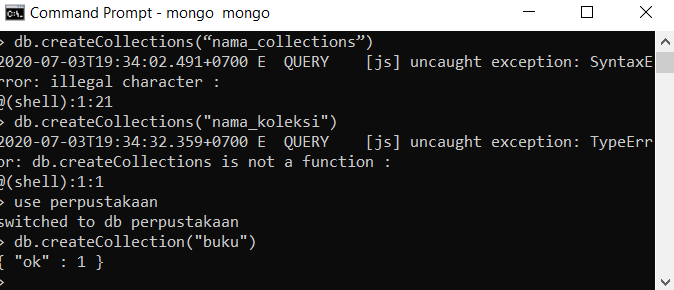
sangatlah Mudah bukan ? tetapi jika kalian ketikan perintah show dbs ,database yang telah dibuat tidak ada di list atau tidak tersimpan , itu karene di mongoDB kita wajib membuat **collection** terlebih dahulu

**Membuat Collection**

Membuat collections bisa dibuat dengan perintah :

db.createCollection(“nama\_collections”)

disini saya mencoba membuat collection buku



Gimana sangat mudah bukan membuat database dan collection di mongo ? oke selanjut kita akan membahas CRUD ,tapi sebelum itu kita akan membahas lebih detail tentang JSON

**Memahami JSON**

Sekarang saya akan membahas singkat tentang **JSON**, pembahasan ini hanya singkat saja yah jika ingin lebih detail tentang json bisa kunjungi link [ini](https://www.json.org/json-id.html)

**JSON(Javascript Object Nation)**  adalah format pertukaran data yang ringan , mudah dibaca dan ditulis oleh manusia , serta mudah diterjemahkan dan dibuat(generate) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa pemogramman javascript, tetapi JSON dapat digunakan di semua bahasaa pemogramaan seperti C,C++,C#,Java ,Javascript,Perl,Python,dll.Oleh karena sifat tersebut menjadikan JSON ideal sebagai bahasa pertukaran-data.

Sebenarnya JSON bukan hanya untuk pertukuran tetapi bisa untuk :

* JSON sebagai format untuk bertukur data client dan server atau antar aplikasi, Contoh Restful API
* JSON sebagai tempat menyimpan data ,seperti database MongoDB
* JSON digunakan untuk menyimpan konfigurasi project , contoh : file composer.json pada php dan package.json pada Node js
* JSON digunakan untuk menyimpan konfigurasi dan penyimpan data pada hugo
* JSON digunakan untuk menyimpan konfigurasi project pada Node js
* JSON digunakan untuk menyimpan data manifest;
* dan masih banyak lagi

Pada point 2 “JSON sebagai tempat menyimpan data seperti MongoDB” yah struktur data MongoDB itu seperti JSON tetapi lebih tepatnya lagi MongoDB mengunakan BSON.

Sekarang kita lihat contoh struktur dari JSON

{

    "name" : "Tegar akmal",

    "age" : 17,

}

Itu adalah stuktur sederhana dari JSON

Berikut penjelasan tentang stuktur JSON :

* {} = json selalu dimulai dengan tanda kurung kurawal dan ditutup dengan kurang }.
* Key = key itu ialah kunci dari value , jika di sturuktur diatas key adalah “name”,”age”
* Value = ialah isi dari key ,jika di sturktur diatas value adalah “tegar akmal”
* JSON selalu mengunakan kata kutip di key dan value
* JSON selalu mengunakan koma untuk memisahkan data yang lebih dari Satu

Berikut contoh lain dari JSON

{

    "name" : "Tegar akmal",

    "age" : 17,

    "hobi" : ["coding","tidur","main games"]

}

Value dari JSON bisa berupa array juga

{

    "name" : "Tegar akmal",

    "age" : 17,

    "hobi" : ["coding","tidur","main games"],

    "socialMedia" : {

        "facebook" : "muhammad Tegar Akmal",

        "github" : "tegar97"

    }

}

Valuenya juga bisa bersarang seperti contoh diatas .

