

Modul Praktikum TBD

Modul IV

[DATABASE NO SQL]



Program Studi Sains Data

Fakultas Sains

INSTITUT TEKNOLOGI SUMATERA

Tahun Ajaran Genap 2023/2024

1. Tujuan

- Memahami database non SQL
- dapat menggunakan mongoDB dalam study kasus sehari hari

2. Teori Dasar

- database nonSQL (mongoDB)

MongoDB merupakan sebuah sistem basis data yang berbasis dokumen (Document Oriented Database) dan termasuk sistem basis data yang menganut paham NoSQL. NoSQL itu bukan berarti anti menggunakan SQL (No-SQL), tapi NoSQL itu singkatan dari Not Only SQL artinya sebuah sistem basis data tidak hanya harus menggunakan perintah SQL untuk melakukan proses manipulasi data. MongoDB tidak memiliki yang namanya tabel, kolom dan baris. Dalam MongoDB yang ada hanyalah koleksi dan dokumen. Koleksi dalam MongoDB bisa kita anggap sebuah Folder(Directory) dan Dokumen bisa kita anggap berkas (File) dalam Folder (Koleksi) tersebut. Dokumen yang terdapat dalam MongoDB dapat memiliki berbeda atribut dengan dokumen lain walaupun berada dalam satu koleksi. Hal ini tidak dapat dilakukan dalam RBMS dimana sebuah baris dalam tabel tidak mungkin memiliki kolom yang berbeda dengan baris yang lain jika berada dalam satu tabel. MongoDB merupakan sistem basis data yang menggunakan konsep key-value, artinya setiap dokumen dalam MongoDB pasti memiliki key. Hal ini berbeda dalam RDBMS yang kita bisa tidak menggunakan primary key ketika membuat sebuah tabel. Sehingga walaupun kita membuat sebuah dokumen tanpa menggunakan primary key, tapi secara otomatis MongoDB memberinya sebuah key. Penggunaan konsep key-value sangat berperan penting, karena hal ini membuat MongoDB menjadi sistem basis data yang sangat cepat jika dibandingkan dengan nonkey-value seperti RDBMS. MongoDB tidak menggunakan bahasa yang biasa digunakan RDBMS (SQL atau PL/SQL).

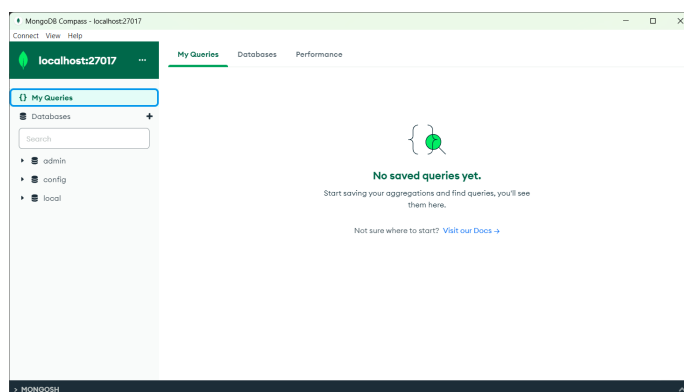
MongoDB menggunakan bahasa BSON, dimana BSON merupakan singkatan dari Binary JSON. Seperti JSON, BSON mendukung embedding dokumen dan dokumen lainnya dalam array dan array. BSON juga mengandung ekstensi yang memungkinkan representasi tipe data yang bukan merupakan bagian dari spec JSON.

3. Instalasi MongoDB

- a. Unduh MongoDB: Kunjungi situs resmi MongoDB <https://www.mongodb.com/try/download/community> dan unduh versi komunitasnya sesuai dengan sistem operasi yang Anda gunakan.
- b. Instalasi: Ikuti petunjuk instalasi yang disediakan untuk sistem operasi Anda. Pastikan untuk mengikuti semua langkah instalasi dengan benar.

4. Mulai Menggunakan MongoDB Compass

- a. Buka MongoDB Compass: Setelah instalasi selesai, buka MongoDB Compass dari menu start atau dengan mengetikkan "MongoDB Compass" di kotak pencarian sistem operasi Anda.
- b. Hubungkan ke Server MongoDB: Ketika Anda membuka MongoDB Compass untuk pertama kali, Anda akan diminta untuk membuat atau memasukkan koneksi ke server MongoDB. Masukkan detail koneksi yang sesuai seperti alamat server, port, dan kredensial yang dibutuhkan.
- c. Jelajahi Database Anda: Setelah terhubung, Anda akan melihat antarmuka grafis yang memungkinkan Anda untuk menjelajahi basis data, koleksi, dan dokumen-dokumen di dalamnya.

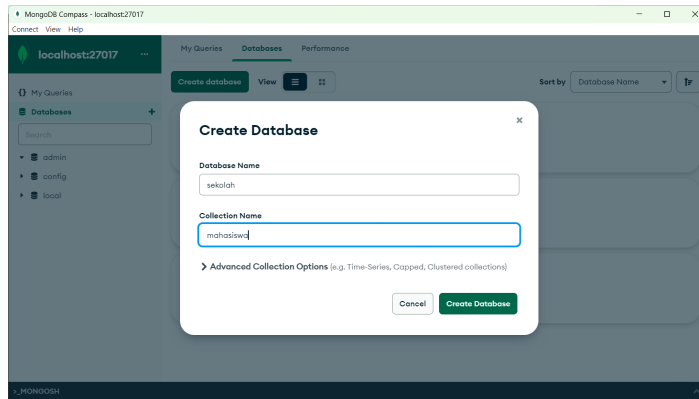


5. Membuat Database baru

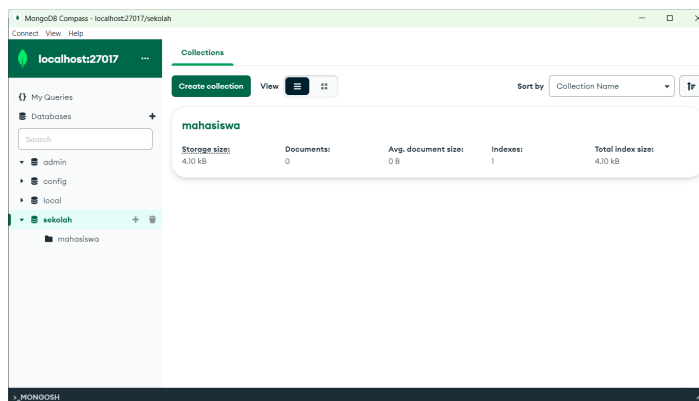
MongoDB Compass sebenarnya lebih berfungsi sebagai alat untuk menjelajahi, mengelola, dan menganalisis data pada database MongoDB yang sudah ada, daripada membuat database atau koleksi baru. Namun, Anda dapat menggunakan MongoDB Compass untuk membuat database dan koleksi jika Anda memiliki izin yang tepat pada server MongoDB yang Anda hubungkan.

- a. Membuat Database
 - Buka Tab "Database": Pada panel kiri, klik tab "Database".

