

Phillip Ball

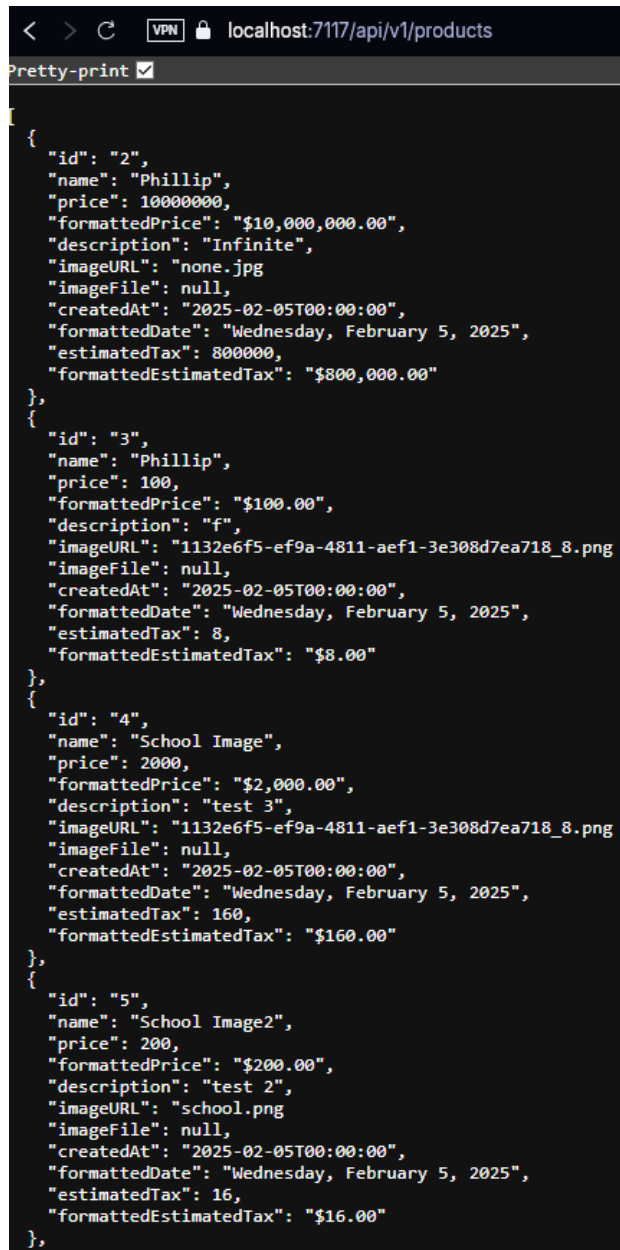
Prof. Jackson

2/21/25

CST-350

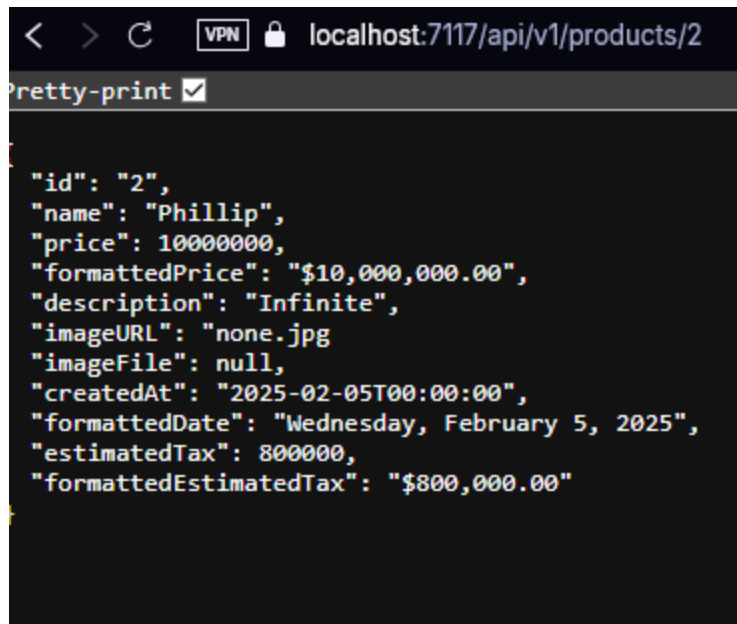
# REST & Right Click

## Screenshots:



```
[
  {
    "id": "2",
    "name": "Phillip",
    "price": 10000000,
    "formattedPrice": "$10,000,000.00",
    "description": "Infinite",
    "imageURL": "none.jpg",
    "imageFile": null,
    "createdAt": "2025-02-05T00:00:00",
    "formattedDate": "Wednesday, February 5, 2025",
    "estimatedTax": 800000,
    "formattedEstimatedTax": "$800,000.00"
  },
  {
    "id": "3",
    "name": "Phillip",
    "price": 100,
    "formattedPrice": "$100.00",
    "description": "f",
    "imageURL": "1132e6f5-ef9a-4811-aef1-3e308d7ea718_8.png",
    "imageFile": null,
    "createdAt": "2025-02-05T00:00:00",
    "formattedDate": "Wednesday, February 5, 2025",
    "estimatedTax": 8,
    "formattedEstimatedTax": "$8.00"
  },
  {
    "id": "4",
    "name": "School Image",
    "price": 2000,
    "formattedPrice": "$2,000.00",
    "description": "test 3",
