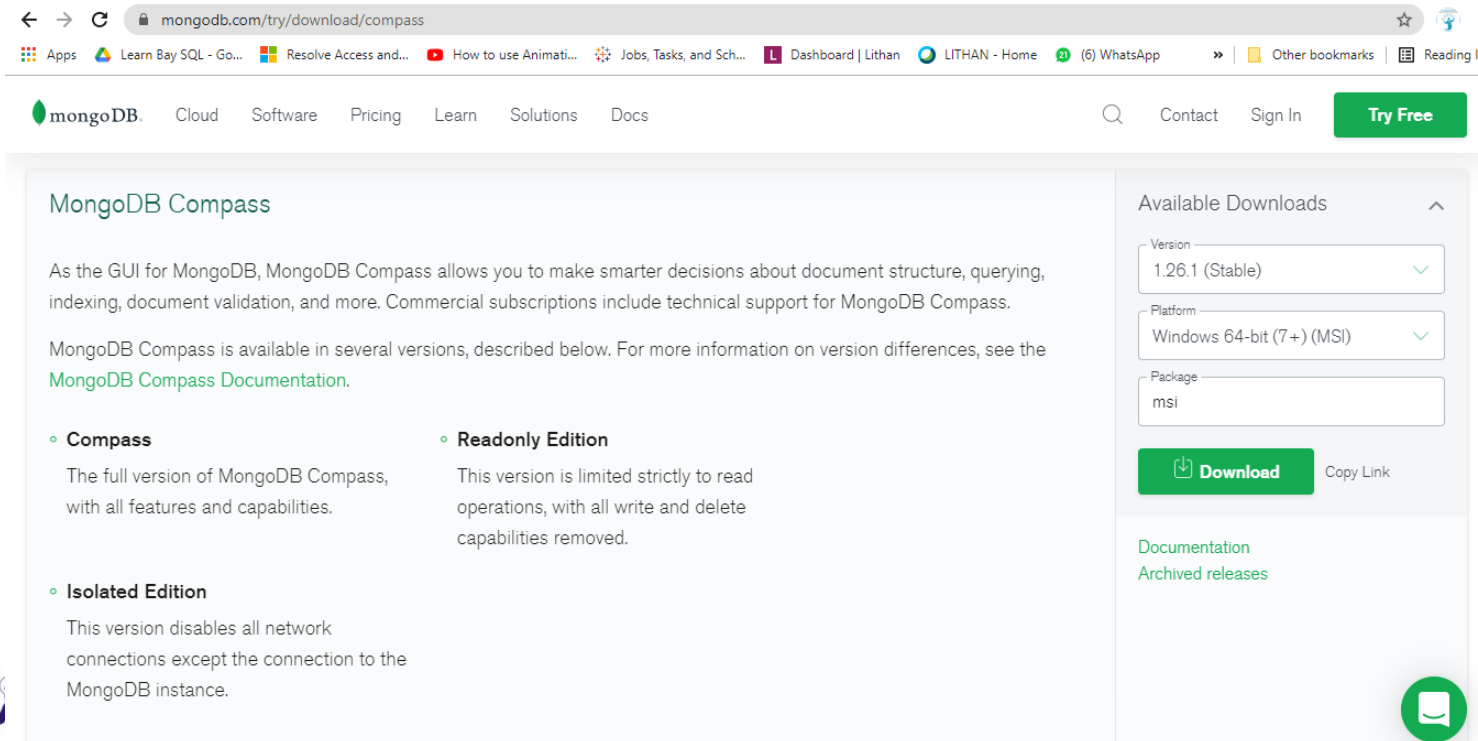


# MongoDB Compass

It's a GUI Interface for handling MongoDB database. It is a convenient tool for performing all CRUD operations without manually writing queries. It helps in many activities such as indexing, document validation, etc.

<https://www.mongodb.com/try/download/compass>



The screenshot shows the MongoDB Compass download page. The browser address bar displays 'mongodb.com/try/download/compass'. The page header includes the MongoDB logo and navigation links: Cloud, Software, Pricing, Learn, Solutions, Docs, Contact, Sign In, and a 'Try Free' button. The main content area is titled 'MongoDB Compass' and describes it as the GUI for MongoDB. It lists three versions: Compass, Readonly Edition, and Isolated Edition, each with a brief description of its capabilities. On the right side, there is a section for 'Available Downloads' with dropdown menus for Version (1.26.1 (Stable)), Platform (Windows 64-bit (7+) (MSI)), and Package (msi). Below these dropdowns are 'Download' and 'Copy Link' buttons. At the bottom right, there are links for 'Documentation' and 'Archived releases'.

**MongoDB Compass**

As the GUI for MongoDB, MongoDB Compass allows you to make smarter decisions about document structure, querying, indexing, document validation, and more. Commercial subscriptions include technical support for MongoDB Compass.

MongoDB Compass is available in several versions, described below. For more information on version differences, see the [MongoDB Compass Documentation](#).

- **Compass**  
The full version of MongoDB Compass, with all features and capabilities.
- **Readonly Edition**  
This version is limited strictly to read operations, with all write and delete capabilities removed.
- **Isolated Edition**  
This version disables all network connections except the connection to the MongoDB instance.