- **Buat Database Baru:** Di bagian kanan atas, Anda akan melihat tombol "Create Database". Klik tombol ini dan masukkan nama untuk database baru Anda.

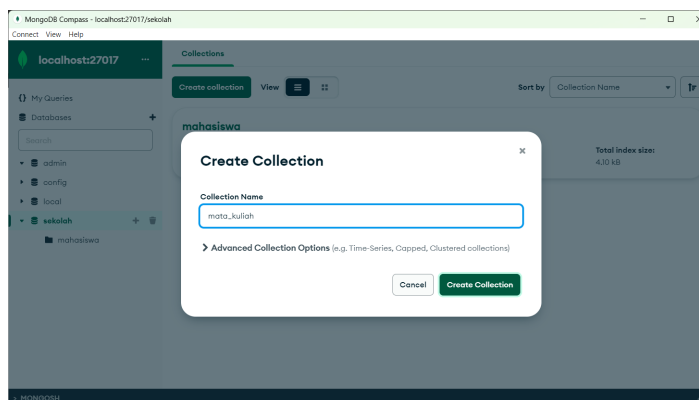


- **Konfirmasi:** Setelah Anda memasukkan nama, Compass akan membuat database baru untuk Anda.

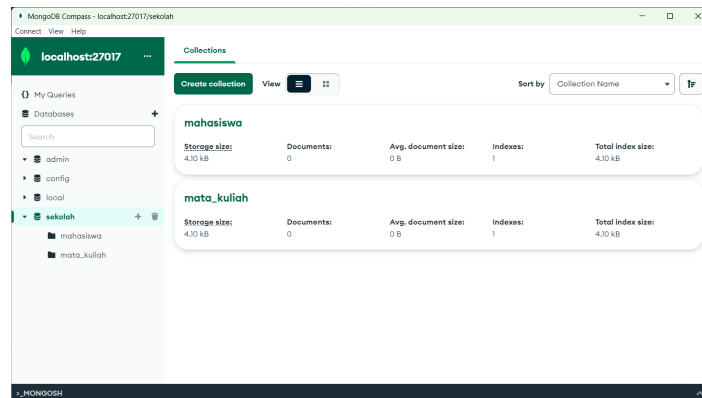


b. Membuat Collection

- **Pilih Database:** Setelah Anda membuat database baru, Anda akan melihatnya muncul dalam daftar database di panel kiri. Klik pada nama database yang baru Anda buat.
- **Buka Tab "Collections":** Setelah Anda memilih database, klik tab "Collections" di bagian atas.



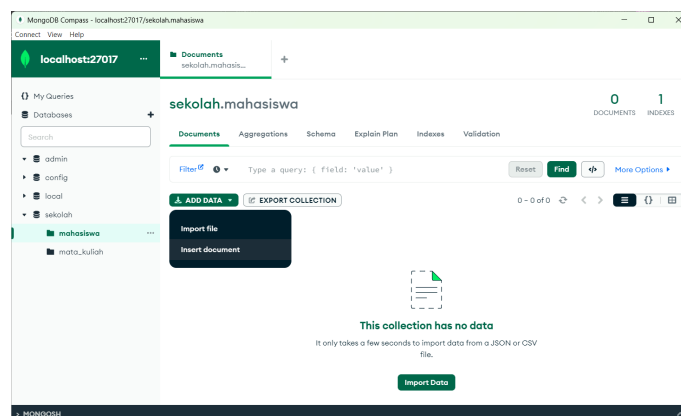
- **Buat Koleksi Baru:** Di bagian kanan atas, Anda akan melihat tombol "Create Collection". Klik tombol ini.
- **Atur Nama Koleksi dan Pilihan:** Masukkan nama untuk koleksi baru Anda dan atur opsi lainnya seperti pengaturan indeks, validasi, dll.
- **Konfirmasi:** Setelah Anda selesai mengatur, Compass akan membuat koleksi baru di dalam database yang Anda pilih.



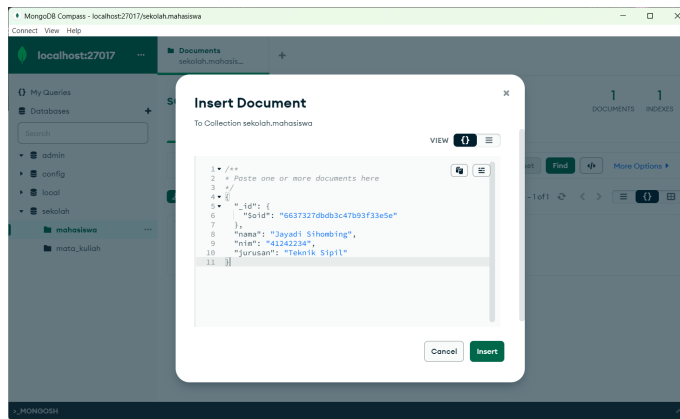
6. Operasi CRUD

MongoDB Compass memungkinkan Anda untuk melakukan operasi CRUD (Create, Read, Update, Delete) dengan mudah:

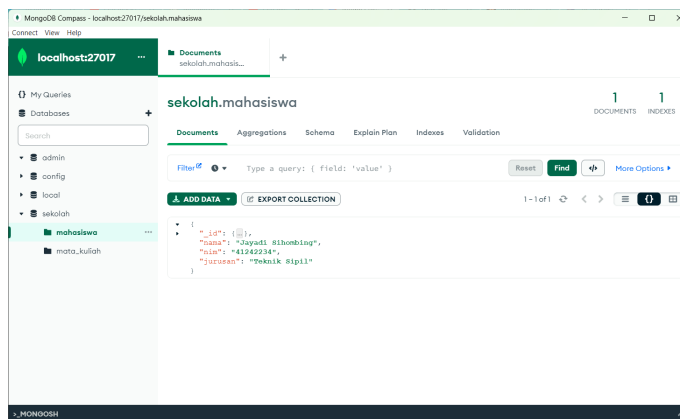
- Create (Buat):** Gunakan fungsi "Insert Document" untuk menambahkan dokumen baru ke dalam koleksi Anda.
- **Pilih insert document**



