9/25/2021

Online Bookstore

Prepared By: H. Mahesh

Table of Contents

1. Project Description

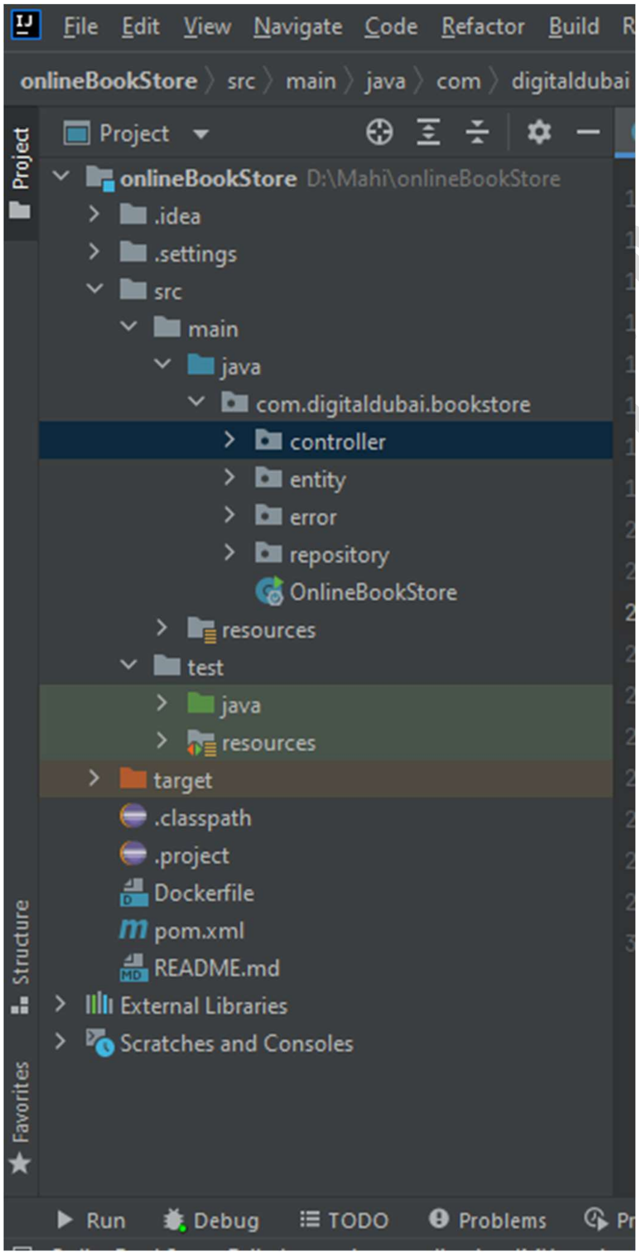
REST API for an online bookstore, where the user can perform the following operations CRUD operations on Books and Checkout operation for single or multiple books which will return the total payable amount.

1.1 Assumptions:

For comic book type on promotion code 0% discount and on Fiction books 20% discount hard coded in code for this assignment This can be configurable in future based on requirement.

2. Project Directory

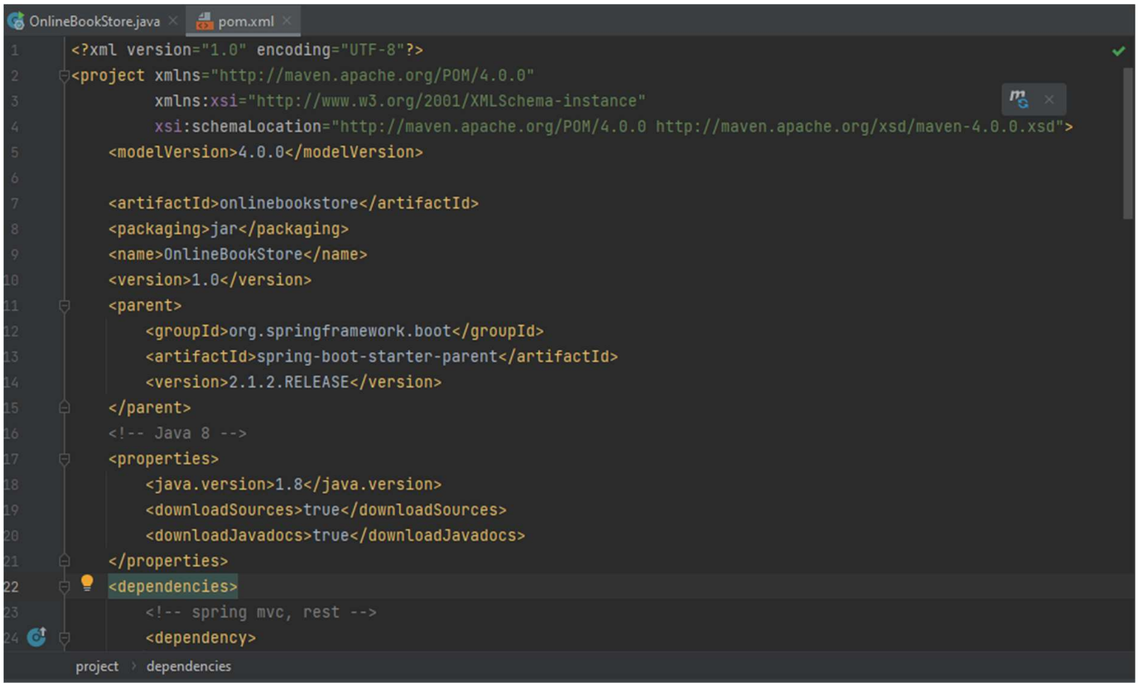
2.1 A standard Maven project structure.