**Available Downloads**

Version: 1.26.1 (Stable) ✓

Platform: Windows 64-bit (7+) (MSI) ✓

Package: msi

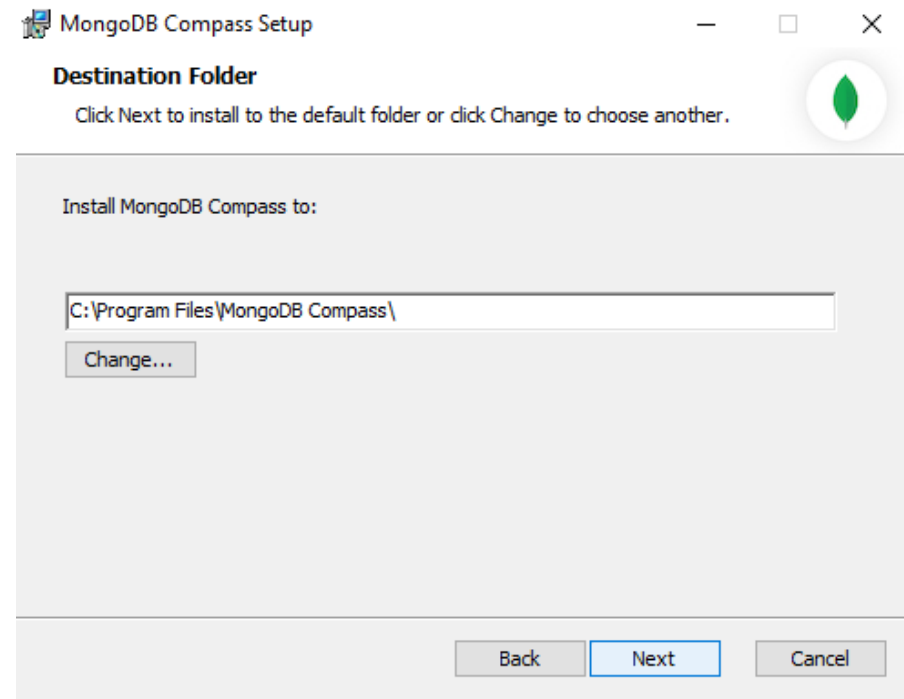
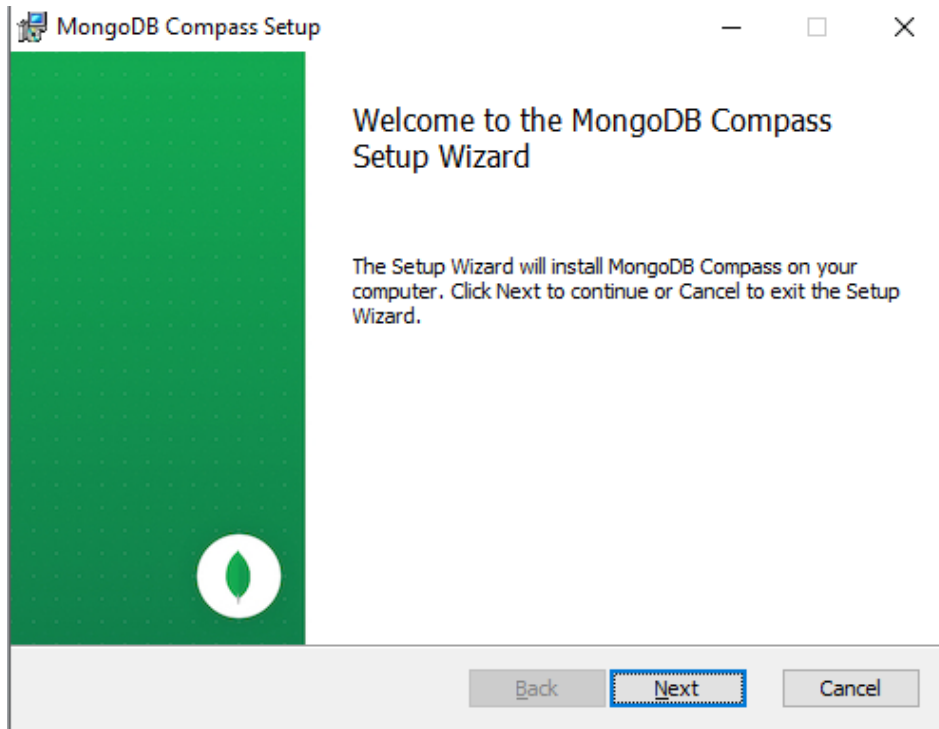
[Download](#) [Copy Link](#)

[Documentation](#)  
[Archived releases](#)



