

E-COMMERCE By Büşra Oran <busraoran@043@gmail.com>

Port for Commerce – 8080

Port for Inventory – 8081

Port for Shopping – 8082

For Inventory Micro service endpoints:

- <http://localhost:8081/inventory/categories>

The screenshot shows a REST client interface with the following details:

- URL:** `http://localhost:8081/inventory/categories`
- Method:** GET
- Response Status:** 200 OK, 619 ms, 257 B
- Response Body (JSON):**

```
{
  "categoryId": 1,
  "categoryName": "Home Accessories"
},
{
  "categoryId": 2,
  "categoryName": "School"
}
```

And when I call the same endpoint by using rest client as shown below, we get the same result as above.



- <http://localhost:8080/commerce/inventory/categories>





The screenshot shows a REST client interface with the following details:



- URL:** `http://localhost:8080/commerce/inventory/categories`
- Method:** GET
- Response Status:** 200 OK, 368 ms, 291 B
- Response Body (JSON):**

```
{
  "categoryId": 1,
  "categoryName": "Home Accessories",
  "productList": []
},
{
  "categoryId": 2,
  "categoryName": "School",
  "productList": []
}
```

- http://localhost:8081/inventory/product/1

New Collection / Get Product By Id Inventory  



Save    



POST  http://localhost:8081/inventory/product/1 **Send** 

Params Authorization Headers (7) Body Pre-request Script Tests Settings **Cookies**

Query Params



	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		


Body Cookies Headers (5) Test Results  200 OK 17 ms 241 B **Save Response** 



Pretty Raw Preview Visualize JSON  

```
1 {
2   "productName": "Carpet",
3   "salesPrice": 453.0,
4   "categoryName": "Home Accessories"
5 }
```

- http://localhost:8080/commerce/inventory/product/1

http://localhost:8080/commerce/inventory/categories  



Save 



GET  http://localhost:8080/commerce/inventory/categories **Send** 

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

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results  200 OK 368 ms 291 B **Save Response** 

Pretty Raw Preview Visualize JSON  

```
1 [
2   {
3     "categoryId": 1,
4     "categoryName": "Home Accessories",
5     "productList": []
6   },
7   {
8     "categoryId": 2,
9     "categoryName": "School",
10    "productList": []
11  }
```

- http://localhost:8081/inventory/products/1

New Collection / Get Products by category id inventory

POST http://localhost:8081/inventory/products/1 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results 200 OK 2.38 s 322 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "productName": "Carpet",
4     "salesPrice": 453.0,
5     "categoryName": "Home Accessories"
6   },
7   {
8     "productName": "Curtain",
9     "salesPrice": 899.0,
10    "categoryName": "Home Accessories"
11  }
```

- http://localhost:8080/commerce/inventory/products/1

New Collection / Get Products by category id

POST http://localhost:8080/commerce/inventory/products/1 Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

KEY	VALUE	DESCRIPTION	Bulk Edit
Key	Value	Description	

Body Cookies Headers (5) Test Results 200 OK 28 ms 322 B Save Response

Pretty Raw Preview Visualize JSON

```
1 [
2   {
3     "productName": "Carpet",
4     "salesPrice": 453.0,
5     "categoryName": "Home Accessories"
6   },
7   {
8     "productName": "Curtain",
9     "salesPrice": 899.0,
10    "categoryName": "Home Accessories"
11  }
```

For shopping Micro Service endpoints:

- <http://localhost:8082/shopping/cart/create>

The screenshot shows a Postman interface for a new collection named "create cart shopping". The request is a POST to <http://localhost:8082/shopping/cart/create>. The "Body" tab is selected, showing a JSON payload: `{ "customerName": "Büşra Öran" }`. The response is a 200 OK status, received in 186 ms with 165 B of data. The response body is empty, displayed in the "Pretty" view.

So, even though we do not give any spesifical input to the controller, it set the other parameters and creates the cart anyway. Now lets dive in to the client part, from the client side, we pass the json as request body to our shopping microservice to create a database object on related table.

- <http://localhost:8080/commerce/shopping/cart/create>

The screenshot shows a Postman interface for a new collection named "create cart". The request is a POST to <http://localhost:8080/commerce/shopping/cart/create>. The "Body" tab is selected, showing a JSON payload: `{ "customerName": "Büşra Öran 4" }`. The response is a 200 OK status, received in 4.22 s with 165 B of data. The response body is empty, displayed in the "Pretty" view.

- `http://localhost:8082/shopping/checkout/1`

New Collection / **cart checkout shopping** Save ... 🔗 💬

POST `http://localhost:8082/shopping/checkout/1` Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results 🌐 200 OK 676 ms 191 B Save Response

Pretty Raw Preview Visualize Text 🔗 🔍

1 Checked out with cart id: 1

- `http://localhost:8080/commerce/shopping/checkout/1`

New Collection / **cart checkout** Save ... 🔗 💬

POST `http://localhost:8080/commerce/shopping/checkout/1` Send

Params Authorization Headers (7) Body Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results 🌐 200 OK 671 ms 191 B Save Response

Pretty Raw Preview Visualize Text 🔗 🔍

1 Checked out with cart id: 1

In the background of this the only thing is changing is the value of `cart_status` column in the database. We have 0 for newly created carts which is the default setter when we first create the cart. If the client chose to checkout then the status is changing to the 1 on the database.

- <http://localhost:8082/shopping/cartProduct/add>

New Collection / add new cartProduct to shopping db

POST <http://localhost:8082/shopping/cartProduct/add> Send

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

● none ● form-data ● x-www-form-urlencoded ● **raw** ● binary ● GraphQL **JSON** Beautify

```
1 {
2   ... "productId":1,
3   ... "salesPrice":10,
4   ... "salesQuantity":24
5 }
```

Body Cookies Headers (5) Test Results 200 OK 422 ms 165 B Save Response

Pretty Raw Preview Visualize JSON

```
1 5
```

- <http://localhost:8082/shopping/cartProduct/2>

<http://localhost:8082/shopping/cartProduct/2> Save

POST <http://localhost:8082/shopping/cartProduct/2> Send

Params Authorization Headers (7) **Body** Pre-request Script Tests Settings Cookies

Query Params

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Body Cookies Headers (5) Test Results 200 OK 228 ms 253 B Save Response

Pretty Raw Preview Visualize JSON

```
1 {
2   "cartProductId": 2,
3   "productId": 1,
4   "salesQuantity": 3,
5   "salesPrice": 100.0,
6   "lineAmount": 300.0
7 }
```

From this point, I could not be able to run successfully the other endpoints that's why this document does not include the other endpoints that are required for to complete successfully the project.