File Edit View Navigate Code Refactor Build R
onlineBookStore 〉 src 〉 main 〉 java 〉 com 〉 digitaldubai
Project
Structure
Favorites
Run
Debug
TODO
Problems
Pr

```
Project
onlineBookStore D:\Mahi\onlineBookStore
  .idea
  .settings
  src
    main
      java
        com.digitaldubai.bookstore
          controller
          entity
          error
          repository
          OnlineBookStore
      resources
    test
      java
      resources
  target
  .classpath
  .project
  Dockerfile
  pom.xml
  README.md
External Libraries
Scratches and Consoles
```

3. Maven Structure

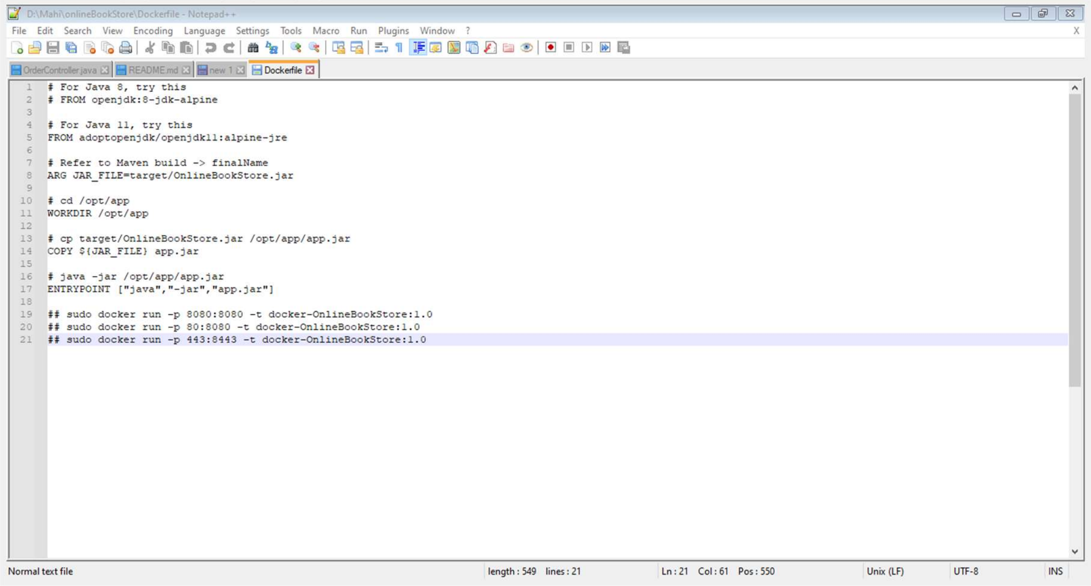
3.1 Project dependencies. Nothing special here, just some Spring Boot dependencies.



```xml
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>

    <artifactId>onlinebookstore</artifactId>
    <packaging>jar</packaging>
    <name>OnlineBookStore</name>
    <version>1.0</version>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.1.2.RELEASE</version>
    </parent>
    <!-- Java 8 -->
    <properties>
        <java.version>1.8</java.version>
        <downloadSources>true</downloadSources>
        <downloadJavadocs>true</downloadJavadocs>
    </properties>
    <dependencies>
        <!-- spring mvc, rest -->
        <dependency>
```

