

\*\*\* I used flask framework in this project.

Part 1 : on visual studio code and using postman to run all APIs :

- Simple Describe :

Catalog Server(python file )port number 5002 :

- It connected with txt file as a simple database , it has all catalog information's like cost , stock , title and the topic.

Order Server(python file ) port number 5003:

- It connected also with a txt file as a simple database it has on all order information's .

ForntEnd (python file ) port number 5000:

- In this file I tried to make it connect with order and catalog and get the information by connect with them .
- 

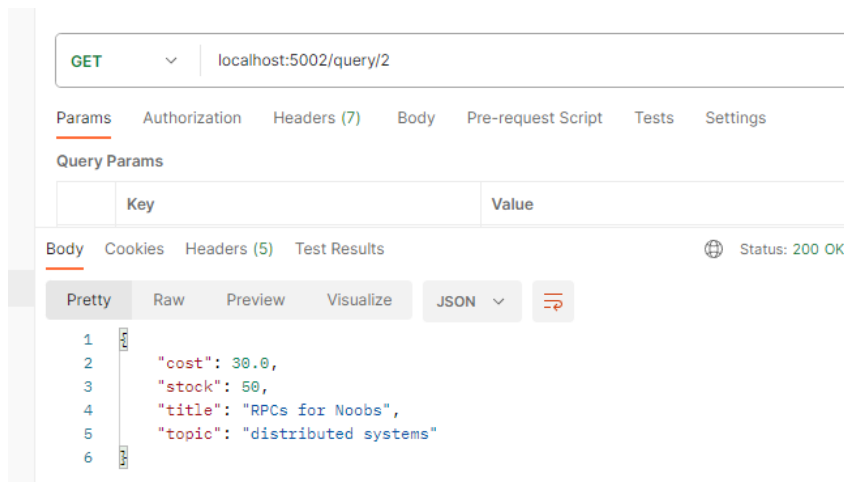
- **Result by using postman , I run each file in different PowerShell to make it run at the same time :**

