

Back-End Development

Express & MongoDB

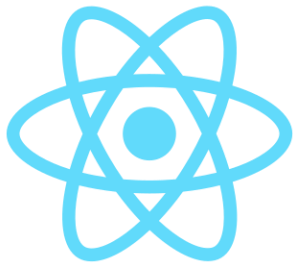
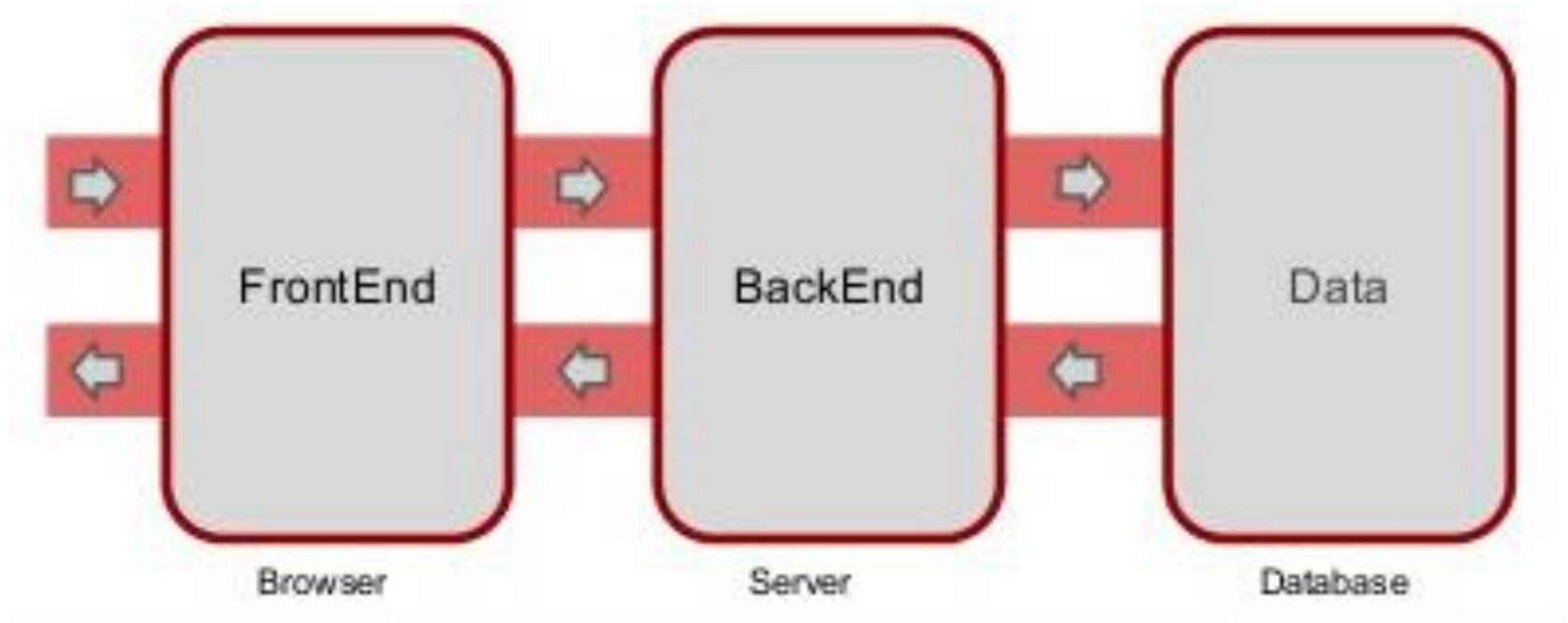
#13b Using Mongoddb module

express



mongoDB

Web App Architecture



#1 Activate MongoDB Server

- Open terminal:

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

#2 Create Db, User & Collection

- Create db toko:

```
$ use toko
```

- Create db user:

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

- Create collection “karyawan”:

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

#3 MongoDB Package

- It's an official MongoDB driver for Node.js. Provides a high-level API on top of mongodb-core that is meant for end users.
- More info: npmjs.com/package/mongodb
- On project dir, install MongoDB package:

```
$ npm install mongodb@2
```

Using MongoDB Package

#1 Make a Connection

```
var MongoClient = require('mongodb').MongoClient;  
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {  
  console.log("Terhubung ke MongoDB!");  
  db.close();  
});
```

PROBLEMS

OUTPUT

TERMINAL

...

1: powershell ▼

```
PS D:\zzz coding\lin_backend> node express-mongoZZZ  
Terhubung ke MongoDB!  
PS D:\zzz coding\lin_backend> █
```

Using MongoDB Package

#2 Insert Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  inputData(db, function() {
    db.close();
  });
});
```

```
var inputData = function(db, callback) {
  var collection = db.collection('karyawan');
  collection.insertMany(
    [{nama: 'Andi', usia: 24, kota: 'Bandung'}],
    function(err, result) {
      console.log("Data sukses tersimpan!");
      callback(result);
    });
}
```

Using MongoDB Package

#3 Insert Many Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  inputData(db, function() {
    db.close();
  });});
```

```
var inputData = function(db, callback) {
  var collection = db.collection('karyawan');
  collection.insertMany(
    [{nama: 'Budi', usia: 21, kota: 'Jakarta'},
     {nama: 'Caca', usia: 25, kota: 'Bandung'},
     {nama: 'Dedi', usia: 28, kota: 'Medan'}],
    function(err, result) {
      console.log("Data sukses tersimpan!");
      callback(result);
    });}
```


