AmazingBooks-H2

- **5 Microservices**
  1. **API Gateway (Resource Server)- Port 8080**
  2. **Bookms (Book Microservice) - Port 8082**
  3. **Issuems(Issue Microservice) - Port 8083**
  4. **Eureka Server (Service Registry) - Port 8761**
  5. **OAuth Server (Authentication) - Port 9000**

- **H2 In-Memory Database is used**
- **Dockerized all the microservices**
- **Metrics are gathered using prometheus & Grafana**

1. **OAuth AuthorizationServer- provides the security**
   - One client application registered(in-memory)
   - Add customers from customerdb(user_details) as in-memory user
2. **API Gateway (Resource Server)- For cross cutting concerns**
   - Through the API Gateway only we are accessing bookms and issuems microservices
3. **Book Microservice - (Bookms)**

   Which stores the Bookdetails of a library

   APIs used to fetch/post/delete/update

   Get Methods
   - **http://lcoalhost:8080/bookms/books - Fetch all the books**
   - **http://lcoalhost:8080/bookms/books/{isbn} - Fetch book by isbn**
   - **http://lcoalhost:8080/bookms/books/availbleBooks - Fetch the available books by checking total copies against issued copies**

   Post Methods
   - **http://lcoalhost:8080/bookms/books - Add a new Book**

   Put Method
   - **http://lcoalhost:8080/bookms/books/{isbn} - edit the book by isbn**

   Delete Method
   - **http://lcoalhost:8080/bookms/books/{isbn} - Delete book by isbn**
4. **Issue Microservice - (Issuems)**

Which stores the issued book details of a library

APIs used to fetch/post/delete/update

Get Methods

- http://lcoalhost:8080/issuems/issue-books **- Fetch all the issued details of books**

- http://lcoalhost:8080/issuems/books/{isbn} **- Fetch issued details of a particular book**

- http://lcoalhost:8080/issuems/issue-books/availbleBooks **- Fetch all the available books by calling bookms**

- http://lcoalhost:8080/issuems/issue-books/customer/{customerid} **- To fetch the details of the books issued to the customer**

- http://lcoalhost:8080/issuems/issue-books/isbn/{isbn} **- To fetch the details of particular issued book**

Post Methods

- http://lcoalhost:8080/issuems/issue-books **- storing Issued detail**

Delete Method

- http://lcoalhost:8080/issuems/issue-books/{id} **- Delete the issue details when cancelling/returning the book**

Docker-Compose file

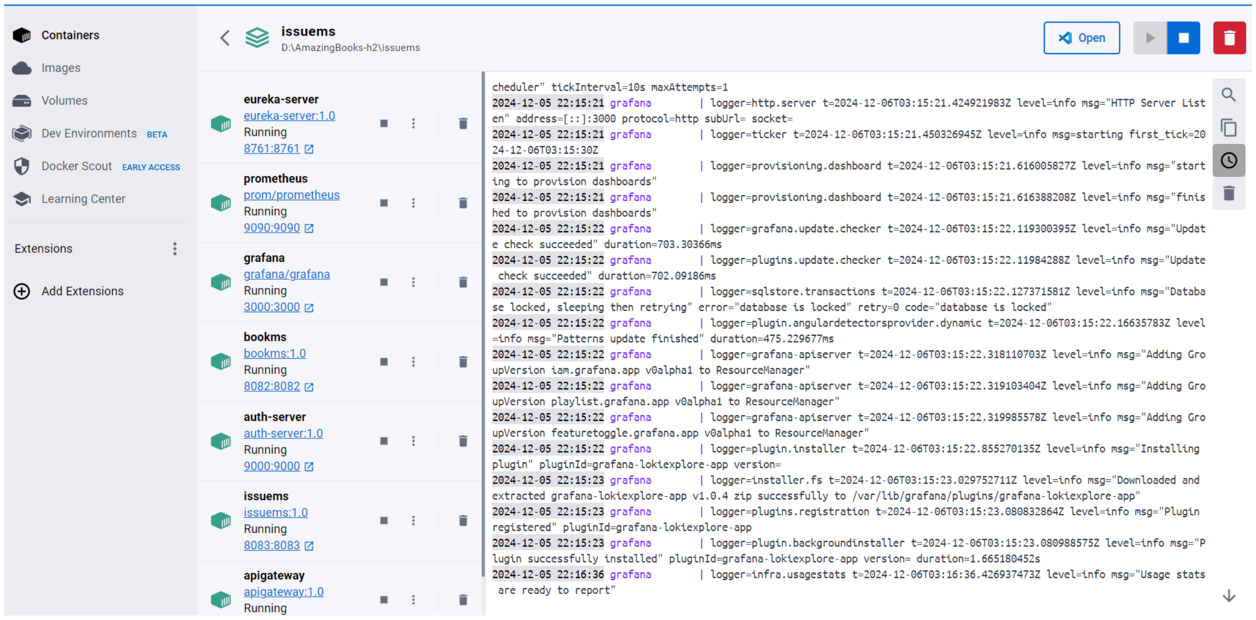
```
version: '3.7'
services:
 eureka-server:
   image: eureka-server:1.0
   container_name: eureka-server
   ports:
     - "8761:8761"
   networks:
     - eureka-network
 bookms:
   container_name: bookms
```

```yaml
  image: bookms:1.0
  ports:
    - "8082:8082"
  networks:
    - eureka-network
  depends_on:
    - eureka-server
issuems:
  image: issuems:1.0
  container_name: issuems
  ports:
    - "8083:8083"
  networks:
    - eureka-network
  depends_on:
    - eureka-server
    - bookms
apigateway:
  image: apigateway:1.0
  container_name: apigateway
  ports:
    - "8080:8080"
  networks:
    - eureka-network
  depends_on:
    - eureka-server
    - bookms
    - issuems
    - auth-server
auth-server:
  image: auth-server:1.0
  container_name: auth-server
  ports:
     - "9000:9000"
  networks:
    - eureka-network
prometheus:
  container_name: prometheus
  image: prom/prometheus
  ports:
    - 9090:9090
  volumes:
    - ./prometheus.yml:/etc/prometheus/prometheus.yml
  command: [
         "--config.file=/etc/prometheus/prometheus.yml"

          ]
  networks:
    - eureka-network
```

```yaml
 grafana:
   container_name: grafana
   image: grafana/grafana
   ports:
     - 3000:3000
   networks:
     - eureka-network
networks:
 eureka-network:
```

1. Create a Docker Image of all the microservices
   - Create a jar file in a target folder
     mvnw clean package -Dmaven.test.skip
   - use a docker file inside of microservice
     docker image build --tag <imagename>:<version> .

