Unit 7: Express & APIs: Lab 1 – Shops Pt. 1

Assigned: 01/11/23

Due: 01/12/23

Completed: 01/12/23

**LMS:** <https://lms.grandcircus.co/mod/assign/view.php?id=22808>

**Google Doc:** <https://docs.google.com/document/d/1dL_kyZezftlaISBq7gfNTqDniJpgWLppWMMfK9R9He4/preview>

**GitHub:** <https://github.com/clintmsmith/GrandCircusLabs/tree/main/Shops_Pt1>

**Skills:** Express, APIs

**Overview**: In this exercise, you will create JSON API endpoints (for computers to read) to interact with an array of shops. Exercise 2 uses the same data for web pages (for humans to read). These exercises are designed to be built together in the same project.

**Setup**: First, set up an Express server.

Text

Description automatically generated

**Build Specifications:**

Create a Shop interface with properties for id, name, and rating. Create an array of shops, like this…

A screenshot of a computer screen

Description automatically generated with medium confidence

const shops: Shop[] = [

    { id: 111, name: "Pepper's Pizza", rating: 4.5 },

    { id: 222, name: "Clive's Chives", rating: 3.4 },

    { id: 333, name: "Betty's Brews", rating: 4.3 },

    { id: 444, name: "Sylvester's Shoes", rating: 3.8 },

    { id: 555, name: "Teddy's Tunes", rating: 4.7 }

];

A screenshot of a computer

Description automatically generated with medium confidence

1. Create an API endpoint to read all shops.

* GET /api/shops. It responds with a JSON array of all shops.

Text

Description automatically generated

Text

Description automatically generated

A picture containing graphical user interface

Description automatically generated

Text

Description automatically generated

2. Create an API endpoint to read details for a particular shop.

* GET /api/shops/:id. It responds with a JSON object of the shop with the given ID (match the id route parameter).
* If there is no matching shop, respond with status code 404 and the JSON object { "error": "Shop not found: ${ID HERE}" }

A picture containing graphical user interface

Description automatically generatedA picture containing logo

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, application

Description automatically generated

Text

Description automatically generated

Graphical user interface

Description automatically generated with low confidence



3. Modify the listing API endpoint to find shops at or above a particular rating.

* Add an optional minRating query string parameter to: GET /api/shops. If specified, it filters the results to shops that have a rating greater than or equal to the given rating. If not specified, include all shops as before.

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, application, Word

Description automatically generated

Text

Description automatically generated

Jonathan also requested we do PUT, POST, and DELETE for complete CRUD functionality.

201 successfully created a thing

204 successful, but no data returned

404 I can’t find the thing you are asking for

**POST**

Text

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Doing a GET to confirm it was added to /shops/

A picture containing chart

Description automatically generated

Text

Description automatically generated

**PUT**

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, text, application

Description automatically generated

The Body confirms the update. Verified this with a GET for /shops/ as well.

Graphical user interface, text, application

Description automatically generated

**DELETE**

I opted to stick with the ‘hard delete’ and used the **filter** method as opposed to **slice**

A screenshot of a computer

Description automatically generated with medium confidence

Graphical user interface, text, application, Word

Description automatically generated



Confirming it was deleted from /shops/

Text

Description automatically generated