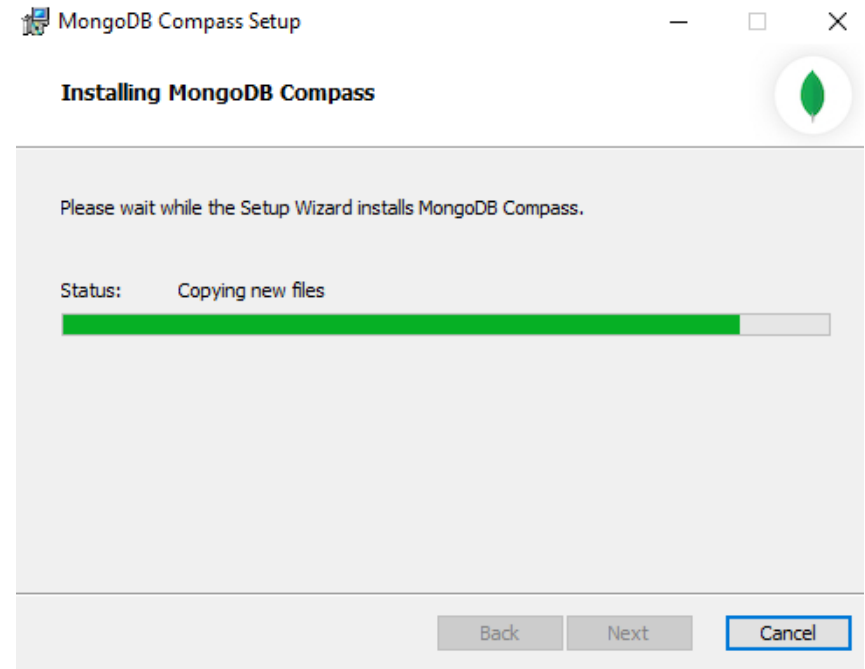
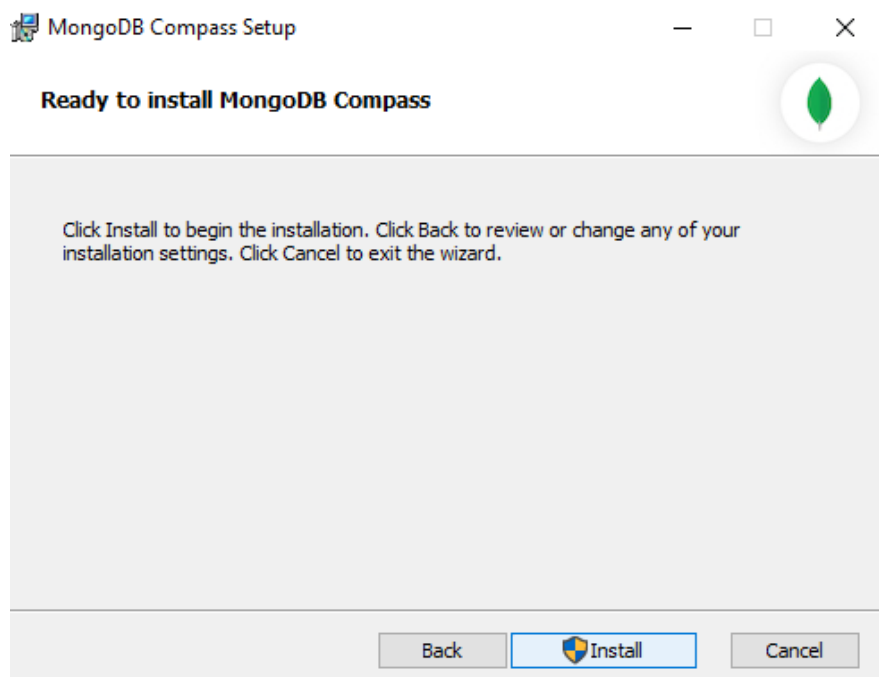
# MongoDB Compass

Double click on the downloaded file to initiate the installation:



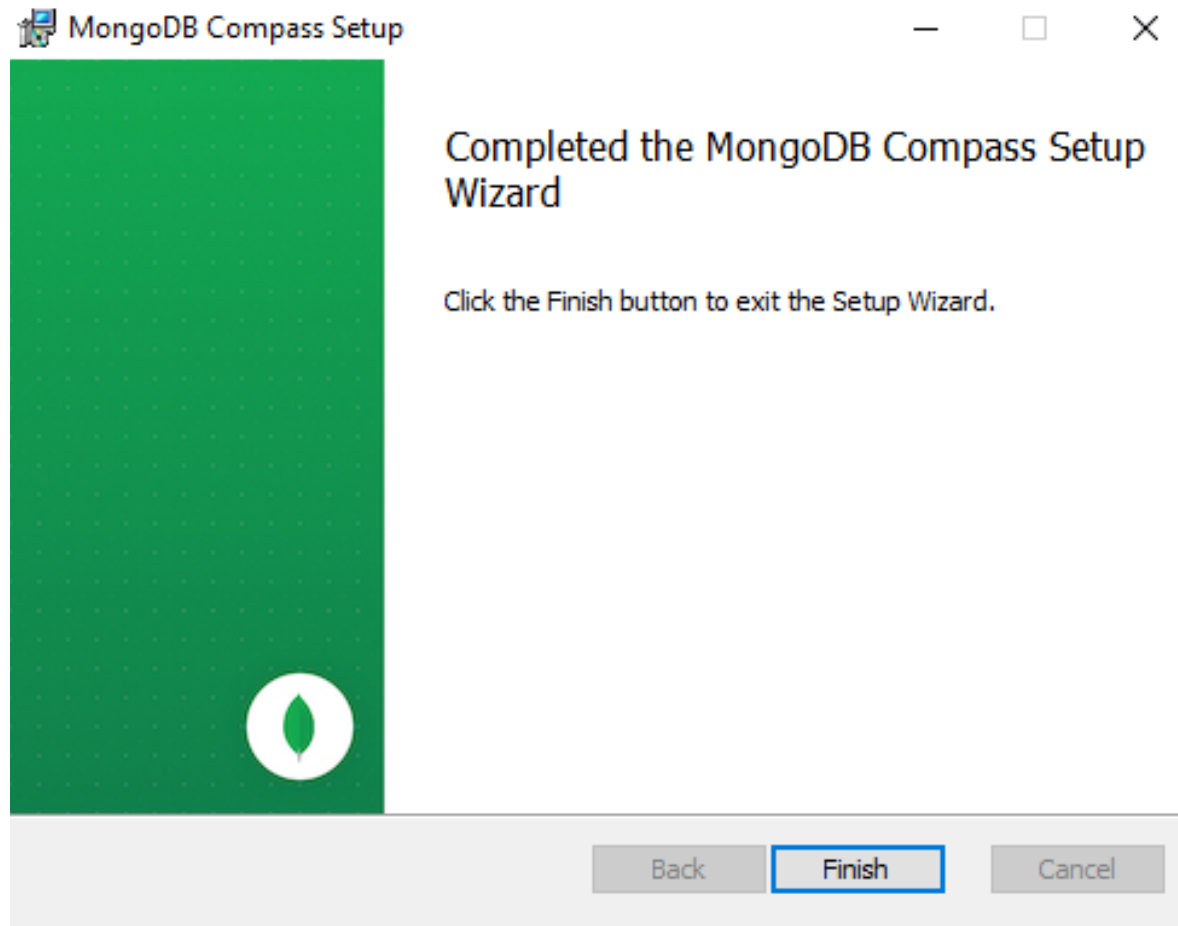
# MongoDB Compass

Click on Install proceed with the installation:



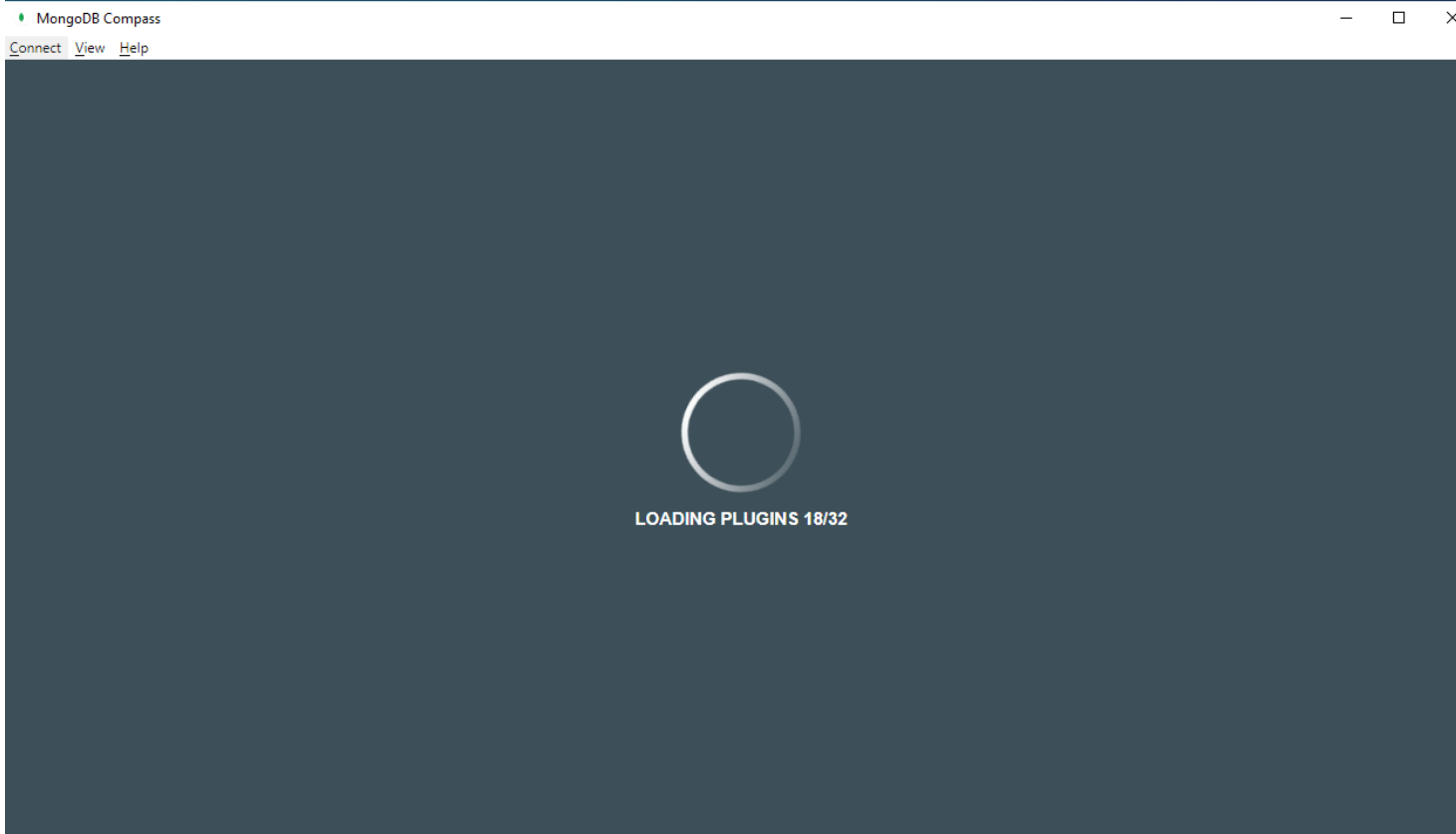
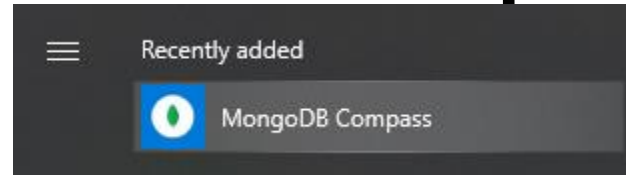
# MongoDB Compass

Click on Finish to complete the installation:



