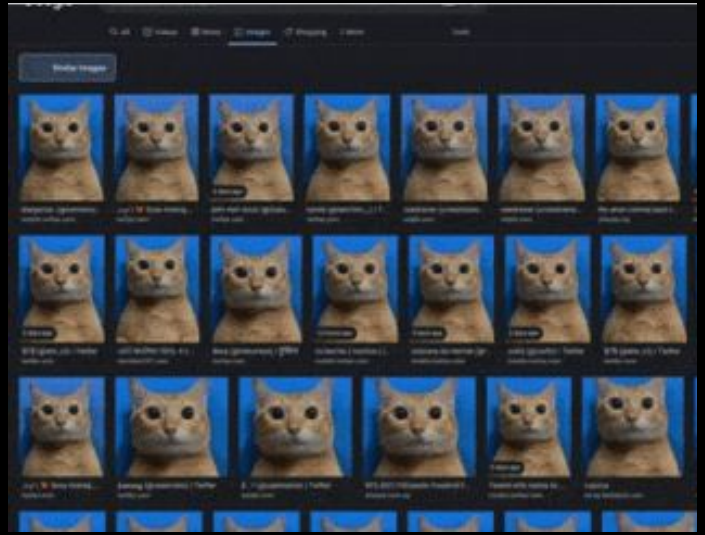
16. Introduction to NoSQL

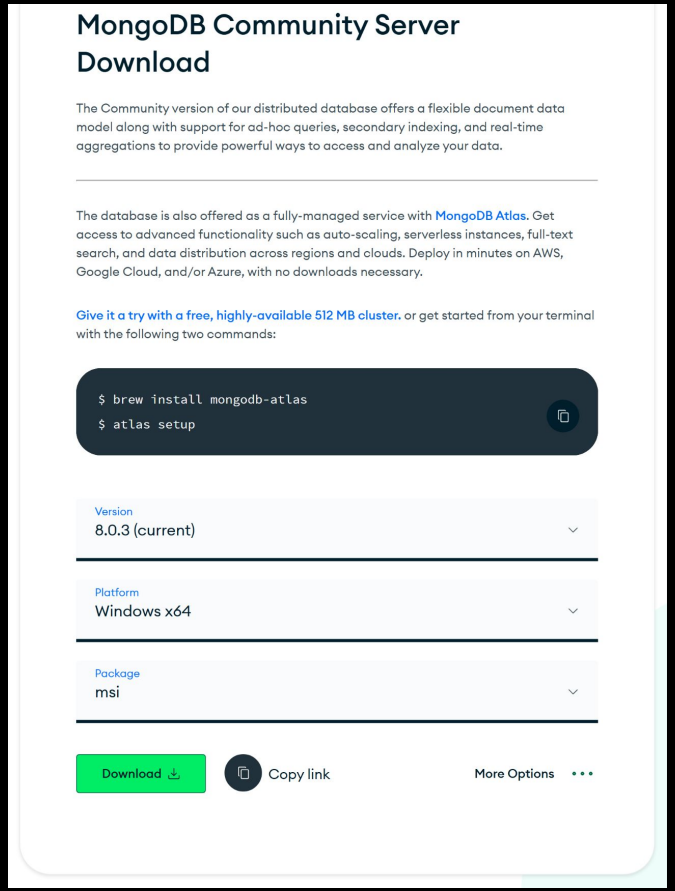
1. What is MongoDB
2. Setting up MongoDB
3. Installing MongoDB Driver
4. Creating MongoDB Connection
5. Saving a Home
6. Install MongoDB Compass
7. Install MongoDB for VSCode
8. Fetching all Homes
9. Fetching one Home



