

**server.js:**

const { MongoClient } = require('mongodb')

const client = new MongoClient('mongodb://localhost:27017')

client.connect()

.then(() => {

console.log('Connected Successfully!')

console.log('Exiting...')

client.close()

})

.catch(error => console.log('Failed to conect', error))

**index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Student Form</title>

</head>

<body>

<form action="/student" method="post">

Name: <input type="text" placeholder="Enter your name" name="name"><br><br>

Email: <input type="text" placeholder="Enter your email" name="email"><br><br>

<button type="submit">Submit</button>

</form>

</body>

</html>

**index.js:**

const express = require('express')

const mongoose = require('mongoose')

const bodyParser = require('body-parser')

const app = express()

app.listen(4000, (res) => {

console.log('Listening on port 4000')

})

app.use(express.static(`${\_\_dirname}/public`))

app.use(bodyParser.json())

app.use(bodyParser.urlencoded({

extended: true

}))

mongoose.connect('mongodb://0.0.0.0:27017/students')

const studentSchema = new mongoose.Schema({

name: String,

email: String

})

const Student = mongoose.model('Student', studentSchema)

app.post('/student', (req, res) => {

let student = new Student(req.body);

student.save()

.then(doc => {

res.send(doc)

console.log(doc)

})

.catch(err => console.log(err))

})

app.get('/students', (req, res) => {

Student.find({})

.then(docs => {

console.log(docs)

res.json(docs)

})

.catch(err => console.log(err))

})

**package.json:**

{

"name": "mongodb-nodejs",

"version": "1.0.0",

"description": "",

"main": "index.js",

"scripts": {

"start": "nodemon index.js",

"test": "echo \"Error: no test specified\" && exit 1"

},

"keywords": [],

"author": "",

"license": "ISC",

"dependencies": {

"body-parser": "^1.20.2",

"express": "^4.18.2",

"mongodb": "^6.1.0",

"mongoose": "^7.5.2"

},

"devDependencies": {

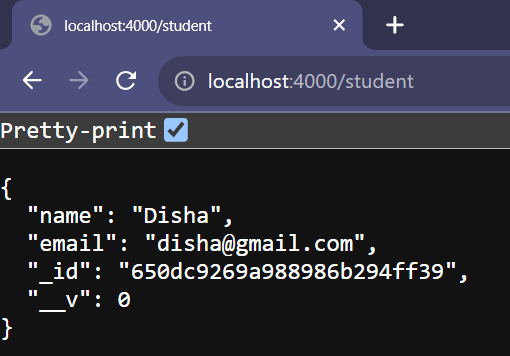
"nodemon": "^3.0.1"

}

}

**OUTPUT:**

****



**Student Data:**

**index.html:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Student Data Management</title>

</head>

<body>

<!-- Form for creating a new student -->

<h2>Create a New Student</h2>

<form action="http://localhost:4000/student" method="POST">

<label for="createName">Name:</label>

<input type="text" id="createName" name="name" required>

<br>

<label for="createEmail">Email:</label>

<input type="email" id="createEmail" name="email" required>

<br>

<button type="submit">Create Student</button>

</form>

<!-- Form for updating a student -->

<h2>Update Student Data</h2>

<form action="/student/update" method="POST">

<label for="updateId">Student ID:</label>

<input type="text" id="updateId" name="id" required>

<br>

<label for="updateName">New Name:</label>

<input type="text" id="updateName" name="name">

<br>

<label for="updateEmail">New Email:</label>

<input type="email" id="updateEmail" name="email">

<br>

<button type="submit">Update Student</button>

</form>

<!-- Form for deleting a student -->

<h2>Delete Student</h2>

<form action="/student/delete" method="POST">

<label for="deleteId">Student ID:</label>

<input type="text" id="deleteId" name="id" required>

<br>

<button type="submit">Delete Student</button>

</form>

</body>

</html>

**index.js:**

const express = require("express");

const mongoose = require('mongoose');

const bodyParser = require("body-parser");

const app = express();

app.listen(4000, () => {

console.log('Listening on port 4000');

});

app.use(express.static(`${\_\_dirname}/public`));

app.use(bodyParser.json());

app.use(bodyParser.urlencoded({

extended: true

}));

mongoose.connect('mongodb://0.0.0.0:27017/student', {

useNewUrlParser: true,

useUnifiedTopology: true

});

const studentSchema = new mongoose.Schema({

name: String,

email: String

});

const Student = mongoose.model('Student', studentSchema);