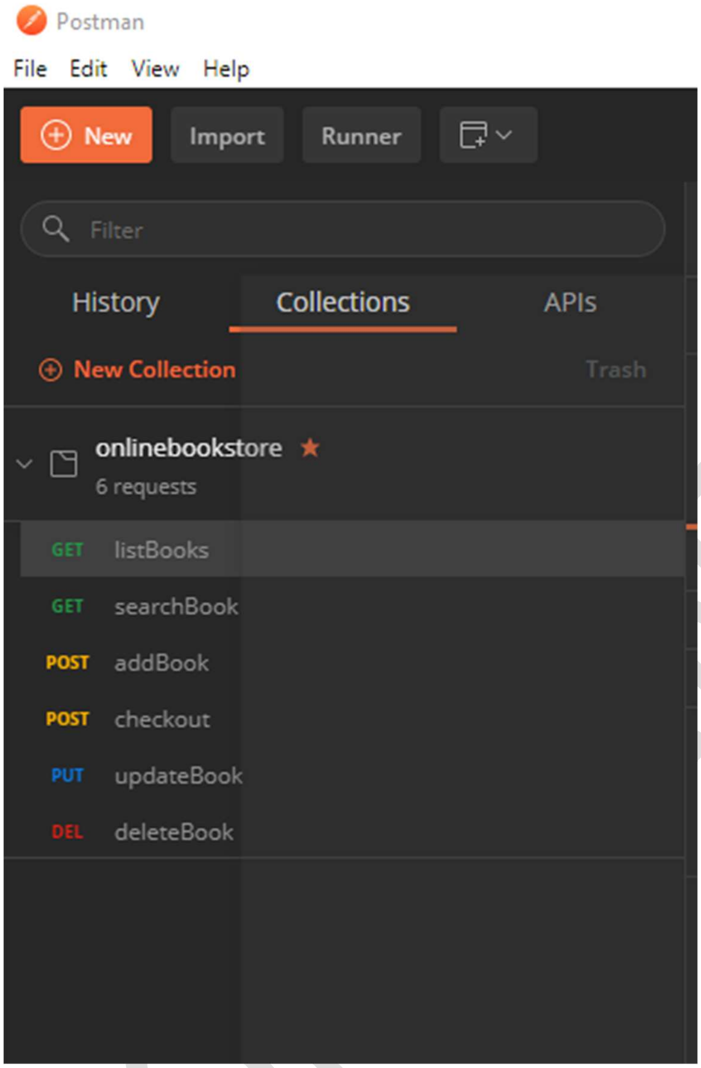
4. Docker File

A Dockerfile is a text file, contains all the commands to assemble the docker image.


D:\Mahi\onlineBookStore\Dockerfile - Notepad++
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
OrderController.java
README.md
new 1
Dockerfile
Normal text file
length : 549 lines : 21
Ln : 21 Col : 61 Pos : 550
Unix (LF)
UTF-8
INS

```dockerfile
# For Java 8, try this
# FROM openjdk:8-jdk-alpine

# For Java 11, try this
FROM adoptopenjdk/openjdk11:alpine-jre

# Refer to Maven build -> finalName
ARG JAR_FILE=target/OnlineBookStore.jar

# cd /opt/app
WORKDIR /opt/app

# cp target/OnlineBookStore.jar /opt/app/app.jar
COPY ${JAR_FILE} app.jar

# java -jar /opt/app/app.jar
ENTRYPOINT ["java","-jar","app.jar"]

## sudo docker run -p 8080:8080 -t docker-OnlineBookStore:1.0
## sudo docker run -p 80:8080 -t docker-OnlineBookStore:1.0
## sudo docker run -p 443:8443 -t docker-OnlineBookStore:1.0
```

