

1. HTTP Server Output (Without Docker, Edlab)

```
elinux1 spring24-lab2-mitalee18-aishwaryax) > Starting server on 127.0.0.1:56363
The Order server is running.
The Catalog server is running.
```

2. HTTP Server Output (Docker, Edlab)

```
spring24-lab2-mitalee18-aishwaryax-order-1 | Order received for toy: Fox and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-order-1 | Order has been processed with OrderNumber 1906
spring24-lab2-mitalee18-aishwaryax-order-1 | Order processed for toy: Fox and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "POST /orders HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-frontend-1 | Thread assigned : 139645607458496, Client ID: 4
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "GET /products/Whale HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-order-1 | Order received for toy: Whale and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-order-1 | Order has been processed with OrderNumber 1907
spring24-lab2-mitalee18-aishwaryax-order-1 | Order processed for toy: Whale and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "POST /orders HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-frontend-1 | Thread assigned : 139645607458496, Client ID: 4
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "GET /products/Whale HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-frontend-1 | Thread assigned : 139645607458496, Client ID: 4
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "GET /products/Whale HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-order-1 | Order received for toy: Elephant and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-order-1 | Order has been processed with OrderNumber 1908
spring24-lab2-mitalee18-aishwaryax-order-1 | Order processed for toy: Elephant and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "POST /orders HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-frontend-1 | Thread assigned : 139645607458496, Client ID: 4
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:56] "GET /products/Tux HTTP/1.1" 200 -
spring24-lab2-mitalee18-aishwaryax-order-1 | Order received for toy: Tux and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-order-1 | Order has been processed with OrderNumber 1909
spring24-lab2-mitalee18-aishwaryax-order-1 | Order processed for toy: Tux and for quantity: 1
spring24-lab2-mitalee18-aishwaryax-frontend-1 | 172.18.0.1 - - [02/Apr/2024 20:53:57] "POST /orders HTTP/1.1" 200 -
```

3. Bring up containers using docker compose

```
elnsrv3 spring24-lab2-mitalee18-aishwaryax) > docker compose up
[+] Running 3/3
  Container spring24-lab2-mitalee18-aishwaryax-catalog-1 Recreated 0.8s
  Container spring24-lab2-mitalee18-aishwaryax-order-1 Recreated 0.8s
  Container spring24-lab2-mitalee18-aishwaryax-frontend-1 Recreated 0.7s
Attaching to spring24-lab2-mitalee18-aishwaryax-catalog-1, spring24-lab2-mitalee18-aishwaryax-frontend-1, spring24-lab2-mitalee18-aishwaryax-order-1
spring24-lab2-mitalee18-aishwaryax-catalog-1 | The Catalog server is running.
spring24-lab2-mitalee18-aishwaryax-order-1 | The Order server is running.
spring24-lab2-mitalee18-aishwaryax-frontend-1 | Starting server on spring24-lab2-mitalee18-aishwaryax-frontend-1:56363
```

4. Tear down container

```
^CGracefully stopping... (press Ctrl+C again to force)
Aborting on container exit...
[+] Stopping 3/3
  Container spring24-lab2-mitalee18-aishwaryax-frontend-1 Stopped 10.8s
  Container spring24-lab2-mitalee18-aishwaryax-order-1 Stopped 10.8s
  Container spring24-lab2-mitalee18-aishwaryax-catalog-1 Stopped 10.8s
canceled
```

5. Postman screenshots for various test cases

Valid order test case

A screenshot of the Postman interface showing a successful POST request to `127.0.0.1:8004/orders`. The request body is a JSON object: `{ "name": "Tux", "quantity": 1 }`. The response status is `200 OK` with a time of `15 ms` and a size of `153 B`. The response body is a JSON object: `{ "data": { "order_number": 5 } }`.

```
1 {
2   "name": "Tux",
3   "quantity": 1
4 }
```

```
1 {
2   "data": {
3     "order_number": 5
4   }
5 }
```

Invalid name test case

A screenshot of the Postman interface showing a successful POST request to `127.0.0.1:8004/orders`. The request body is a JSON object: `{ "name": "INVALID", "quantity": 1 }`. The response status is `200 OK` with a time of `15 ms` and a size of `183 B`. The response body is a JSON object: `{ "error": { "code": 304, "message": "could not post order" } }`.

```
1 {
2   "name": "INVALID",
3   "quantity": 1
4 }
```

```
1 {
2   "error": {
3     "code": 304,
4     "message": "could not post order"
5   }
6 }
```

Invalid quantity (insufficient) test case

A screenshot of the Postman interface showing a POST request to `127.0.0.1:8004/orders`. The request body is a JSON object: `{ "name": "Tux", "quantity": 10000 }`. The response status is `200 OK` with a time of `15 ms` and a size of `183 B`. The response body is a JSON object: `{ "error": { "code": 304, "message": "could not post order" } }`.

```
1 {
2   "name": "Tux",
3   "quantity": 10000
4 }
```

Valid test case

GET

127.0.0.1:56363/products/Tux

Send

Params

Authorization

Headers (7)

Body

Pre-request Script

Tests

Settings

Cookies

Query Params

	Key	Value	Description	...	Bulk Edit
	Key	Value	Description		

ody

Cookies

Headers (3)

Test Results

Status: 200 OK

Time: 73 ms

Size: 184 B

Save as example

...