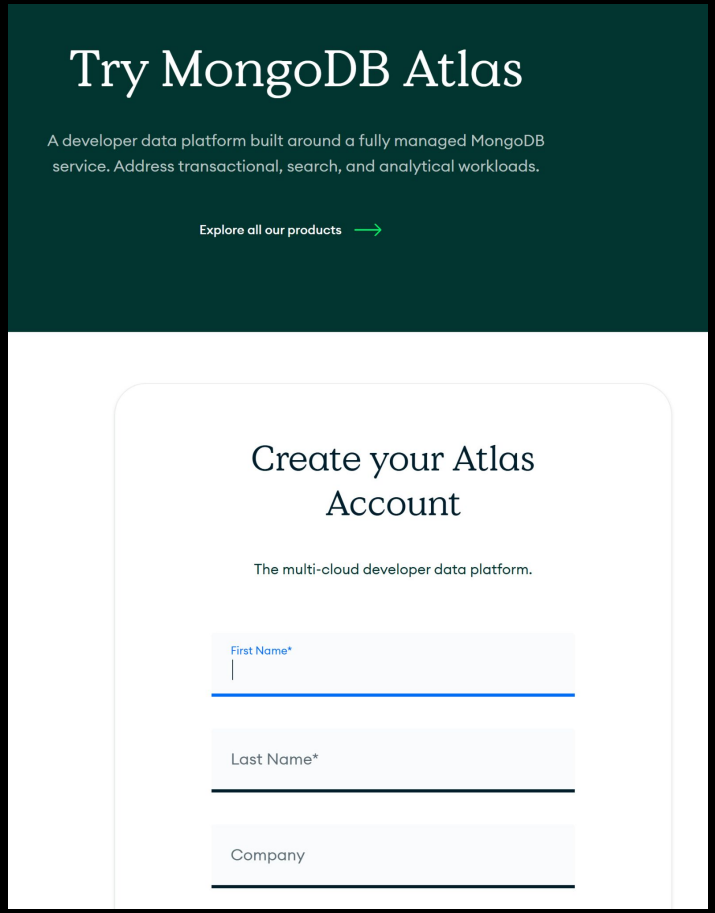
16.1 What is MongoDB

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.