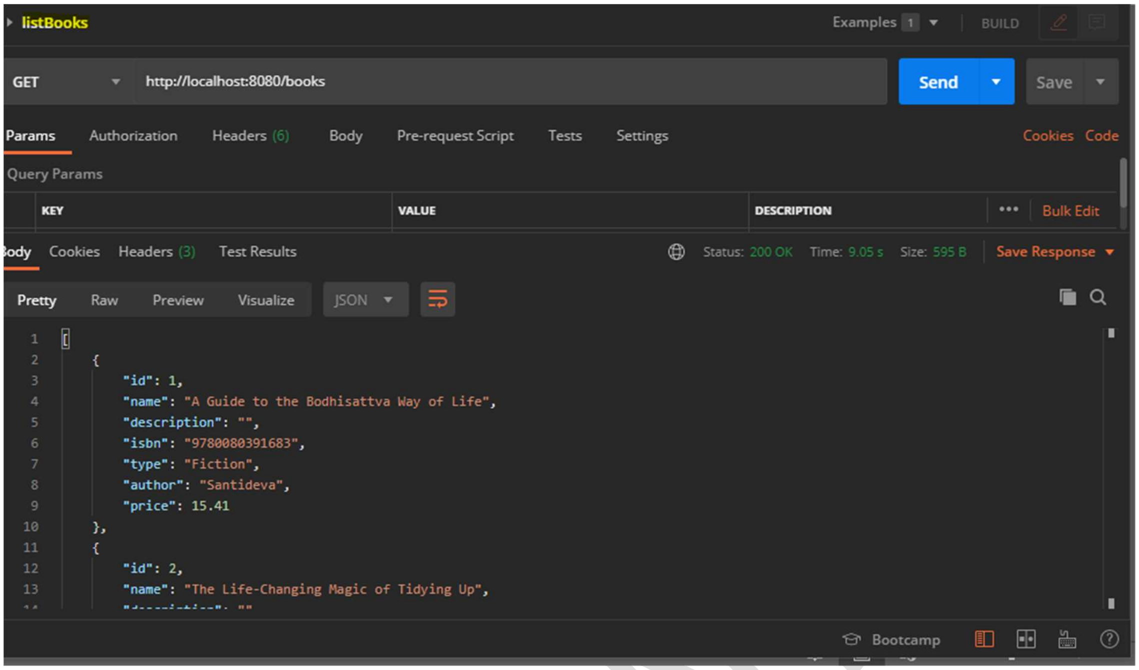
5. Postman CRUD Operation



onlinebookstore ★
6 requests

GET listBooks
GET searchBook
POST addBook
POST checkout
PUT updateBook
DEL deleteBook

5.1 LIST all Books (GET Operation)



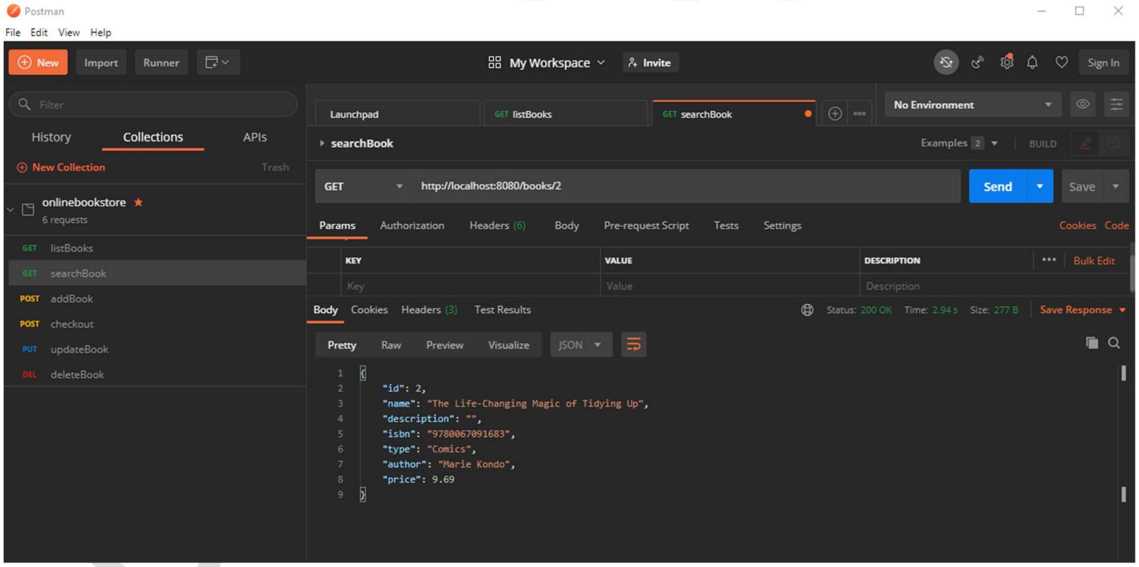
GET http://localhost:8080/books

Send

Status: 200 OK Time: 9.05 s Size: 595 B

```
[
    {
        "id": 1,
        "name": "A Guide to the Bodhisattva Way of Life",
        "description": "",
        "isbn": "9780080391683",
        "type": "Fiction",
        "author": "Santideva",
        "price": 15.41
    },
    {
        "id": 2,
        "name": "The Life-Changing Magic of Tidying Up",
        "description": ""
```