- **Isi data sesuai dengan kebutuhan**

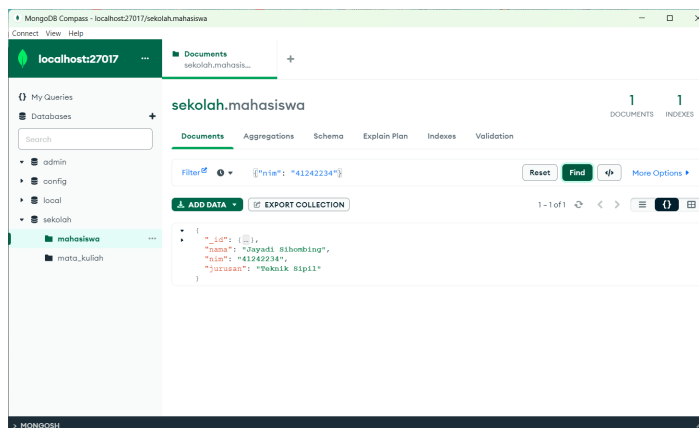


● Lihat hasil

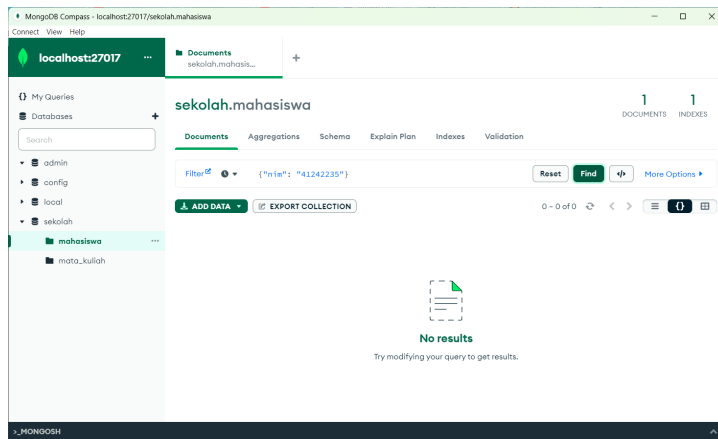


- b. Read (Baca): Gunakan fungsi "Find" untuk mencari dokumen dalam koleksi Anda. Anda dapat menggunakan filter dan proyeksi untuk menyaring hasilnya.

● Input filter dengan format {key: value} lalu klik find



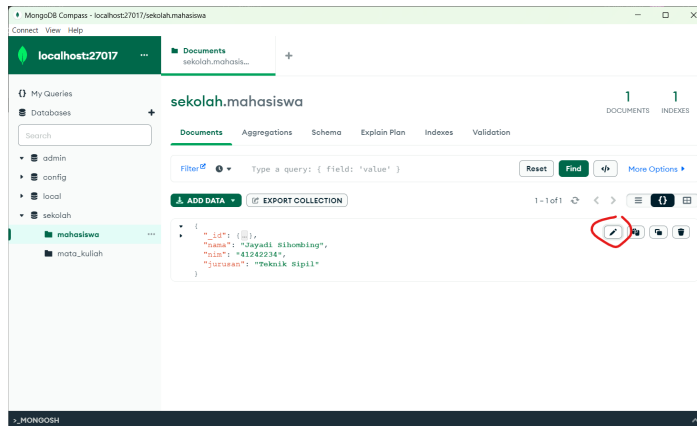
● Contoh lain



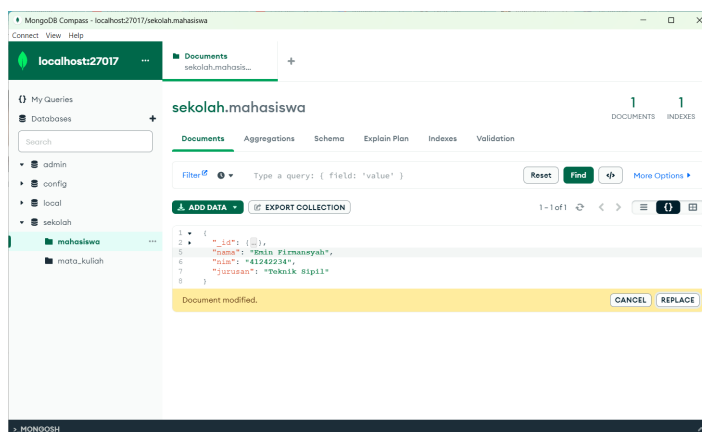
Filter dengan nim = 41232235 tidak ditemukan

- c. Update (Perbarui): Anda dapat memperbarui dokumen dengan mengklik dokumen yang ingin Anda ubah dan mengedit nilainya langsung dari antarmuka.

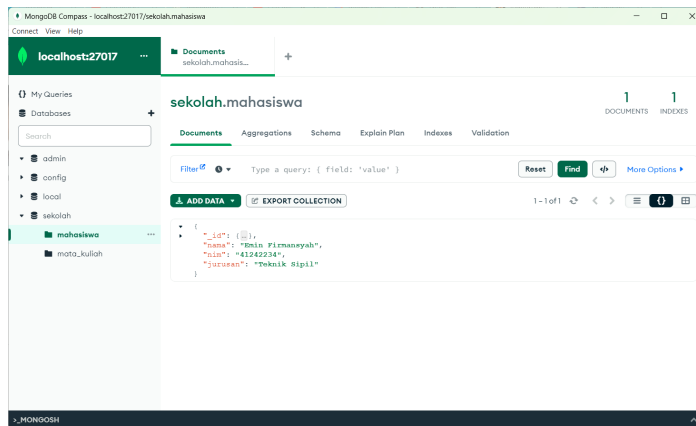
● Klik tombol edit disamping data



● Ubah value lalu klik tombol replace

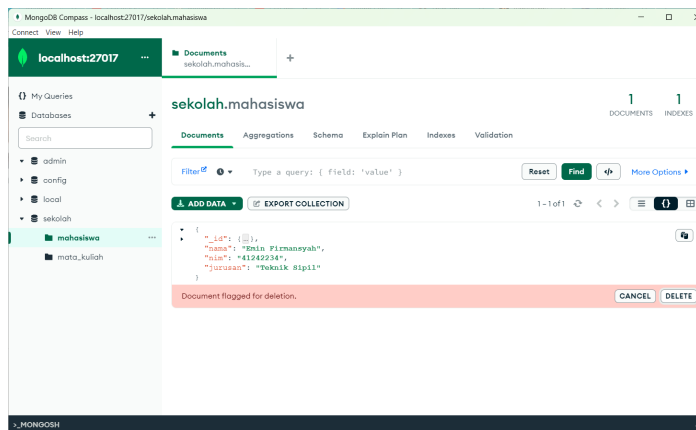


● Hasil data berhasil diupdate



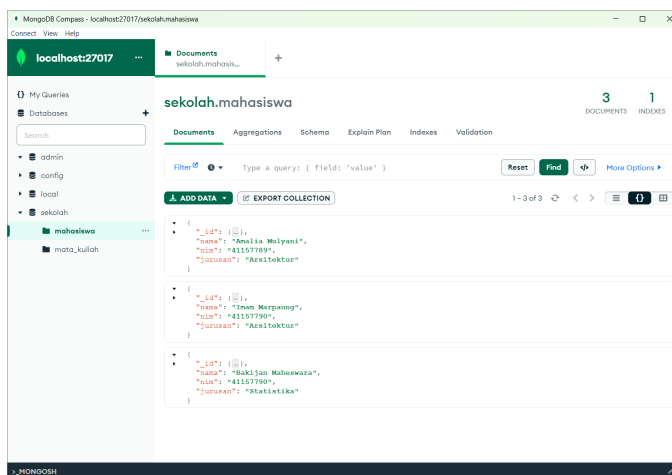
d. Delete (Hapus): Gunakan fungsi "Delete" untuk menghapus dokumen dari koleksi Anda.