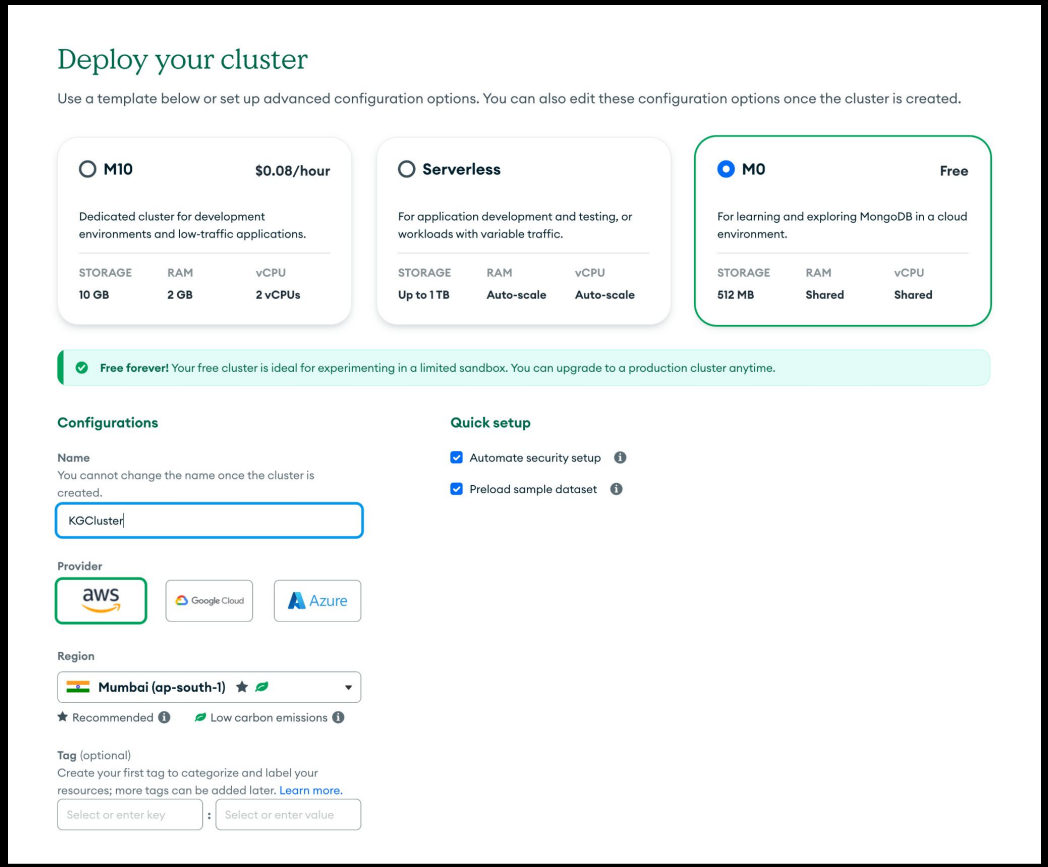
16.2 Setting up MongoDB



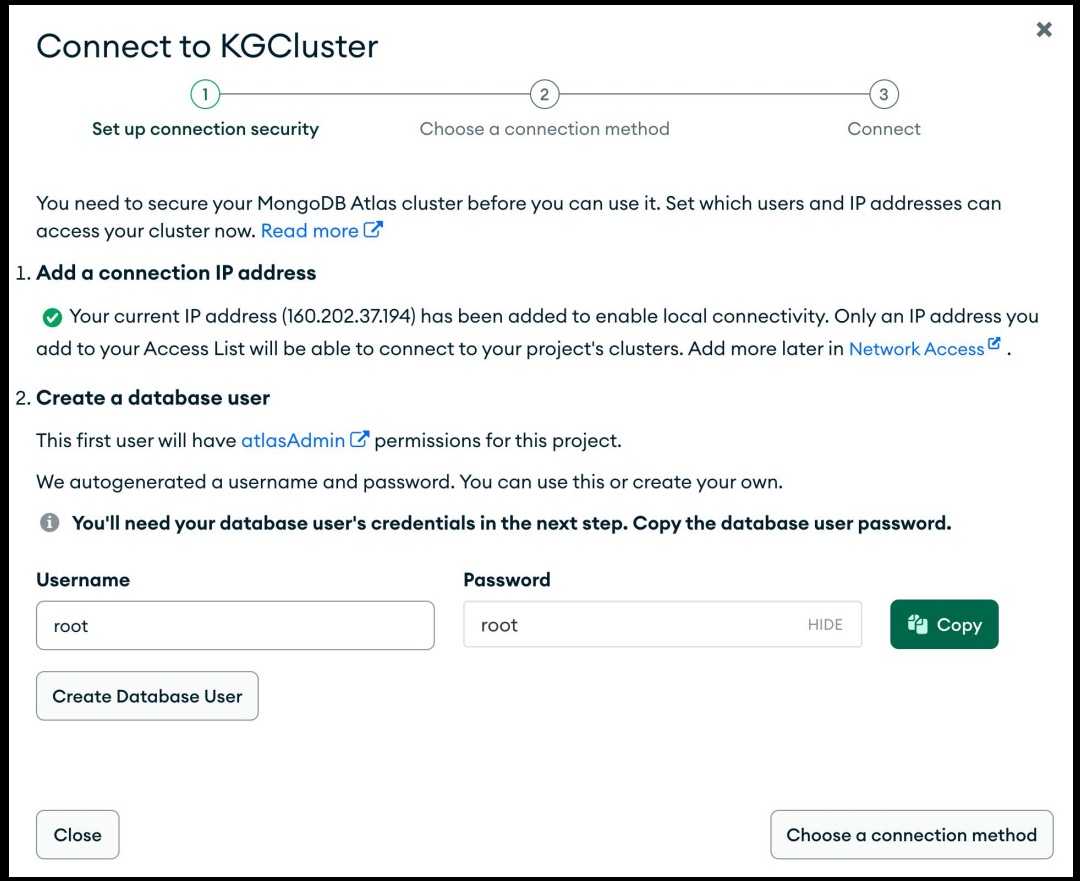
VS



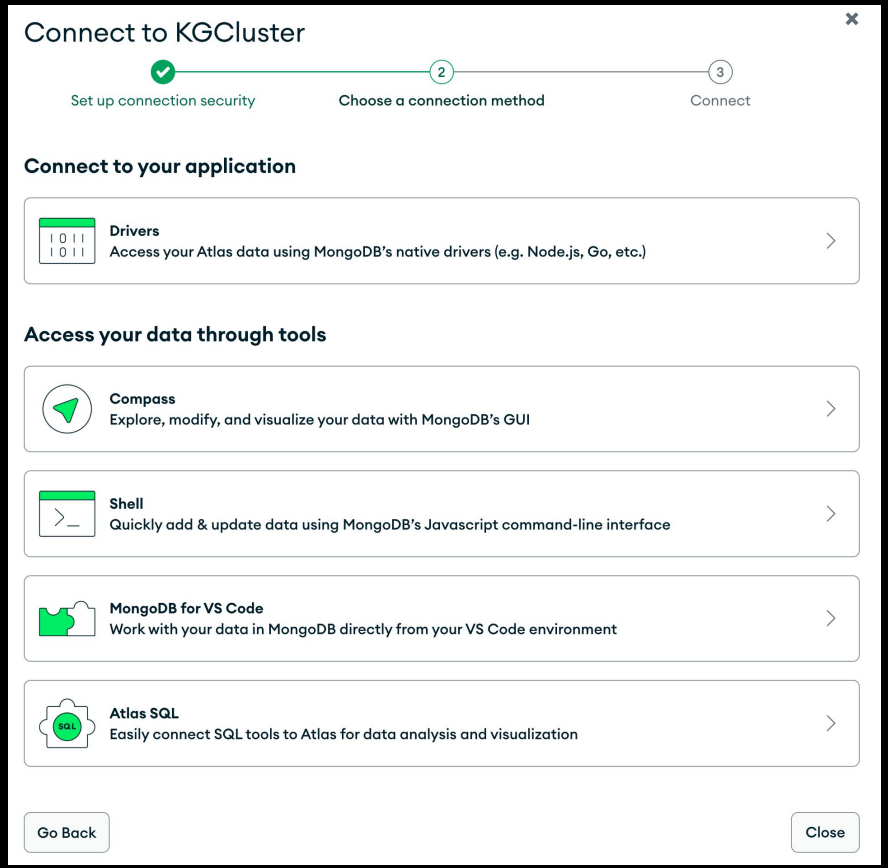
16.2 Setting up MongoDB

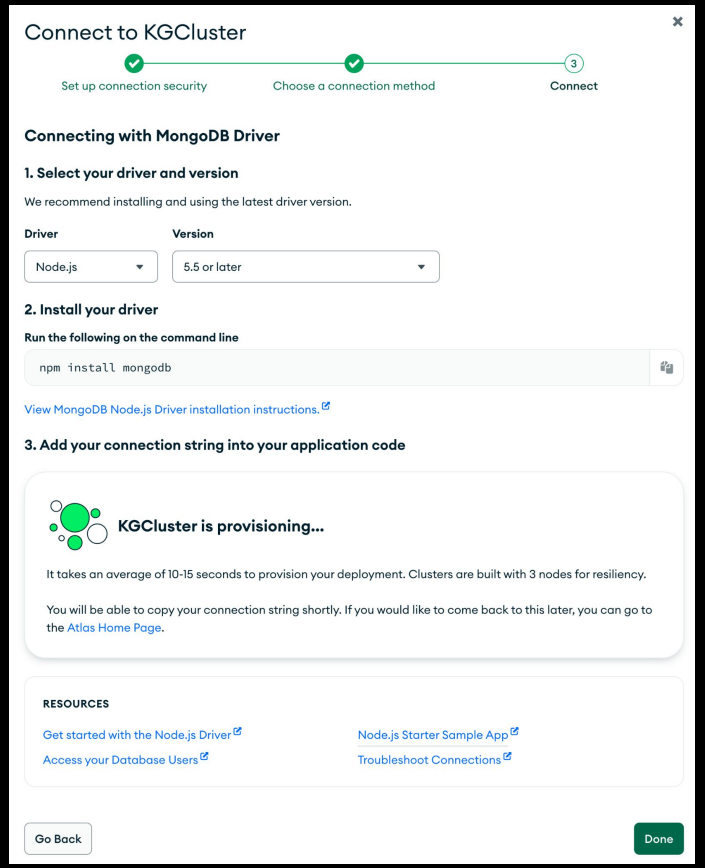


16.2 Setting up MongoDB

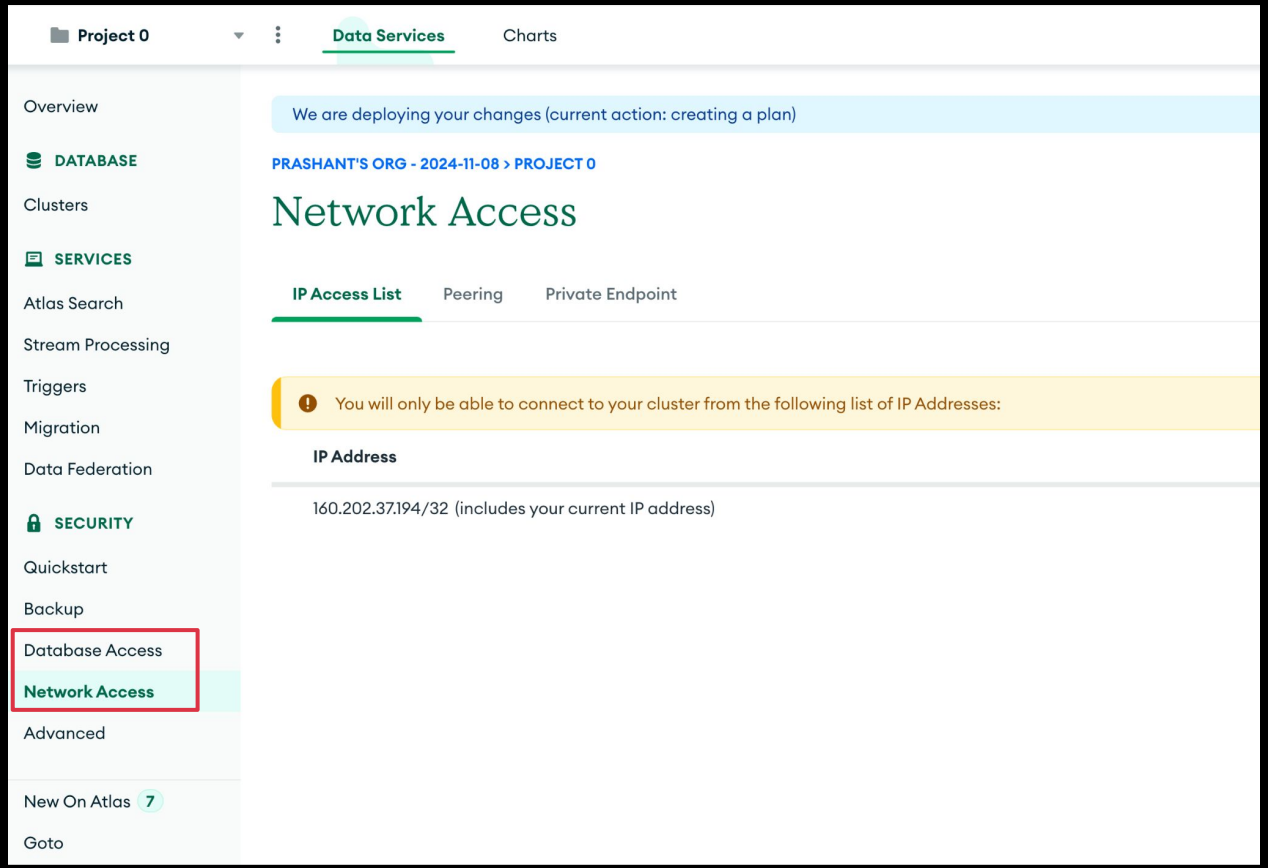


16.2 Setting up MongoDB





16.2 Setting up MongoDB



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

database.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  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

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/no
2:15)
    at exports.getIndex (/Users/prashantjain/workspace/stuff/Test Project/
toreController.js:5:8)
    at Layer.handle [as handle_request] (/Users/prashantjain/workspace/st