5.2 Search Book (GET Operation)



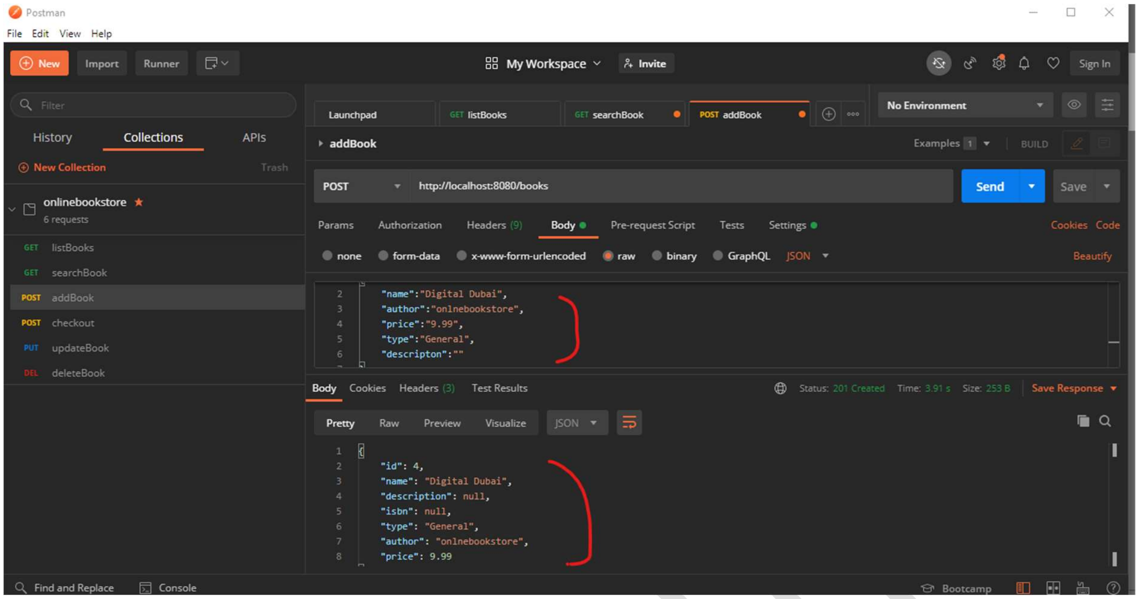
GET http://localhost:8080/books/2

Send

Status: 200 OK Time: 2.94 s Size: 277 B

```
{
    "id": 2,
    "name": "The Life-Changing Magic of Tidying Up",
    "description": "",
    "isbn": "9780067091683",
    "type": "Comics",
    "author": "Marie Kondo",
    "price": 9.69
}
```

5.3 Add Book (POST Operation)



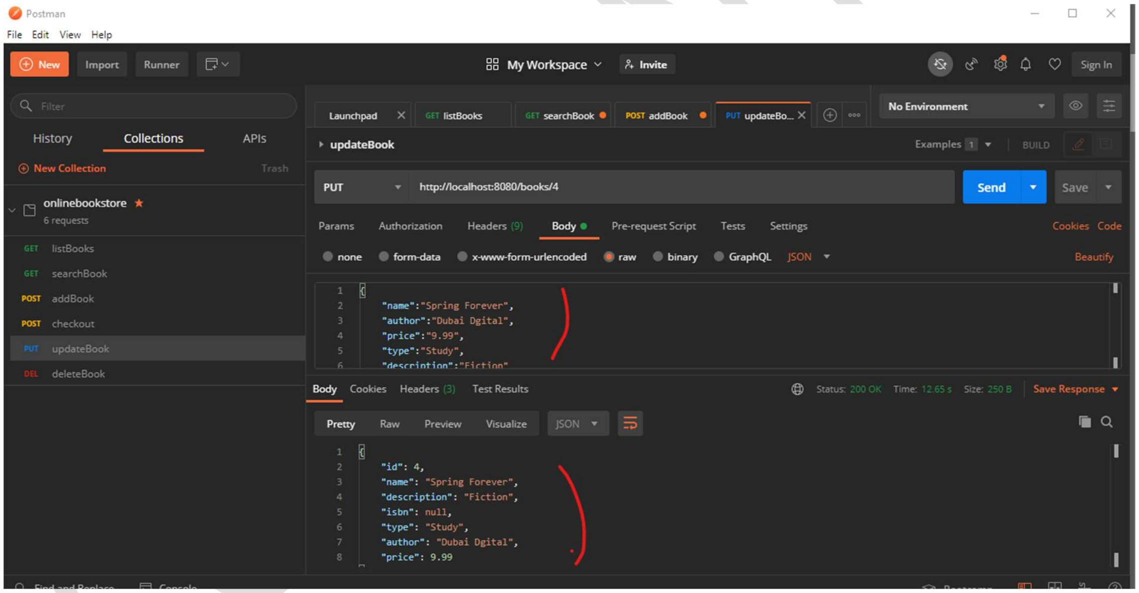
POST http://localhost:8080/books

```
    "name":"Digital Dubai",
    "author":"onlnebookstore",
    "price":"9.99",
    "type":"General",
    "descripton":""
```

Status: 201 Created Time: 3.91 s Size: 253 B

```
{
    "id": 4,
    "name": "Digital Dubai",
    "description": null,
    "isbn": null,
    "type": "General",
    "author": "onlnebookstore",
    "price": 9.99
```