    "imageURL": "1132e6f5-ef9a-4811-aef1-3e308d7ea718_8.png",
    "imageFile": null,
    "createdAt": "2025-02-05T00:00:00",
    "formattedDate": "Wednesday, February 5, 2025",
    "estimatedTax": 160,
    "formattedEstimatedTax": "$160.00"
  },
  {
    "id": "5",
    "name": "School Image2",
    "price": 200,
    "formattedPrice": "$200.00",
    "description": "test 2",
    "imageURL": "school.png",
    "imageFile": null,
    "createdAt": "2025-02-05T00:00:00",
    "formattedDate": "Wednesday, February 5, 2025",
    "estimatedTax": 16,
    "formattedEstimatedTax": "$16.00"
  }
],
```

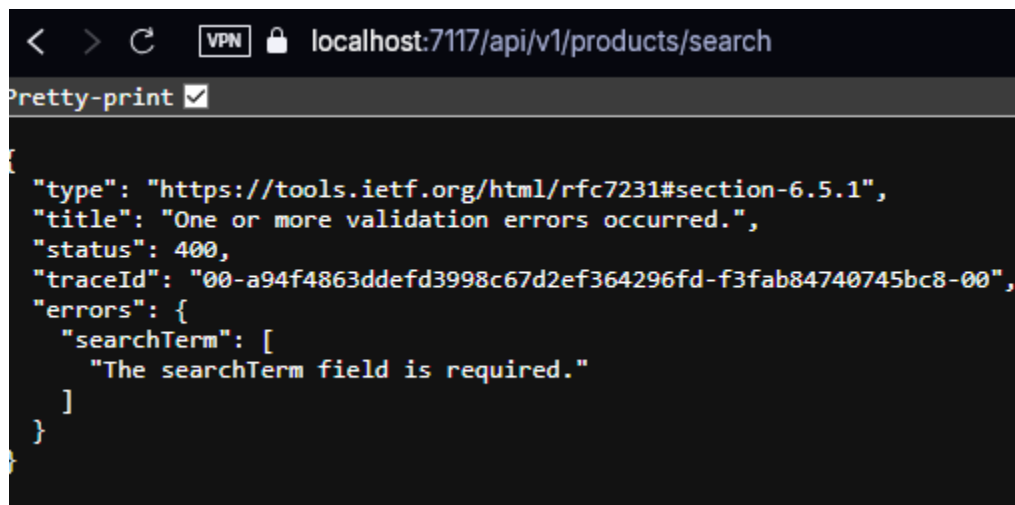
1: Browser request for GetAllProducts



```
< > ↻ VPN localhost:7117/api/v1/products/2
Pretty-print ☒

{
  "id": "2",
  "name": "Phillip",
  "price": 10000000,
  "formattedPrice": "$10,000,000.00",
  "description": "Infinite",
  "imageUrl": "none.jpg",
  "imageFile": null,
  "createdAt": "2025-02-05T00:00:00",
  "formattedDate": "Wednesday, February 5, 2025",
  "estimatedTax": 800000,
  "formattedEstimatedTax": "$800,000.00"
}
```

