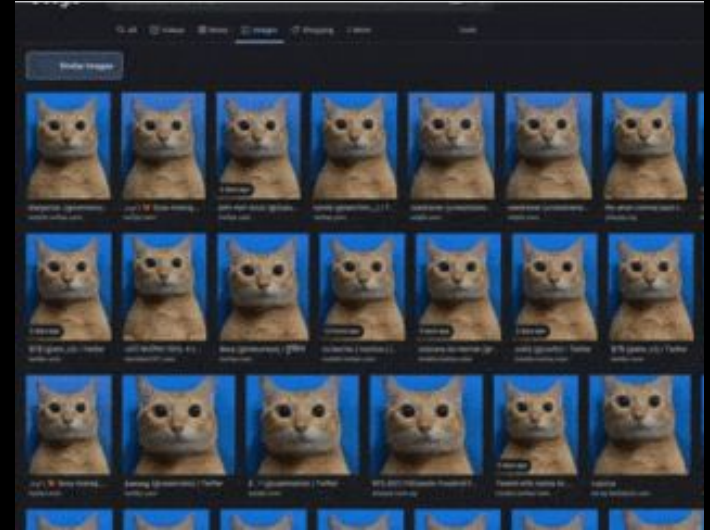




16. Introduction to NoSQL

1. What is MongoDB
2. Setting up MongoDB
3. Installing MongoDB Driver
4. Creating MongoDB Connection
5. Saving a Home
6. Install MongoDB Compass
7. Install MongoDB for VSCode
8. Fetching all Homes
9. Fetching one Home





16.1 What is MongoDB

1. **MongoDB** is the **product** and the **company** that builds it.
2. The name comes from the work **Humongous**.
3. **NoSQL Document Database**: Stores data in flexible, JSON-like documents.
4. **Dynamic Schema**: Allows **fields to vary across documents** without predefined schemas.
5. **High Performance**: Optimized for **fast read and write** operations.
6. **Scalability**: Supports **horizontal scaling** through sharding.
7. **High Availability**: Provides replication with automatic failover.
8. **Rich Query Capabilities**: Offers **powerful querying, indexing, and aggregation**.
9. **Geospatial and Text Search**: Includes support for location-based and full-text queries.
10. **Cross-Platform Compatibility**: Works with various **operating systems and programming languages**.
11. **Easy Integration**: Integrates smoothly with modern development stacks.

```
{
  "_id": "4f5b5c85-d8d3-4f58-8acf-3f5e5e4e59ea",
  "Items": [
    {
      "ProductId": 1,
      "ProductName": "Elden Ring",
      "Price": "49.97",
      "Quantity": 1
    },
    {
      "ProductId": 2,
      "ProductName": "FIFA 23",
      "Price": "69.97",
      "Quantity": 1
    }
  ]
}
```

The MongoDB logo, consisting of a green leaf icon and the text "MongoDB" in a green, sans-serif font.



16.2 Setting up MongoDB

MongoDB Community Server Download

The Community version of our distributed database offers a flexible document data model along with support for ad-hoc queries, secondary indexing, and real-time aggregations to provide powerful ways to access and analyze your data.

The database is also offered as a fully-managed service with [MongoDB Atlas](#). Get access to advanced functionality such as auto-scaling, serverless instances, full-text search, and data distribution across regions and clouds. Deploy in minutes on AWS, Google Cloud, and/or Azure, with no downloads necessary.

[Give it a try with a free, highly-available 512 MB cluster.](#) or get started from your terminal with the following two commands:

```
$ brew install mongodb-atlas
$ atlas setup
```

Version
8.0.3 (current)

Platform
Windows x64

Package
msi

Download

Copy link

More Options

VS

Try MongoDB Atlas

A developer data platform built around a fully managed MongoDB service. Address transactional, search, and analytical workloads.

Explore all our products →

Create your Atlas Account

The multi-cloud developer data platform.

First Name*

Last Name*

Company



16.2 Setting up MongoDB

Deploy your cluster

Use a template below or set up advanced configuration options. You can also edit these configuration options once the cluster is created.

☐ M10

\$0.08/hour

Dedicated cluster for development environments and low-traffic applications.

STORAGE	RAM	vCPU
10 GB	2 GB	2 vCPUs

☐ Serverless

For application development and testing, or workloads with variable traffic.

STORAGE	RAM	vCPU
Up to 1TB	Auto-scale	Auto-scale

☒ M0

Free

For learning and exploring MongoDB in a cloud environment.

STORAGE	RAM	vCPU
512 MB	Shared	Shared

✔ **Free forever!** Your free cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

Configurations

Name

You cannot change the name once the cluster is created.