// Create a new student

app.post('/student', (req, res) => {

let student = new Student(req.body);

student.save()

.then(doc => {

res.send(doc);

console.log(doc);

})

.catch(err => console.log(err));

});

// Retrieve all students

app.get('/student', (req, res) => {

Student.find({})

.then(docs => {

console.log(docs);

res.json(docs);

})

.catch(err => console.log(err));

});

// Update a student by ID

app.post('/student/update', (req, res) => {

const { id, name, email } = req.body;

// Convert id to ObjectId

const mongoose = require('mongoose');

const ObjectId = mongoose.Types.ObjectId;

const objectId = new ObjectId(id);

Student.findByIdAndUpdate(objectId, { name, email }, { new: true })

.then(updatedStudent => {

if (!updatedStudent) {

res.status(404).send('Student not found');

} else {

res.json(updatedStudent);

}

})

.catch(err => {

console.error(err);

res.status(500).send('Error updating student data');

});

});

// Delete a student by ID

app.post('/student/delete', (req, res) => {

const { id } = req.body;

// Convert id to ObjectId

const mongoose = require('mongoose');

const ObjectId = mongoose.Types.ObjectId;

const objectId = new ObjectId(id);

Student.findByIdAndDelete(objectId)

.then(deletedStudent => {

if (!deletedStudent) {

res.status(404).send('Student not found');

} else {

res.json(deletedStudent);

}

})

.catch(err => {

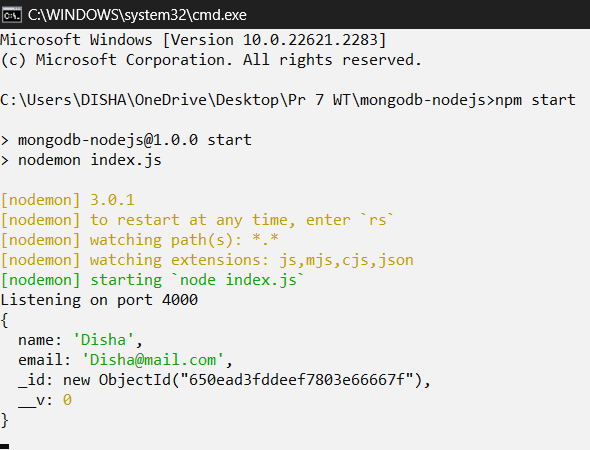
console.error(err);

res.status(500).send('Error deleting student data');

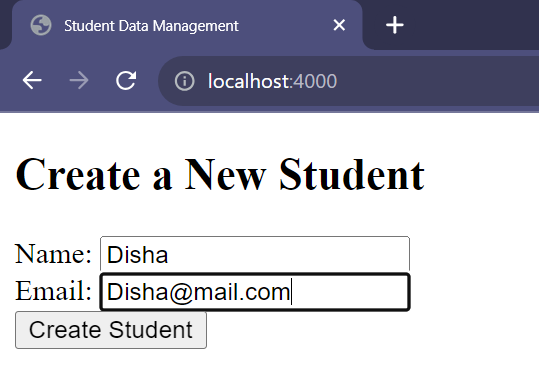
});

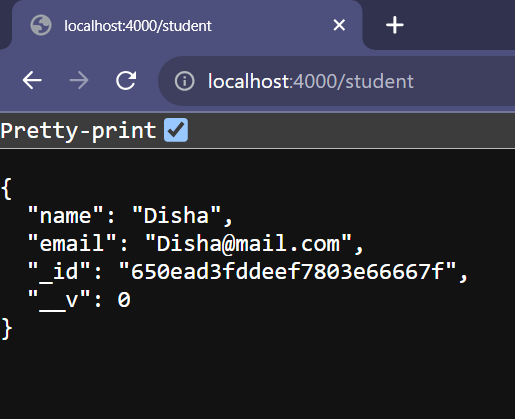
});