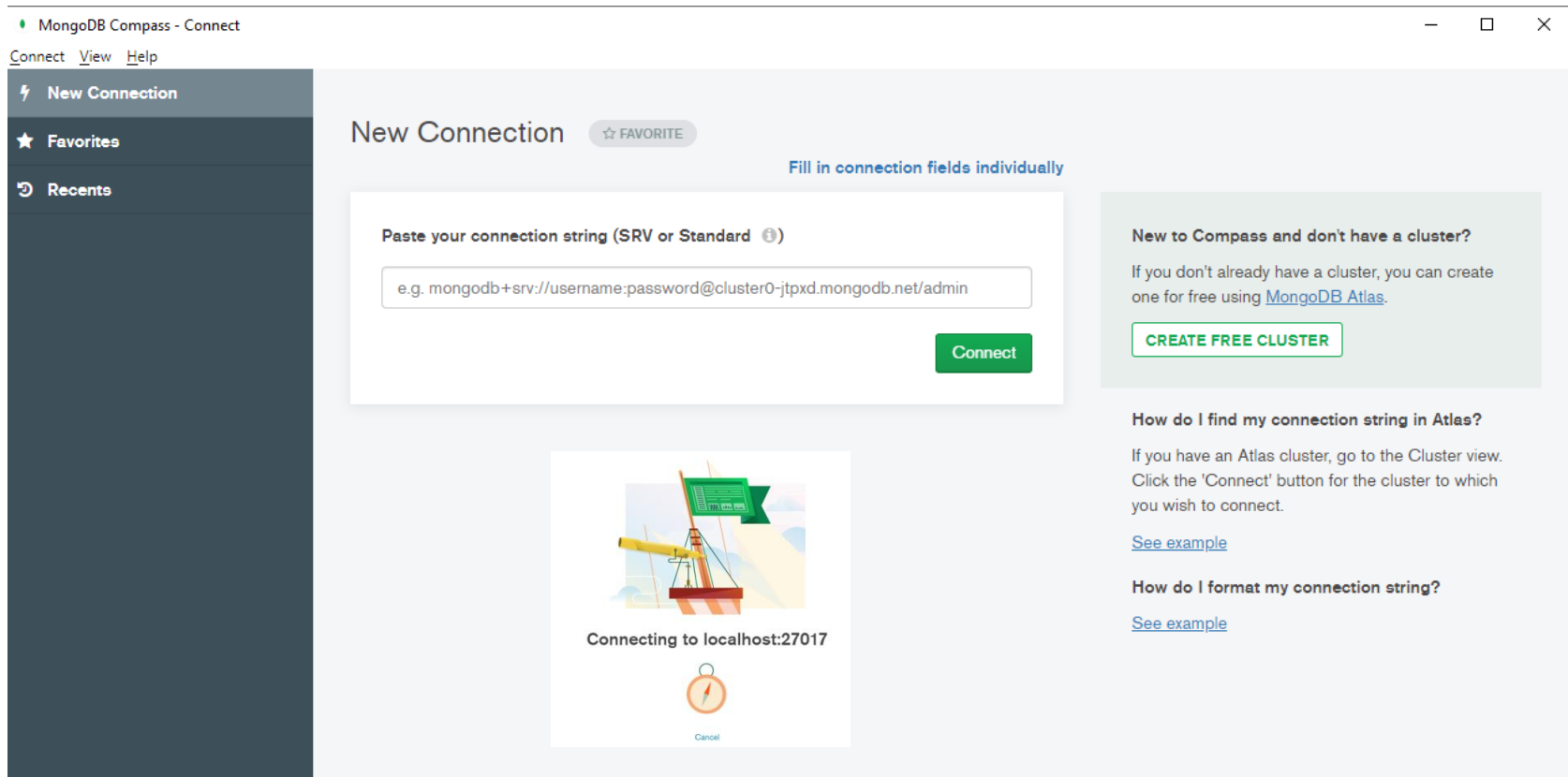
# MongoDB Compass

Go to Start Menu and click on



# MongoDB Compass

Connect to the cluster (server). We will be connecting to the local host.  
Provide the connection string and click on connect



NOTE: Mongo server must be running on your system to connect to the host



# MongoDB Compass

Once connected you will be able to see all the databases in the mongo

MongoDB Compass - localhost:27017

Connect View Help

Local

9 DBS 40 COLLECTIONS

☆ FAVORITE

HOST  
localhost:27017

CLUSTER  
Standalone

EDITION  
MongoDB 4.4.5 Enterprise

Q Filter your data

- > admin
- > batch1
- > batch2
- > config
- > demo
- > img
- > local
- > mydb
- > test

+

Databases Performance

CREATE DATABASE

Database Name ^	Storage Size	Collections	Indexes	
admin	32.0KB	0	1	
batch1	224.0KB	7	8	
batch2	64.0KB	2	2	
config	12.0KB	0	2	
demo	32.0KB	1	1	
img	5.3MB	2	4	
local	36.0KB	1	1	
mydb	36.0KB	2	2	
test	2.7MB	25	30	



# MongoDB Compass

MongoDb Compass allows us to perform below operations:

- Create Database
- Create Collection
- Perform CRUD operations
- Add data to collection
  - Import File
  - Insert Document
- Use Options
  - Filter
  - Project
  - Sort
- Aggregations
- Indexes
- Validation





# MongoDB Compass

## Create Database

MongoDB Compass - localhost:27017

Connect View Help

Local

9 DBS 40 COLLECTIONS

☆ FAVORITE

HOST  
localhost:27017

CLUSTER  
Standalone

EDITION  
MongoDB 4.4.5 Enterprise

Filter your data

- > admin
- > batch1
- > batch2
- > config
- > demo
- > img
- > local
- > mydb
- > test

+

Databases Performance

CREATE DATABASE

Database Name ^

- admin
- batch1
- batch2
- config
- demo
- img
- local
- mydb
- test

Indexes

1	
8	
2	
2	
1	
4	
1	
2	
25	
30	

Minimise

### Create Database

Database Name

Test2

Collection Name

emp

☐ Capped Collection ⓘ

☐ Use Custom Collation ⓘ

Before MongoDB can save your new database, a collection name must also be specified at the time of creation. [More Information](#)