Pretty

Raw

Preview

Visualize

JSON

...

...

```
1 {
2   "data": {
3     "name": "Tux",
4     "price": 2555.0,
5     "quantity": 2942
6   }
7 }
```

Invalid Query

GET

127.0.0.1:56363/products/NONE

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 (3)

Test Results

Status: 200 OK

Time: 13 ms

Size: 180 B

Save as example

...

Pretty

Raw

Preview

Visualize

JSON

...

...

```
1 {
2   "error": {
3     "code": 404,
4     "message": "product not found"
5   }
6 }
```

GRPC Postman Outputs

Catalog Service - Query - Valid Case

The screenshot shows the Postman interface for a gRPC call. The top bar indicates the endpoint is `localhost:56361` and the method is `Catalog / Query`. The `Message` tab is active, showing a JSON request:

```
{  "itemName": "Tux"}
```

. The `Response` tab is also active, showing a successful response with status code `0 OK` and a time of `49 ms`. The response body is a JSON object:

```
{  "cost": 2555,  "stock": 2940,  "name": "Tux"}
```

.

Catalog Service - Query - InValid Toy Case

The screenshot shows the Postman interface for a gRPC call. The top bar indicates the endpoint is `localhost:56361` and the method is `Catalog / Query`. The `Message` tab is active, showing a JSON request:

```
{  "itemName": "Invalid"}
```

. The `Response` tab is also active, showing an error response with status code `3 INVALID_ARGUMENT` and a time of `41 ms`. Below the response, there is an illustration of a person holding a sign that says "Invalid argument". The text below the illustration reads: "Invalid argument. You have specified an invalid argument. Check the message and invoke the method again." A red button labeled "Invalid argument provided" is visible at the bottom.

Catalog Service - Buy - Valid Case

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

- URL:** localhost:56361
- Method:** Catalog / Buy
- Message Tab:** Contains a JSON request body:

```
{  "itemName": "Tux",  "quantity": 1}
```
- Response Tab:** Shows a successful response with status code 200 OK and a time of 58 ms. The response body is:

```
{  "response": 1}
```

Catalog Service - Buy - Insufficient Quantity Case

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

- URL:** localhost:56361
- Method:** Catalog / Buy
- Message Tab:** Contains a JSON request body:

```
{  "itemName": "Tux",  "quantity": 1000000000}
```
- Response Tab:** Shows an error response with status code 400 and a time of 54 ms. The response body is:

```
{  "response": 0}
```

Catalog Service - Buy - Invalid Toy Case

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

- URL:** localhost:56361
- Method:** Catalog / Buy
- Message Tab:** Contains a JSON request body:

```
{  "itemName": "Invalid",  "quantity": 1}
```
- Response Tab:** Shows an error response with status code 400 and a time of 58 ms. The response body is:

```
{  "response": -1}
```

Order Service - postOrder - Valid Case

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

- URL:** localhost:56362
- Method:** Order / buyOrder
- Message Tab:** Contains a JSON request body:

```
1 {  
2   "ToyName": "Elephant",  
3   "ToyQuantity": 1  
4 }
```
- Response Tab:** Shows the response with status code 200 OK and time 146 ms. The response body is:

```
1 {  
2   "Message": "Order processed successfully",  
3   "OrderNumber": 4  
4 }
```

Order Service - postOrder - Insufficient Quantity Case

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

- URL:** localhost:56362
- Method:** Order / buyOrder
- Message Tab:** Contains a JSON request body:

```
1 {  
2   "ToyName": "Elephant",  
3   "ToyQuantity": 1000000000  
4 }
```
- Response Tab:** Shows the response with status code 200 OK and time 58 ms. The response body is:

```
1 {  
2   "Message": "Insufficient Quantity",  
3   "OrderNumber": -1  
4 }
```

Order Service - postOrder - Invalid Toy Case

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

- URL:** localhost:56362
- Method:** Order / buyOrder
- Message Tab:** Contains a JSON request body:

```
1 {  
2   "ToyName": "Invalid",  
3   "ToyQuantity": 1  
4 }
```
- Response Tab:** Shows the response with status code 200 OK and time 59 ms. The response body is:

```
1 {  
2   "Message": "Invalid Toy Name",  
3   "OrderNumber": -1  
4 }
```