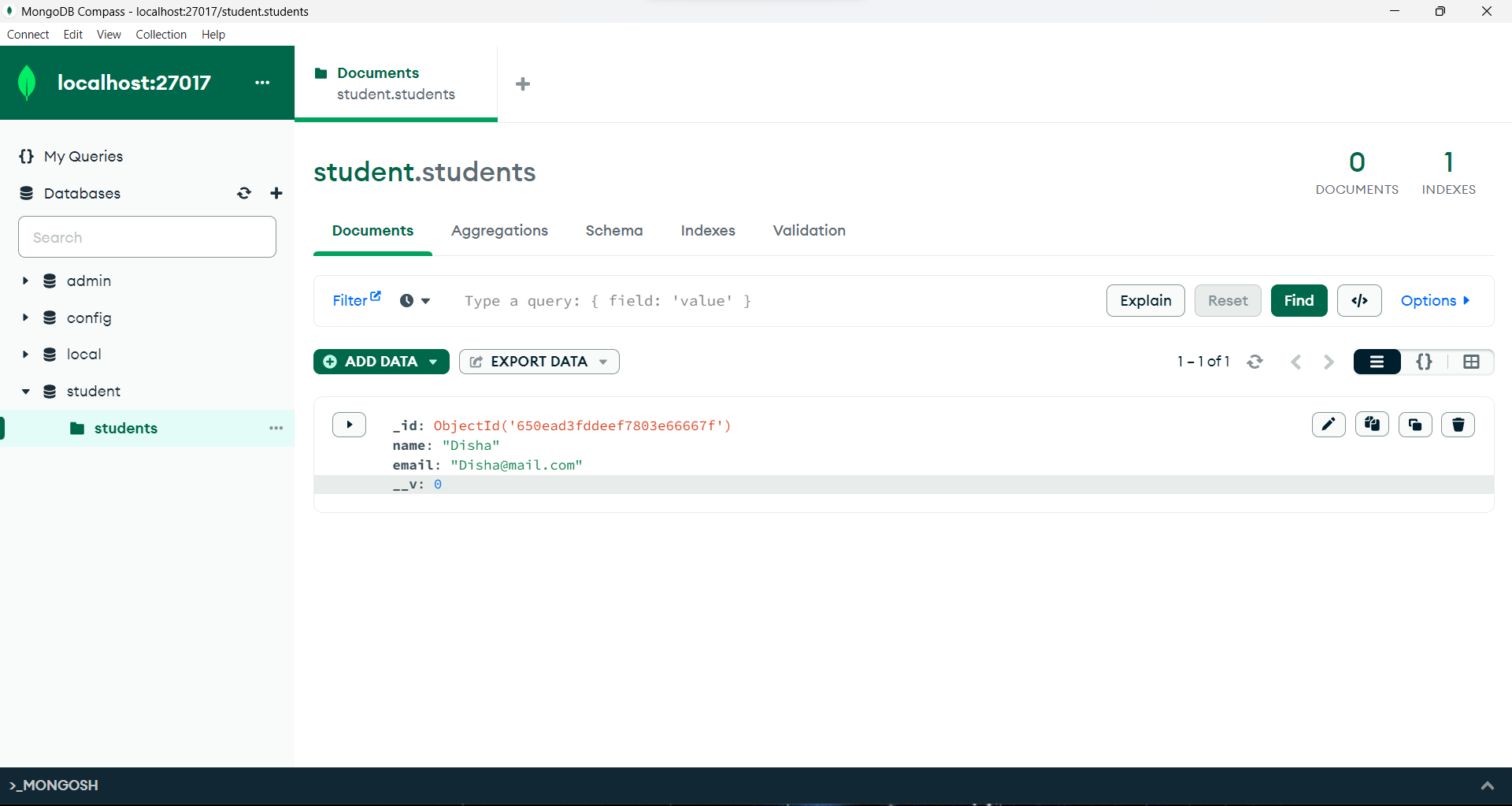
**OUTPUT:**

****

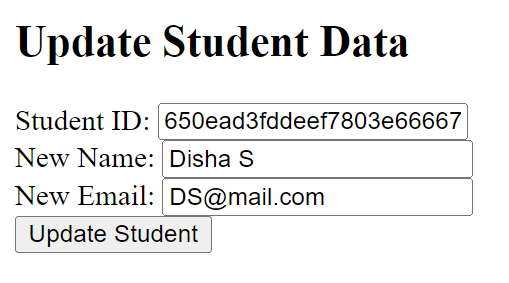
1. **Insert:**