• **Catalog :**

Windows PowerShell

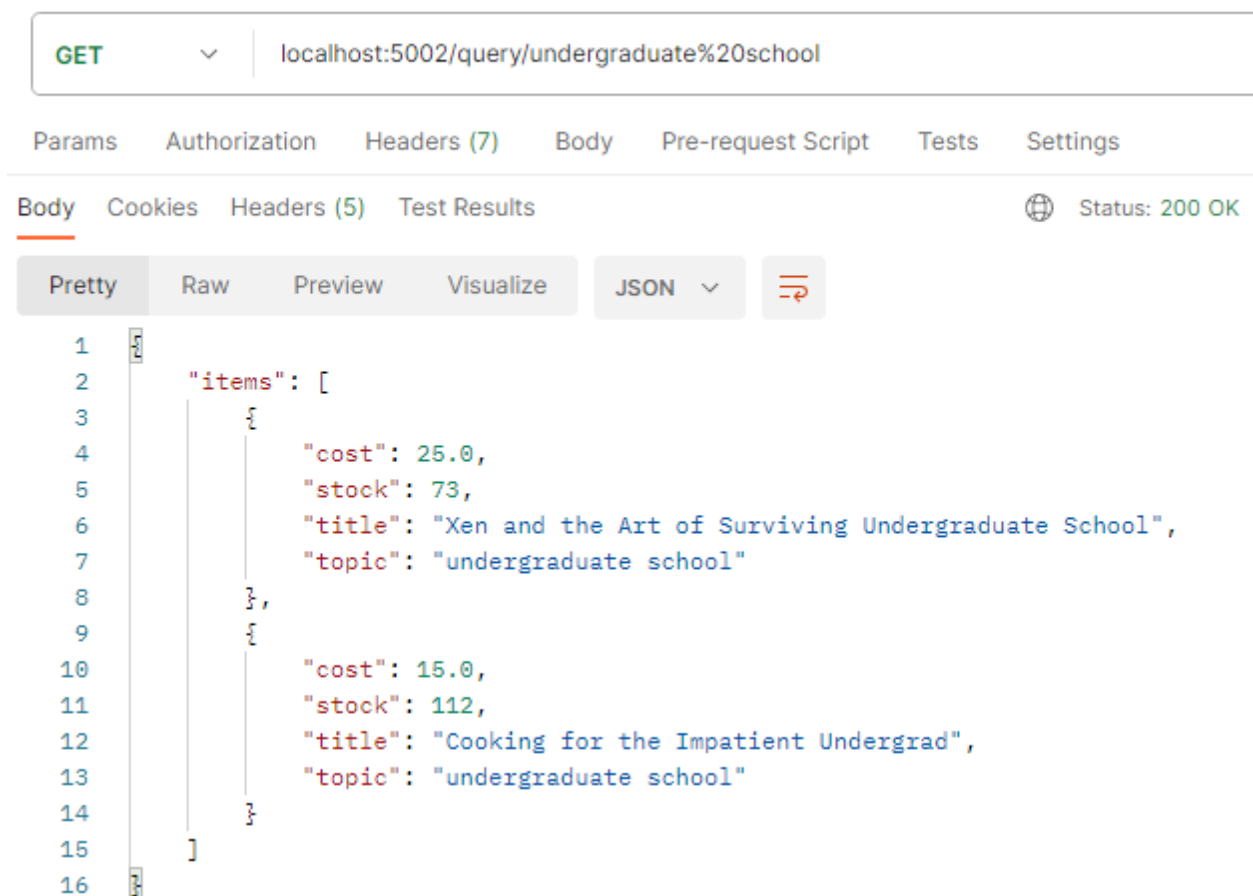
```
PS C:\Users\hp\Documents\GitHub\DOS-Lab1_DaliaDwaikat\dos> python catalogServer.py
{1: {'title': 'How to get a good grade in DOS in 40 minutes a day', 'stock': 86, 'cost': 20},
 2: {'title': 'RPCs for Noobs', 'stock': 50, 'cost': 30.0, 'topic': 'distributed systems'},
 3: {'title': 'the Art of Surviving Undergraduate School', 'stock': 73, 'cost': 25.0, 'topic': 'undergraduate school'},
 4: {'title': 'Cooking for the Impatient Undergrad', 'stock': 112, 'cost': 15.0, 'topic': 'undergraduate school'}}
* Serving Flask app 'catalogServer'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production server.
* Running on http://127.0.0.1:5002
Press CTRL+C to quit
127.0.0.1 - - [19/Nov/2023 20:21:15] "GET /query/2 HTTP/1.1" 200 -
```

1- Query for element 2, this the result on postman :



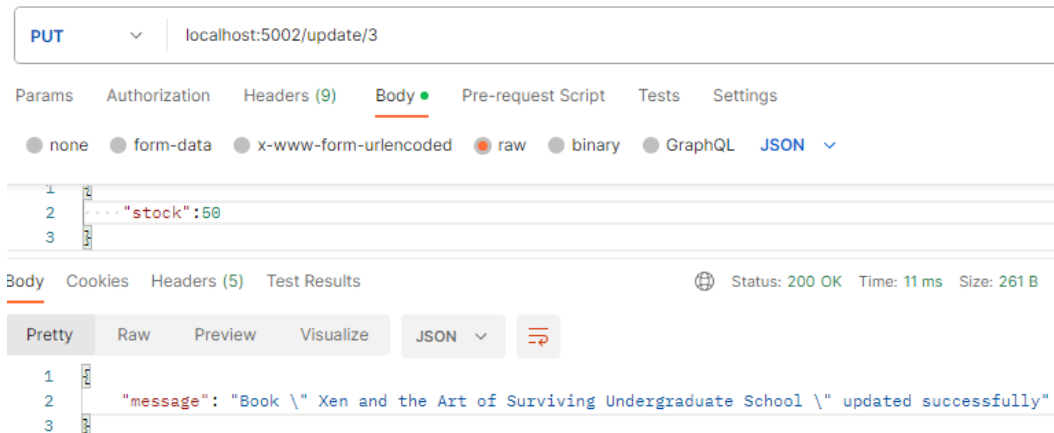
2- Query by topic for (undergraduate school):