Berikut ialah type data yang didukung JSON



**JSON VS BSON**

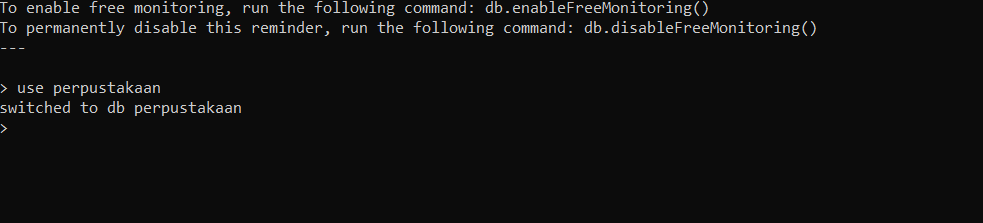
**/- MASIH RISET -/**

**CREATE READ,UPDATE DELETE (CRUD) MONGO DB**

Sekarang kita akan membahas tentang CRUD dari Mongodb , Method CRUD di MongoDB sangat simpel disbanding sql.

Pastikan kalian telah terhubung dengan MongoShell dan telah masuk ke database,didalam kasus ini saya akan memakai database perpustakaan

Untuk masuk kedatabase cukup dengan **use namaDatabase**



**CREATE**

Yang pertama saya akan membahas create terlebih dahulu. Create berfungsi untuk memasukan data kedatabase.

Di MongoDB Untuk memasukan data ada 3 method

Tips : “Untuk mempermudah menulis codenya kalian bisa pakai code editor seperti visual studio code, sublime ,dll lalu codenya copas ke terminal ”

**A.InsertOne()**

Berfungsi untuk memasukan **1 data saja** .

Untuk pengunaan InsertOne seperti ini

db.namaCollection.insertOne({key : “value”})

disini saya akan membuat collection dengan nama “buku” dan isi dari collection nya adalah nama,pengarang,harga

db.buku.insertOne({

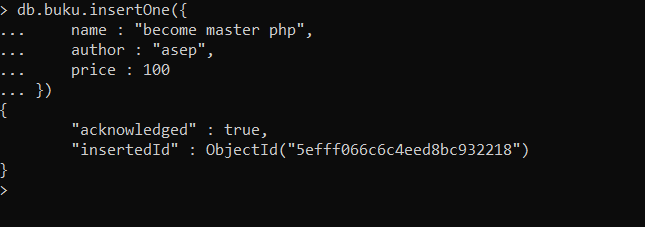
    name : "become master php",

    author : "asep",

    price : 100

})

Maka jika berhasil akan muncul seperti ini



**B.InsertMany()**

Jika tadi hanya bisa memasukan 1 data saja insertMany bisa memasukan data lebih dari 1.

Untuk pengunaan insertMany hampir sama dengan insertOne yang membedakan hanya ada array sebelum object / ( {} )

db.namaCollection.insertMany([{key : “value”}])

Disini saya akan memasukan 3 buku maka codenya jadi seperti :

db.buku.insertMany([

    {

        name : "cara membuat nasi goreng",

        author : "bacdim",

        price : 200

    },

    {

        name : "become master mongoDB",

        author : "tegar",

        price : 0

    },

    {

        name : "become master node js",

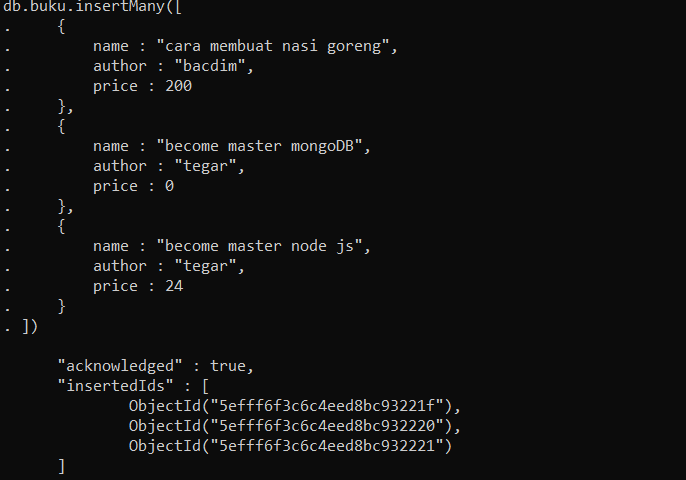
        author : "tegar",

        price : 24

    }

])

