

## TUGAS BASIS DATA LANJUT

### MEMBUAT DATA BASE ONE TO MANY TO MANY DI MONGODB

Nama : YOLLAVINIA

NIM : 17050623007

#### #ONE TO MANY

1. Membuat connections dalam database "DOSEN"

```
Command Prompt - mongo
> show dbs
admin    0.000GB
config  0.000GB
kuliah   0.000GB
local    0.000GB
> use kuliah
switched to db kuliah
> db
kuliah
> show collections
dosen
```

Database sudah muncul dosen

2. Membuat collections "matkul"

```
> db.matkul.insert({matkul:'matematika', id_dosen:db.dosen.f
> b.matkul.insert({matkul:'matematika', id_dosen:db.dosen.f
> db.matkul.insert({matkul:'matematika', id_dosen:db.dosen.fi
nd()[0]._id, jumlah_sks:'2', jenis_matkul:'wajib'})
writeResult({ "nInserted" : 1 })
>
```

3. Hasil relasi

```
Command Prompt - mongo
{id_dosen" : "ABC",
"nama_dosen" : "dandy",
"alamat" : "tulungagung",
"tgl_lahir" : "05-09-1990",
"email" : "dandi@gmail.com"
}
> db.matkul.insert.find().pretty()
2019-02-27T20:58:07.307+0700 E QUERY [js] TypeError: db.ma
tkul.insert.find is not a function :
@(shell):1:1
> db.matkul.find().pretty()
{
  "_id" : ObjectId("5c7696c1da8dd1a63853bbc5"),
  "matkul" : "matematika",
  "id_dosen" : ObjectId("5c765b47da0c4e92d07b0459"),
  "jumlah_sks" : "2",
  "jenis_matkul" : "wajib"
}
>
```

## Many to Many

### 1. Membuat database "DOSEN PEMBIMBING"

```
Command Prompt - mongo
2019-02-27T23:03:28.028-0700 I CONTROL [initandlisten] ** Read and write access to data and configuration is u
nrestricted.
2019-02-27T23:03:28.028-0700 I CONTROL [initandlisten] **
2019-02-27T23:03:28.028-0700 I CONTROL [initandlisten] ** WARNING: This server is bound to localhost.
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] ** Remote systems will be unable to connect to this ser
ver.
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] ** Start the server with --bind_ip <address> to specify
which IP
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] ** addresses it should serve responses from, or with --
bind_ip_all to
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] ** bind to all interfaces. If this behavior is desired,
start the
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] ** server with --bind_ip 127.0.0.1 to disable this warn
ing.
2019-02-27T23:03:28.029-0700 I CONTROL [initandlisten] **
--
Enable MongoDB's free cloud-based monitoring service, which will then receive and display
metrics about your deployment (disk utilization, CPU, operation statistics, etc).
The monitoring data will be available on a MongoDB website with a unique URL accessible to you
and anyone you share the URL with. MongoDB may use this information to make product
improvements and to suggest MongoDB products and deployment options to you.
To enable free monitoring, run the following command: db.enableFreeMonitoring()
To permanently disable this reminder, run the following command: db.disableFreeMonitoring()
--
> use dosen_pembimbing
switched to db dosen_pembimbing
>
```

### 2. Mengloneksikan database "Gudang", distribusi dan "toko'.

```
Command Prompt - mongo
> show collection
2019-02-28T13:28:21.448+0700 E QUERY [js] Error: don't know how to show [collection] :
shellHelper.show@src/mongo/shell/utils.js:1066:11
shellHelper@src/mongo/shell/utils.js:766:15
@shellHelper:1:1
> show collections
>
> db
dosen_pembimbing
> show collections
> db.dosen_pembimbing.insert({nama_mahasiswa:'yolla', NIM:'18098', PRODI:'MI', SKS:'29'})
writeResult({ "nInserted" : 1 })
> db.mahasiswa.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:db.dosen_pembimbing.findO[0]._id, asal:'jombang', umur:'19'})
2019-02-28T13:48:21.202+0700 E QUERY [js] SyntaxError: missing name after . operator @shell:1:63
> db.mahasiswa.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:db.dosen_pembimbing.findO[0]._id, asal:'jombang', umur:'19'})
2019-02-28T13:49:30.294+0700 E QUERY [js] SyntaxError: missing } after property list @shell:1:117
> db.mahasiswa.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:db.dosen_pembimbing.findO[0]._id, asal:'jombang', umur:'19'})
writeResult({ "nInserted" : 1 })
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:55:50.466+0700 E QUERY [js] SyntaxError: missing : after property id @shell:1:123
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:56:29.809+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:64
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
```