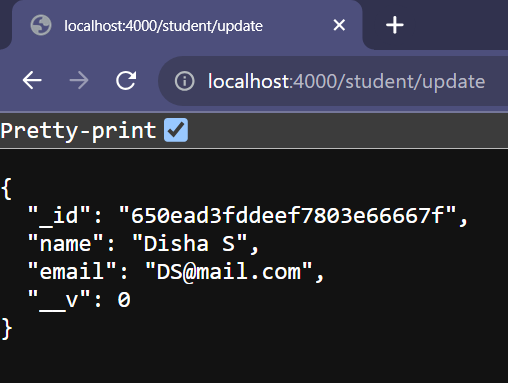
****

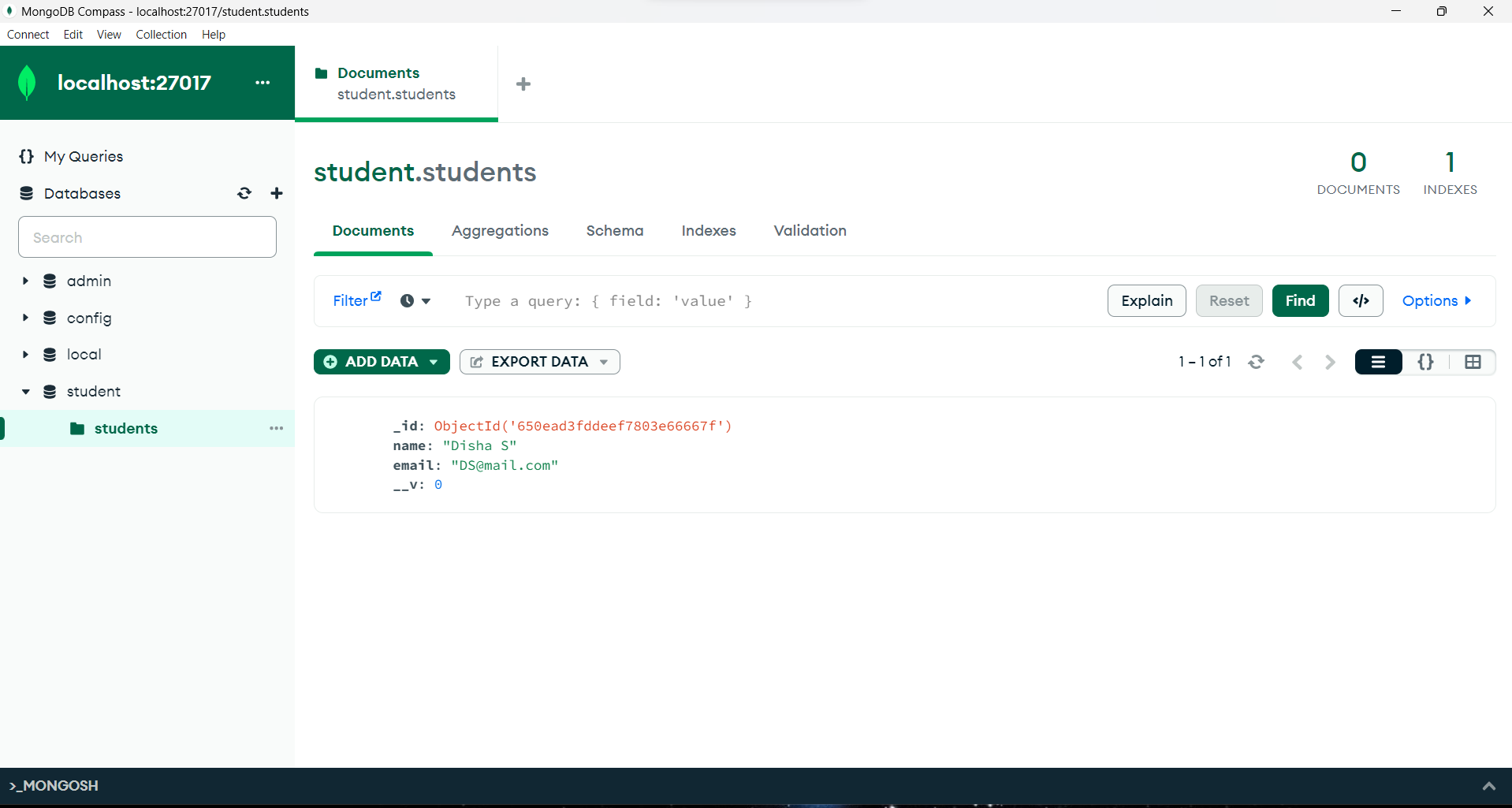
****

****

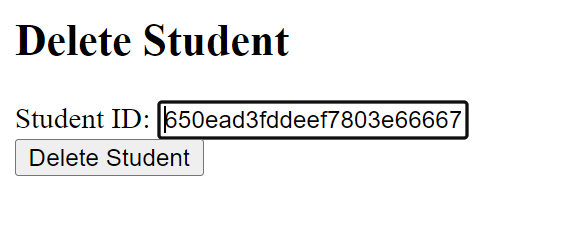
1. **Update:**