Maka kalau berhasil akan muncul seperti ini



Peraturan yang harus di ingat dalam penulisan :

1. Karena ini bson maka **key** tidak perlu mengunakan tanda kutip (“ ”)
2. Untuk key yang bertype **string wajib mengunakan tanda kutip** sedangkan yang bertype integer , bollean ,dll tidak perlu
3. ObjectId /id otomatis dibuat tetapi kita bisa buat id sendiri,tetapi lebih disarankan pakai id bawaan dari mongoDB
4. Value bisa berupa array dan embedded document , nanti kita akan membahas ini.

**READ**

Sekarang kita akan membahas tentang Read dari MongoDB, ada 2 methods untuk mencari data di MongoDB

**A.find()**

Yang pertama ada methods find() , methods ini berfungsi untuk mencari data di mongoDB dan bisa me filter data .

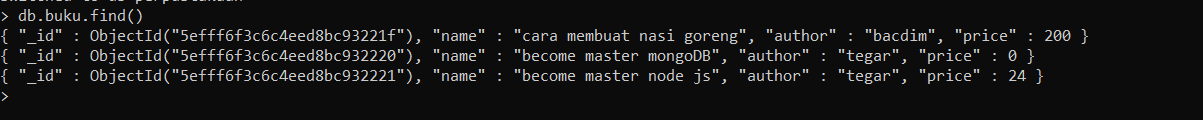
Untuk cara pengunaan sangat mudah :

db.namaCollection.find()

disini saya mencoba mencari data data buku yang telah kita buat tadi , maka cukup mengetikan

db.buku.find()

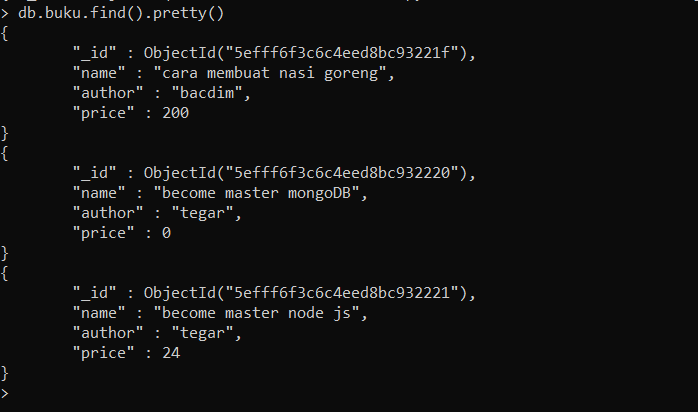
Maka akan keluar hasil seperti ini



Jika dilihat lihat tampilan data yang keluar tidak enak dipandang , nah mongoDB Menyiapkan method pretty().Pengunaan nya cukup seperti ini

db.buku.find().pretty()

Maka hasil akan lebih cantik dan mudah dibaca kan



Didalam methods find() kita juga bisa mefilter data yang mau ditampilkan .

db.buku.find({key : value})

Contohnya saya ingin mefilter data buku yang harganya 200 ,maka cukup seperti ini saja

db.buku.find({price : 200})

Nanti teman teman akan lebih mendalami tentang methods find di BAB LEBIH MEMPERDALAMI TENTANG READ OPERATION , dibab sana teman teman akan mempelajari tentang filter , cursor object,projection, dll, Disini saya ingin focus dulu tentang crud oke .

**b.findOne()**

Method find One berfungsi untuk mencari / membaca **1 data saja .**

Untuk pengunaan dari findOne cukup mudah

db.buku.findOne()

itu akan mencari data **terbaru** dari collection , teman teman bisa juga me filter datanya

db.buku.findOne({key : value})

**READ**