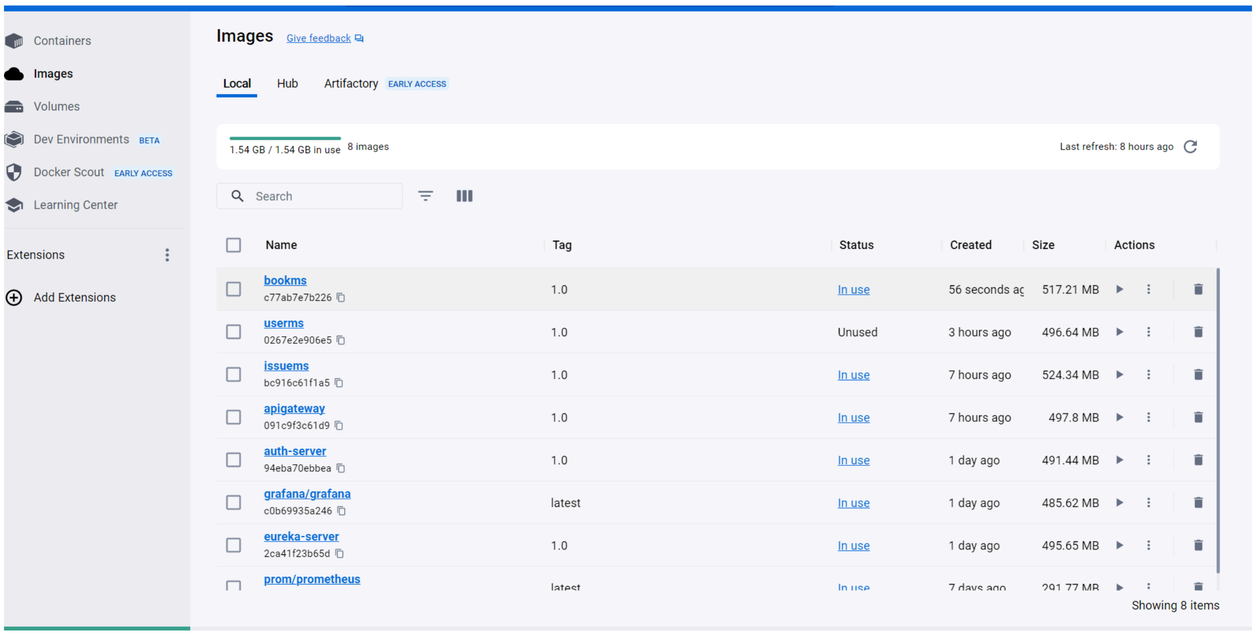
2. Run the Docker compose file to create a containers
   - Docker-compose up -d



issuems
D:\AmazingBooks-h2\issuems

```
eureka-server
eureka-server:1.0
Running
8761:8761

prometheus
prom/prometheus
Running
9090:9090

grafana
grafana/grafana
Running
3000:3000

bookms
bookms:1.0
Running
8082:8082

auth-server
auth-server:1.0
Running
9000:9000

issuems
issuems:1.0
Running
8083:8083

apigateway
apigateway:1.0
Running
```

```
cheduler" tickInterval=10s maxAttempts=1
2024-12-05 22:15:21 grafana       | logger=http.server t=2024-12-06T03:15:21.424921983Z level=info msg="HTTP Server List
en" address=[::]:3000 protocol=http subUrl= socket=
2024-12-05 22:15:21 grafana       | logger=ticker t=2024-12-06T03:15:21.450326945Z level=info msg=starting first_tick=20
24-12-06T03:15:30Z
2024-12-05 22:15:21 grafana       | logger=provisioning.dashboard t=2024-12-06T03:15:21.616005827Z level=info msg="start
ing to provision dashboards"
2024-12-05 22:15:21 grafana       | logger=provisioning.dashboard t=2024-12-06T03:15:21.616388208Z level=info msg="finis
hed to provision dashboards"
2024-12-05 22:15:22 grafana       | logger=grafana.update.checker t=2024-12-06T03:15:22.119300395Z level=info msg="Updat
e check succeeded" duration=703.30366ms
2024-12-05 22:15:22 grafana       | logger=plugins.update.checker t=2024-12-06T03:15:22.11984288Z level=info msg="Update
 check succeeded" duration=702.09186ms
2024-12-05 22:15:22 grafana       | logger=sqlstore.transactions t=2024-12-06T03:15:22.127371581Z level=info msg="Databa
se locked, sleeping then retrying" error="database is locked" retry=0 code="database is locked"
2024-12-05 22:15:22 grafana       | logger=plugin.angulardetectorsprovider.dynamic t=2024-12-06T03:15:22.16635783Z level
=info msg="Patterns update finished" duration=475.229677ms
2024-12-05 22:15:22 grafana       | logger=grafana-apiserver t=2024-12-06T03:15:22.318110703Z level=info msg="Adding Gro
upVersion iam.grafana.app v0alpha1 to ResourceManager"
2024-12-05 22:15:22 grafana       | logger=grafana-apiserver t=2024-12-06T03:15:22.319103404Z level=info msg="Adding Gro
upVersion playlist.grafana.app v0alpha1 to ResourceManager"
2024-12-05 22:15:22 grafana       | logger=grafana-apiserver t=2024-12-06T03:15:22.319985578Z level=info msg="Adding Gro
upVersion featuretoggle.grafana.app v0alpha1 to ResourceManager"
2024-12-05 22:15:22 grafana       | logger=plugin.installer t=2024-12-06T03:15:22.855270135Z level=info msg="Installing
plugin" pluginId=grafana-lokiexplore-app version=
2024-12-05 22:15:23 grafana       | logger=installer.fs t=2024-12-06T03:15:23.029752711Z level=info msg="Downloaded and
extracted grafana-lokiexplore-app v1.0.4 zip successfully to /var/lib/grafana/plugins/grafana-lokiexplore-app"
2024-12-05 22:15:23 grafana       | logger=plugins.registration t=2024-12-06T03:15:23.080832864Z level=info msg="Plugin
registered" pluginId=grafana-lokiexplore-app
2024-12-05 22:15:23 grafana       | logger=plugin.backgroundinstaller t=2024-12-06T03:15:23.080988575Z level=info msg="P
lugin successfully installed" pluginId=grafana-lokiexplore-app version= duration=1.665180452s
2024-12-05 22:16:36 grafana       | logger=infra.usagestats t=2024-12-06T03:16:36.426937473Z level=info msg="Usage stats
 are ready to report"
```