5.4 Update Book (POST Operation)



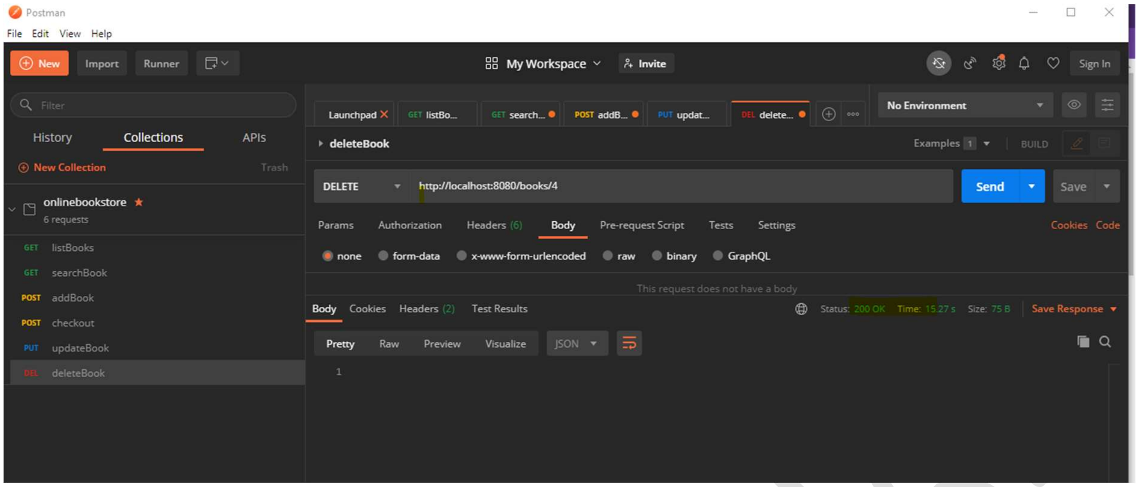
PUT http://localhost:8080/books/4

```
{
    "name":"Spring Forever",
    "author":"Dubai Dgital",
    "price":"9.99",
    "type":"Study",
    "description":"Fiction"
```

Status: 200 OK Time: 12.65 s Size: 250 B

```
{
    "id": 4,
    "name": "Spring Forever",
    "description": "Fiction",
    "isbn": null,
    "type": "Study",
    "author": "Dubai Dgital",
    "price": 9.99
```