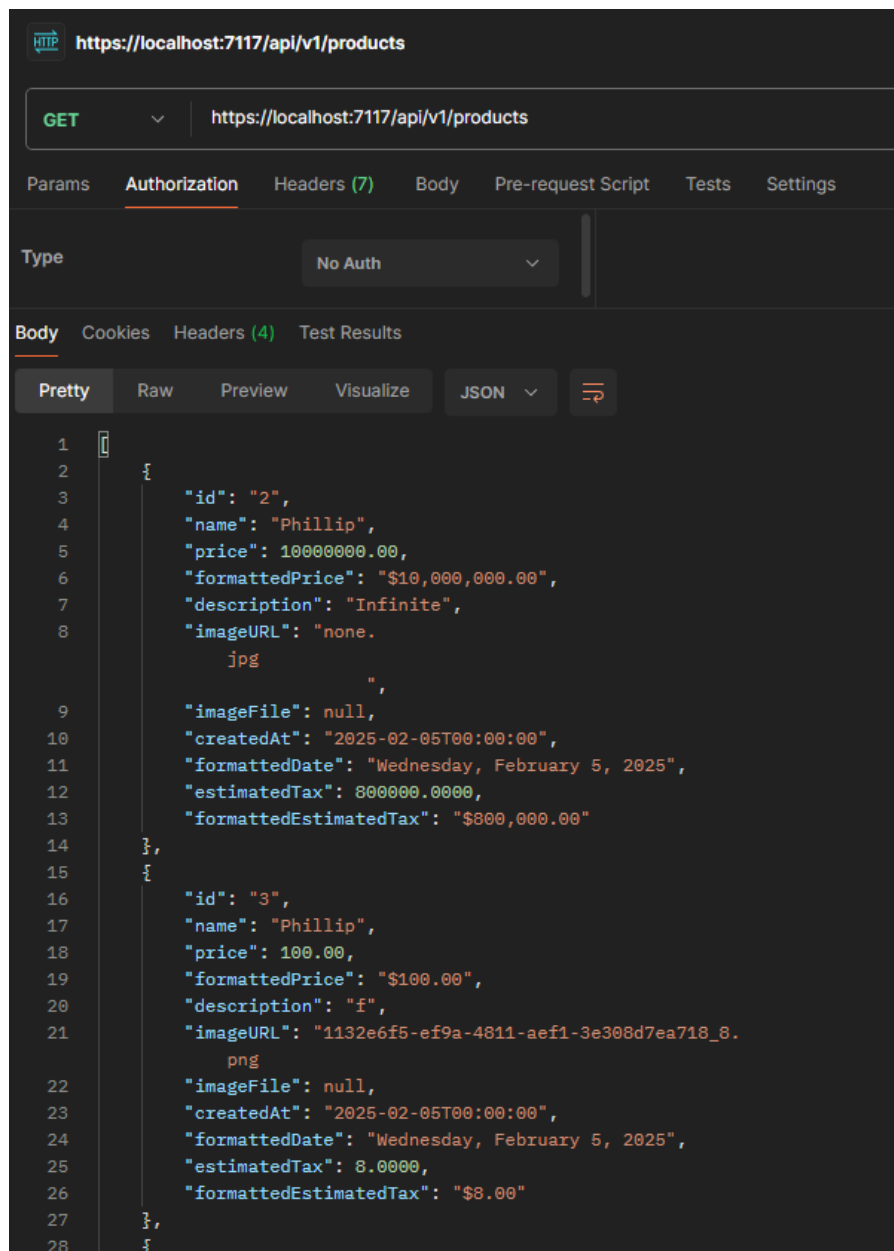
2: Browser request for GetProductById



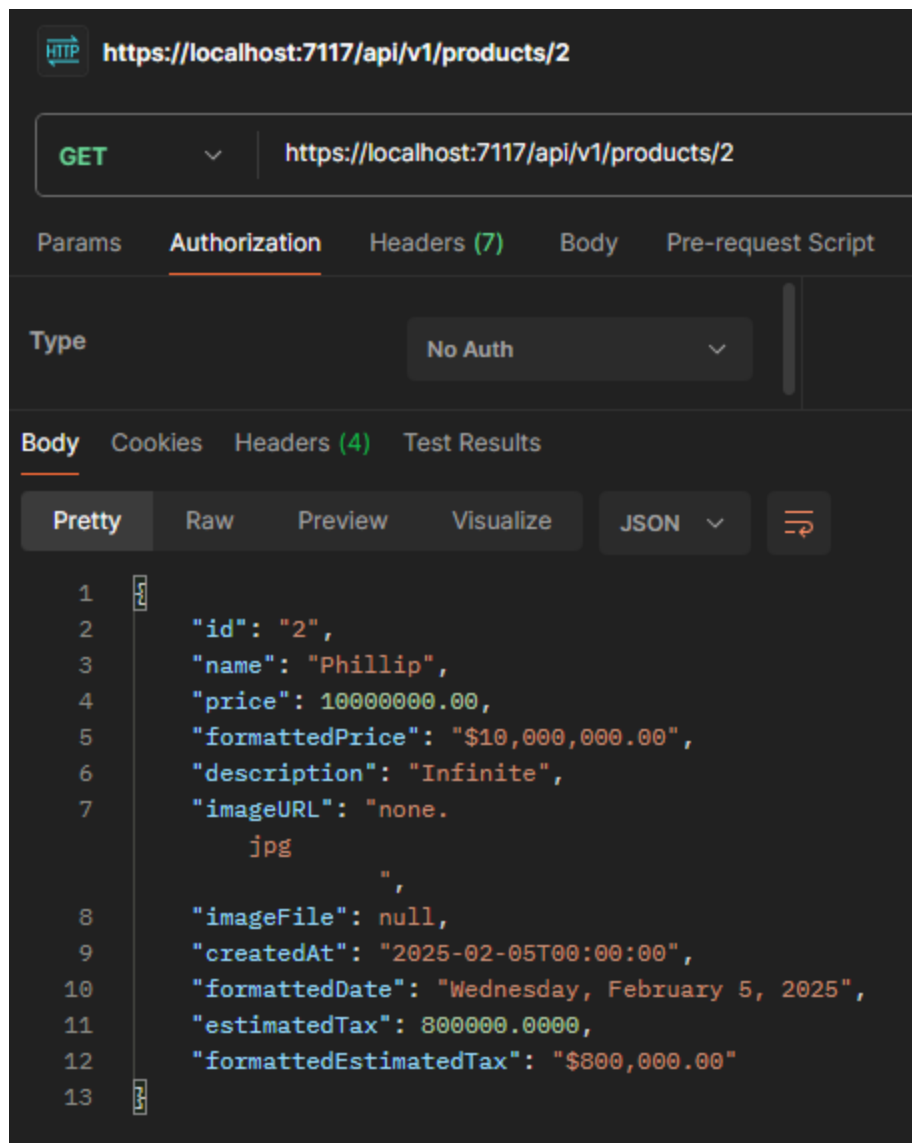
```
< > ↻ VPN localhost:7117/api/v1/products/search
Pretty-print ☒

{
  "type": "https://tools.ietf.org/html/rfc7231#section-6.5.1",
  "title": "One or more validation errors occurred.",
  "status": 400,
  "traceId": "00-a94f4863ddef3998c67d2ef364296fd-f3fab84740745bc8-00",
  "errors": {
    "searchTerm": [
      "The searchTerm field is required."
    ]
  }
}
```