Multithreading approach with client maintaining sessions in HTTP service

Thread assigned : 13072687104, Client ID: 2
Thread assigned : 13096452096, Client ID: 5
Thread assigned : 13113241600, Client ID: 3
Thread assigned : 13130031104, Client ID: 8
Thread assigned : 13146820608, Client ID: 1
Thread assigned : 13163610112, Client ID: 4
Thread assigned : 13180399616, Client ID: 9
Thread assigned : 13096452096, Client ID: 5
Thread assigned : 13197189120, Client ID: 10
Thread assigned : 13096452096, Client ID: 1
Thread assigned : 13146820608, Client ID: 6
Thread assigned : 13197189120, Client ID: 10
Thread assigned : 13213978624, Client ID: 7
Thread assigned : 13197189120, Client ID: 6
Thread assigned : 13072687104, Client ID: 2
Thread assigned : 13096452096, Client ID: 1
Thread assigned : 13113241600, Client ID: 3
Thread assigned : 13163610112, Client ID: 8

Client ID and Thread ID mappings are mostly maintained but in some cases they are not due to thread being already assigned to some other Client ID.

Persistence

```
✓ Container spring24-lab2-mitalee18-aishwaryax-catalog-1 Recreated
10.6s
✓ Container spring24-lab2-mitalee18-aishwaryax-order-1 Recreated
10.8s
✓ Container spring24-lab2-mitalee18-aishwaryax-frontend-1 Recreated
11.4s
Attaching to catalog-1, frontend-1, order-1
frontend-1 | Starting server on spring24-lab2-mitalee18-aishwaryax-frontend-1:56363
order-1 | The Order server is running.
catalog-1 | The Catalog server is running.
frontend-1 | 192.168.65.1 -- [03/Apr/2024 02:47:39] "GET /products/Tux HTTP/1.1" 200 -
order-1 | Order received for toy: Tux and for quantity: 1
order-1 | Order has been processed with OrderNumber 4
order-1 | Order processed for toy: Tux and for quantity: 1
frontend-1 | 192.168.65.1 -- [03/Apr/2024 02:49:12] "POST /orders HTTP/1.1" 200 -
frontend-1 exited with code 0
frontend-1 exited with code 137
order-1 exited with code 0
order-1 exited with code 137
catalog-1 exited with code 0
aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % docker-compose up
[+] Running 4/3
✓ Network spring24-lab2-mitalee18-aishwaryax_default C... 0.0s
✓ Container spring24-lab2-mitalee18-aishwaryax-catalog-1 Created 0.0s
✓ Container spring24-lab2-mitalee18-aishwaryax-order-1 Created 0.0s
✓ Container spring24-lab2-mitalee18-aishwaryax-frontend-1 Created 0.0s
Attaching to catalog-1, frontend-1, order-1
order-1 | The Order server is running.
catalog-1 | The Catalog server is running.
frontend-1 | Starting server on spring24-lab2-mitalee18-aishwaryax-frontend-1:56363
frontend-1 | 192.168.65.1 -- [03/Apr/2024 02:51:27] "GET /products/Tux HTTP/1.1" 200 -
order-1 | Order received for toy: Tux and for quantity: 1
order-1 | Order has been processed with OrderNumber 5
order-1 | Order processed for toy: Tux and for quantity: 1
frontend-1 | 192.168.65.1 -- [03/Apr/2024 02:51:40] "POST /orders HTTP/1.1" 200 -
```

```

● aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % curl http://localhost:56363/products/Tux
{"data": {"name": "Tux", "price": 2555.0, "quantity": 2942}}
● aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % curl -X POST -d '{"name": "Tux", "quantity": 1}' -H "Content-Type: application/json" http://localhost:56363/orders
{"data": {"order_number": 4}}
● aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % docker-compose down
[+] Running 4/4
  ✓ Container spring24-lab2-mitalee18-aishwaryax-frontend-1 Removed 10.2s
  ✓ Container spring24-lab2-mitalee18-aishwaryax-order-1 Removed 10.1s
  ✓ Container spring24-lab2-mitalee18-aishwaryax-catalog-1 Removed 10.2s
  ✓ Network spring24-lab2-mitalee18-aishwaryax_default R... 0.1s
● aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % curl http://localhost:56363/products/Tux
{"data": {"name": "Tux", "price": 2555.0, "quantity": 2941}}
● aishwaryasahoo@Aishwaryas-MacBook-Air spring24-lab2-mitalee18-aishwaryax % curl -X POST -d '{"name": "Tux", "quantity": 1}' -H "Content-Type: application/json" http://localhost:56363/orders
{"data": {"order_number": 5}}

```