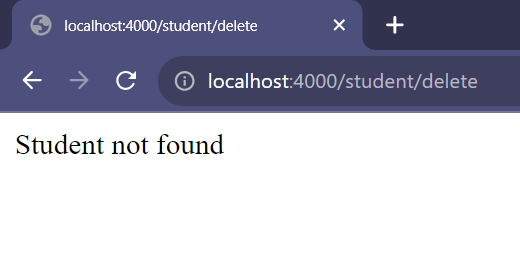
****

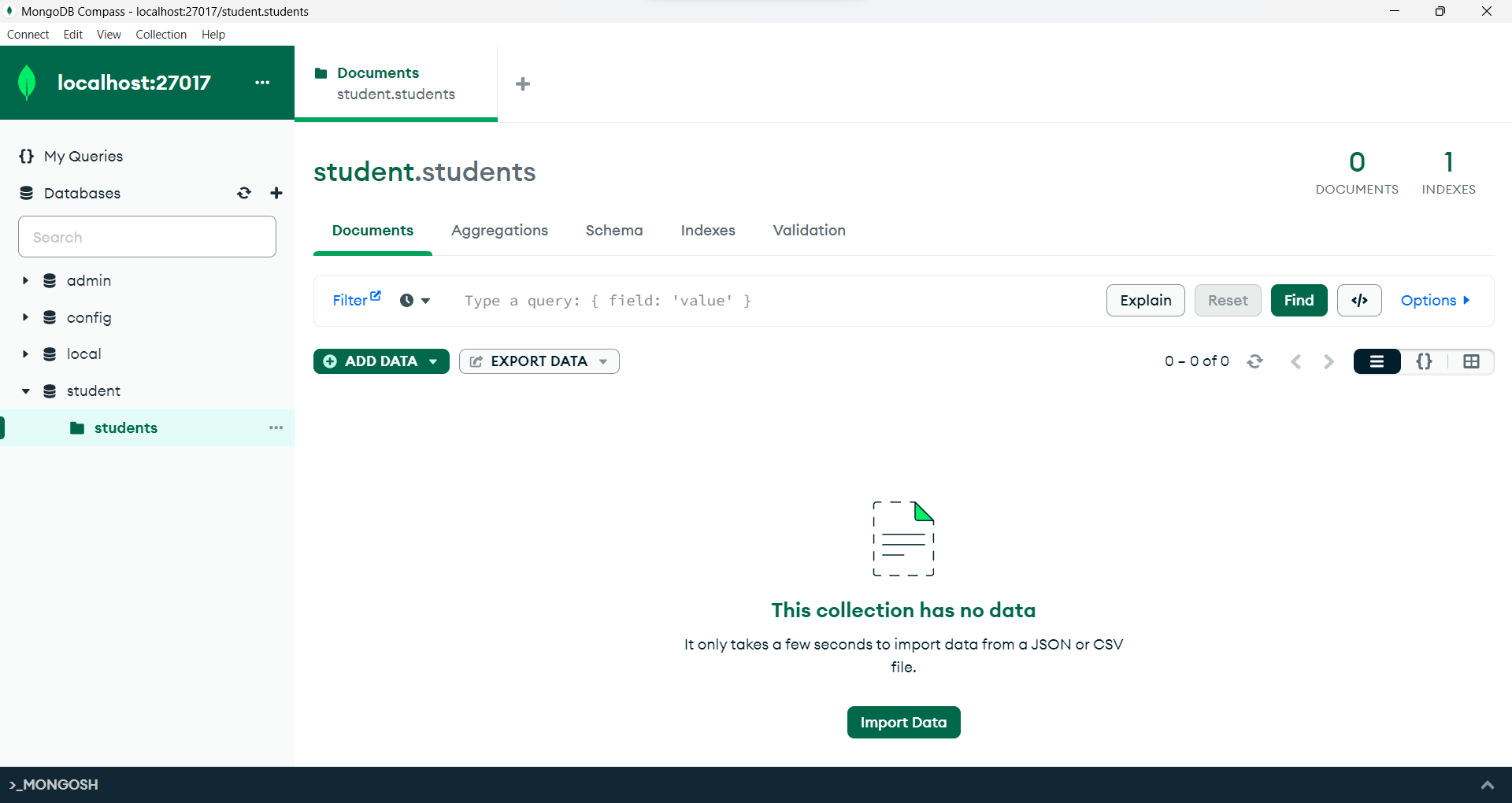
****

****

1. **Delete:**

****

****

****