- Klik ikon tombol delete disamping data lalu klik tombol delete



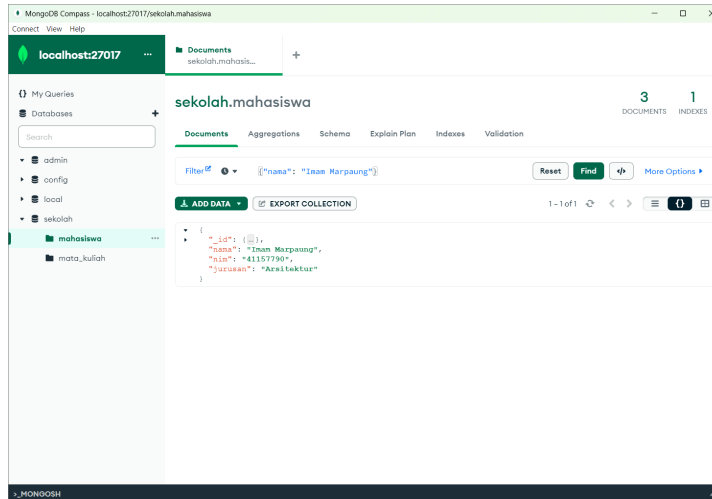
7. Pelajari Querying, Indexing, dan Validation

a. Querying (Pertanyaan)

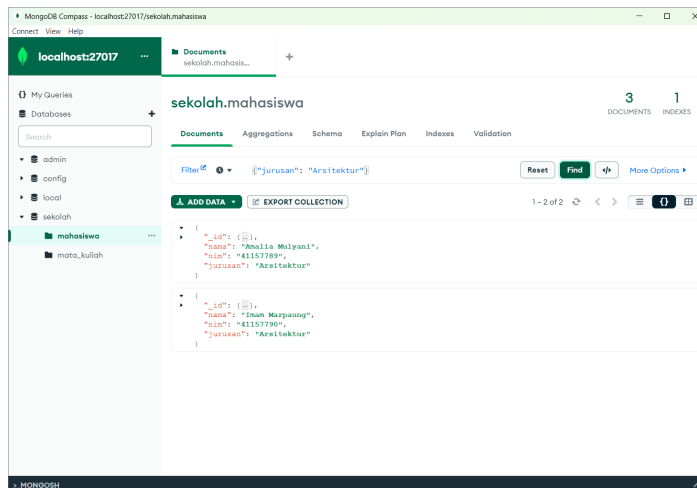


- Gunakan Fitur Find: Di bagian atas, Anda akan melihat kotak teks "Filter" di sebelah tombol "Find". Masukkan kriteria pencarian Anda di sini. Anda dapat menggunakan sintaks MongoDB standar untuk membangun kriteria pencarian Anda.

- Jalankan Query: Setelah Anda memasukkan kriteria pencarian Anda, tekan tombol "Find" untuk menjalankan query. Hasilnya akan ditampilkan di panel bawah.
- Filter data berdasarkan nama



- Filter data berdasarkan jurusan

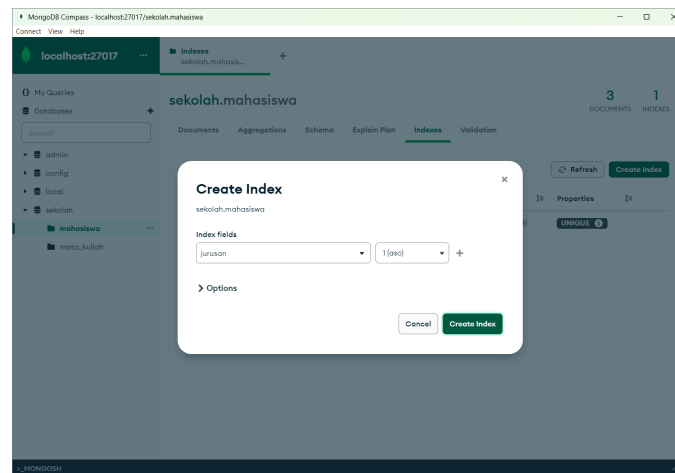


b. Indexing (Pengeindeksan)

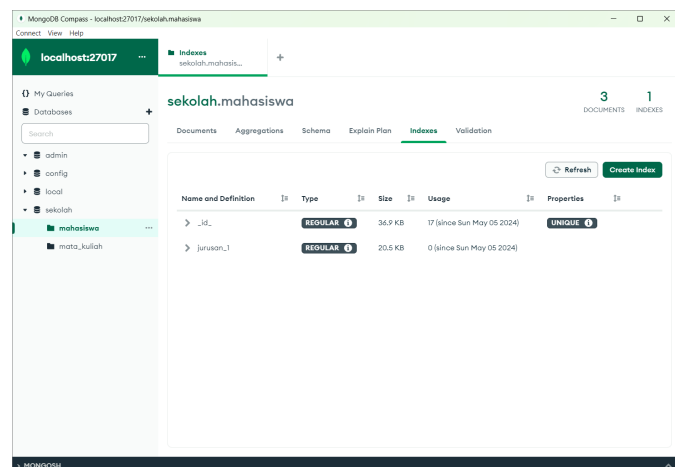
Pengeindeksan memungkinkan basis data untuk dengan cepat menemukan dokumen yang cocok dengan kriteria pencarian tertentu, mengurangi waktu yang dibutuhkan untuk menjalankan query dan meningkatkan kinerja keseluruhan sistem.

- Pilih Database dan Koleksi: Di panel kiri, pilih database dan koleksi di mana Anda ingin membuat indeks.
- Buka Tab "Indexes": Di bagian atas, klik tab "Indexes".
- Buat Indeks Baru: Di bagian kanan atas, Anda akan melihat tombol "Create Index". Klik tombol ini.

- **Atur Indeks:** Masukkan nama kolom yang ingin Anda indeks, serta pilihan lainnya seperti arah indeks (asc atau desc) dan jenis indeks (misalnya, indeks tunggal atau gabungan).



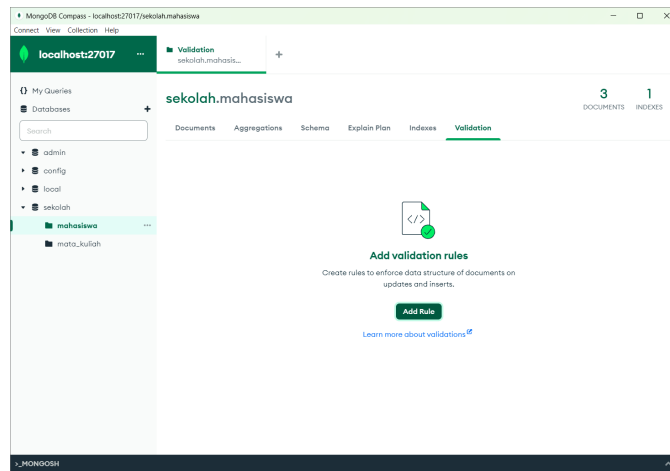
- **Konfirmasi:** Setelah Anda selesai mengatur, Compass akan membuat indeks baru untuk koleksi yang Anda pilih.