HTTP Dos / catalog / query-topic



3- Update for item 3 , I want update number of stock to 50 (the result on postman and the txt file also is updated):

- Postman :



- Txt file :

Before :

```
catalog.txt
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,86,20.0,distrib
3 2,RPCs for Noobs,50,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,73,25.0,undergrad
5 4,Cooking for the Impatient Undergrad,112,15.0,undergraduate school
6 |
```

After:

```
catalog.txt
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,86,20.0,distrib
3 2,RPCs for Noobs,50,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,50,25.0,undergrad
5 4,Cooking for the Impatient Undergrad,112,15.0,undergraduate school
6 |
```

- Order :

```

Windows PowerShell
PS C:\Users\hp\Documents\GitHub\DOS-Lab1_DaliaDwaikat\dos> python orderServer.py
1: {'title': 'How to get a good grade in DOS in 40 minutes a day', 'sold': 3}, 2: {'
, 3: {'title': 'Xen and the Art of Surviving Undergraduate School', 'sold': 0}, 4: {'
Undergrad', 'sold': 6}}
ata geted succesfully
* Serving Flask app 'orderServer'
* Debug mode: off
ARNING: This is a development server. Do not use it in a production deployment. Use
* Running on http://127.0.0.1:5003
ress CTRL+C to quit

```

1- Purchase : (I want purchase book number 4 ), and the result will see on postman alson on the txt files catalog and order :

POST localhost:5003/purchase/4

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

body Cookies Headers (5) Test Results Status: 200 OK Time: 12 ms Siz

Pretty Raw Preview Visualize JSON

```

1
2 "message": "Book \" Cooking for the Impatient Undergrad \" purchased successfully"
3

```

Before :

```

catalog.txt
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,86,20.0,distrib
3 2,RPCs for Noobs,50,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,50,25.0,undergra
5 4,Cooking for the Impatient Undergrad,112,15.0,undergraduate school
6

orders.txt
1 id,title,sold
2 1,How to get a good grade in DOS in 40 minu
3 2,RPCs for Noobs,1
4 3,Xen and the Art of Surviving Undergraduat
5 4,Cooking for the Impatient Undergrad,6
6

```

After:

catalog.txt

```
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,86,20.0,distribu
3 2,RPCs for Noobs,50,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,50,25.0,undergrad
5 4,Cooking for the Impatient Undergrad,111,15.0,undergraduate school
6 |
```

orders.txt

```
1 id,title,sold
2 1,How to get a good grade in DOS in 40 minutes
3 2,RPCs for Noobs,1
4 3,Xen and the Art of Surviving Undergraduate Sc
5 4,Cooking for the Impatient Undergrad,7
6
```

- **Front** : (to run it correctly , I run each file on different PowerShell at the same time ):

Windows PowerShell

```
PS C:\Users\hp\Documents\GitHub\DOS-Lab1_DaliaDwaikat\dos> python frontend.py
* Serving Flask app 'frontend'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a produc
* Running on http://127.0.0.1:5000
Press CTRL+C to quit
```

- 1- Search (I want search about book number 3):  
the result on postman :

