

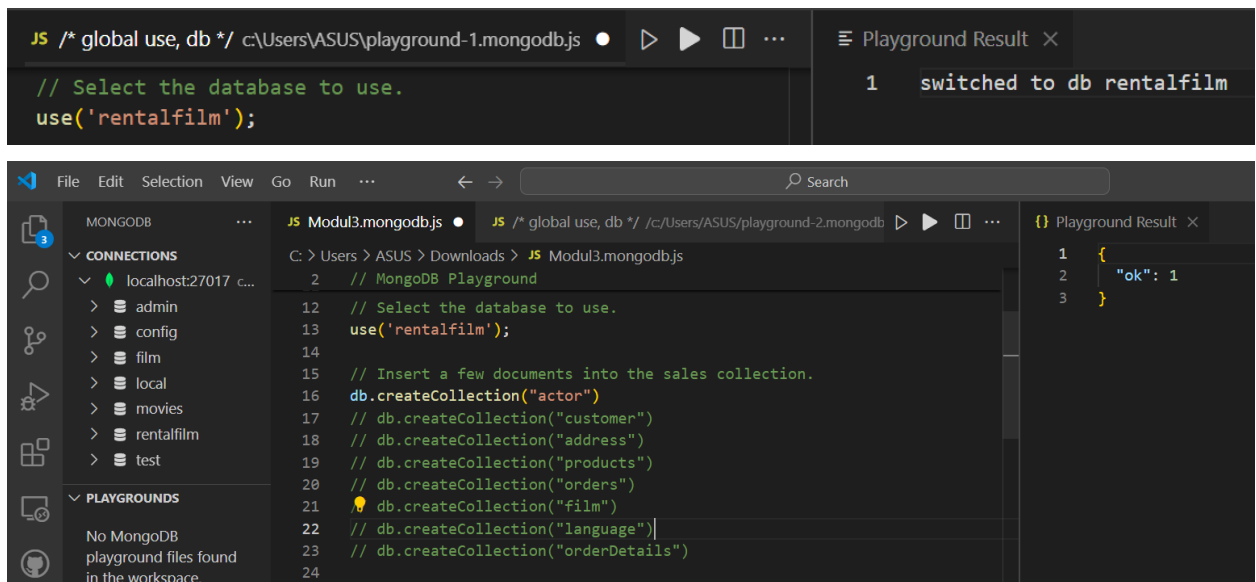
Nama : Frila Cahya Wardani

Kelas : TI – 21 – PA

NPM : 212310014

Lab. Basis Data Lanjut Modul 3

1. Buatlah database rentalfilm di mongoDB seperti yang telah dilakukan di pertemuan pertama.



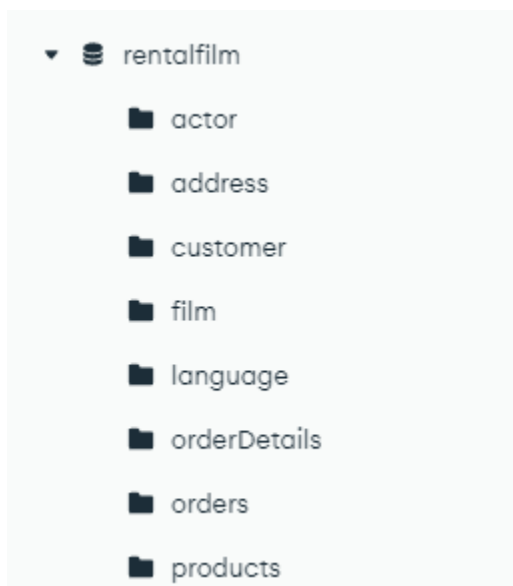
The image shows two screenshots related to MongoDB database setup. The top screenshot is a MongoDB Playground interface where the command `use('rentalfilm');` has been executed, resulting in the message "1 switched to db rentalfilm". The bottom screenshot is a VS Code editor showing a JavaScript file named `Modul3.mongodb.js`. The code in the file includes comments and commands to select the database and create several collections: `actor`, `customer`, `address`, `products`, `orders`, `film`, `language`, and `orderDetails`. The VS Code interface also shows a sidebar with a list of connections, including `localhost:27017` with a `rentalfilm` database selected. The `PLAYGROUNDS` section in the sidebar indicates that no MongoDB playground files were found in the workspace.

```
JS /* global use, db */ c:\Users\ASUS\playground-1.mongodb.js

// Select the database to use.
use('rentalfilm');
```

```
JS Modul3.mongodb.js JS /* global use, db */ /c:/Users/ASUS/playground-2.mongodb.js
2 // MongoDB Playground
12 // Select the database to use.
13 use('rentalfilm');
14
15 // Insert a few documents into the sales collection.
16 db.createCollection("actor")
17 // db.createCollection("customer")
18 // db.createCollection("address")
19 // db.createCollection("products")
20 // db.createCollection("orders")
21 // db.createCollection("film")
22 // db.createCollection("language")
23 // db.createCollection("orderDetails")
24
```

```
1 {
2   "ok": 1
3 }
```