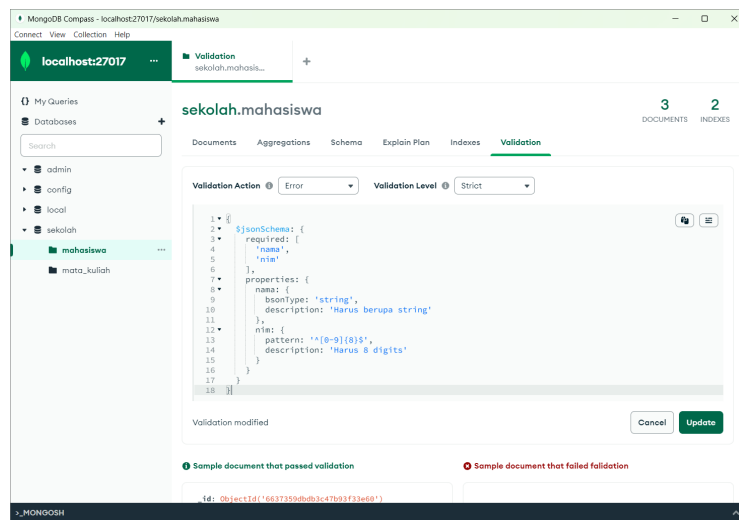
c. Validation

MongoDB Validation memungkinkan Anda untuk menentukan aturan validasi yang harus dipatuhi oleh dokumen sebelum dokumen tersebut disimpan dalam koleksi. Dengan menggunakan validasi, Anda dapat memastikan bahwa data yang dimasukkan ke dalam basis data memenuhi kriteria tertentu, sehingga menjaga konsistensi dan integritas data.

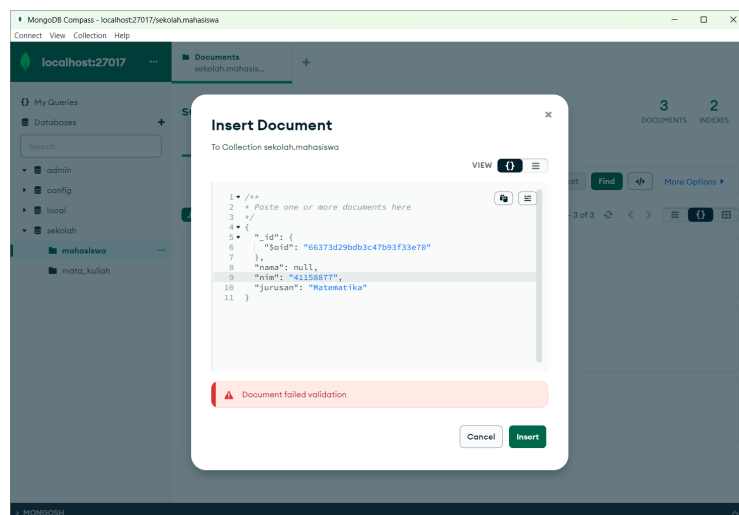
- **Pilih Database dan Koleksi:** Di panel kiri, pilih database dan koleksi di mana Anda ingin membuat validation.
- **Buka Tab "Indexes":** Di bagian atas, klik tab "Validation".
- **Buat Indeks Baru:** Anda akan melihat tombol "Add Rule". Klik tombol ini.



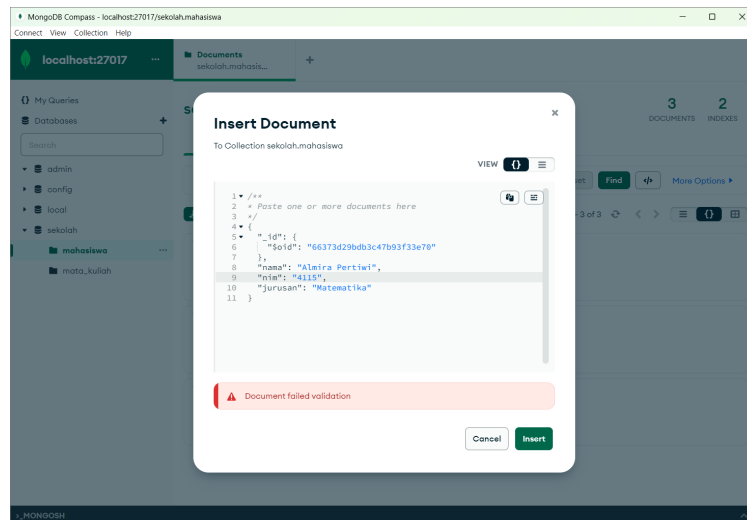
- Atur Rules dan klik tombol Update



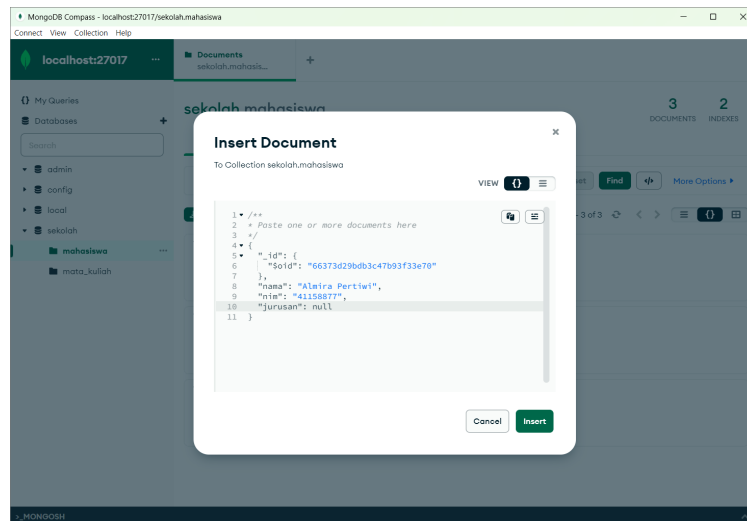
- Gagal insert document karena validation nama harus diisi



- Gagal insert document karena validation nim harus 8 digits



- Berhasil insert document walaupun jurusan tidak diisi



8. Integrasi dengan Python

Terakhir, pelajari cara mengintegrasikan MongoDB dengan aplikasi Anda. Anda dapat menggunakan driver MongoDB yang tersedia untuk berbagai bahasa pemrograman seperti Python, JavaScript, Java, dll.

