Name: Andrew Parsons
TMU ID: 500992021
Due Date: Apr 8, 2023
Github Repo Link: https://github.com/Parsonswlu/parsonswlu.github.io/tree/main/CCPS530_Lab8

CCPS 530 - Web Systems Development - Lab 8

1. **Enhance the express URI (/bookinventory/add) to insert a book into a MongoDB collection.**

- The code was refactored to add a document to a MongoDB database (db = 'ccps530_lab7', collection = 'book') instead of adding it to a JSON variable

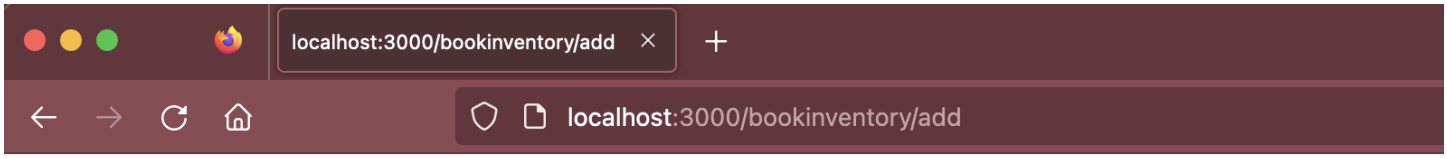
```
app.post('/bookinventory/addbook', function (req, res) {
 const client = new MongoClient(uri);

 var new_doc = {
   'title': req.body.title,
   'author': req.body.author,
   'publisher': req.body.publisher,
   'date': req.body.date,
   'website': req.body.website
 };

 async function addBook() {
   try {
     const db = client.db(dbname);
     const coll = db.collection(collname);

     const query = await coll.insertOne(new_doc);
     console.log('1 document inserted');
     res.send('<h1>Andrew Parsons Book Inventory</h1><ul><li><a href="/">Home Page</a></li><li><a
href="/bookinventory/list">Current Book List</a></li><li><a href="/bookinventory/add">Add Another
Book</a></li></ul><h2>Title: ' + req.body.title + ' is added!</h2>');
   } finally {
     // Ensures that the client will close when you finish/error
     await client.close();
   }
 }
 addBook().catch(console.dir);
});
```

Before hitting Submit



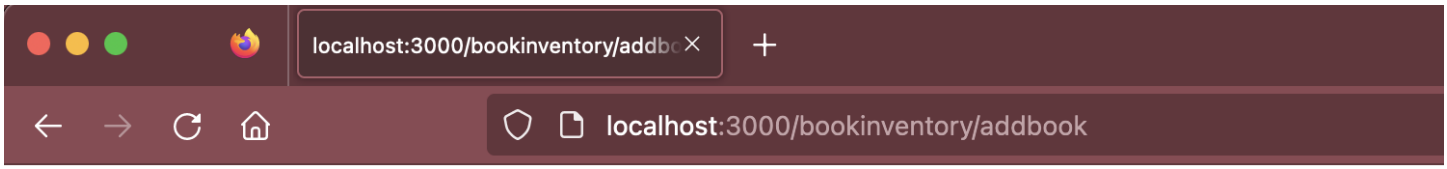
Andrew Parsons Book Inventory

- Home Page
- Current Book List

Insert a book:

Title: test_title1
Author: test_author1
Publisher: test_publisher1
Date: test_date1
Website: test_website1
Submit

After hitting Submit:

