

## Practical 4

### MongoDB Indexes

```
C:\Users\Admin>mongosh
```

```
test> use students
```

```
switched to db students
```

```
students>
```

```
db.studgrades.insertMany([{"name":"Barry",subject:"maths",score:"92"},{"name":"Barry",subject:"maths",score:"92"},{"name":"Kent",subject:"Physics",score:"87"},{"name":"harry",subject:"maths",score:"99",notes:"exceptional"},{"name":"Alex",subject:"Literature",score:"78"},{"name":"Tom",subject:"History",score:"65",notes:"adequate"}]);
```

```
{
  acknowledged: true,
  insertedIds: {
    '0': ObjectId('678a2996c9185666448bf202'),
    '1': ObjectId('678a2996c9185666448bf203'),
    '2': ObjectId('678a2996c9185666448bf204'),
    '3': ObjectId('678a2996c9185666448bf205'),
    '4': ObjectId('678a2996c9185666448bf206'),
    '5': ObjectId('678a2996c9185666448bf207')
  }
}
```

```
students> db.studgrades.find({}, {_id:0});
```

```
[
  { name: 'Barry', subject: 'maths', score: '92' },
  { name: 'Barry', subject: 'maths', score: '92' },
  { name: 'Kent', subject: 'Physics', score: '87' },
  {
    name: 'harry',
    subject: 'maths',
    score: '99',
    notes: 'exceptional'
  },
  { name: 'Alex', subject: 'Literature', score: '78' },
  { name: 'Tom', subject: 'History', score: '65', notes: 'adequate' }
]
```

```
students> db.studgrades.find().pretty();
```

```
[
  {
    _id: ObjectId('678a2996c9185666448bf202'),
    name: 'Barry',
```

```

    subject: 'maths',
    score: '92'
  },
  {
    _id: ObjectId('678a2996c9185666448bf203'),
    name: 'Barry',
    subject: 'maths',
    score: '92'
  },
  {
    _id: ObjectId('678a2996c9185666448bf204'),
    name: 'Kent',
    subject: 'Physics',
    score: '87'
  },
  {
    _id: ObjectId('678a2996c9185666448bf205'),
    name: 'harry',
    subject: 'maths',
    score: '99',
    notes: 'exceptional'
  },
  {
    _id: ObjectId('678a2996c9185666448bf206'),
    name: 'Alex',
    subject: 'Literature',
    score: '78'
  },
  {
    _id: ObjectId('678a2996c9185666448bf207'),
    name: 'Tom',
    subject: 'History',
    score: '65',
    notes: 'adequate'
  }
]

```

```

students> db.studgrades.createIndex({name:1},{name:"student name index"});
student name index

```

```

students> db.studgrades.getIndexes();
[
  { v: 2, key: { _id: 1 }, name: '_id_' },
  { v: 2, key: { name: 1 }, name: 'student name index' }
]

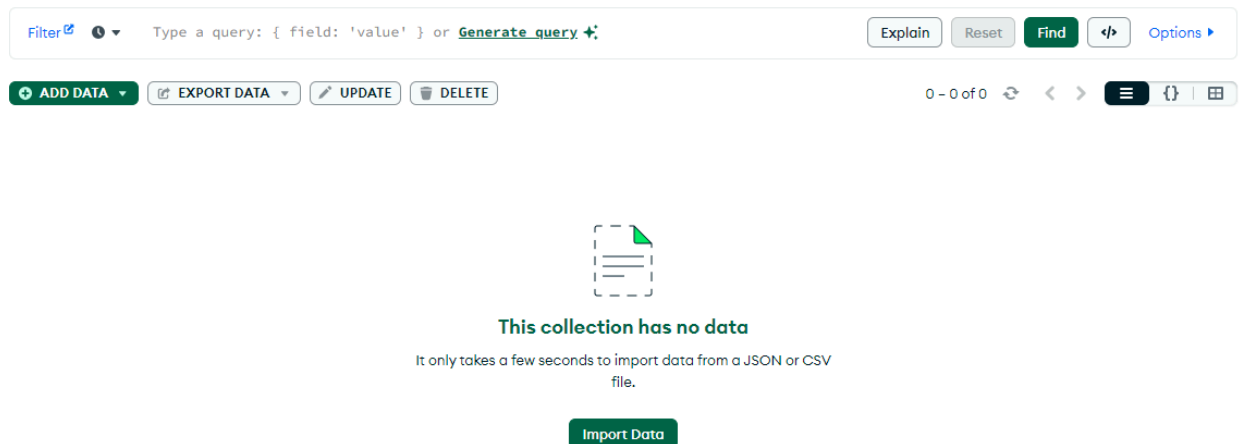
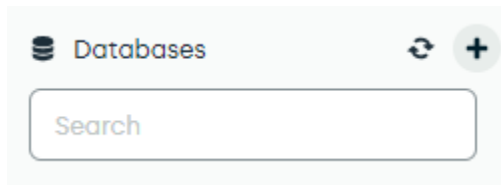
```

```
]
students> db.studgrades.dropIndex("student name index"); OR
db.studgrades.dropIndex({name:1});
{ nIndexWas: 2, ok: 1 }

students> db.studgrades.dropIndexes();
{
  nIndexWas: 1,
  msg: 'non-_id indexes dropped for collection',
  ok: 1
}
```

## Creating Indexes in MongoDB Compass

1. Create a new database by clicking the + button



## 2. Import data

The screenshot shows the MongoDB Compass interface for the 'airport' collection in the 'db1' database. The left sidebar shows the database structure with 'airport' selected. The main panel displays two documents. The first document has an \_id of '678a2caf4cb15ad9b6c587ea' and contains fields for origin and destination airports, cities, passengers, seats, flights, distance, fly date, and populations. The second document has an \_id of '678a2caf4cb15ad9b6c587eb' and contains similar fields for a different route. The top right shows 3.6m documents and 1 index.

## 3. Navigate to Indexes and click on create indexes

The screenshot shows the 'Indexes' tab for the 'airport' collection. The 'Create Index' button is highlighted in green. The 'Refresh' button is also visible. The top right shows 3.6m documents and 1 index. The bottom of the interface shows a 'VIEWING' tab with 'INDEXES' selected.

## 4.

The screenshot shows the 'Create Index' dialog box. The database is 'db1.airport'. The 'Index fields' section has a dropdown menu for 'Select or type a field name' and a dropdown menu for 'Select a type'. The 'Options' section is expanded, and the 'Create Index' button is highlighted in green. The 'Cancel' button is also visible.

5. Select any field and type of your choice. You can explore more properties from options.

db1.airport

3.6m

1

DOCUMENTSINDEXES

DocumentsAggregationsSchemaIndexesValidation

Create IndexRefreshVIEWINGINDEXESSearch Indexes

Name and Definition	Type	Size	Usage	Properties
> Dest_airport_lat_1	REGULAR	24.8 MB	0 (since Fri Jan 17 2025)	
> Destination_city_text	TEXT	54.9 MB	0 (since Fri Jan 17 2025)	
> Fly_date_1	REGULAR	21.8 MB	0 (since Fri Jan 17 2025)	
> _id_	REGULAR	39.9 MB	2 (since Fri Jan 17 2025)	UNIQUE