

**An- Najah National
University**

Faculty of Engineering



Department Name:
Computer Engineering

Dos Project_Part1

Instructor Name: Dr. Samer arande
Student Name: Jawna Amjad Atwe – Rahaf Ali Nawwas
Academic Year: 2022/2023

WE HAVE 3 SERVER:

1. Front-end server → my first linux in virtual box
 2. catalog server → my windows
 3. order server → my second linux in virtual box
-

Then we want start with Front-end →

In this server we added 3 classes the first class book ,it contain all element which for any book , and second class front-end service :

```
public class front_end_service {  
    private final RestTemplate restTemplate = new RestTemplate();  
    public Book Search(String topic) {  
        return restTemplate.getForObject("http://localhost:8080/BookTopic/"+topic, Book.class);  
    }  
  
    public Book info(String item) {  
        return restTemplate.getForObject("http://localhost:8080/BookItem/"+item, Book.class);  
    }  
  
    public Book purchase(String item) {  
        restTemplate.put("Book/purchase/"+item, Book.class);  
        return null;  
    }  
}
```

and third class is front-end controller which contain info, search and purchase.

Then catalog →

My data base in widows

It contain interface book_JPA , book class , book service and catalog service controller

Catalog service controller it include get by name ,add book ,update book info and get by topic.

Then order→

It contain book class , purchase service and purchase controller
purchase controller it include purchase item

implementation

1. we want get information for book which it name RPCs for Noobs

The screenshot shows a REST client interface with a GET request to `http://localhost:8080/info/RPCs%20for%20Noobs`. The response is a JSON object displayed in the 'Body' tab, formatted as follows:

```
1 {
2   "price": 20,
3   "stock": 5,
4   "topic": "distributed systems",
5   "bookName": "RPCs for Noobs"
6 }
```

2. we want purchase a book which it name RPCs for Noobs

The screenshot shows a REST client interface with a PUT request to `http://localhost:8080/purchase/RPCs%20for%20Noobs`. The 'Params' tab is active, showing a single query parameter 'Key' with the value 'Value'. The 'Body' tab is also visible, showing a single line of text: 'Done and the number of books become 4'.

KEY	VALUE
Key	Value

Body: Done and the number of books become 4

3.now we want get all books which topic is distributed system

The screenshot shows a REST client interface with a GET request to `http://localhost:8080/search/distributed systems`. The 'Params' tab is active, showing a single query parameter 'Key' with the value 'Value'. The 'Body' tab is also visible, showing a JSON array of two book objects.

KEY	VALUE
Key	Value

Body:

```
[{"price": 10, "stock": 6, "topic": "distributed systems", "bookName": "How to get a good grade in DOS in 40 minutes a day"}, {"price": 20, "stock": 3, "topic": "distributed systems", "bookName": "RPCs for Noobs"}]
```

4.now we want get all books which topic is undergraduate

GET <http://localhost:8080/search/undergraduate school>

Params Authorization Headers (6) 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   {  
3     "price": 110,  
4     "stock": 5,  
5     "topic": "undergraduate school",  
6     "bookName": "Xen and the Art of Surviving Undergraduate School"  
7   },  
8   {  
9     "price": 155,  
10    "stock": 1,  
11    "topic": "undergraduate school",  
12    "bookName": "Cooking for the Impatient Undergrad"  
13  }  
}
```

5. We Purchased all the books with their name RPCs for Noobs

<http://localhost:8080/purchase/RPCs%20for%20Noobs>

PUT <http://localhost:8080/purchase/RPCs%20for%20Noobs>

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 Text

```
1 Done and the number of books become2
```

6.now we want get info about this book

GET http://localhost:8080/info/RPCs%20for%20Noobs

Params Authorization Headers (6) 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   "price": 20,
3   "stock": 0,
4   "topic": "distributed systems",
5   "bookName": "RPCs for Noobs"
6 }
```

We notice that the stock is reduced to 0

7.now we try purchase another book but it stock is zero

PUT http://localhost:8080/purchase/RPCs%20for%20Noobs

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 Text

```
1 Does not exist
```

Thank you☺