5.5 Delete Book (DELETE Operation)



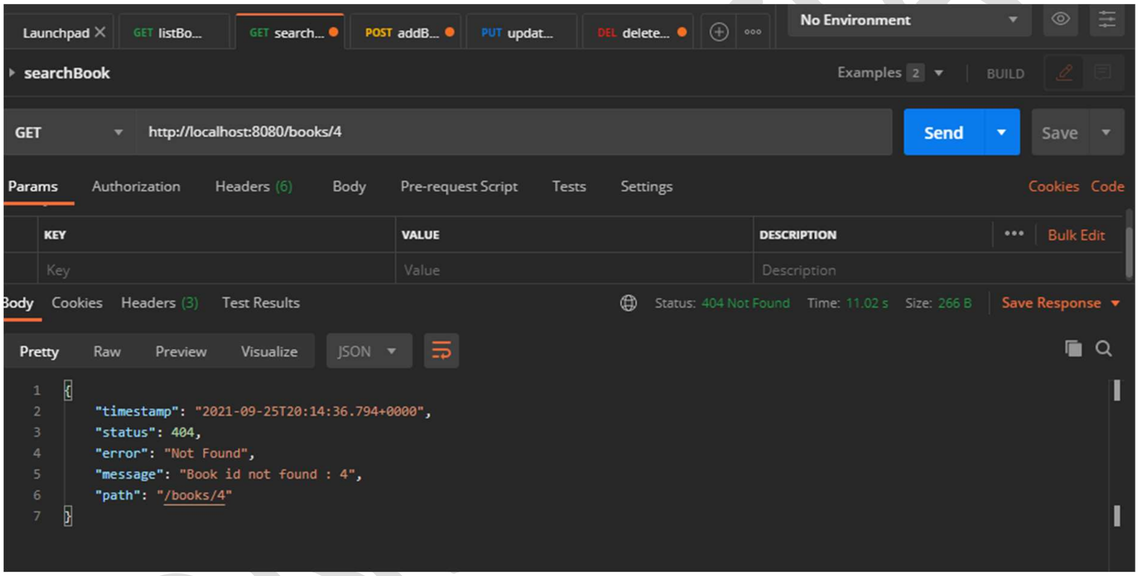
deleteBook

DELETE http://localhost:8080/books/4

Body: ● none ○ form-data ○ x-www-form-urlencoded ○ raw ○ binary ○ GraphQL

This request does not have a body

Body Cookies Headers (2) Test Results — Status: 200 OK Time: 15.27 s Size: 75 B



searchBook

GET http://localhost:8080/books/4

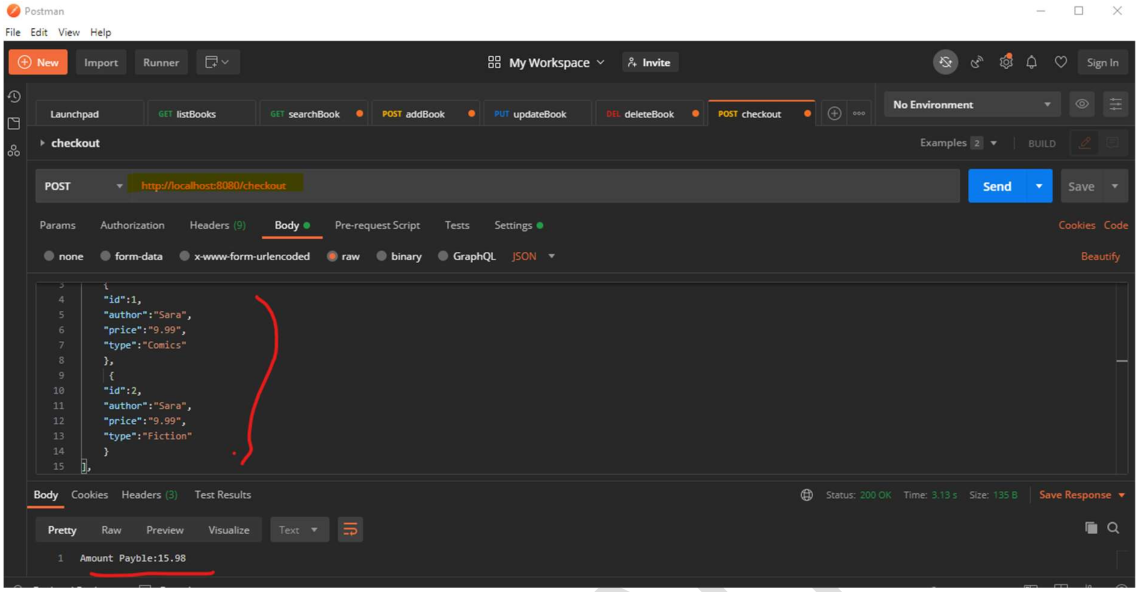
KEY	VALUE	DESCRIPTION
Key	Value	Description

Body — Status: 404 Not Found Time: 11.02 s Size: 266 B

```
{
    "timestamp": "2021-09-25T20:14:36.794+0000",
    "status": 404,
    "error": "Not Found",
    "message": "Book id not found : 4",
    "path": "/books/4"
}
```

5.6. Check out Books (POST Operation)

a) With Discount



POST http://localhost:8080/checkout

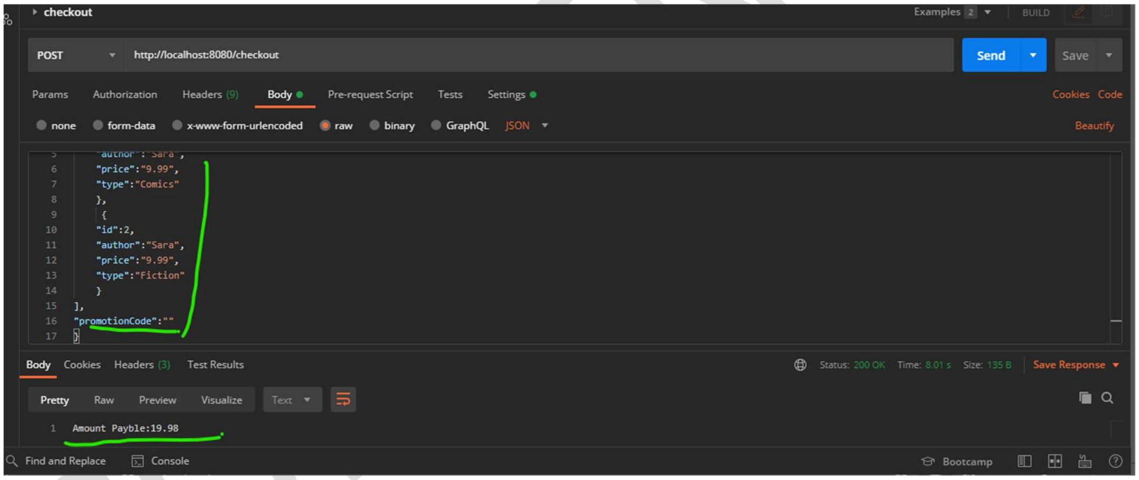
Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
    {
    "id":1,
    "author":"Sara",
    "price":"9.99",
    "type":"Comics"
    },
     {
    "id":2,
    "author":"Sara",
    "price":"9.99",
    "type":"Fiction"
    }
],
```

Status: 200 OK Time: 3.13 s Size: 135 B