CANCEL CREATE DATABASE



# MongoDB Compass

## Add data to collection

MongoDB Compass - localhost:27017/test.salary

Connect View Collection Help

The screenshot shows the MongoDB Compass interface with a modal dialog box open for inserting data into the 'test.salary' collection. The dialog box has a title bar 'Insert to Collection test.salary' and a 'VIEW' section with a JSON icon selected. The main area contains a text editor with the following content:

```
1 /**
2  * Paste one or more documents here
3  */
4 {
5   "id": {
6     "$oid": "6097ca9d6fed09228188b68f"
7   }
8 }
```

At the bottom of the dialog box are 'CANCEL' and 'INSERT' buttons. The background interface shows the 'test.salary' collection with a filter bar, an 'ADD DATA' button, and a table of documents. The table has columns for '\_id', 'EID', 'DEPT', 'DESI', and 'SALARY'. The first document is visible with the following values:

_id	EID	DEPT	DESI	SALARY
Object	1060	TECH	ASSISTANT	2500



# MongoDB Compass

## Aggregations

MongoDB Compass - localhost:27017/test.salary

Connect View Collection Help

Local

9 DBS 40 COLLECTIONS

☆ FAVORITE

HOST  
localhost:27017

CLUSTER  
Standalone

EDITION  
MongoDB 4.4.5 Enterprise

Filter your data

- admin
- batch1
- batch2
- config
- demo
- img
- local
- mydb
- test
- abc
- abcd

test.salary

Aggregations

DOCUMENTS 55 TOTAL SIZE 4.2KB AVG. SIZE 78B INDEXES 1 TOTAL SIZE 32.0KB AVG. SIZE 32.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

COLLATION Untitled- Modified SAVE SAMPLE MODE AUTO PREVIEW

55 Documents in the Collection

Preview of Documents in the Collection

Select an operator to construct expressions used in the aggregation pipeline stages. [Learn more](#)

Output after \$group stage (Sample of 7 documents)

```
1 /**
2  * _id: The id of the group.
3  * fieldN: The first field name.
4  */
5 {
6   _id: "$DEPT",
7   TOTALCOST: { $sum: "$SALARY" },
8   AvgSal: { $avg: "$SALARY" },
9   MinSal: { $min: "$SALARY" },
10  MaxSal: { $max: "$SALARY" }
11 }
```

Document 1:

\_id: ObjectId("608a7474a538f278ddd80e5a")  
EID: 1004  
DEPT: "MIS"  
DESI: "Manager"  
SALARY: 134789

Document 2:

\_id: ObjectId("608a7474a538f278ddd80e5b")  
EID: 1001  
DEPT: "OPS"  
DESI: "Director"  
SALARY: 380000

Document 3:

\_id: "HR"  
TOTALCOST: 1350000  
AvgSal: 150000  
MinSal: 150000  
MaxSal: 150000

Document 4:

\_id: "mis"  
TOTALCOST: 153767  
AvgSal: 153767  
MinSal: 153767  
MaxSal: 153767



# MongoDB Compass

## Validations

MongoDB Compass - localhost:27017/test.salary

Connect View Collection Help

Local

9 DBS 40 COLLECTIONS

☆ FAVORITE

HOST  
localhost:27017

CLUSTER  
Standalone

EDITION  
MongoDB 4.4.5 Enterprise

Filter your data

- admin
- batch1
- batch2
- config
- demo
- img
- local
- mydb
- test
- abc
- abod

test.salary Validation

test.salary

DOCUMENTS 55 TOTAL SIZE 4.2KB AVG. SIZE 78B INDEXES 1 TOTAL SIZE 32.0KB AVG. SIZE 32.0KB

Documents Aggregations Schema Explain Plan Indexes Validation

```
1 {  
2   SALARY: 1  
3   $gt: 20000  
4 }  
5 }
```

✓ Sample Document That Passed Validation

```
_id: ObjectId("608a7474a538f278ddd80e5a")  
EID: 1004  
DEPT: "MIS"  
DESI: "Manager"  
SALARY: 134789
```