Integrasi MongoDB dengan Python sangat umum dilakukan untuk pengembangan aplikasi berbasis data. MongoDB menyediakan driver resmi untuk Python yang disebut PyMongo, yang memungkinkan Anda untuk terhubung ke server MongoDB, menjalankan operasi database, dan mengelola data secara efisien menggunakan Python. Berikut adalah langkah-langkah dasar untuk mengintegrasikan MongoDB dengan Python menggunakan PyMongo:

a. Instalasi PyMongo

Pastikan Anda telah menginstal PyMongo. Jika belum, Anda dapat menginstalnya menggunakan pip dengan menjalankan perintah berikut di terminal atau command

prompt: "pip install pymongo"

```
C:\WINDOWS\system32\cmd. x + v
C:\Users\backenddev>pip install pymongo
Collecting pymongo
  Downloading pymongo-4.7.1-cp310-cp310-win_amd64.whl.metadata (22 kB)
Collecting dnspython<3.0.0,>=1.16.0 (from pymongo)
  Downloading dnspython-2.6.1-py3-none-any.whl.metadata (5.8 kB)
  Downloading pymongo-4.7.1-cp310-cp310-win_amd64.whl (484 kB)
    484.5/484.5 kB 689.4 kB/s eta 0:00:00
  Downloading dnspython-2.6.1-py3-none-any.whl (307 kB)
    307.7/307.7 kB 4.8 MB/s eta 0:00:00
Installing collected packages: dnspython, pymongo
Successfully installed dnspython-2.6.1 pymongo-4.7.1
C:\Users\backenddev>
```

b. Menghubungkan ke Server MongoDB

```
File Edit Selection ... TBD Modul MongoDB
EXPLORER
  OPEN EDITORS
    connect-mongo...
  TBD MODUL MONGODB
    connect-mongod...
connect-mongodb.py x
1 import pymongo
2
3 # Buat koneksi ke server MongoDB
4 client = pymongo.MongoClient("mongodb://localhost:27017/")
5
6 # Pilih database yang ingin Anda gunakan
7 mydb = client["sekolah"]
8
9 print ("mongodb berhasil terhubung")
10

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
PS D:\TBD Modul MongoDB> py hello-mongodb.py
mongodb berhasil terhubung
PS D:\TBD Modul MongoDB>
```

c. Menjalankan Operasi CRUD

● Read

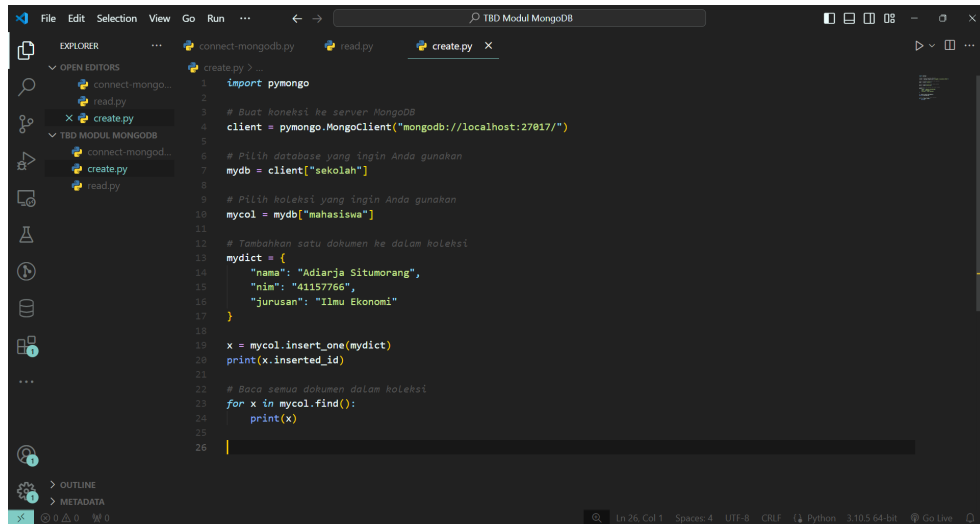
Code

```
File Edit Selection View Go ... TBD Modul MongoDB
EXPLORER
  OPEN EDITORS
    connect-mongo...
    read.py
  TBD MODUL MONGODB
    connect-mongo...
    read.py
connect-mongodb.py read.py x
1 import pymongo
2
3 # Buat koneksi ke server MongoDB
4 client = pymongo.MongoClient("mongodb://localhost:27017/")
5
6 # Pilih database yang ingin Anda gunakan
7 mydb = client["sekolah"]
8
9 # Pilih koleksi yang ingin Anda gunakan
10 mycol = mydb["mahasiswa"]
11
12 # Baca semua dokumen dalam koleksi
13 for x in mycol.find():
14     print(x)
15
16
```

Hasil

```
PS D:\TBD Modul MongoDB> py read.py
{'_id': ObjectId('6637359dbdb3c47b93f33e60'), 'nama': 'Amalia Mulyani', 'nim': '41157789', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373638bdb3c47b93f33e65'), 'nama': 'Imam Marpaung', 'nim': '41157790', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373665bdb3c47b93f33e68'), 'nama': 'Bakijan Maheswara', 'nim': '41157790', 'jurusan': 'Statistika'}
{'_id': ObjectId('66373d29bdb3c47b93f33e70'), 'nama': 'Almira Pertiwi', 'nim': '41158877', 'jurusan': None}
PS D:\TBD Modul MongoDB>
```

● Create Code

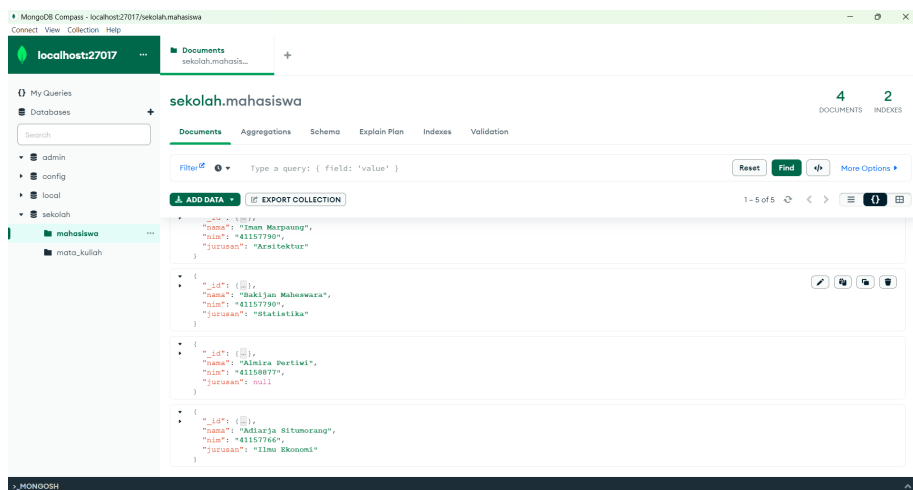


```
1 import pymongo
2
3 # Buat koneksi ke server MongoDB
4 client = pymongo.MongoClient("mongodb://localhost:27017/")
5
6 # Pilih database yang ingin Anda gunakan
7 mydb = client["sekolah"]
8
9 # Pilih koleksi yang ingin Anda gunakan
10 mycol = mydb["mahasiswa"]
11
12 # Tambahkan satu dokumen ke dalam koleksi
13 mydict = {
14     "nama": "Adiarja Situmorang",
15     "nim": "41157766",
16     "jurusan": "Ilmu Ekonomi"
17 }
18
19 x = mycol.insert_one(mydict)
20 print(x.inserted_id)
21
22 # Baca semua dokumen dalam koleksi
23 for x in mycol.find():
24     print(x)
25
26
```