Provider



Region

Mumbai (ap-south-1) ★

★ Recommended ⓘ Low carbon emissions ⓘ

Tag (optional)

Create your first tag to categorize and label your resources; more tags can be added later. [Learn more.](#)

Quick setup

☒ Automate security setup ⓘ

☒ Preload sample dataset ⓘ



16.2 Setting up MongoDB

Connect to KGCluster ✕

1

Set up connection security

2

Choose a connection method

3

Connect

You need to secure your MongoDB Atlas cluster before you can use it. Set which users and IP addresses can access your cluster now. [Read more](#)

1. Add a connection IP address

- ✓ Your current IP address (160.202.37.194) has been added to enable local connectivity. Only an IP address you add to your Access List will be able to connect to your project's clusters. Add more later in [Network Access](#).

2. Create a database user

This first user will have [atlasAdmin](#) permissions for this project.

We autogenerated a username and password. You can use this or create your own.

i You'll need your database user's credentials in the next step. Copy the database user password.

Username

root

Password

root

HIDE



Copy

Create Database User

Close

Choose a connection method



16.2 Setting up MongoDB

Connect to KGCluster


1

2

3

Set up connection securityChoose a connection methodConnect


Connect to your application



Drivers
Access your Atlas data using MongoDB's native drivers (e.g. Node.js, Go, etc.)


>

Access your data through tools




Compass
Explore, modify, and visualize your data with MongoDB's GUI

>




Shell
Quickly add & update data using MongoDB's Javascript command-line interface

>



MongoDB for VS Code
Work with your data in MongoDB directly from your VS Code environment

>



Atlas SQL
Easily connect SQL tools to Atlas for data analysis and visualization

>

Go Back

Close

Connect to KGCluster

1

2

3

Set up connection securityChoose a connection methodConnect

Connecting with MongoDB Driver

1. Select your driver and version

We recommend installing and using the latest driver version.

Driver	Version
<div>Node.js</div>	<div>5.5 or later</div>


2. Install your driver

Run the following on the command line

```
npm install mongodb
```

[View MongoDB Node.js Driver installation instructions.](#)

3. Add your connection string into your application code



KGCluster is provisioning...

It takes an average of 10-15 seconds to provision your deployment. Clusters are built with 3 nodes for resiliency.

You will be able to copy your connection string shortly. If you would like to come back to this later, you can go to the [Atlas Home Page](#).

RESOURCES

[Get started with the Node.js Driver](#)[Node.js Starter Sample App](#)

[Access your Database Users](#)[Troubleshoot Connections](#)

Go Back

Done



16.2 Setting up MongoDB

Project 0

Data Services

Charts

Overview

DATABASE

Clusters

SERVICES

Atlas Search

Stream Processing

Triggers

Migration

Data Federation

SECURITY

Quickstart

Backup

Database Access

Network Access

Advanced

New On Atlas 7

Goto

We are deploying your changes (current action: creating a plan)

PRASHANT'S ORG - 2024-11-08 > PROJECT 0

Network Access

IP Access List

Peering

Private Endpoint

! You will only be able to connect to your cluster from the following list of IP Addresses:

IP Address

160.202.37.194/32 (includes your current IP address)



16.3 Installing MongoDB Driver

✓ TERMINAL

- prashantjain@Prashants-Mac-mini Chapter 13 – MVC % npm install mongodb

added 12 packages, and audited 234 packages in 2s

49 packages are looking for funding
run `npm fund` for details

2 low severity vulnerabilities

To address all issues, run:
npm audit fix

Run `npm audit` for details.
- prashantjain@Prashants-Mac-mini Chapter 13 – MVC % █



16.3 Installing MongoDB Driver

JS database.js M X

Test Project > node > Chapter 13 - MVC > utils > JS database.js > ...

```
1  const mongodb = require("mongodb");
2
3  const MongoClient = mongodb.MongoClient;
4
5  const url = "mongodb+srv://root:root@kgcluster.ie6mb.mongodb.net/?
  retryWrites=true&w=majority&appName=KGCluster";
6
7  const mongoConnect = (callback) => {
8    MongoClient.connect(url)
9      .then((client) => {
10       console.log("Connected to MongoDB");
11       callback(client);
12     })
13     .catch((err) => {
14       console.log(err);
15     });
16  };
17
18  module.exports = mongoConnect;
```