Database try

▼ Dos

▼ frontend

GET search

GET info

POST purchase

▼ catalog

GET query-item

GET query-topic

PUT update

GET hello

▼ order

POST New Request

GET New Request

> New Collection

GET localhost:5000/search/3

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

Query Params

Key	Value
Key	Value

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "cost": 25.0,
3   "stock": 50,
4   "title": "Xen and the Art of Surviving Undergraduate School",
5   "topic": "undergraduate school"
6 }
```

Status: 200 OK

2- Info : (I want to get info about element number 4 ):

Database try

▼ Dos

▼ frontend

GET search

GET info

POST purchase

▼ catalog

GET query-item

GET query-topic

PUT update

GET hello

▼ order

POST New Request

GET New Request

> New Collection

GET localhost:5000/info/4

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

Query Params

Key	Value	Description
Key	Value	Description

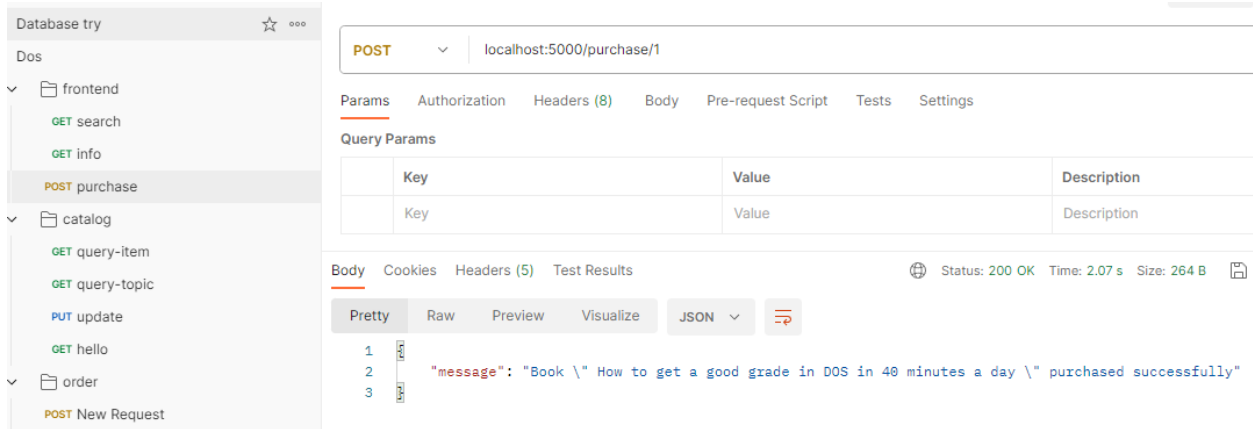
Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "cost": 15.0,
3   "stock": 111,
4   "title": "Cooking for the Impatient Undergrad",
5   "topic": "undergraduate school"
6 }
```

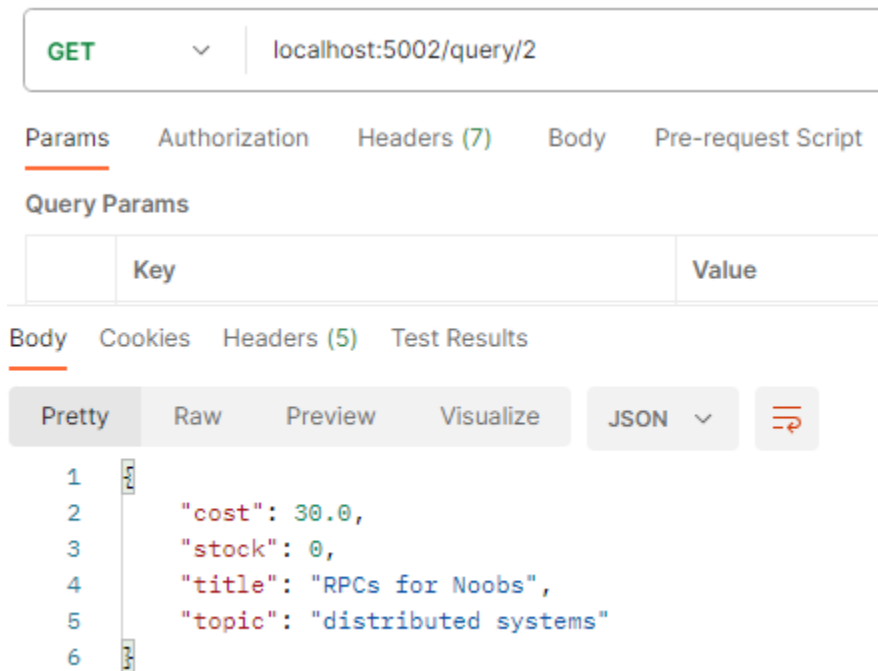
Status: 200 OK Time: 2.04 s

3- Purchase : I want purchase a book “How to get a good grade in DOS in 40 minutes a day” and this is number 1 :



- If I want purchase a book I out of stock this is what appeared as result in postman :

I update the information by using catalog server for item 2 and make the stock books is 0 , and I tried to purchase the book from frontend or from order server is the same result the give it :



The result :