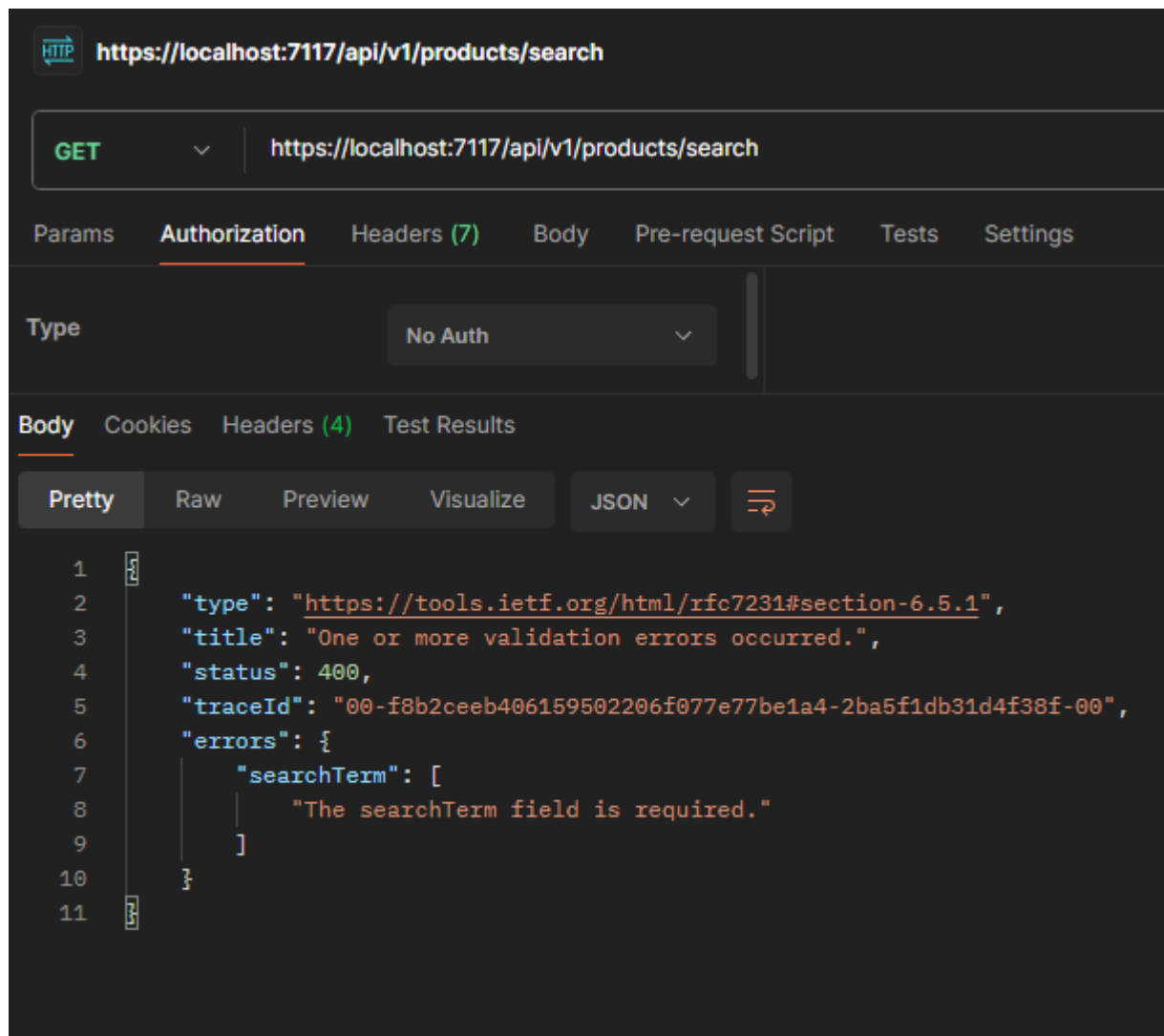
3: Browser request for 'search' with no term applied