VC/node_modules/express/lib/router/layer.js:95:5)
    at next (/Users/prashantjain/workspace/stuff/Test Project/node/Chapte
router/route.js:149:13)
    at Route.dispatch (/Users/prashantjain/workspace/stuff/Test Project/n
press/lib/router/route.js:119:3)
    at Layer.handle [as handle_request] (/Users/prashantjain/workspace/st
VC/node_modules/express/lib/router/layer.js:95:5)
    at /Users/prashantjain/workspace/stuff/Test Project/node/Chapter 13 -
/index.js:284:15
    at Function.process_params (/Users/prashantjain/workspace/stuff/Test
odules/express/lib/router/index.js:346:12)
    at next (/Users/prashantjain/workspace/stuff/Test Project/node/Chapte
router/index.js:280:10)
    at Function.handle (/Users/prashantjain/workspace/stuff/Test Project/
xpress/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 | app.js M | home.js M

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

```
const mongodb = require("mongodb");

const MongoClient = mongodb.MongoClient;

const url = "mongodb+srv://root:root@kgcluster.ie6mb.mongodb.net/?retryWrites=true&
w=majority&appName=KGCluster";

let _db;

const mongoConnect = (callback) => {
  MongoClient.connect(url)
  .then((client) => {
    console.log("Connected to MongoDB");
    _db = client.db("airbnb");
    callback();
  })
  .catch((err) => {
    console.log(err);
    throw err;
  });
};

const getDb = () => {
  if (!_db) {
    throw new Error("Database not connected");
  }
  return _db;
};

exports.mongoConnect = mongoConnect;
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')
}
```