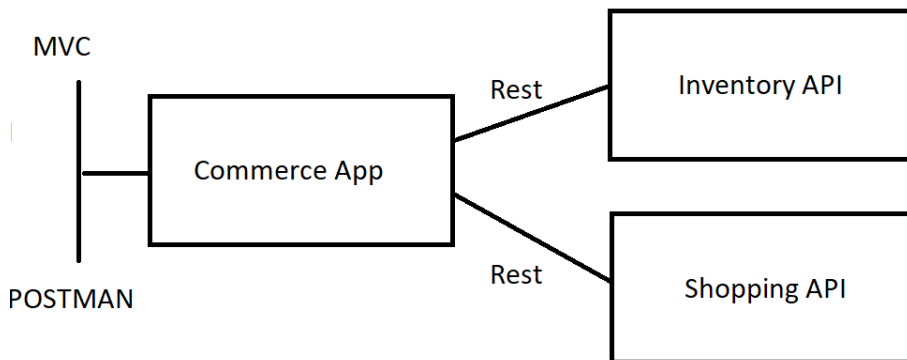
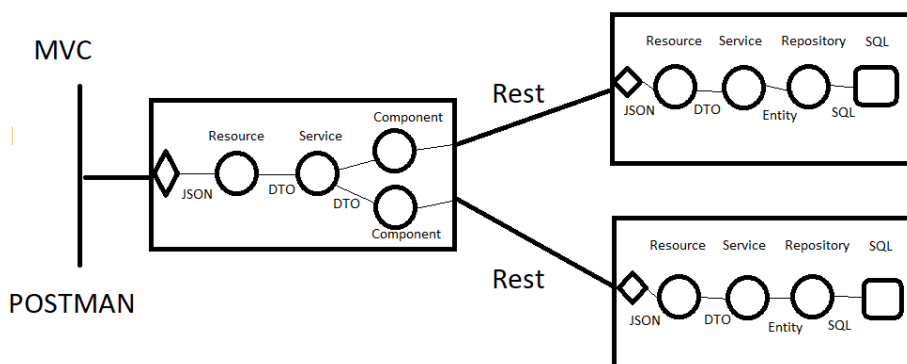


Project General View

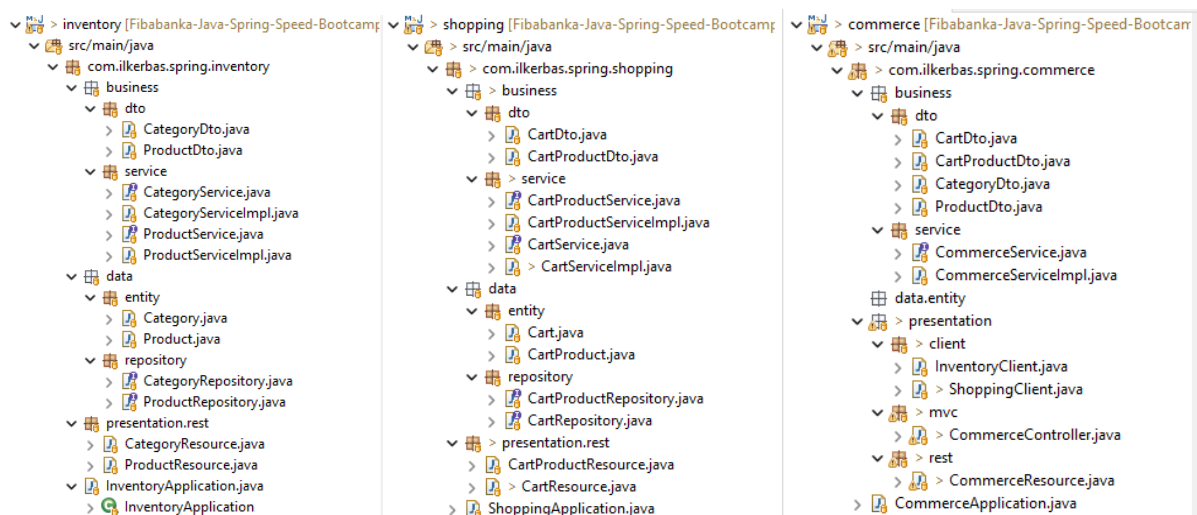
1.



2.