Using MongoDB Package

#4 Find All Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  findData(db, function() {
    db.close();
  });});
```

```
var findData = function(db, callback) {
  var collection = db.collection('karyawan');
  collection.find({}).toArray(function(err, docs) {
    console.log("Berikut data yang tersimpan:");
    console.log(docs)
    callback(docs);
  });
}
```

Using MongoDB Package

#5 Find a Specific Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  findData(db, function() {
    db.close();
  });});
```

```
var findData = function(db, callback) {
  var collection = db.collection('karyawan');
  collection.find({nama: 'Andi'}).toArray((err, docs)=>{
    console.log("Berikut data yang tersimpan:");
    console.log(docs)
    callback(docs);
  });
}
```

Using MongoDB Package

#6a Insert Then Find Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  inputData(db, function() {
    findData(db, function(){
      db.close();});
  });});
```

```
var inputData = function(db, callback) {
  var collection = db.collection('karyawan');
  collection.insertMany(
    [{nama: 'Dedi'}, {nama: 'Euis'}, {nama: 'Fafa'}],
    function(err, result) {
      console.log("Data sukses tersimpan!");
      callback(result);
    });
}
```

Using MongoDB Package

#6b Insert Then Find Data

```
var findData = function(db, callback){  
  var collection = db.collection('karyawan');  
  collection.find({}).toArray(function(err, docs){  
    console.log("Berikut data yang tersimpan:");  
    console.log(docs)  
    callback(docs);  
  });  
}
```

PROBLEMS

OUTPUT

TERMINAL

...

1: powershell



Terhubung ke MongoDB!

Data sukses tersimpan!

Berikut data yang tersimpan:

```
[ { _id: 5ab2ba125150911fc8ce707c, nama: 'Andi' },  
  { _id: 5ab2ba125150911fc8ce707d, nama: 'Budi' },  
  { _id: 5ab2ba125150911fc8ce707e, nama: 'Caca' },  
  { _id: 5ab2bd900cb21a1cfcda9db5, nama: 'Dedi' },  
  { _id: 5ab2bd900cb21a1cfcda9db6, nama: 'Euis' },  
  { _id: 5ab2bd900cb21a1cfcda9db7, nama: 'Fafa' } ]
```

PS D:\zzz coding\lin_backend>

Using MongoDB Package

#7 Update Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  updateData(db, function() {
    db.close();
  });});
```

```
const updateData = function(db, callback) {
  const collection = db.collection('karyawan');
  collection.updateOne({nama: 'Andi'},
    {$set: {kota: 'Solo'}}, (err, out) => {
    console.log("Data sukses terupdate!");
    callback(out);
  });
}
```

Using MongoDB Package

#8 Update Data

```
var edit = (db)=>{  
  var col = db.collection('datamongo')  
  col.update(  
    {nama: 'Deni'},  
    {$set: {kota: 'Surabaya'}},  
    {multi: true},  
    (error, hasil)=>{  
      console.log(hasil)  
    })  
  })  
}
```

Using MongoDB Package

#9 Remove Data

```
var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';
```

```
MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
  removeData(db, function() {
    db.close();
  });});
```

```
const removeData = function(db, callback) {
  const collection = db.collection('karyawan');
  collection.deleteOne({nama: 'Budi'}, (err, out) => {
    console.log("Data sukses dihapus!");
    callback(out);
  });
}
```

Using MongoDB Package

#10 Remove Data by Id

```
Var mongodb = require('mongodb')

var hapus = (db) => {
var col = db.collection('datamongo')
col.remove(
{_id: new mongodb.ObjectId(
"5bb2ed2e97413a2eb0867a93"
)}},
(error, hasil)=>{
console.log(hasil)
}
)
}
```


Express & MongoDB

express



mongoDB

Express & MongoDB

#1 Connect

```
const express = require('express');
const app = express();

var MongoClient = require('mongodb').MongoClient;
var url = 'mongodb://lintang:1234@localhost:27017/toko';

MongoClient.connect(url, function(err, db) {
  console.log("Terhubung ke MongoDB!");
});

### create route here ###

app.listen(3210, ()=>{
  console.log('Server aktif di port 3210')
});
```

```
PS D:\zzz coding\lin_backend> node z_express_mongo
Server aktif di port 3210
Terhubung ke MongoDB!
█
```

Express & MongoDB

#2 GET & POST

```
app.get('/data', (req, res)=>{
  MongoClient.connect(url, (err, db)=>{
    var collection = db.collection('karyawan');
    collection.find({}).toArray((err, docs)=>{
      console.log(docs);
      res.send(docs);
    });
  });
});
```

```
app.post('/data', (req, res)=>{
  MongoClient.connect(url, (err, db)=>{
    var data = {nama:'Susi', usia:22};
    var collection = db.collection('karyawan');
    collection.insert(data, (err, result)=>{
      console.log(result);
      res.send(result);
    });
  });
});
```

Express & mLab

express



mLab

Back-End Development

Express & MongoDB

#13b Using Mongoddb module

express



mongoDB