

Need to implement

1. RestController

2. ProductService

3. ProductDao with one custom query

3. Create ProductNotFoundException in com.verizon.exeption package

Perform All 1. void AddProduct(),

2. List<Product> GetAllProducts(),

3. Product GetProductById(Integer id) throws

ProductNotFoundException

4. List<Product> GetAllProductsBetweenPrice(Integer low, Integer high);

5. void UpdateProduct(Integer id, Product product) throws

ProductNotFoundException

6. void DeleteProduct(Integer id) throws ProductNotFoundException

1. void AddProduct() --> POSTING DATA

POST http://localhost:7777/api/ POST http://localhost:9999/product POST http://localhost:7777/api/ POST http://localhost:7777/api/ + ...

HTTP http://localhost:9999/product Save </>

POST http://localhost:9999/product Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

none form-data x-www-form-urlencoded raw binary JSON

```
1 {
2   "product_id": 1,
3   "product_name": "Handwash",
4   "price": 100
5 }
```

Body Cookies Headers (5) Test Results 200 OK 14 ms 181 B Save Response

Pretty Raw Preview Visualize Text

1 Added to database

2. List<Product> GetAllProducts(),

POST http://localhost:7777/api/ GET http://localhost:9999/product POST http://localhost:7777/api/ POST http://localhost:7777/api/ + ...

HTTP http://localhost:9999/product Save </>

GET http://localhost:9999/product Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary

This request does not have a body

Body Cookies Headers (5) Test Results 200 OK 15 ms 321 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "product_name": "Soap",
4     "price": 120,
5     "product_id": 0
6   },
7   {
8     "product_name": "Surf",
9     "price": 70,
10    "product_id": 2
11  },
12  {
13    "product_name": "Handwash",
14    "price": 100,
15    "product_id": 1
16  }
17 }
```

3. Product GetProductById(Integer id) throws ProductNotFoundException

POST http://localhost:7777/api/ GET http://localhost:9999/product/2 POST http://localhost:7777/api/ POST http://localhost:7777/api/ + ...

http://localhost:9999/product/2 Save </>

GET http://localhost:9999/product/2 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary

This request does not have a body

Body Cookies Headers (5) Test Results 200 OK 16 ms 213 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "product_name": "Surf",
3   "price": 70,
4   "product_id": 2
5 }
```

4. List<Product> GetAllProductsBetweenPrice(Integer low,Integer high);

POST http://localhost:7777/api/ GET http://localhost:9999/product/price-range?low=100&high=500 POST http://localhost:7777/api/ POST http://localhost:7777/api/ + ...

http://localhost:9999/product/price-range?low=100&high=500 Save </>

GET http://localhost:9999/product/price-range?low=100&high=500 Send

Params Authorization Headers (6) Body Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary

This request does not have a body

Body Cookies Headers (5) Test Results 200 OK 43 ms 271 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "product_name": "Soap",
4     "price": 120,
5     "product_id": 0
6   },
7   {
8     "product_name": "Handwash",
9     "price": 100,
10    "product_id": 1
11  }
12 ]
```

5. void UpdateProduct(Integer id,Product product)throws ProductNotFoundException

POST http://localhost:7777/api PUT http://localhost:9999/product/2 POST http://localhost:7777/api POST http://localhost:7777/api + ...

HTTP **http://localhost:9999/product/2** Save </>

PUT http://localhost:9999/product/2 Send

Params Authorization Headers (8) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary JSON Beautify

```
1 {
2   "product_id": 2,
3   "product_name": "Handwash",
4   "price": 300
5 }
```

Body Cookies Headers (5) Test Results 200 OK 23 ms 218 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "product_name": "Handwash",
3   "price": 300,
4   "product_id": 2
5 }
```

POST http://localhost:7777/api GET http://localhost:9999/product POST http://localhost:7777/api POST http://localhost:7777/api + ...

HTTP **http://localhost:9999/product** Save </>

GET http://localhost:9999/product Send

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings Cookies

none form-data x-www-form-urlencoded raw binary

This request does not have a body

Body Cookies Headers (5) Test Results 200 OK 12 ms 326 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   {
3     "product_name": "Soap",
4     "price": 120,
5     "product_id": 0
6   },
7   {
8     "product_name": "Handwash",
9     "price": 300,
10    "product_id": 2
11  },
12 }
```

6. void DeleteProduct(Integer id)throws ProductNotFoundException

The screenshot displays an HTTP client interface with a list of requests at the top: `POST http://localhost:7777/api/`, `DEL http://localhost:9999/pro`, `POST http://localhost:7777/api/`, and `POST http://localhost:7777/api/`. The selected request is a `DELETE` to `http://localhost:9999/product/2`. The interface includes tabs for `Params`, `Authorization`, `Headers (6)`, `Body`, `Pre-request Script`, `Tests`, and `Settings`. The `Body` tab is active, showing `none` selected for the request body. A `Send` button is visible. Below the request details, the response is shown with a status of `200 OK`, a time of `22 ms`, and a size of `170 B`. The response body is `Deleted`.

HTTP `http://localhost:9999/product/2` Save </>

DELETE `http://localhost:9999/product/2` Send

Params Authorization Headers (6) **Body** Pre-request Script Tests Settings Cookies

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary

This request does not have a body

Body Cookies Headers (5) Test Results 200 OK 22 ms 170 B Save Response

Pretty Raw Preview Visualize Text 🔍

1 Deleted