```
Amount Payble:15.98
```

b) Without Discount



POST http://localhost:8080/checkout

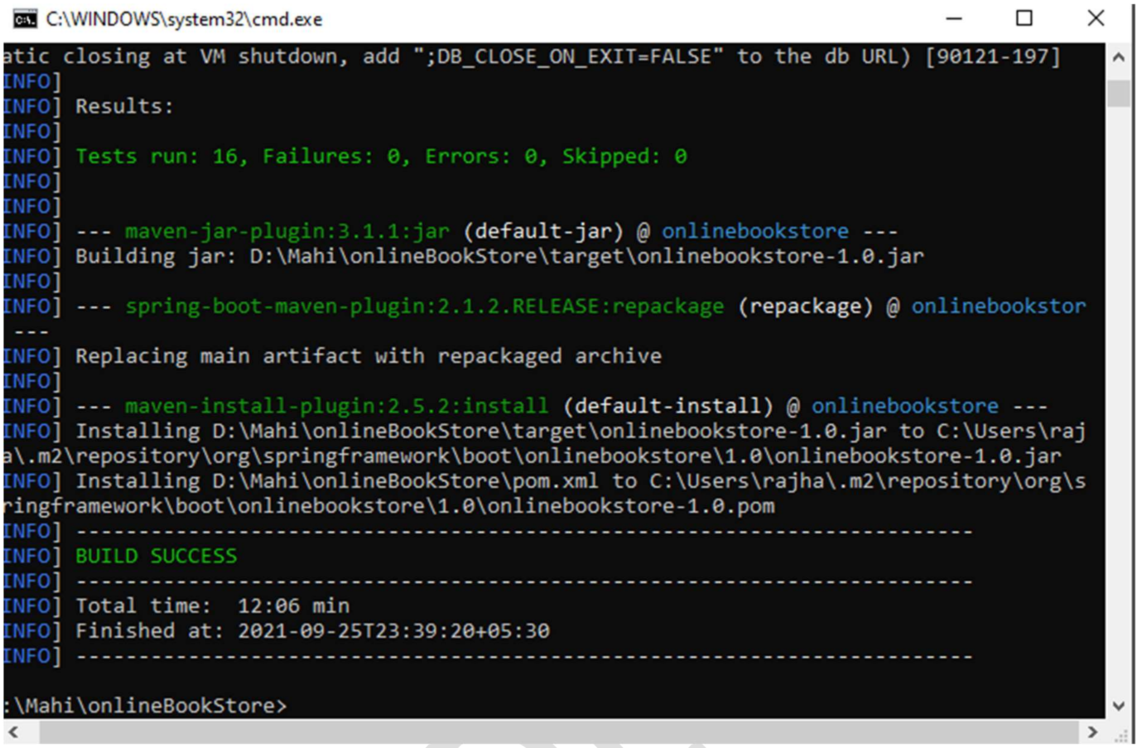
Body: ○ none ○ form-data ○ x-www-form-urlencoded ● raw ○ binary ○ GraphQL JSON

```
    "author":"Sara",
    "price":"9.99",
    "type":"Comics"
    },
     {
    "id":2,
    "author":"Sara",
    "price":"9.99",
    "type":"Fiction"
    }
],
"promotionCode":""
}
```

Status: 200 OK Time: 8.01 s Size: 135 B

```
Amount Payble:19.98
```

6. Junit Test cases



```
atic closing at VM shutdown, add ";DB_CLOSE_ON_EXIT=FALSE" to the db URL) [90121-197]
INFO]
INFO] Results:
INFO]
INFO] Tests run: 16, Failures: 0, Errors: 0, Skipped: 0
INFO]
INFO]
INFO] --- maven-jar-plugin:3.1.1:jar (default-jar) @ onlinebookstore ---
INFO] Building jar: D:\Mahi\onlineBookStore\target\onlinebookstore-1.0.jar
INFO]
INFO] --- spring-boot-maven-plugin:2.1.2.RELEASE:repackage (repackage) @ onlinebookstor
 ---
INFO] Replacing main artifact with repackaged archive
INFO]
INFO] --- maven-install-plugin:2.5.2:install (default-install) @ onlinebookstore ---
INFO] Installing D:\Mahi\onlineBookStore\target\onlinebookstore-1.0.jar to C:\Users\raj
a\.m2\repository\org\springframework\boot\onlinebookstore\1.0\onlinebookstore-1.0.jar
INFO] Installing D:\Mahi\onlineBookStore\pom.xml to C:\Users\rajha\.m2\repository\org\s
ringframework\boot\onlinebookstore\1.0\onlinebookstore-1.0.pom
INFO] ------------------------------------------------------------------------
INFO] BUILD SUCCESS
INFO] ------------------------------------------------------------------------
INFO] Total time:  12:06 min
INFO] Finished at: 2021-09-25T23:39:20+05:30
INFO] ------------------------------------------------------------------------

:\Mahi\onlineBookStore>
```