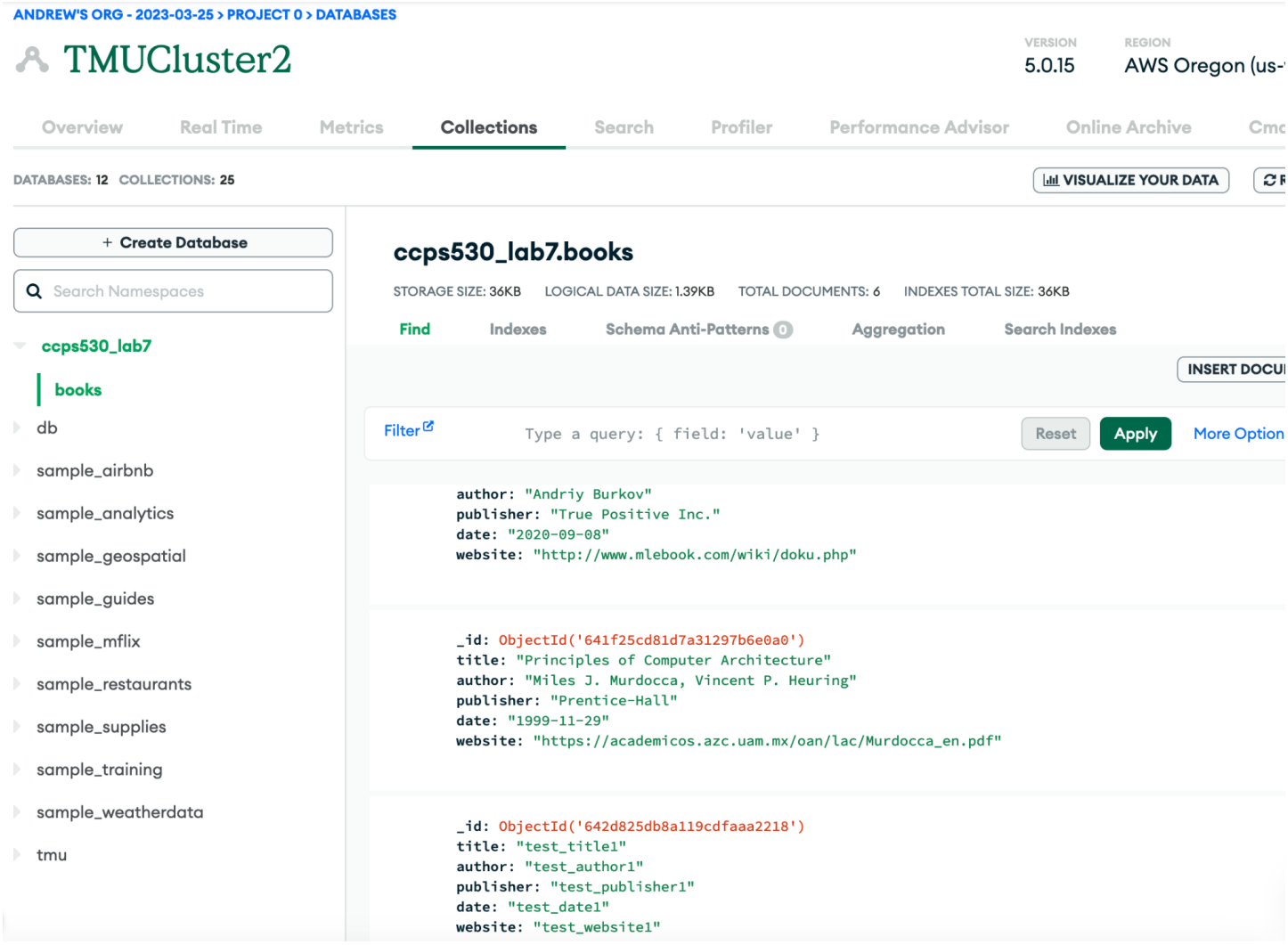


Andrew Parsons Book Inventory

- Home Page
- Current Book List
- Add Another Book

Title: test_title1 is added!

MongoDB website - db: ‘ccps530_lab7’, collection: ‘books’



2. **Enhance the express URI (/bookinventory/list) to retrieve all books from a MongoDB collection.**