2. Masukkan tiga buah data di setiap collections pada database rentalfilm menggunakan query.

```
JS Modul3.mongodbjs • JS /* global use, db */ /c:/Users/ASUS/playground-2.mongodbjs > ...
C: > Users > ASUS > Downloads > JS Modul3.mongodbjs > ...

25 db.customers.insert([
26   {
27     id: "Fernanda",
28     fullname: "Fernanda Ramos",
29     email: "fernadaramos4@uol.com.br",
30     age: 24
31   },
32   {
33     id: "Mark",
34     fullname: "Mark Philips",
35     email: "mphilips12@shaw.ca",
36     city: "San Francisco"
37   },
38   {
39     id: "Jennifer",
40     fullname: "Jennifer Peterson",
41     email: "jenniferp@rogers.ca",
42     occupation: "teacher"
43   }
44 ])

Playground Result ×
1 {
2   "acknowledged": true,
3   "insertedIds": {
4     "0": {
5       "$oid": "6630e17347f715273d220047"
6     },
7     "1": {
8       "$oid": "6630e17347f715273d220048"
9     },
10    "2": {
11      "$oid": "6630e17347f715273d220049"
12    }
13  }
14 }
```

rentalfilm > customers

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

[ADD DATA](#) [EXPORT DATA](#) [UPDATE](#) [DELETE](#)

```
_id: ObjectId('663120d3c6b740f95d7523d0')
id: "Fernanda"
fullname: "Fernanda Ramos"
email: "fernadaramos4@uol.com.br"
age: 24
```

```
_id: ObjectId('663120d3c6b740f95d7523d1')
id: "Mark"
fullname: "Mark Philips"
email: "mphilips12@shaw.ca"
city: "San Francisco"
```

```
_id: ObjectId('663120d3c6b740f95d7523d2')
id: "Jennifer"
fullname: "Jennifer Peterson"
email: "jenniferp@rogers.ca"
occupation: "teacher"
```

```
JS Modul3.mongodbj.js • JS /* global use, db */ /c:/Users/ASUS/playground-2.mongodbi ▶ ▶ □ ... {} Playground Result ✕
C: > Users > ASUS > Downloads > JS Modul3.mongodbj.js > id

44 db.actor.insert([
45   {
46     First_Name: "Frila",
47     Last_Name: "Cw",
48     Last_Update: "2024-03-11 10:46:54"
49   },
50   {
51     id: 2,
52     First_Name: "Frila",
53     Last_Name: "Cahya",
54     Last_Update: "2024-03-11 10:46:54"
55   },
56   {
57     id: 3,
58     First_Name: "Frila",
59     Last_Name: "Wardani",
60     Last_Update: "2024-03-11 10:46:54"
61   }
])

1 {
2   "acknowledged": true,
3   "insertedIds": {
4     "0": {
5       "$oid": "6630e2de600da93639ba9db9"
6     },
7     "1": {
8       "$oid": "6630e2de600da93639ba9dba"
9     },
10    "2": {
11      "$oid": "6630e2de600da93639ba9dbb"
12    }
13  }
14 }
```