HTTP Dos / frontend / purchase

POST localhost:5000/purchase/2

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

Query Params

	Key	Value
	Key	Value

















Body Cookies Headers (5) Test Results Status: 404 NOT FOUND

Pretty Raw Preview Visualize JSON
















```
1 {
2   "error": "Item is out of stock"
3 }
```

## Part 2 : running the server inside docker desktop :

I created 3 images as a separated webservices and each one is run in different port and server :

	<a href="#">dwaikat/flask-front-server</a>	latest	<a href="#">In use</a>	2 days ago	1.11 GB			
	<a href="#">dwaikat/flask-order-server</a>	latest	<a href="#">In use</a>	3 days ago	1.11 GB			
	<a href="#">daliadwaikat/dalia-dwaikatdockerdos</a>	tagname	<a href="#">In use</a>	3 days ago	1.11 GB			
	<a href="#">dwaikat/flask-catalog-server</a>	latest	<a href="#">In use</a>	3 days ago	1.11 GB			

I run it the three servers at the same time :

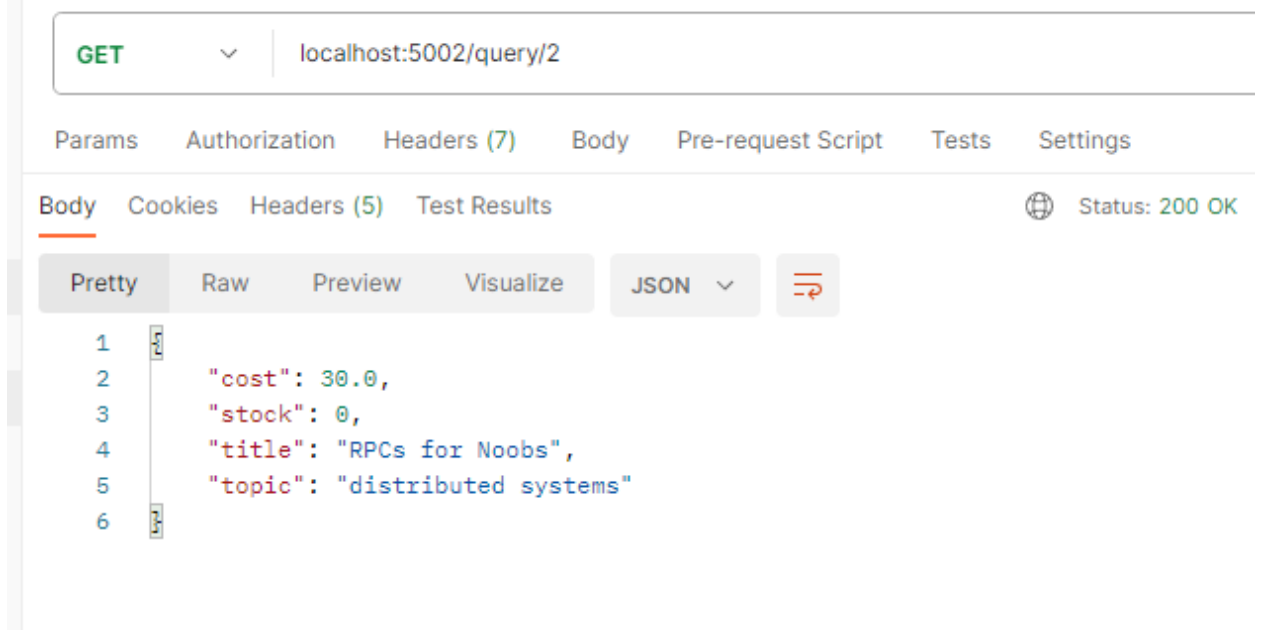
		<a href="#">jovial_euler</a>	<a href="#">dwaikat/flask-front</a>	Running	0.02%	<a href="#">5000:5000</a>	34 seconds ago			
		<a href="#">quizzical_w</a>	<a href="#">dwaikat/flask-orde</a>	Running	0.01%	<a href="#">5003:5003</a>	35 seconds ago			
		<a href="#">frosty_thom</a>	<a href="#">dwaikat/flask-catal</a>	Running	0.01%	<a href="#">5002:5002</a>	38 seconds ago			

