

## Bagian Kelima

# MongoDB dan Mongoose

-> MongoDB adalah NOSQL database, tanpa menggunakan relation system.

-> Mongoose adalah Package library yang menghubungkan NodeJS dan Aplikasi Express untuk berkomunikasi secara langsung dengan MongoDB database. Tanpa library ini akan cukup sulit menghubungkan antara NodeJS, Express dan MongoDB ini.

## Membuat Aplikasi Baru

Kita akan membuat aplikasi untuk belajar mempraktekan mongodb dan mongoose ini, ikuti step berikut ini:

-> Buka terminal anda yang pakai windows bisa gunakan GIT Bash atau CMDER dari cmdr.net (alternatif CMD commandline).

-> Arahkan ke folder kerja kita, jika saya kelasNodeJS > projects.

-> Pada folder projects buat folder baru nama project dengan mkdir belajarmongoose.

-> kemudian CD ke folder belajarmongoose.

-> npm init

-> npm install mongoose express --save

-> Buka folder belajarmongoose dengan Editor.

-> Buat file baru dengan nama server.js

-> Buat code seperti dibawah ini:

```
var express = require('express');
var mongoose = require('mongoose');

var app = express();

app.listen(3000, function(){
  if (err) {
    console.log(err);
  } else {
    console.log("Server berjalan pada port 3000")
  }
});
```

-> run server pada terminal: nodemon server.

-> Pastikan tidak ada error.

-> Buat halaman root dengan menambahkan baris code dibawah ini:

```
app.get('/', function(req, res, next){
  res.json('Selamat Datang di Halaman Beranda')
})
```

-> Kemudian buat Dinamis Url :

```
app.get('/:name', function(req, res, next){
  res.json(req.params.name)
});
```

-> di pada browser localhost:3000 dan localhost:3000/products

Koneksi ke Cloud Database mlab.com

- > Kita akan menggunakan extenal / cloud database yaitu mlab.com
- > Silahkan login ke mlab.com
- > Create New database.
- > Pilih Single-node > Sandbox
- > Berinama database dengan "mongoosecoba".
- > Klik button "Create new MongoDB deployment".
- > Klik database nya.
- > Create new User.
- > Klik Users > klik Add database user.
- > Masukkan Database username misalnya 'root', password dan confirm password > klik Create.
- > Copy link
- > Membuat 1 baris code untuk koneksi database:

```
mongoose.connect('..link dipaste disini..',  
function(err){  
  if (err){  
    console.log(err);  
  } else {  
    console.log("Connected to the database")  
  }  
});
```

- > Kemudian cek di terminal 😊

## Membuat Collections

Collection-collection adalah nama dari grup dari data dalam database, bila di RDBS seperti tabel-tabel.

Kita akan membuat User Collection (table User dalam SQL), caranya adalah sbb:

```
var UserSchema = new mongoose.Schema({  
  name: String,  
  age: Number  
});
```

Schema adalah semacam blueprint, setiap Schema "map" ke setiap Collection pada mongodb. Schema seperti kumpulan dari Field atau Atribut.

Untuk map Schema ke Collection kita harus membuat method model, yang akan mengaitkan antara Collection dengan Schema, seperti sbb :

```
var User = mongoose.model('User', UserSchema);
```

## Membuat new User

Kita akan membuat User baru dengan menggunakan method GET dengan URL baru, seperti dibawah ini:

```
app.get('/create-user', function(req, res, next){  
  var user = new User();  
  user.name = "Brandon";  
  user.age = 17;  
  res.json(user);  
});
```

Kemudian test di browser dengan URL localhost:3000/create-user.

->\* Jangan lupa comment out latihan DINAMIC URL nya dulu.

Bila kita cek di mlab.com data tidak tersimpan karena pada code diatas kita tidak memberikan perintah "save", edit code diatas menjadi sbb:

```
app.get('/create-user', function(req, res, next){  
  var user = new User();  
  user.name = "Brandon";  
  user.age = 17;  
  user.save()  
});
```

Atau gunakan check error seperti dibawah ini:

```
      user.save(function(err){  
if (err) next(err);  
res.json(user);  
});
```

Bila kita cek di mlab maka user baru tsb akan tersimpan.

---

Copyright ©GandrungStudio 2018 Dilarang memperbanyak dan mendistribusikan Materi Training ini tanpa ijin dari PT.Gandrung Media Corporation. Pelanggaran akan hak cipta ini akan kami ajukan ke Institusi Hukum yang berlaku.