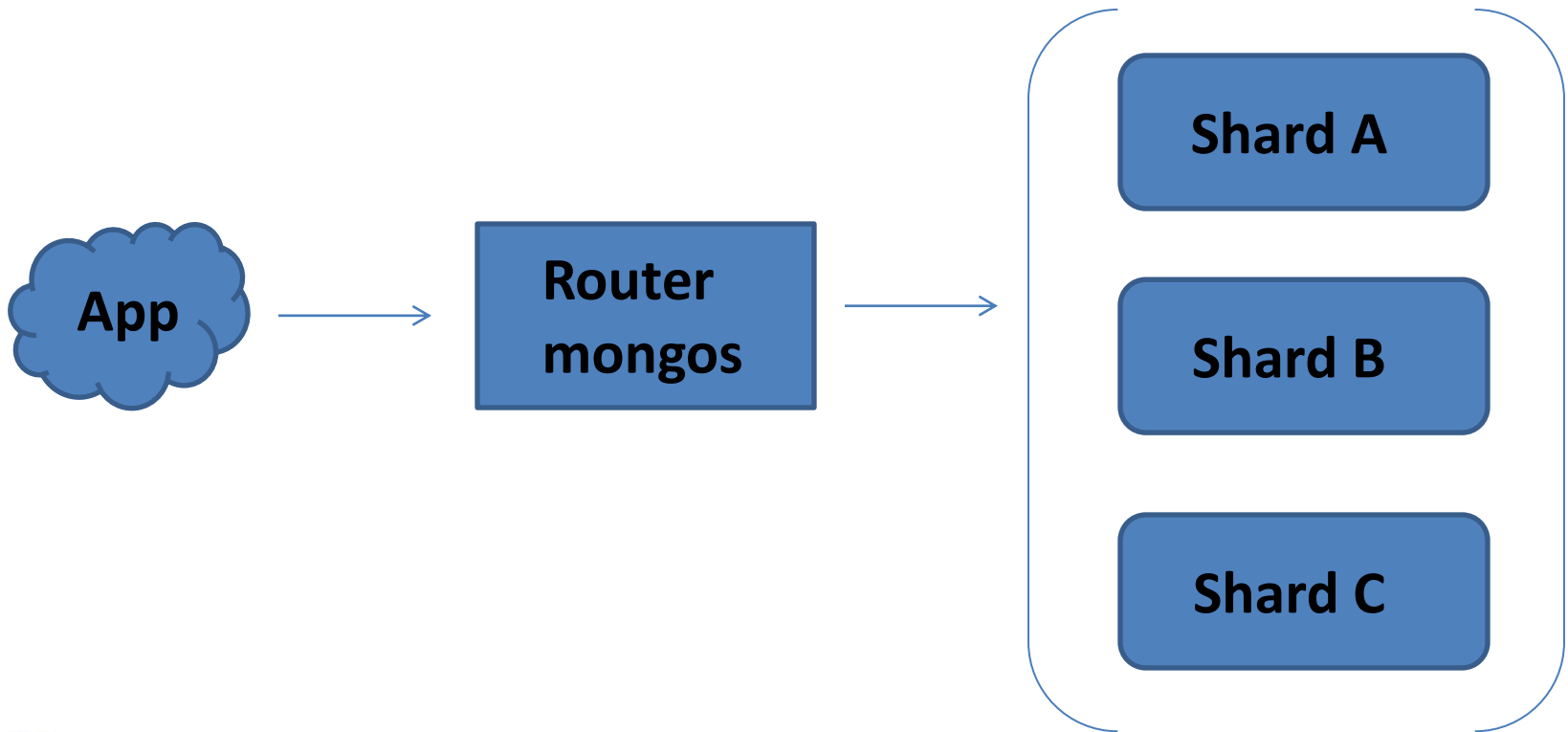
✗ Sample Document That Failed Validation

No Preview Documents



# Sharding MongoDB

Sharding is the process of partitioning your data across multiple servers. It is a type of database partitioning that separates very large database into faster, smaller and more easily manageable parts called shards.



# Sharding MongoDB

MongoDB uses the shard key to distribute the collection's documents across shards. The shard key consists of a field or multiple fields in the documents.