### 1- Catalog server:

After I wrote the docker file then i created the image by using commands docker , I tried to run the program on it :



a. Query:for element 2



GET localhost:5002/query/2

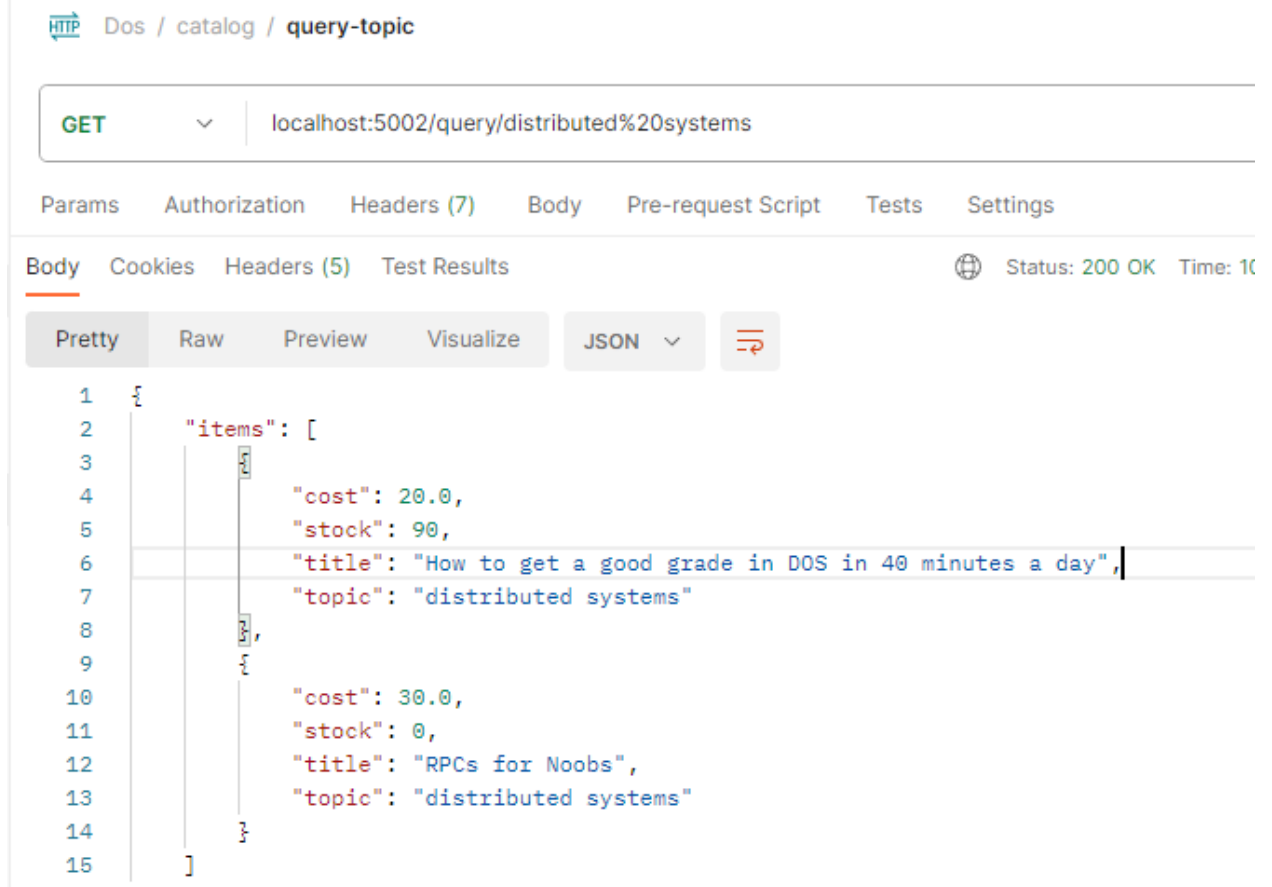
Params Authorization Headers (7) Body Pre-request Script Tests Settings