Structure of Applications



PORTS

Inventory → 8081, Shopping → 8082, Commerce → 8080

2 MySQL database

EXPECTED TASKS

English

All the given requests are working as expected.

MVC controller was used as client. Requests are sent to MVC controller by POSTMAN.

Türkçe

Verilen tüm istekler beklenildiği gibi çalışmaktadır.

Client olarak MVC controller kullanıldı. MVC Controllera istekler POSTMAN tarafından gönderildi.

- GET /commerce/inventory/categories

MVC Controller Method

```
// find all categories
@GetMapping("/inventory/categories")
@ResponseBody
public List<CategoryDto> getAllCategories() {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/inventory/categories/";

    @SuppressWarnings("unchecked")
    ArrayList<CategoryDto> categories = restTemplate.getForObject(url, ArrayList.class);

    return categories;
}
```

Request

The screenshot shows a Postman interface with a GET request to `http://localhost:8080/commerce/mvc/inventory/categories`. The response is a JSON array of three category objects. The status is 200 OK, time is 465 ms, and size is 288 B.

KEY	VALUE	DESCRIPTION	...	Bulk Edit
categoryId	1			
categoryName	technology			
categoryId	2			
categoryName	gift			
categoryId	3			
categoryName	baby			

GET **/commerce/inventory/products/{categoryId}**

MVC Controller Method

```
// find all products by given category id
@GetMapping("/inventory/products/{categoryId}")
@ResponseBody
public List<ProductDto> getProductsByCategoryId(@PathVariable("categoryId") long categoryId) {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/inventory/products/" + categoryId;

    @SuppressWarnings("unchecked")
    List<ProductDto> productDtos = restTemplate.getForObject(url, ArrayList.class);

    return productDtos;
}
```

Request

GET

http://localhost:8080/commerce/mvc/inventory/products/3

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

Status: 200 OK

Time: 34 ms

Size: 314 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1 {
2   {
3     "productId": 4,
4     "productName": "toy car",
5     "salesPrice": 57.0,
6     "categoryId": 3
7   },
8   {
9     "productId": 5,
10    "productName": "toy house",
11    "salesPrice": 124.0,
12    "categoryId": 3
13  }
14 }
```

GET /commerce/inventory/product/{productId}

MVC Controller Method

```
// find product by given id
@GetMapping("/inventory/product/{productId}")
@ResponseBody
public ProductDto getProductById(@PathVariable("productId") long productId) {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/inventory/product/" + productId;

    ProductDto productDto = restTemplate.getForObject(url, ProductDto.class);
    return productDto;
}
```

Request

GET

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

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

⌚ Status: 200 OK Time: 213 ms Size: 235 B Save Response

Pretty

Raw

Preview

Visualize

JSON

1

2

3

4

5

6

Text

```
"productId": 1,
"productName": "mac",
"salesPrice": 40000.0,
"categoryId": 1
```

GET /commerce/shopping/cart/create > cartId

MVC Controller Method

```
// create a cart
@GetMapping("/shopping/cart/create")
@ResponseBody
public long postCart() {

    CartDto postedCartDto = new CartDto();
    postedCartDto.setCartStatus(false);
    postedCartDto.setCustomerName("fibabanka");
    postedCartDto.setTotalAmount(2222);
    postedCartDto.setCartProductDtos(new ArrayList<CartProductDto>());

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/shopping/cart/create";

    Long cartId = restTemplate.postForObject(url, postedCartDto, Long.class);

    return cartId;
}
```

}

Request

GET

http://localhost:8080/commerce/mvc/shopping/cart/create

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

Status: 200 OK

Time: 48 ms

Size: 166 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

18

Database Entity

	cart_id	cart_status	customer_name	total_amount
▶	2	1	ilker	192
	7	0	IlkerBas	3245
	18	0	fibabanka	2222
*	NULL	NULL	NULL	NULL

POST /commerce/shopping/cart/add < CartProductDto

MVC Controller Method

```
// add product to cart
@GetMapping("/shopping/cart/add")
@ResponseBody
public CartProductDto addProduct() {

    CartProductDto cartProductDto = new CartProductDto();
    cartProductDto.setLineAmount(88);
    cartProductDto.setProductId(5);
    cartProductDto.setSalesPrice(2345);
    cartProductDto.setSalesQuantity(3);
    cartProductDto.setCartId(18);