### 3. Table yang telah dihubungkan

```
Command Prompt - mongo
2019-02-28T13:49:30.294+0700 E QUERY [js] SyntaxError: missing } after property list @shell:1:117
> db.mahasiswa.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:db.dosen_pembimbing.findO[0]._id, asal:'jombang', umur:'19'})
writeResult({ "nInserted" : 1 })
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:55:50.466+0700 E QUERY [js] SyntaxError: missing : after property id @shell:1:123
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:56:29.809+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:64
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:58:12.805+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:64
> db.mata_kuliah.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:59:03.790+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:74
> db.mata_kuliah.insert({NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:59:49.499+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:24
> db.mata_kuliah.insert({nama_mahasiswa:db.mahasiswa.findO[0]._id, NIM:db.mahasiswa.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
writeResult({ "nInserted" : 1 })
> show collections
dosen_pembimbing
mahasiswa
mata_kuliah
```

```
Command Prompt - mongo
KS:'2', id_matkul:'AA'})
2019-02-28T13:56:29.809+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:64
> db.mata_kuliah.insert({nama:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:58:12.805+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:64
> db.mata_kuliah.insert({nama_mahasiswa:db.dosen_pembimbing.findO[0]._id, NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:59:03.790+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:74
> db.mata_kuliah.insert({NIM:dosen_pembimbing.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
2019-02-28T13:59:49.499+0700 E QUERY [js] ReferenceError: dosen_pembimbing is not defined :
@shell:1:24
> db.mata_kuliah.insert({nama_mahasiswa:db.mahasiswa.findO[0]._id, NIM:db.mahasiswa.findO[0]._id, matakuliah:'PBO', SKS:'2', id_matkul:'AA'})
writeResult({ "nInserted" : 1 })
> show collections
dosen_pembimbing
mahasiswa
mata_kuliah
> db.detail.insert({nama_mahasiswa:db.mata_kuliah.findO[0]._id, NIM:db.mata_kuliah.findO[0]._id, SKS:db.mata_kuliah.findO[0].id, id:1})
writeResult({ "nInserted" : 1 })
> show collections
detail
dosen_pembimbing
mahasiswa
mata_kuliah
```

### 4. Hasil dari MANY TO MANY

Command Prompt - mongo

```
> db.detail.insert({nama_mahasiswa:db.mata_kuliah.find()[0]._id, NIM:db.mata_kuliah.find()[0]._id, SKS:db.mata_kuliah.find()[0]._id,})
writeResult({ "nInserted" : 1 })
> show collections
detail
dosen_pembimbing
mahasiswa
mata_kuliah
> db.dosen_pembimbing.find().pretty()
{
  "_id" : ObjectId("5c77821e0ad86fb6c689560f"),
  "nama_mahasiswa" : "yolla",
  "NIM" : "18098",
  "PRODI" : "MI",
  "SKS" : "29"
}
> db.mahasiswa.find().pretty()
{
  "_id" : ObjectId("5c7784e70ad86fb6c6895610"),
  "nama_mahasiswa" : ObjectId("5c77821e0ad86fb6c689560f"),
  "NIM" : ObjectId("5c77821e0ad86fb6c689560f"),
  "asal" : "Jombang",
  "umur" : "19"
}
> db.mata_kuliah.find().pretty()
{
  "_id" : ObjectId("5c7787500ad86fb6c6895611"),
  "nama_mahasiswa" : ObjectId("5c7784e70ad86fb6c6895610"),
  "NIM" : ObjectId("5c7784e70ad86fb6c6895610"),
  "matakuliah" : "P80",
  "SKS" : "2",
  "id_matkul" : "AA"
}
> db.detail.find().pretty()
{
  "_id" : ObjectId("5c7788be0ad86fb6c6895612"),
  "nama_mahasiswa" : ObjectId("5c7787500ad86fb6c6895611"),
  "NIM" : ObjectId("5c7787500ad86fb6c6895611"),
  "SKS" : ObjectId("5c7787500ad86fb6c6895611")
}
```