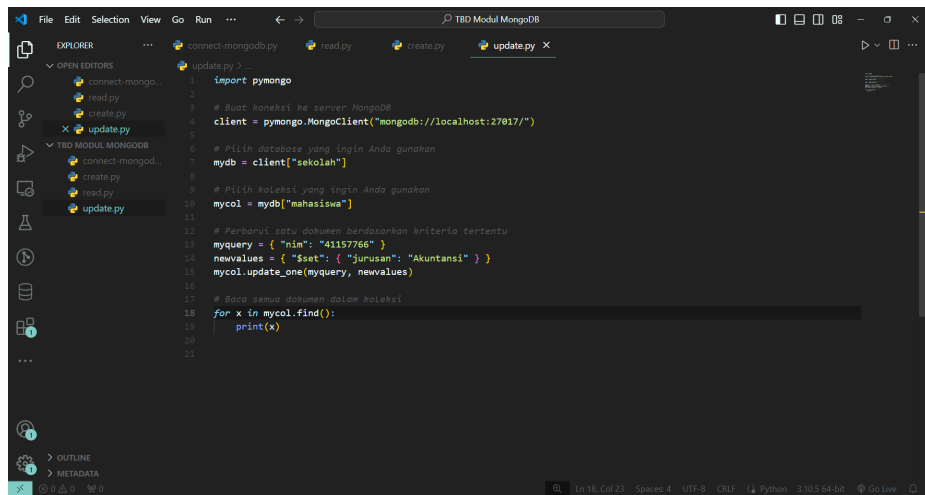
Hasil

Adiarja Situmorang berhasil ditambahkan

```
PS D:\TBD Modul MongoDB> py create.py
663741a913074e2b545f5fe2
{'_id': ObjectId('6637359dbdb3c47b93f33e60'), 'nama': 'Amalia Mulyani', 'nim': '41157789', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373638bdb3c47b93f33e65'), 'nama': 'Imam Marpaung', 'nim': '41157790', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373665bdb3c47b93f33e68'), 'nama': 'Bakijan Maheswara', 'nim': '41157790', 'jurusan': 'Statistika'}
{'_id': ObjectId('66373d29bdb3c47b93f33e70'), 'nama': 'Almira Pertiwi', 'nim': '41158877', 'jurusan': None}
{'_id': ObjectId('663741a913074e2b545f5fe2'), 'nama': 'Adiarja Situmorang', 'nim': '41157766', 'jurusan': 'Ilmu Ekonomi'}
PS D:\TBD Modul MongoDB>
```



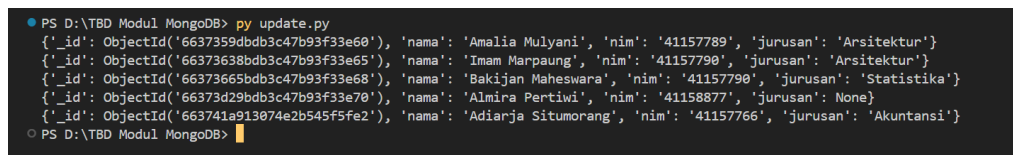
● Update Code



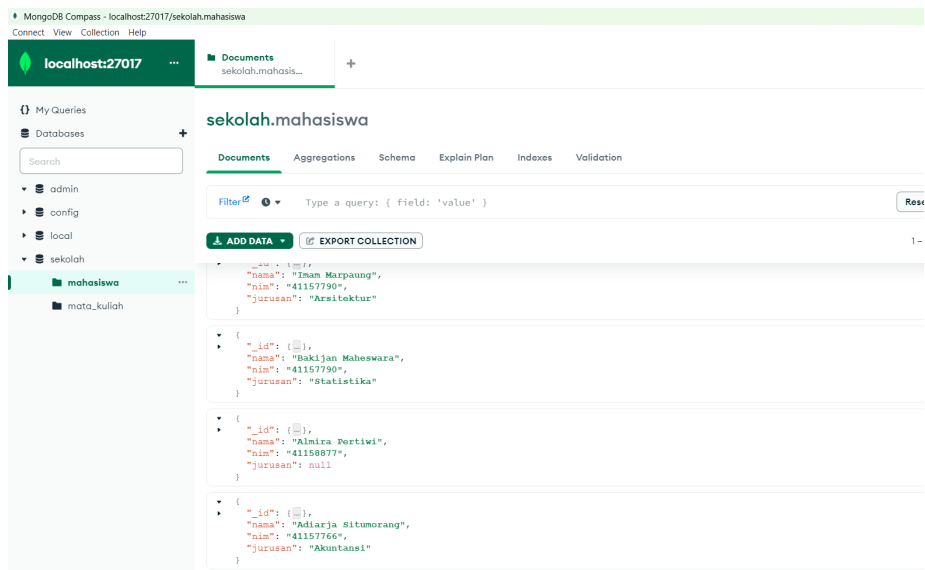
```
1 import pymongo
2
3 # Buat koneksi ke server MongoDB
4 client = pymongo.MongoClient("mongodb://localhost:27017/")
5
6 # Pilih database yang ingin Anda gunakan
7 mydb = client["sekolah"]
8
9 # Pilih koleksi yang ingin Anda gunakan
10 mycol = mydb["mahasiswa"]
11
12 # Perbarui satu dokumen berdasarkan kriteria tertentu
13 myquery = { "nim": "41157766" }
14 newvalues = { "$set": { "jurusan": "Akuntansi" } }
15 mycol.update_one(myquery, newvalues)
16
17 # Baca semua dokumen dalam koleksi
18 for x in mycol.find():
19     print(x)
```

Hasil

Mahasiswa dengan nim 41157766 berhasil diupdate

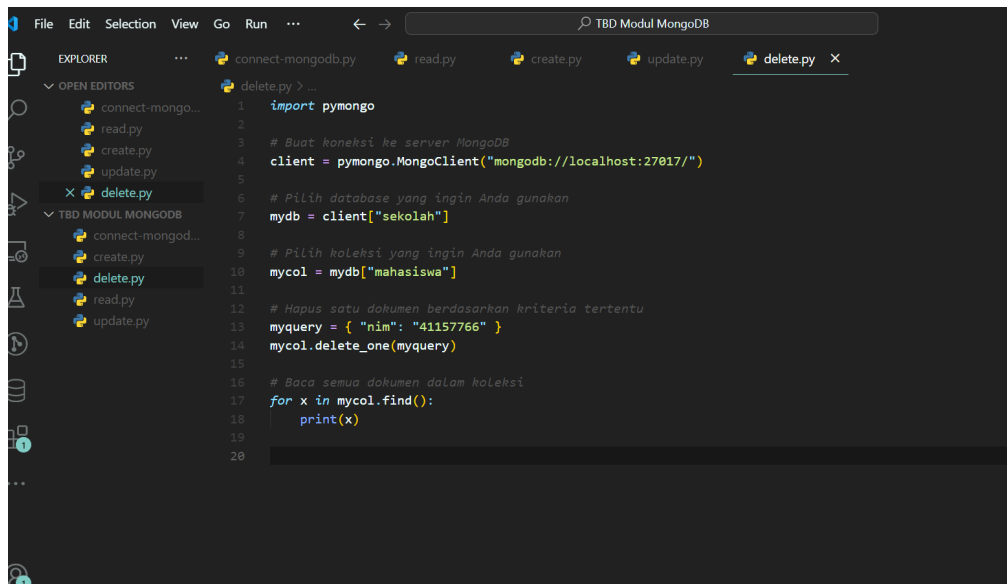


```
PS D:\TBD Modul MongoDB> py update.py
{'_id': ObjectId('6637359dbdb3c47b93f33e60'), 'nama': 'Amalia Mulyani', 'nim': '41157789', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373638bdb3c47b93f33e65'), 'nama': 'Imam Marpaung', 'nim': '41157790', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373665bdb3c47b93f33e68'), 'nama': 'Bakijan Maheswara', 'nim': '41157790', 'jurusan': 'Statistika'}
{'_id': ObjectId('66373d29bdb3c47b93f33e70'), 'nama': 'Almira Pertiwi', 'nim': '41158877', 'jurusan': None}
{'_id': ObjectId('663741a913074e2b545f5fe2'), 'nama': 'Adiarja Situmorang', 'nim': '41157766', 'jurusan': 'Akuntansi'}
```