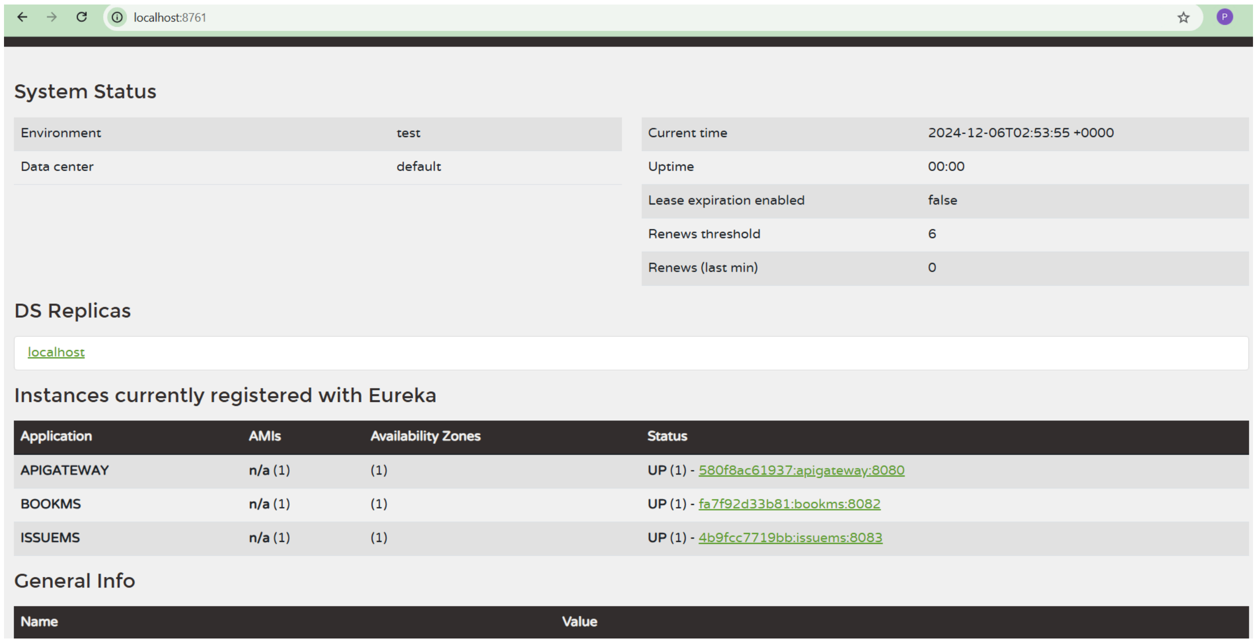
1.54 GB / 1.54 GB in use 8 images

Last refresh: 8 hours ago

Name	Tag	Status	Created	Size
bookms c77ab7e7b226	1.0	In use	56 seconds ag	517.21 MB
userms 0267e2e906e5	1.0	Unused	3 hours ago	496.64 MB
issuems bc916c61f1a5	1.0	In use	7 hours ago	524.34 MB
apigateway 091c9f3c61d9	1.0	In use	7 hours ago	497.8 MB
auth-server 94eba70ebbea	1.0	In use	1 day ago	491.44 MB
grafana/grafana c0b69935a246	latest	In use	1 day ago	485.62 MB
eureka-server 2ca41f23b65d	1.0	In use	1 day ago	495.65 MB
prom/prometheus	latest	In use	7 days ago	291.77 MB

Showing 8 items

Eureka-Server


localhost:8761

System Status

Environment	test
Data center	default

Current time	2024-12-06T02:53:55 +0000
Uptime	00:00
Lease expiration enabled	false
Renews threshold	6
Renews (last min)	0