16.3 Installing MongoDB Driver

```
database.js M app.js M X
Test Project > node > Chapter 13 - MVC > app.js > ...
25
26 | const mongoConnect = require("../utils/database");
27 | const PORT = 3000;
28 | mongoConnect((client) => {
29 |   console.log(client);
30 |   app.listen(PORT, () => {
31 |     console.log(`Server running on address http://localhost:${PORT}`);
32 |   });
33 | });
```

```
Connected to MongoDB
<ref *1> MongoClient {
  _events: [Object: null prototype] {},
  _eventsCount: 0,
  _maxListeners: undefined,
  mongoLogger: undefined,
  s: {
    url: 'mongodb+srv://root:root@kgcluster.ie6mb.mongodb.net/?retryWrites=true&w=majority&appName=KGCluster',
    bsonOptions: {
      raw: false,
      useBigInt64: false,
      promoteLongs: true,
      promoteValues: true,
      promoteBuffers: false,
      ignoreUndefined: false,
      bsonRegExp: false,
      serializeFunctions: false,
      fieldsAsRaw: {},
      enableUtf8Validation: true
    },
    namespace: MongoDBNamespace { db: 'admin', collection: undefined },
    hasBeenClosed: false,
    sessionPool: ServerSessionPool { client: [Circular *1], sessions: [List] },
    activeSessions: Set(0) {},
    authProviders: MongoClientAuthProviders { existingProviders: [Map] },
    options: [Getter],
    readConcern: [Getter],
    writeConcern: [Getter],
    readPreference: [Getter],
    isMongoClient: [Getter]
  },
  connectionLock: undefined,
  topology: Topology {
    _events: [Object: null prototype] {
      topologyDescriptionChanged: [Array],
      connectionPoolCreated: [Function (anonymous)],

```

Server running on address http://localhost:3000

TypeError: db.execute is not a function

```
    at Home.fetchAll (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/index.js:2:15)
    at exports.getIndex (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/index.js:5:8)
    at Layer.handle [as handle_request] (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/layer.js:95:5)
    at next (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/route.js:149:13)
    at Route.dispatch (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/route.js:119:3)
    at Layer.handle [as handle_request] (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/layer.js:95:5)
    at /Users/prashantjain/workspace/stuff/Test Project/node/Chapter 13 - MVC/index.js:284:15
    at Function.process_params (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/index.js:346:12)
    at next (/Users/prashantjain/workspace/stuff/Test Project/node/Chapter 13 - MVC/index.js:280:10)
    at Function.handle (/Users/prashantjain/workspace/stuff/Test Project/node_modules/express/lib/router/index.js:175:3)
```



16.3 Installing MongoDB Driver

```
module.exports = class Home {  
  constructor(name, description, price, location, rating,  
    imageUrl) {  
    this.name = name;  
    this.description = description;  
    this.price = price;  
    this.location = location;  
    this.rating = rating;  
    this.imageUrl = imageUrl;  
  }  
  
  save() {  
  }  
  
  static fetchAll() {  
  }  
  
  static findById(id) {  
  }  
  
  static deleteById(id) {  
  }  
};
```



16.4 Creating MongoDB Connection

```
database.js M X app.js M home.js M
Test Project > node > Chapter 13 - MVC > utils > database.js > ...
1 | const mongodb = require("mongodb");
2 |
3 | const MongoClient = mongodb.MongoClient;
4 |
5 | const url = "mongodb+srv://root:root@kgcluster.ie6mb.mongodb.net/?retryWrites=true&
w=majority&appName=KGCluster";
6 |
7 | let _db;
8 |
9 | const mongoConnect = (callback) => {
10 |   MongoClient.connect(url)
11 |     .then((client) => {
12 |       console.log("Connected to MongoDB");
13 |       _db = client.db("airbnb");
14 |       callback();
15 |     })
16 |     .catch((err) => {
17 |       console.log(err);
18 |       throw err;
19 |     });
20 | };
21 |
22 | const getDb = () => {
23 |   if (!_db) {
24 |     throw new Error("Database not connected");
25 |   }
26 |   return _db;
27 | };
28 |
29 | exports.mongoConnect = mongoConnect;
30 | exports.getDb = getDb;
```



16.5 Saving a Home

```
save() {  
  const db = getDb();  
  // insert Many takes an array of objects  
  return db.collection("homes").insertOne(this).then((result) => {  
    console.log(result);  
  });  
}
```

```
{  
  acknowledged: true,  
  insertedId: new ObjectId('672df86c3d20696a2b523c79')  
}
```

The screenshot shows a web browser at localhost:3000/host/add-home. The page has a red navigation bar with links: airbnb, Homes-List, Favourites, Bookings, Host Homes, and Add Home. The main content area is titled "Register Your Home on AirBnB" and contains a form with the following fields: Name (Utsav), Description (the best house), Year (1999), Location (ghaziabad), Rating (4.5), and Image Path (/images/house1.png). A red "Add Home" button is at the bottom of the form.