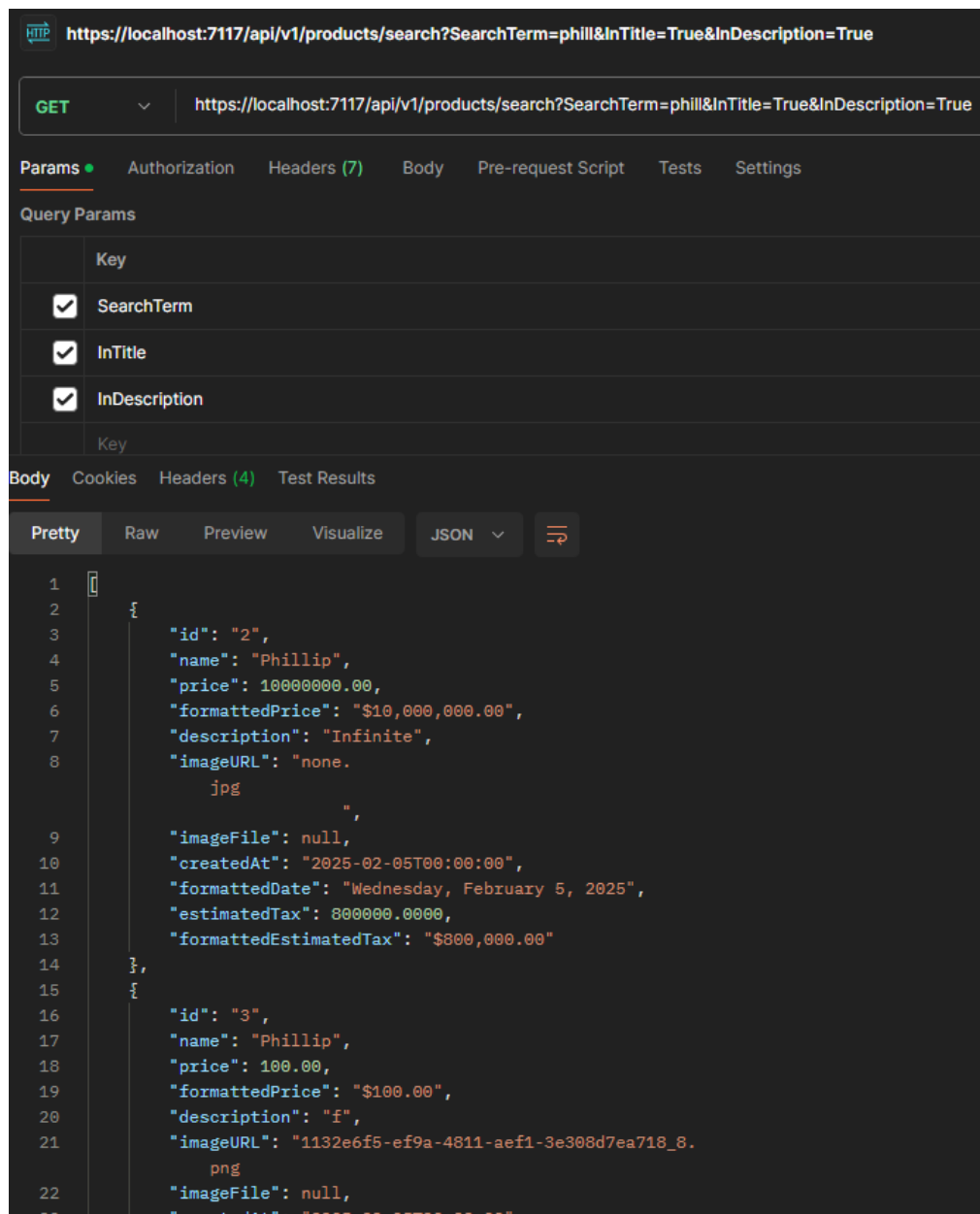
4: PostMan request of GetAllProducts



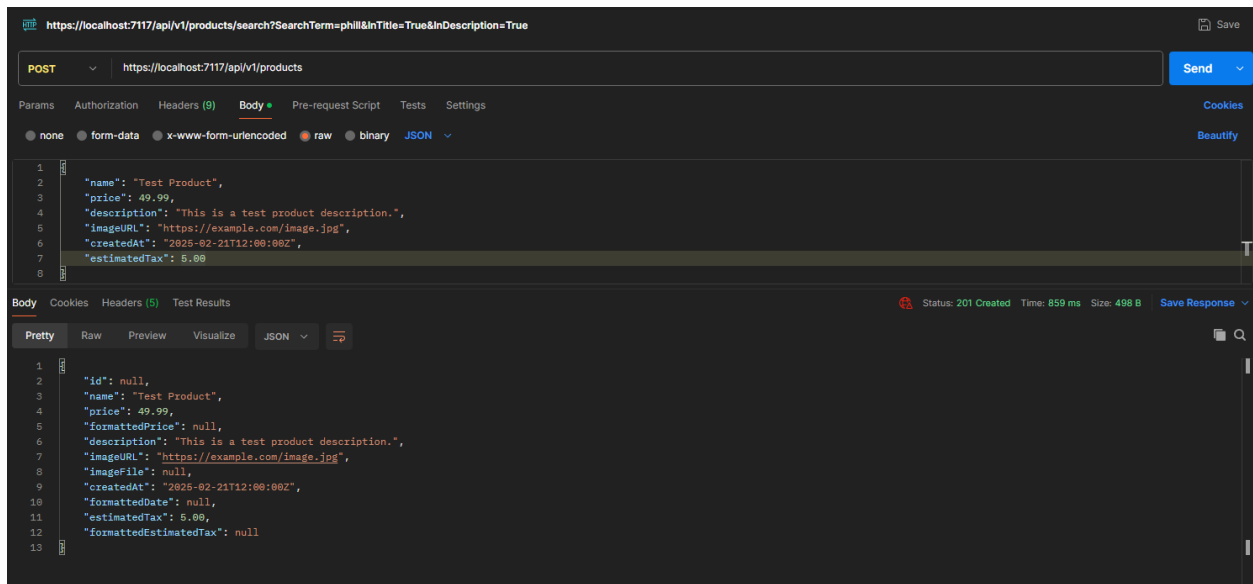
5: PostMan request for GetProductById



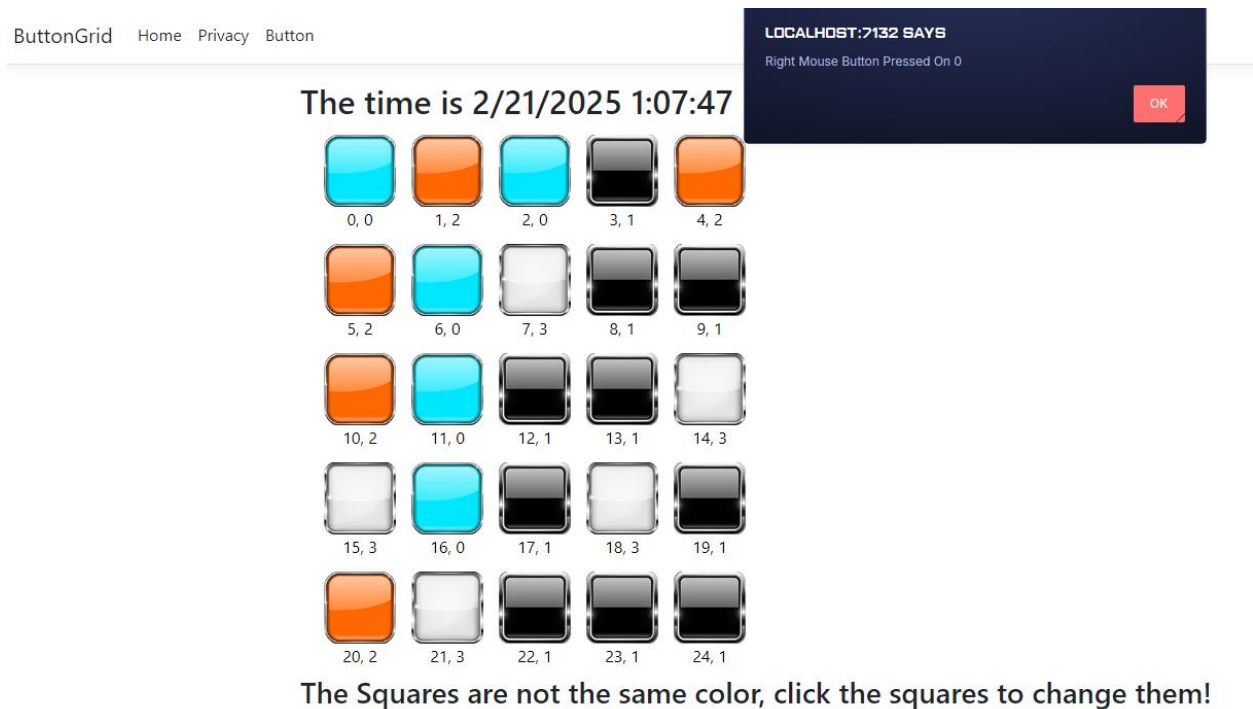
6: PostMan request for 'search' with no term applied



7: PostMan request for 'search' with a term applied



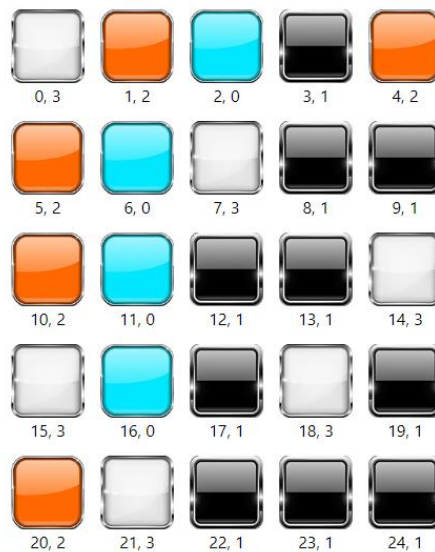
## 8: Creating a product using PostMan



## 9: Right click response on the first button



The time is 2/21/2025 1:07:47



The Squares are not the same color, click the squares to change them!

10: Left Click response on the first button

## What I learned:

I learned that splitting API requests into a REST service allows for smooth testing. I learned the difference between having requests through an MVC controller compared to a REST one. We implemented methods for creating, updating, and deleting products inside of the REST controller. We learned how to use PostMan for API testing. We learned how to handle right click functions using a different AJAX request to the controller based on the JS mouse input recieved.