DS Replicas

localhost

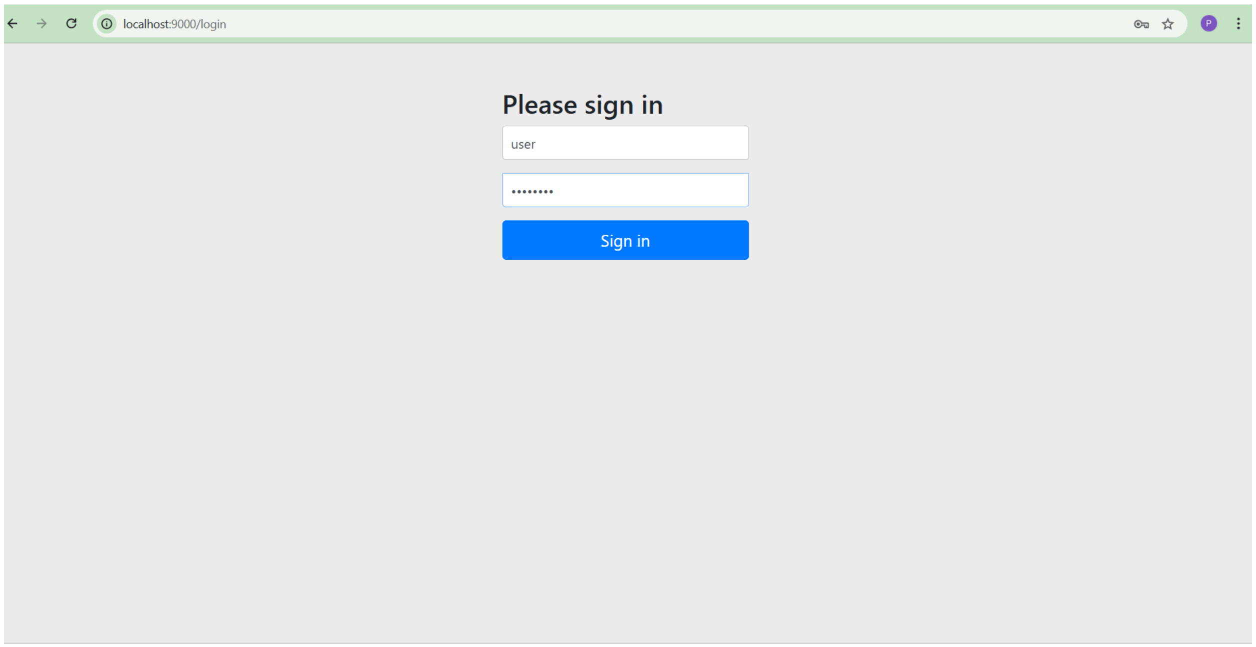
Instances currently registered with Eureka

Application	AMIs	Availability Zones	Status
APIGATEWAY	n/a (1)	(1)	UP (1) - 580f8ac61937:apigateway:8080
BOOKMS	n/a (1)	(1)	UP (1) - fa7f92d33b81:bookms:8082
ISSUEMS	n/a (1)	(1)	UP (1) - 4b9fcc7719bb:issuems:8083

General Info

Name	Value

OAuth2 Authorization Server- It gets the details from Customerdb and authenticates it. Customerdb has all the userdetails with roles.

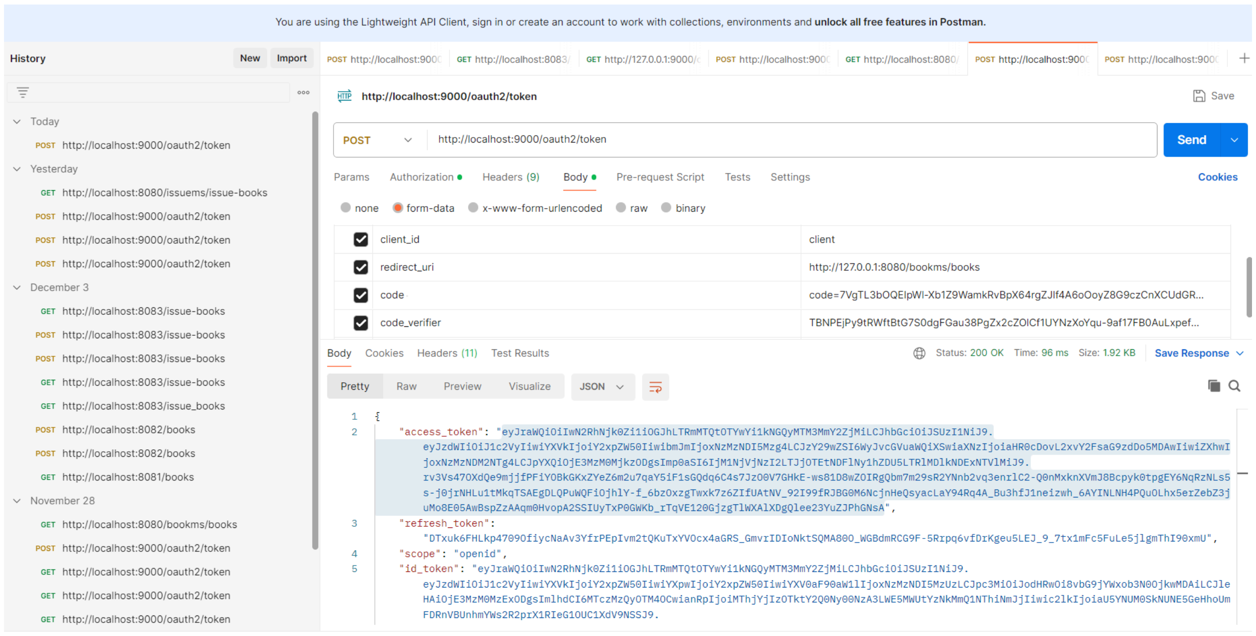

localhost:9000/login

Please sign in

user

••••••••

<u>**Authorization Token is generated**</u>


You are using the Lightweight API Client, sign in or create an account to work with collections, environments and **unlock all free features in Postman.**