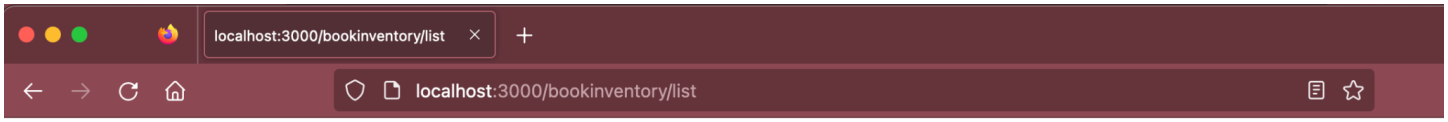
- The code was refactored to pull all documents from a MongoDB database (db = 'ccps530_lab7', collection = 'book') instead of printing the results of a stored JSON variable.

```
app.get('/bookinventory/list', function (req, res) {
 const client = new MongoClient(uri);
 var bookinventory = [];
 async function getBookInventory() {
   try {
     const db = client.db(dbname);
     const coll = db.collection(collname);

     bookinventory = await coll.find({}).toArray();
     var html = '<p>'
     for (var i = 0; i < bookinventory.length; i++) {
       html = html + 'Title: ' + bookinventory[i].title + '<br>';
       html = html + 'Author: ' + bookinventory[i].author + '<br>';
       html = html + 'Publisher: ' + bookinventory[i].publisher + '<br>';
       html = html + 'Date: ' + bookinventory[i].date + '<br>';
       html = html + 'Website: ' + bookinventory[i].website + '<br>';
       html = html + '<br>';
     }
     html += '</p>'

     res.send('<h1>Andrew Parsons Book Inventory</h1><ul><li><a href="/">Home Page</a></li><li><a
href="/bookinventory/add">Add a Book</a></li></ul><h2>Current Book List: </h2>' + html);
   } finally {
     // Ensures that the client will close when you finish/error
     await client.close();
   }
 }
 getBookInventory().catch(console.dir);
});
```

localhost:3000/bookinventory/list



Andrew Parsons Book Inventory

- Home Page
- Add a Book

Current Book List:

Title: Deep Learning for Coders with fastai & PyTorch: AI Applications Without a PhD
Author: Jeremy Howard & Sylvain Gugger
Publisher: O'Reilly Media, Inc.
Date: 2021-11-05
Website: https://course.fast.ai/Resources/book.html

Title: Quantum Computation and Quantum Information: 10th Anniversity Edition
Author: Michael A. Nielsen and Isaac L. Chuang
Publisher: Cambridge University Press
Date: 2011-01-30
Website: https://www.cambridge.org/highereducation/books/quantum-computation-and-quantum-information/01E10196D0A682A6AEFFEA52D53BE9AE#overview

Title: Spatial Computing
Author: Shashi Shekhar and Pamela Vold
Publisher: The MIT Press
Date: 2020-02-18
Website: https://mitpress.mit.edu/9780262538046/spatial-computing/

Title: Machine Learning Engineering
Author: Andriy Burkov
Publisher: True Positive Inc.
Date: 2020-09-08
Website: http://www.mlebook.com/wiki/doku.php

Title: Principles of Computer Architecture
Author: Miles J. Murdocca, Vincent P. Heuring
Publisher: Prentice-Hall
Date: 1999-11-29
Website: https://academicos.azc.uam.mx/oan/lac/Murdocca_en.pdf

Title: test_title1
Author: test_author1
Publisher: test_publisher1
Date: test_date1
Website: test_website1

Technical Report

1. Write a technical report explaining your design choices and what you used to test the RESTful endpoints from numbers 1 and 2.

- The key to designing the code in Lab 8 was to understand how to combine the code from Lab 6 and 7 in a way that could replace using a local JSON file with a call to the MongoDB collection.
- This involved wrapping an async function within each of the relevant **app.get()** Express calls, and submitting the html response only after the database function had been called and completed.
- One key aspect that took a bit of testing was to realize that the MongoClient(uri) object had to be re-instantiated for every call to the DB - and it too had to be wrapped within the **app.get**() Express call.
- Additionally, the entirety of the HTML response had to wrapped within the body of the **async function** to the MongoDB collection to ensure that the async call had been completed prior to populating the HTML for the Express URI - otherwise it threw errors as the result of a race condition.
- See code from responses to 1 and 2 above for specific implementation.

2. How long did you spend on this lab? The length of time includes readings and research and code experimentation. State time involved in readings and research as well as code experimentation sessions.

- I spent approximately 2.5 hours working on this lab.
- It took about 2 hours to merge the code from labs 6 and 7, as well as finding the correct syntax in order to return all documents from a MongoDB collection. After that it was mostly rearranging the order of the code in order to ensure all functions and responses fired correctly and within the scope that was required.
- It then took about another 30 minutes to do the validation, take screenshots, write up the technical report, etc.