Body Cookies Headers (5) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON

```
1 {
2   "cost": 30.0,
3   "stock": 0,
4   "title": "RPCs for Noobs",
5   "topic": "distributed systems"
6 }
```

b. Query for distributed system :



HTTP Dos / catalog / query-topic

GET localhost:5002/query/distributed%20systems

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

Body Cookies Headers (5) Test Results Status: 200 OK Time: 10

Pretty Raw Preview Visualize JSON

```
1 {
2   "items": [
3     {
4       "cost": 20.0,
5       "stock": 90,
6       "title": "How to get a good grade in DOS in 40 minutes a day",
7       "topic": "distributed systems"
8     },
9     {
10      "cost": 30.0,
11      "stock": 0,
12      "title": "RPCs for Noobs",
13      "topic": "distributed systems"
14    }
15  ]
16 }
```

c. Update data (update data for element 3):

Before: inside file

Containers

Images

Volumes

Dev Environments **BETA**

Docker Scout

Learning center

Extensions

Add Extensions

STATUS  
Running (7 minutes ago)

Logs Inspect Bind mounts Exec **Files** Stats

Name	Note	Size	Last modified
.dockerenv		0 Bytes	3 days ago
app	MODIFIED		3 days ago
catalog	MODIFIED		2 days ago
__pycache__			4 days ago
catalog_requirements.txt		5 Bytes	3 days ago
catalog.txt	MODIFIED	300 Bytes	2 days ago

/app/catalog/catalog.txt

Plain Text

```
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,90,20.0,distributed systems
3 2,RPCs for Noobs,0,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,72,25.0,undergraduate school
5 4,Cooking for the Impatient Undergrad,108,15.0,undergraduate school
6
```

Engine running RAM 4.40 GB CPU 8.96% Signed in

After :