POST http://localhost:9000/oauth2/token

Body: form-data

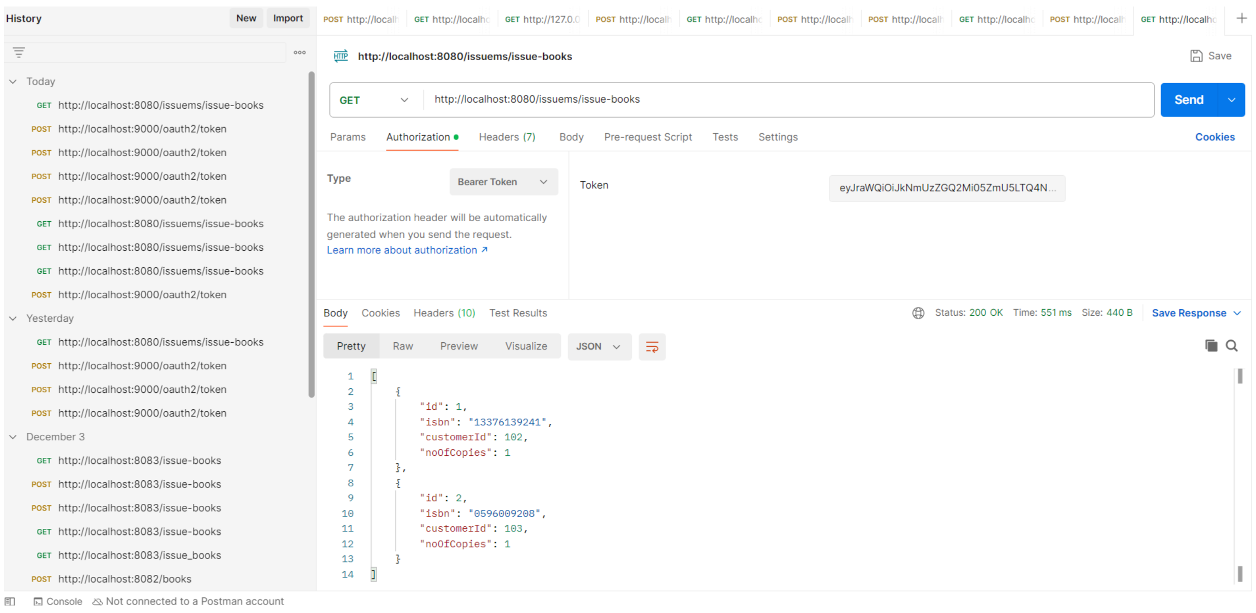
client_id	client
redirect_uri	http://127.0.0.1:8080/bookms/books
code	code=7VgTL3bOQElpWI-Xb1Z9WamkRvBpX64rgZJlf4A6oOoyZ8G9czCnXCUdGR...
code_verifier	TBNPEjPy9tRWftBtG7S0dgFGau38PgZx2cZOlCf1UYNzXoYqu-9af17FB0AuLxpef...

Status: 200 OK Time: 96 ms Size: 1.92 KB

```
{
    "access_token": "eyJraWQiOiIwN2RhNjk0Zi1iOGJhLTRmMTQtOTYwYi1kNGQyMTM3MmY2ZjMiLCJhbGciOiJSUzI1NiJ9.
        eyJzdWIiOiJ1c2VyIiwiYXVkIjoiY2xpZW50IiwibmJmIjoxNzMzNDI5Mzg4LCJzY29wZSI6WyJvcGVuaWQiXSwiaXNzIjoiaHR0cDovL2xvY2FsaG9zdDo5MDAwIiwiZXhwI
        joxNzMzNDM2NTg4LCJpYXQiOjE3MzM0MjkzODgsImp0aSI6IjM1NjVjNzI2LTJjOTEtNDFlNy1hZDU5LTRlMDlkNDExNTVlMiJ9.
        rv3Vs47OXdQe9mjjfPFiYOBkGKxZYeZ6m2u7qaY5iF1sGQdq6C4s7JzO0V7GHkE-ws81D8wZOIRgQbm7m29sR2YNnb2vq3enrlC2-Q0nMxknXVmJ8Bcpyk0tpgEY6NqRzNLs5
        s-j0jrNHLu1tMkqTSAEgDLQPuWQFiOjhlY-f_6bzOxzgTwxk7z6ZIfUAtNV_92I99fRJBG0M6NcjnHeQsyacLaY94Rq4A_Bu3hfJ1neizwh_6AYINLNH4PQuOLhx5erZebZ3j
        uMo8E05AwBspZzAAqm0HvopA2SSIUyTxP0GWKb_rTqVE120GjzgTlWXAlXDgQlee23YuZJPhGNsA",
    "refresh_token":
        "DTxuk6FHLkp47090fiycNaAv3YfrPEpIvm2tQKuTxYVOcx4aGRS_GmvrIDIoNktSQMA800_WGBdmRCG9F-5Rrpq6vfDrKgeu5LEJ_9_7tx1mFc5FuLe5jlgmThI90xmU",
    "scope": "openid",
    "id_token": "eyJraWQiOiIwN2RhNjk0Zi1iOGJhLTRmMTQtOTYwYi1kNGQyMTM3MmY2ZjMiLCJhbGciOiJSUzI1NiJ9.
        eyJzdWIiOiJ1c2VyIiwiYXVkIjoiY2xpZW50IiwiYXpwIjoiY2xpZW50IiwiYXV0aF90aW1lIjoxNzMzNDI5MzUzLCJpc3MiOiJodHRwOi8vbG9jYWxob3N0OjkwMDAiLCJle
        HAiOjE3MzM0MzExODgsImlhdCI6MTczMzQyOTM4OCwianRpIjoiMThjYjIzOTktY2Q0Ny00NzA3LWE5MWUtYzNkMmQ1NThiNmJjIiwic2lkIjoiaU5YNUM0SkNUNE5GeHhoUm
        FDRnVBUnhmYWs2R2prX1RIeG1OUC1XdV9NSSJ9.
```

Get Methods (provide the authorization token for fetching resource server API gateway)

http://lcoalhost/:8080/issuems/issue-books- To get the issue details of the books


GET http://localhost:8080/issuems/issue-books

Authorization — Type: Bearer Token

The authorization header will be automatically generated when you send the request.
Learn more about authorization ↗