● Delete

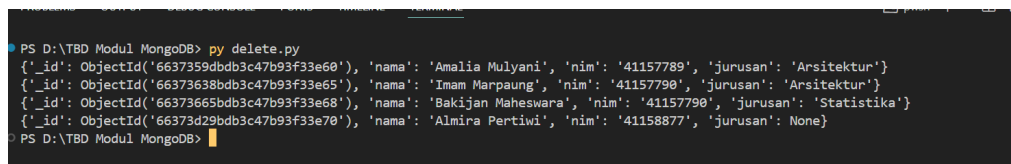
Code



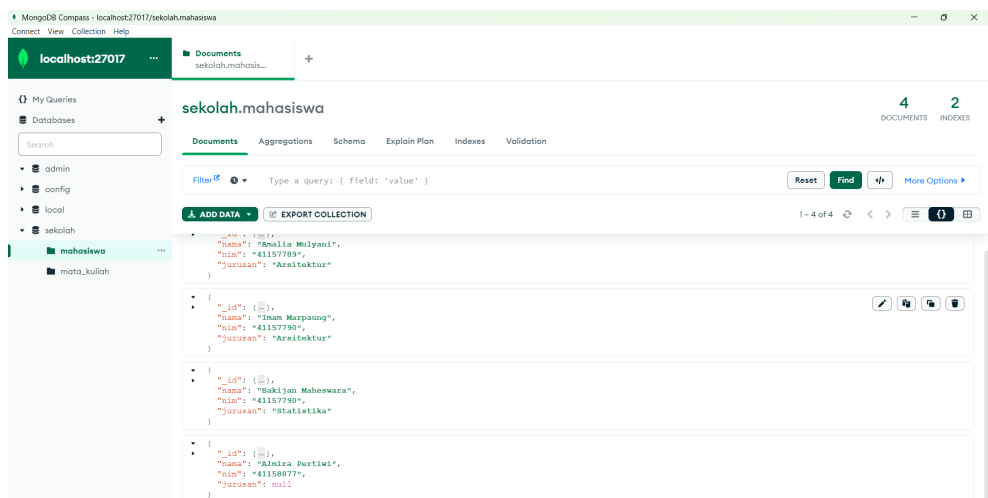
```
1 import pymongo
2
3 # Buat koneksi ke server MongoDB
4 client = pymongo.MongoClient("mongodb://localhost:27017/")
5
6 # Pilih database yang ingin Anda gunakan
7 mydb = client["sekolah"]
8
9 # Pilih koleksi yang ingin Anda gunakan
10 mycol = mydb["mahasiswa"]
11
12 # Hapus satu dokumen berdasarkan kriteria tertentu
13 myquery = { "nim": "41157766" }
14 mycol.delete_one(myquery)
15
16 # Baca semua dokumen dalam koleksi
17 for x in mycol.find():
18     print(x)
19
20
```

Hasil

Mahasiswa dengan nim 51157766 berhasil dihapus dari dokumen



```
PS D:\TBD Modul MongoDB> py delete.py
{'_id': ObjectId('6637359dbdb3c47b93f33e60'), 'nama': 'Amalia Mulyani', 'nim': '41157789', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373638bdb3c47b93f33e65'), 'nama': 'Imam Marpaung', 'nim': '41157790', 'jurusan': 'Arsitektur'}
{'_id': ObjectId('66373665bdb3c47b93f33e68'), 'nama': 'Bakijan Maheswara', 'nim': '41157790', 'jurusan': 'Statistika'}
{'_id': ObjectId('66373d29bdb3c47b93f33e70'), 'nama': 'Almira Pertiwi', 'nim': '41158877', 'jurusan': None}
PS D:\TBD Modul MongoDB>
```



Tugas Individu

1. Buat database dengan nama “NamaLengkapAnda”
2. Buat collection dengan nama “buku” dengan setruktur data seperti berikut
 - id
 - judul
 - pengarang
 - penerbit
 - tahun_terbit
 - kategori
3. Buat validation di collection “buku” dengan ketentuan seperti berikut
 - Semua key wajib diisi
 - Key tahun_terbit harus berupa angka / integer
 - Key selain tahun_terbit harus berupa string
4. Dengan menggunakan mongodb compass tambahkan buku dengan data seperti berikut

id	judul	pengarang	penerbit	tahun_terbit	kategori
	War and Peace	Leo Tolstoy	The Russian Messenger	1869	Fiksi
	The Catcher in the Rye	J. D. Salinger	Little, Brown and Company	1951	Fiksi
	The Hobbit	J. R. R. Tolkien	George Allen & Unwin	1937	Fiksi
	Frankenstein	Mary Shelley	Lackington, Hughes, Harding, Mavor, & Jones	1818	Fiksi
	Alice's Adventures in Wonderland	Lewis Carroll	Macmillan	1865	Fiksi

5. Buat skrip Python untuk menambahkan beberapa buku dan anggota perpustakaan ke dalam basis data MongoDB menggunakan PyMongo dengan data buku seperti berikut

id	judul	pengarang	penerbit	tahun_terbit	kategori
	The Hitchhiker's Guide to the Galaxy	Douglas Adams	Pan Books	1979	Fiksi Sains
	The Da Vinci Code	Dan Brown	Doubleday	2003	Misteri
	The Hunger Games	Suzanne Collins	Scholastic Corporation	2008	Fiksi Distopia
	Harry Potter and the Sorcerer's Stone	J. K. Rowling	Bloomsbury	1997	Fantasi
	A Brief History of Time	Stephen Hawking	Bantam Books	1988	Ilmu Pengetahuan

6. Dengan menggunakan mongodb compass tampilkan buku berdasarkan kategori Fiksi
7. Buat skrip Python untuk menampilkan buku berdasarkan kategori Fiksi
8. Buat skrip Python untuk menambahkan 3 buku kategori berbeda yang ingin anda buat.
Isi pengarang dengan nama anda dan tahun terbit 2024
9. Buat skrip Python untuk memperbarui buku dengan pengarang Stpehen Hawking
rubah kategori menjadi “Popular Science”
10. Buat skrip Python untuk menghapus buku yang dikarang oleh Lewis Carroll