\_\_pycache\_\_

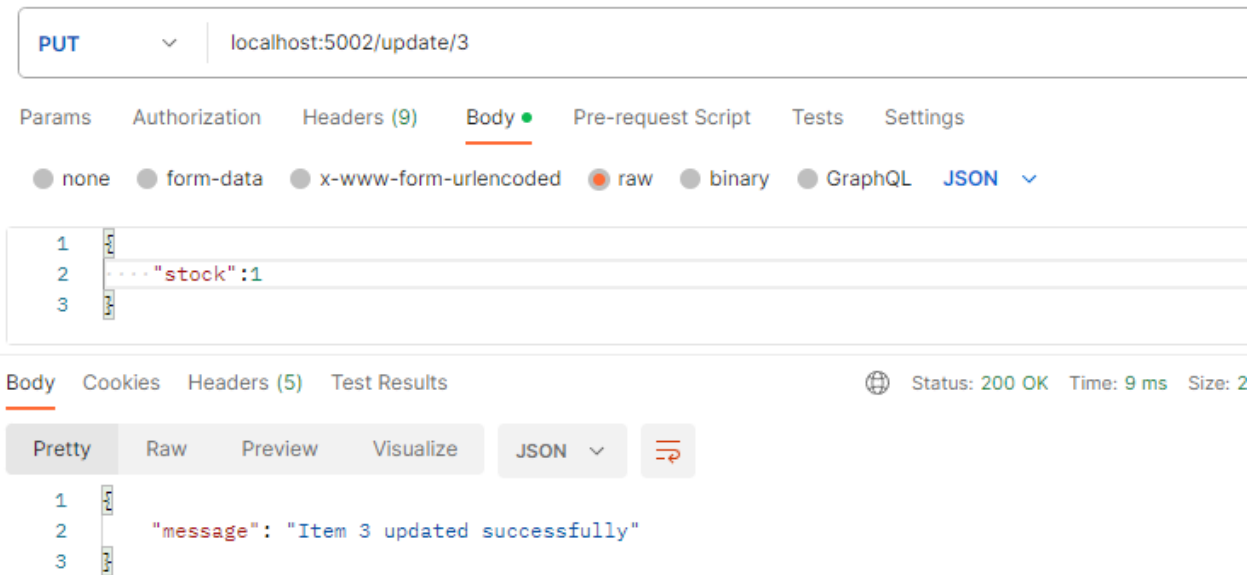
catalog\_requirements.txt 5 Bytes

catalog.txt MODIFIED 299 Bytes

/app/catalog/catalog.txt

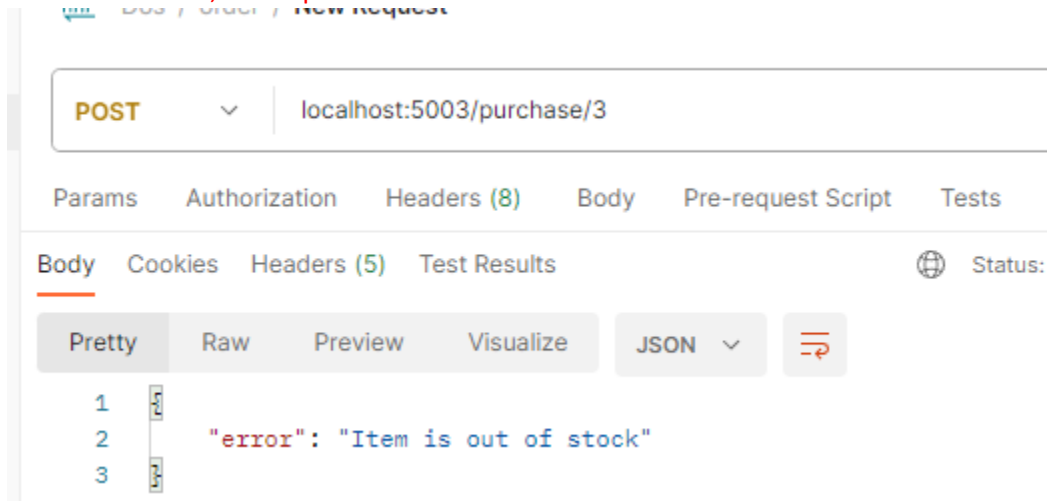
```
1 id,title,stock,cost,topic
2 1,How to get a good grade in DOS in 40 minutes a day,90,20.0,distributed systems
3 2,RPCs for Noobs,0,30.0,distributed systems
4 3,Xen and the Art of Surviving Undergraduate School,1,25.0,undergraduate school
5 4,Cooking for the Impatient Undergrad,108,15.0,undergraduate school
6
```

Postman :



## 2- Order Server :

- Purchase book 3 , I have purchased 2 second times :



## 3- Front Server :

- Info :

GET localhost:5000/info/4

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

Query Params

Key	Value
Key	Value

Body Cookies Headers (5) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON

```

1  {
2    "cost": 15.0,
3    "stock": 111,
4    "title": "Cooking for the Impatient Undergrad",
5    "topic": "undergraduate school"
6  }

```

- Search :

> Database try
 < Dos
 < frontend
 GET search
 GET info
 POST purchase
 < catalog
 GET query-item
 GET query-topic
 PUT update
 GET hello
 < order
 POST New Request
 GET New Request
 > New Collection

GET localhost:5000/search/3
 Params Authorization Headers (7) Body Pre-request Script Tests Settings
 Query Params
 

Key	Value
Key	Value

 Body Cookies Headers (5) Test Results Status: 200 OK
 Pretty Raw Preview Visualize JSON
 

```

1  {
2    "cost": 25.0,
3    "stock": 50,
4    "title": "Xen and the Art of Surviving Undergraduate School",
5    "topic": "undergraduate school"
6  }

```

- Purchase :

POST localhost:5000/purchase/1

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

Query Params

Key	Value	Description
Key	Value	Description

Body Cookies Headers (5) Test Results Status: 200 OK Time: 2.07 s Size: 264 B

Pretty Raw Preview Visualize JSON

```

1  {
2    "message": "Book \" How to get a good grade in DOS in 40 minutes a day \" purchased successfully"
3  }

```

Note there is a small problem inside front , It stopped on works suddenly but before that It was work correctly , that is only inside the docker just , I will fix it until discussed .

I pushed my images to docker hub , it is public repository : [daliadwaikat/dalia-dwaikatdockerdos](https://hub.docker.com/r/daliadwaikat/dalia-dwaikatdockerdos)