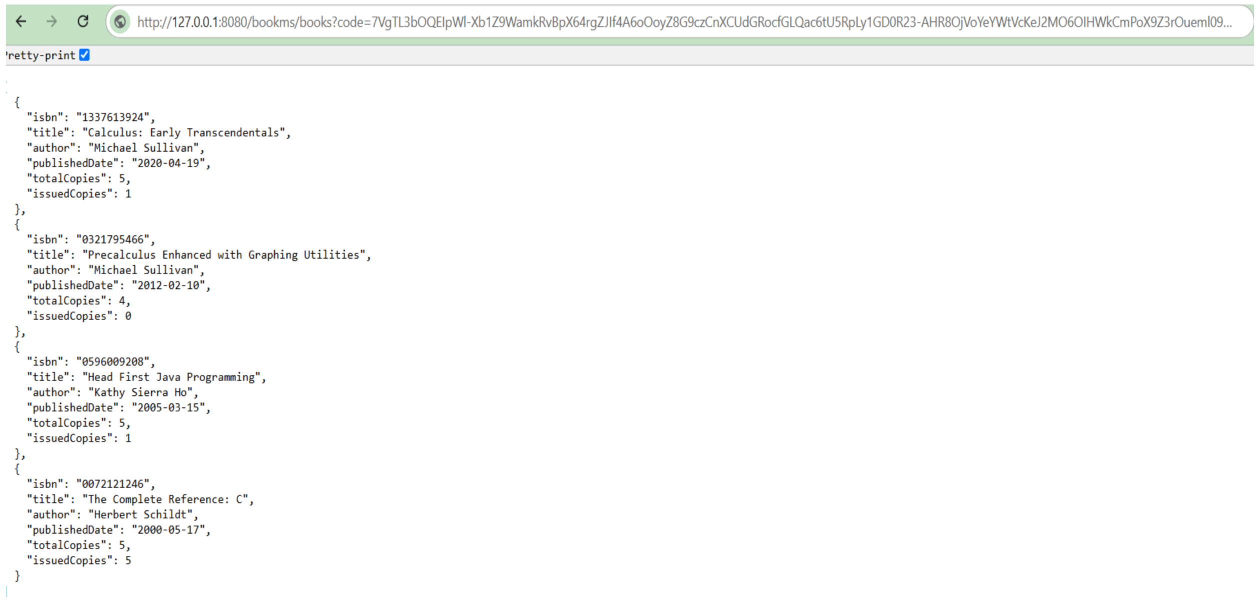
Token: eyJraWQiOiJkNmUzZGQ2Mi05ZmU5LTQ4N...

Status: 200 OK Time: 551 ms Size: 440 B

```
[
    {
        "id": 1,
        "isbn": "13376139241",
        "customerId": 102,
        "noOfCopies": 1
    },
    {
        "id": 2,
        "isbn": "0596009208",
        "customerId": 103,
        "noOfCopies": 1
    }
]
```

When you access the API gateway in a browser. OAuth2 authenticate the user with login form. If the user is a valid user it provides authorization code.

http://lcoalhost/:8080/bookms/books- To get the details of the books(through API Gateway)

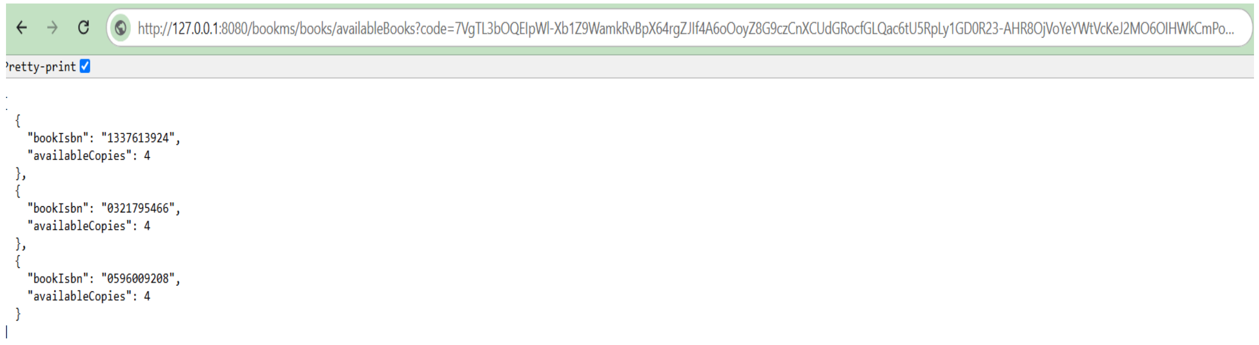


http://127.0.0.1:8080/bookms/books?code=7VgTL3bOQElpWl-Xb1Z9WamkRvBpX64rgZJIf4A6oOoyZ8G9czCnXCUdGRocfGLQac6tU5RpLy1GD0R23-AHR8OjVoYeYWtVcKeJ2MO6OIHWkCmPoX9Z3rOueml09...

Pretty-print ✓

```
  {
    "isbn": "1337613924",
    "title": "Calculus: Early Transcendentals",
    "author": "Michael Sullivan",
    "publishedDate": "2020-04-19",
    "totalCopies": 5,
    "issuedCopies": 1
  },
  {
    "isbn": "0321795466",
    "title": "Precalculus Enhanced with Graphing Utilities",
    "author": "Michael Sullivan",
    "publishedDate": "2012-02-10",
    "totalCopies": 4,
    "issuedCopies": 0
  },
  {
    "isbn": "0596009208",
    "title": "Head First Java Programming",
    "author": "Kathy Sierra Ho",
    "publishedDate": "2005-03-15",
    "totalCopies": 5,
    "issuedCopies": 1
  },
  {
    "isbn": "0072121246",
    "title": "The Complete Reference: C",
    "author": "Herbert Schildt",
    "publishedDate": "2000-05-17",
    "totalCopies": 5,
    "issuedCopies": 5
  }
```

http://localhost:8080/books/availableBooks- It displays the available copies of each book

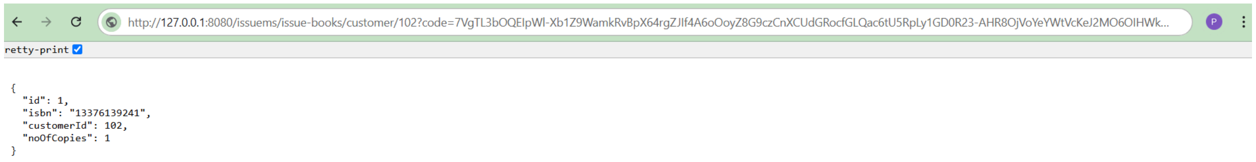


http://127.0.0.1:8080/bookms/books/availableBooks?code=7VgTL3bOQElpWl-Xb1Z9WamkRvBpX64rgZJIf4A6oOoyZ8G9czCnXCUdGRocfGLQac6tU5RpLy1GD0R23-AHR8OjVoYeYWtVcKeJ2MO6OIHWkCmPo...