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/shopping/cart/add";

    CartProductDto cartProduct = restTemplate.postForObject(url, cartProductDto, CartProductDto.class);

    return cartProduct;
}
```

Request

GET

http://localhost:8080/commerce/mvc/shopping/cart/add

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

Status: 200 OK

Time: 978 ms

Size: 288 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

```
1  {
2    "cartProductId": 20,
3    "productId": 5,
4    "salesQuantity": 3,
5    "salesPrice": 2345,
6    "lineAmount": 88,
7    "cartId": 18,
8    "productName": "toy house"
9  }
```

Database Entity

	cart_product_id	line_amount	product_id	sales_price	sales_quantity	cart_id
▶	20	88	5	2345	3	18
*	NULL	NULL	NULL	NULL	NULL	NULL

	product_id	product_name	sales_price	category_id
▶	1	mac	40000	1
	2	asus	23870	1
	3	mask	120	2
	4	toy car	57	3
	5	toy house	124	3
*	NULL	NULL	NULL	NULL

DELETE /commerce/shopping/cart/{cartId}/remove/{productId}

MVC Controller Method

```
// delete cart product by given cartId and productId
@DeleteMapping("/shopping/cart/{cartId}/remove/{productId}")
@ResponseBody
public void deleteCartProduct(@PathVariable("cartId") long cartId, @PathVariable("productId") long productId) {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/shopping/cart/" + cartId + "/remove/" + productId;

    restTemplate.delete(url);

}
```

Request

DELETE

http://localhost:8080/commerce/mvc/shopping/cart/18/remove/20

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 (4)

Test Results

Status: 200 OK

Time: 81 ms

Size: 123 B

Save Response

Pretty

Raw

Preview

Visualize

Database Entity

	cart_product_id	line_amount	product_id	sales_price	sales_quantity	cart_id
*	NULL	NULL	NULL	NULL	NULL	NULL

	cart_id	cart_status	customer_name	total_amount
▶	2	1	ilker	192
	7	0	IlkerBas	3245
	18	1	fibabanka	2222
*	NULL	NULL	NULL	NULL

GET **/commerce/shopping/checkout/{cartId} 0 -> 1**

MVC Controller Method

```
// checkout
@GetMapping("/shopping/checkout/{cartId}")
@ResponseBody
public boolean checkout(@PathVariable("cartId") long cartId) {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/shopping/checkout/" + cartId;

    boolean result = restTemplate.getForObject(url, Boolean.class);

    return result;
}
```

Request

GET

http://localhost:8080/commerce/mvc/shopping/checkout/18

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

Status: 200 OK

Time: 67 ms

Size: 168 B

Save Response

Pretty

Raw

Preview

Visualize

JSON

1

true

Database Entity

	cart_id	cart_status	customer_name	total_amount
▶	2	1	ilker	192
	7	0	IlkerBas	3245
	18	1	fibabanka	2222
*	NULL	NULL	NULL	NULL

GET **/commerce/shopping/cart/find > CartDto**

MVC Controller Method

```
// find cart by id
@GetMapping("/shopping/cart/find/{cartId}")
@ResponseBody
public CartDto find(@PathVariable("cartId") long cartId) {

    RestTemplate restTemplate = new RestTemplate();

    String url = "http://localhost:8080/commerce/shopping/cart/find/" + cartId;

    CartDto cartDto = restTemplate.getForObject(url, CartDto.class);

    return cartDto;
}
```

Request

GET

http://localhost:8080/commerce/mvc/shopping/cart/find/18

Send

ParamsAuthorizationHeaders (6)BodyPre-request ScriptTestsSettingsCookies

BodyCookiesHeaders (5)Test Results

Status: 200 OKTime: 57 msSize: 380 BSave Response

PrettyRawPreviewVisualizeJSON

```
1  {
2    "cartId": 18,
3    "customerName": "fibabanka",
4    "totalAmount": 2222,
5    "cartStatus": false,
6    "cartProductDtos": [
7      {
8        "cartProductId": 20,
9        "productId": 0,
10       "salesQuantity": 3,
11       "salesPrice": 2345,
12       "lineAmount": 88,
13       "cartId": 18,
14       "productName": null
15     }
16   ]
17 }
```