rentalfilm > actor

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#) ↗

ADD DATA EXPORT DATA UPDATE DELETE

```
_id: ObjectId('6630e2de600da93639ba9db9')
id: 1
First_Name: "Frila"
Last_Name: "Cw"
Last_Update: "2024-03-11 10:46:54"

_id: ObjectId('6630e2de600da93639ba9dba')
id: 2
First_Name: "Frila"
Last_Name: "Cahya"
Last_Update: "2024-03-11 10:46:54"

_id: ObjectId('6630e2de600da93639ba9dbb')
id: 3
First_Name: "Frila"
Last_Name: "Wardani"
Last_Update: "2024-03-11 10:46:54"
```

```
JS Modul3.mongodbj.js • JS /* global use, db */ /c:/Users/ASUS/playground-2.mongodbi ▶ ▶ □ ... {} Playground Result ✕
C: > Users > ASUS > Downloads > JS Modul3.mongodbj.js > Tag

63 db.address.insertOne(
64   {
65     Nama_Produk: "Kulkas",
66     Harga: 10000000,
67     Berat_Produk: 8.9,
68     Rusak: false,
69     Tanggal_Masuk: new Date(),
70     Tag: ["Elektronik", "Perabotan"],
71     Dimensi: {
72       tinggi: 80,
73       panjang: 50,
74       lebar: 20
75     },
76     Id_Product: ObjectId()
77   }
78 )

1 {
2   "acknowledged": true,
3   "insertedId": {
4     "$oid": "6630e3b5d8c8998d7e26e230"
5   }
6 }
```


3. Lakukan minimal dua buah query update data dan delete data.

<pre>125 db.customers.update(126 { id: "Jennifer" }, 127 { \$set: { fullname: 'Jennifer Aniston' } } 128) 129 130 db.customers.update(131 { id: "Mark" }, 132 { \$set: { fullname: 'Mark Daihatsu' } } 133)</pre>	<pre>1 { 2 "acknowledged": true, 3 "insertedId": null, 4 "matchedCount": 1, 5 "modifiedCount": 1, 6 "upsertedCount": 0 7 }</pre>
---	--

```
_id: ObjectId('663120d3c6b740f95d7523d1')
id: "Mark"
fullname: "Mark Daihatsu"
email: "mphilips12@shaw.ca"
city: "San Francisco"
```

```
_id: ObjectId('663120d3c6b740f95d7523d2')
id: "Jennifer"
fullname: "Jennifer Aniston"
email: "jenniferp@rogers.ca"
occupation: "teacher"
```

<pre>135 db.address.remove(136 {Nama_Produk: "Kursi"} 137) 138 139 db.address.remove(140 {Nama_Produk: "Meja"} 141)</pre>	<pre>1 { 2 "acknowledged": true, 3 "deletedCount": 1 4 }</pre>
---	--

rentalfilm > address

Documents 0 Aggregations Schema Indexes 1 Validation

Type a query: { field: 'value' } or [Generate query](#)

ADD DATA EXPORT DATA UPDATE DELETE

```
_id: ObjectId('663121e56460de0c2af0f70b')
Nama_Produk: "TV"
Harga: 5000000
Berat_Produk: 4.5
Rusak: false
Tanggal_Masuk: 2024-04-30T16:52:53.391+00:00
Tag: Array (2)
Dimensi: Object
Id_Product: ObjectId('663121e56460de0c2af0f708')
```

```
_id: ObjectId('663121fb3e42e9381b203479')
Nama_Produk: "Kulkas"
Harga: 10000000
Berat_Produk: 8.9
Rusak: false
Tanggal_Masuk: 2024-04-30T16:53:15.680+00:00
Tag: Array (2)
Dimensi: Object
Id_Product: ObjectId('663121fb3e42e9381b203478')
```