Pretty-print ✓

```
  {
    "bookIsbn": "1337613924",
    "availableCopies": 4
  },
  {
    "bookIsbn": "0321795466",
    "availableCopies": 4
  },
  {
    "bookIsbn": "0596009208",
    "availableCopies": 4
  }
```

http://localhost:8080/issuems/issue-books/customer/102 - Display the books taken by customer 102



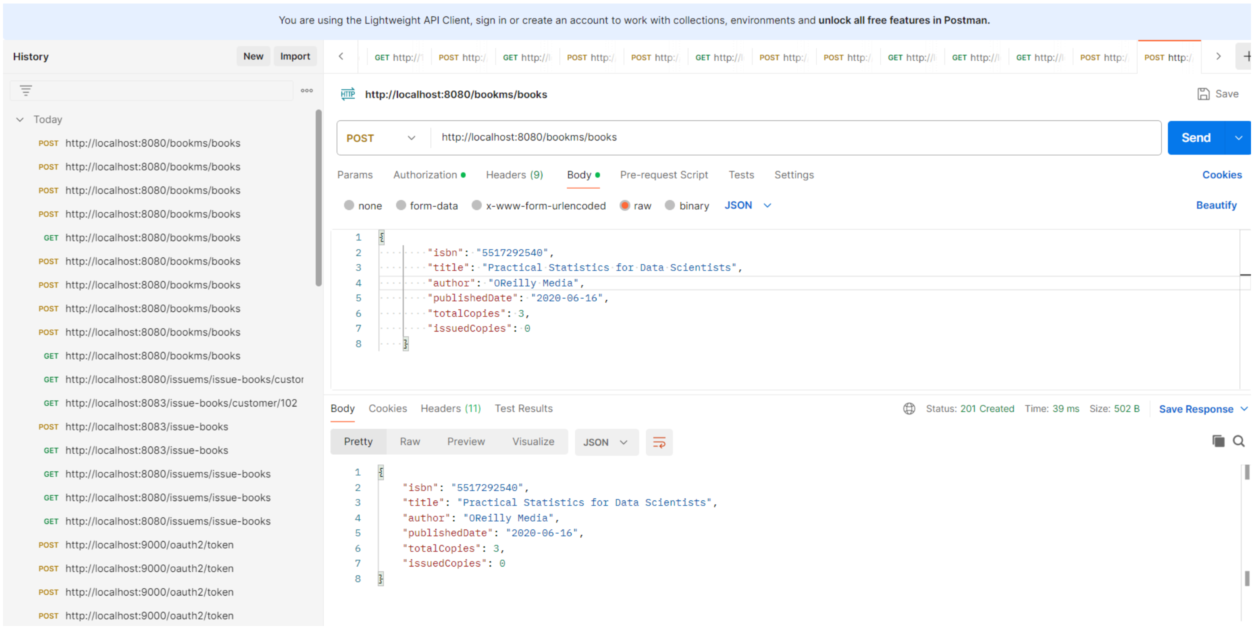
http://127.0.0.1:8080/issuems/issue-books/customer/102?code=7VgTL3bOQEIpWI-Xb1Z9WamkRvBpX64rgZJIf4A6oOoyZ8G9czCnXCUdGRocfGLQac6tU5RpLy1GD0R23-AHR8OjVoYeYWtVcKeJ2MO6OIHWk...

retty-print ✓

```
{
  "id": 1,
  "isbn": "13376139241",
  "customerId": 102,
  "noOfCopies": 1
}
```

POSTS

http://localhost:8080/bookms/books - Adding a new book to the table



POST http://localhost:8080/bookms/books

```
{
    "isbn": "5517292540",
    "title": "Practical Statistics for Data Scientists",
    "author": "OReilly Media",
    "publishedDate": "2020-06-16",
    "totalCopies": 3,
    "issuedCopies": 0
}
```

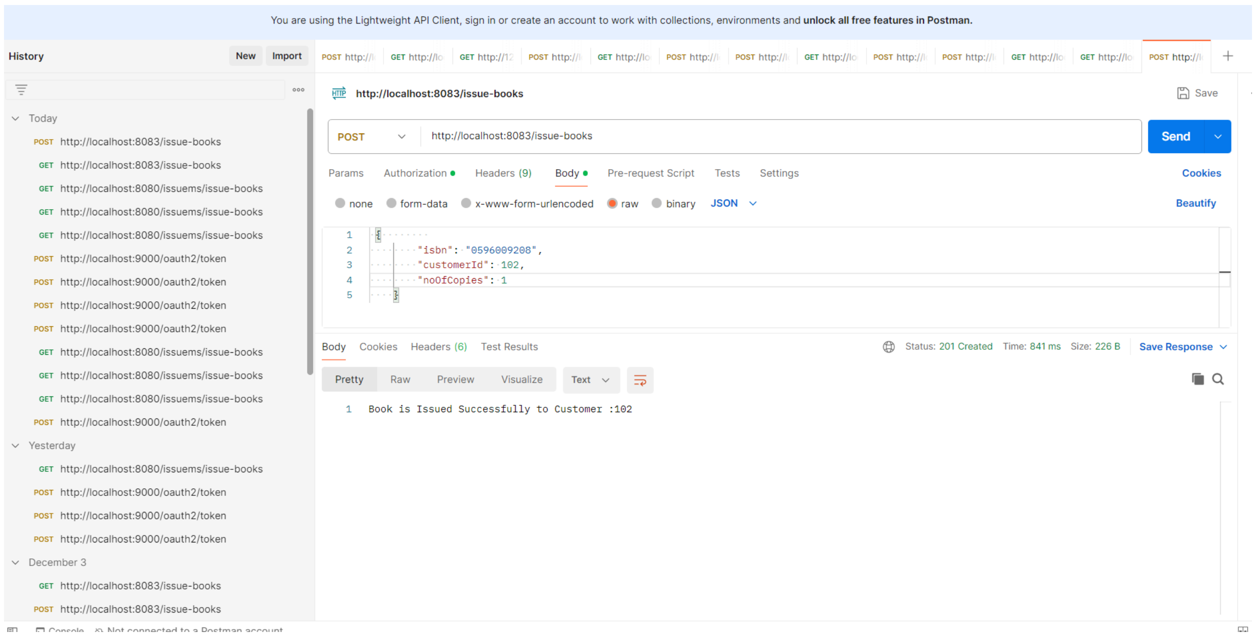
Status: 201 Created   Time: 39 ms   Size: 502 B

```
{
    "isbn": "5517292540",
    "title": "Practical Statistics for Data Scientists",
    "author": "OReilly Media",
    "publishedDate": "2020-06-16",
    "totalCopies": 3,
    "issuedCopies": 0
}
```

