**Coding Challenge**

In an e-commerce system that supports a web storefront, implement a Shopping Cart service

Shopping Cart service should support the following functionality:

Add/remove items to the cart

Retrieve contents of the cart

Calculate and display taxes for the items in the cart

Check-out / Purchase all the items in the cart

Include stubs for any other backend services needed to support the shopping cart functionality and the Check-out/purchase workflow.

In addition to implementing this service, also implement a test, calling each of the functions supported by the Shopping Cart service.

Make whatever assumptions needed to proceed with this challenge. The goal of this challenge is to assess the candidate’s knowledge of programming a service, with object oriented principles, and knowledge of the object-oriented programming language used.

**Approach:**

**The problem is solved using following tech stack.**

**Tech stack**: Spring boot, Java 8, Rest, Spring Data JPA, H2 in memory DB

         Junit classes are written to test methods.

Auto DDL is used for this assignment for the ease of development (In real time production ready projects it’s not recommended)

**Endpoints:**

Kindly refer the exported ‘**ShoppingCart.postman\_collection.json**’ to test the APIs in postman.

Add product to cart

<http://localhost:8080/addProduct>

  {

         "productid": 13,

        "name": "BB",

        "description" : "CC product",

        "price": 1000,

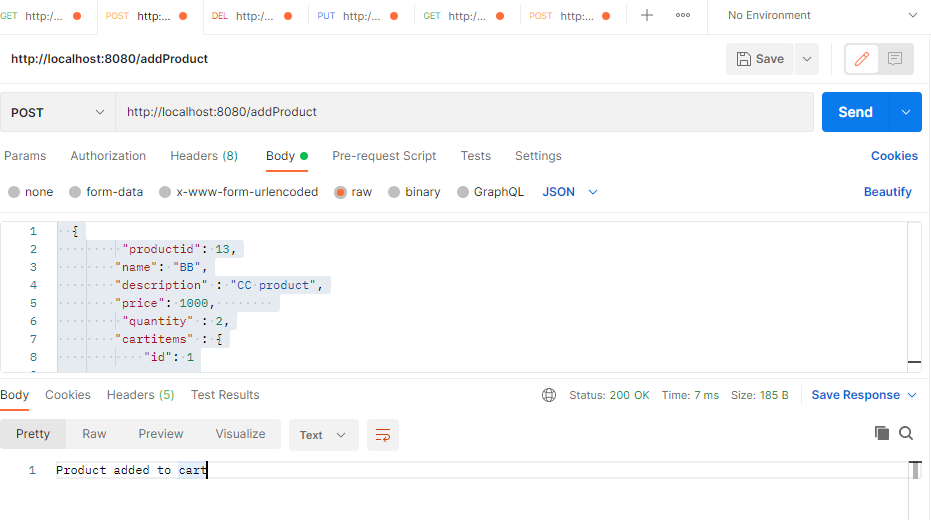
         "quantity" : 2,

        "cartitems" : {

            "id": 1

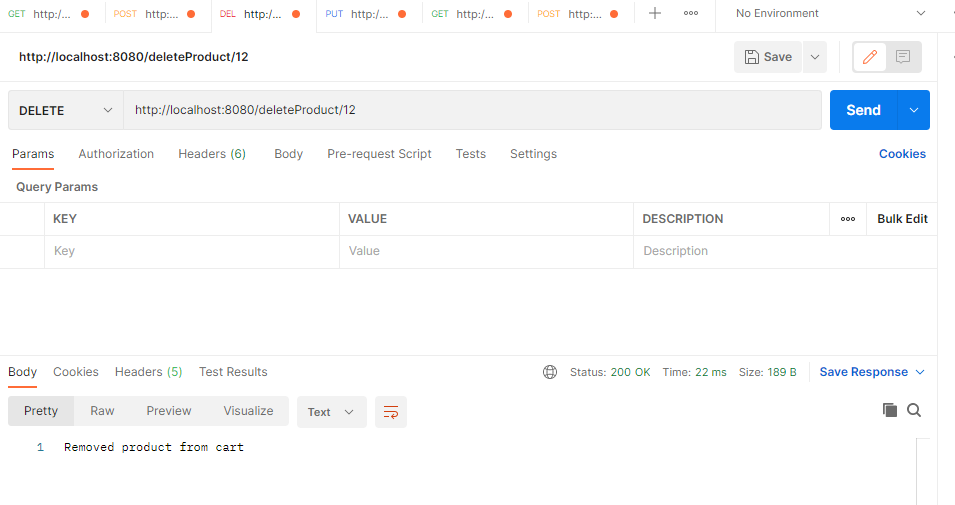
         }

    }



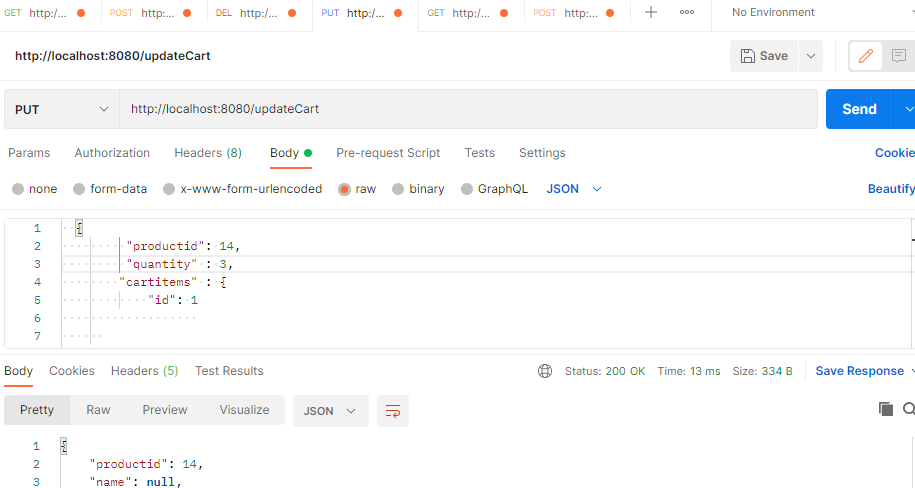
Remove product from cart

<http://localhost:8080/deleteProduct/12>



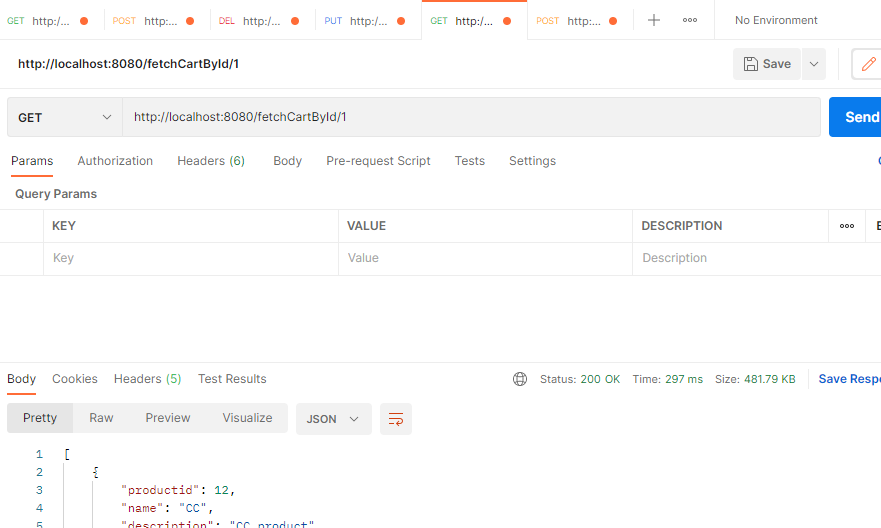
Update cart ( product quantity)

<http://localhost:8080/updateCart>



Fetch all product items of cart

<http://localhost:8080/fetchCartById/1>



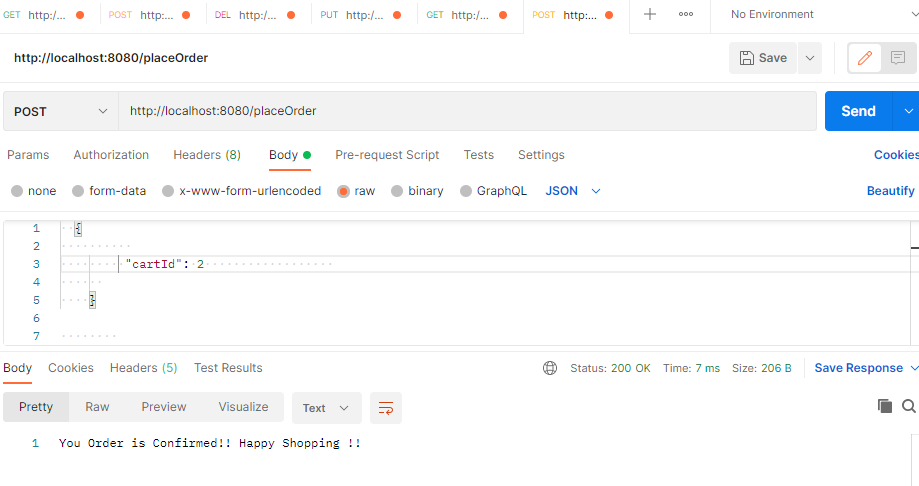
Checkout

<http://localhost:8080/placeOrder>

 {

         "cartId": 1

    }



When re-submitted the checkout request!