## Why Sharding ?

Scalable - data is growing continuously

High Availability

Ability to control data distribution

Application Transparent

Cost effective

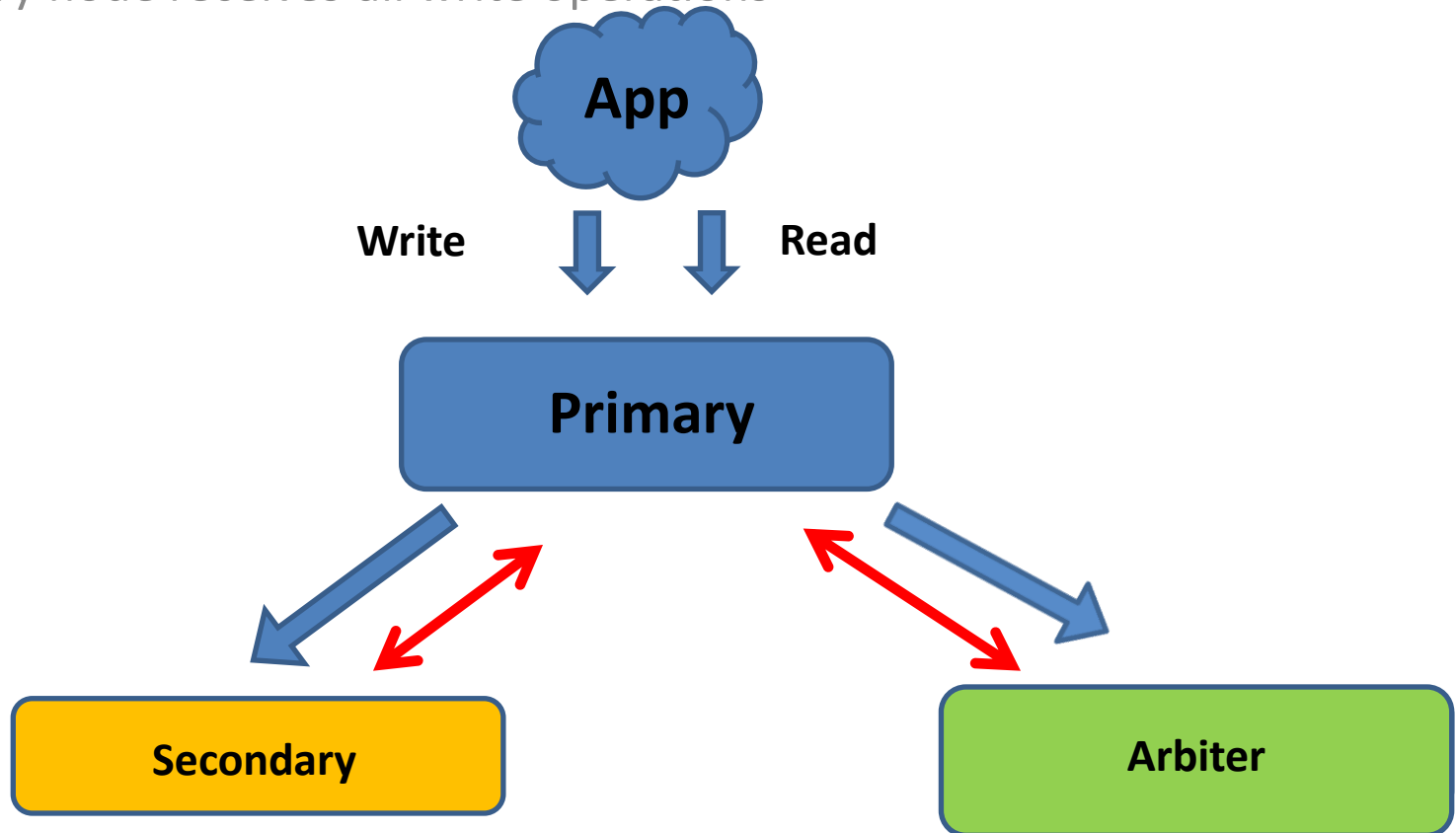
No database downtime



# Replication in MongoDB

A replica set is a group of mongod instances (server) that maintain the same data set. A replica set contains several data bearing nodes and optionally one arbiter node. Of the data bearing nodes, one and only one member is deemed the primary node, while the other nodes are deemed secondary nodes.

The primary node receives all write operations



# Replication in MongoDB

## Major features of replica:

Asynchronous Replication -Secondary replicate the primary's and apply the operations to their data sets asynchronously.

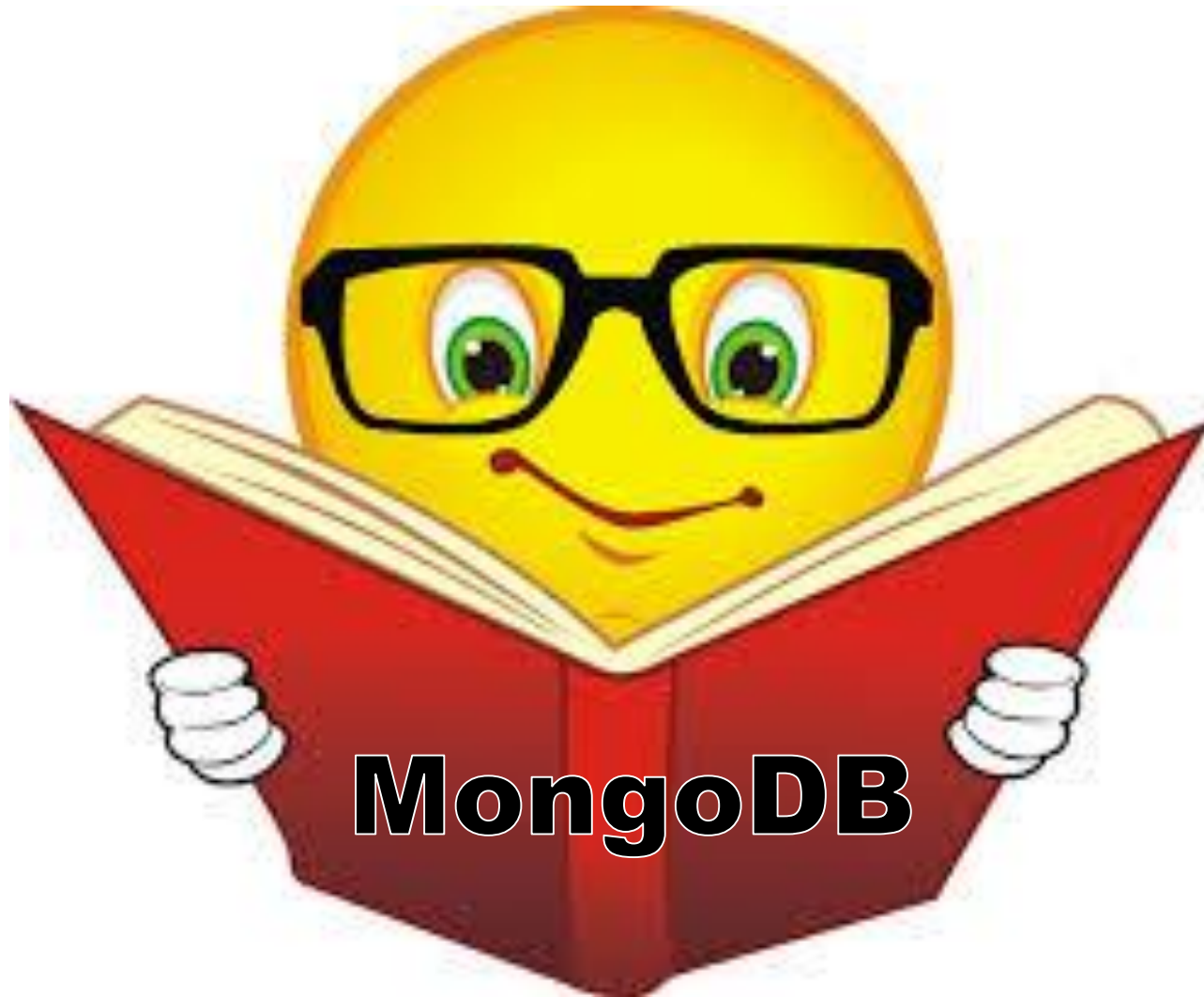
Automatic Failover (electionTimeoutMillis period (10 seconds by default))

Read Preference - can specify a read preference to send read operations to secondaries.

Mirrored Reads – operations can be in the cache of secondary.







*Thanks!*

**EVERY ENDING  
IS REALLY JUST A  
NEW BEGINNING**

Rajeev Garg  
Data Analytics Trainer