You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.
History
New
Import
Today
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
GET http://localhost:8080/issuems/issue-books/custor
GET http://localhost:8083/issue-books/customer/102
POST http://localhost:8083/issue-books
GET http://localhost:8083/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
Save
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
Beautify
Body
Cookies
Headers (11)
Test Results
Save Response
Pretty
Raw
Preview
Visualize
JSON

http://localhost:8080/issuems/issue-books- If a customer requested a copies of books

First it calls the bookms microservice to check the availability of requested book by calculating(totalcopies-issuedcopies)
If so the book is issued to the customer


You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.
History
New
Import
Today
POST http://localhost:8083/issue-books
GET http://localhost:8083/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
Yesterday
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
December 3
GET http://localhost:8083/issue-books
POST http://localhost:8083/issue-books
Save
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
JSON
Beautify
Body
Cookies
Headers (6)
Test Results
Save Response
Pretty
Raw
Preview
Visualize
Text

http://localhost:8083/issue-books

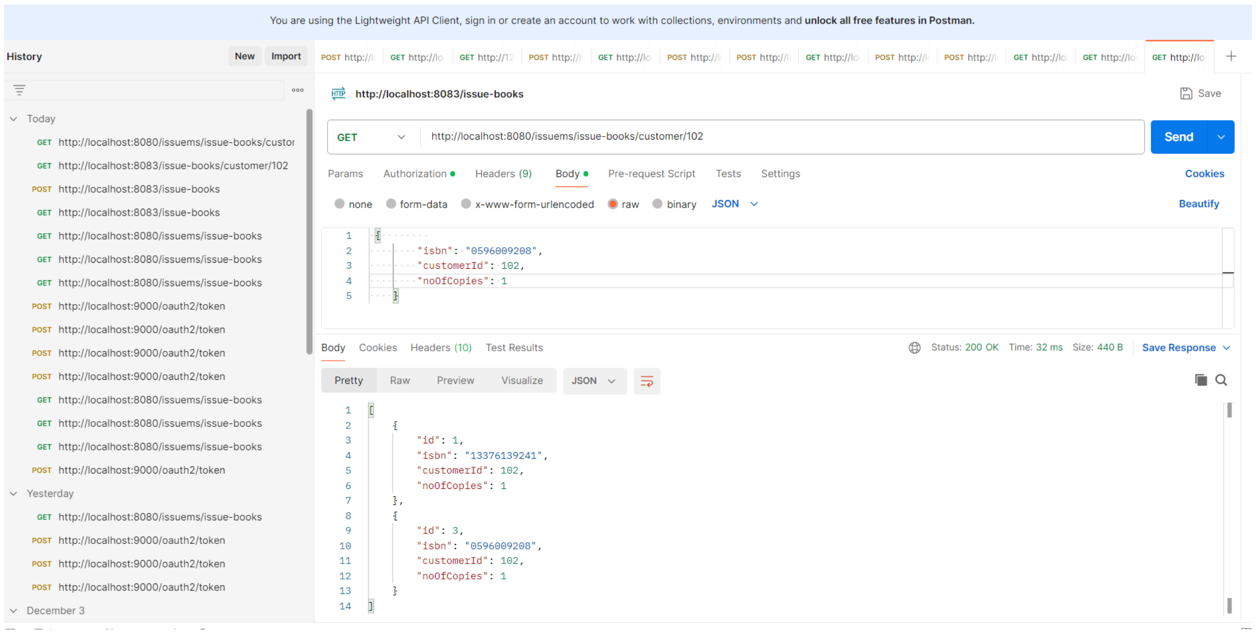
POST http://localhost:8083/issue-books

```
{
    "isbn": "0596009208",
    "customerId": 102,
    "noOfCopies": 1
}
```

Status: 201 Created Time: 841 ms Size: 226 B

```
Book is Issued Successfully to Customer :102
```

<span style="text-decoration: underline">After Issuing the book to customer 102 -Get Method to show all the books taken by customer 103</span>


You are using the Lightweight API Client, sign in or create an account to work with collections, environments and unlock all free features in Postman.
History
New
Import
Today
GET http://localhost:8080/issuems/issue-books/custor
GET http://localhost:8083/issue-books/customer/102
POST http://localhost:8083/issue-books
GET http://localhost:8083/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
Yesterday
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
December 3
Save
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
JSON
Beautify
Body
Cookies
Headers (10)
Test Results
Save Response
Pretty
Raw
Preview
Visualize
JSON

http://localhost:8083/issue-books

GET http://localhost:8080/issuems/issue-books/customer/102

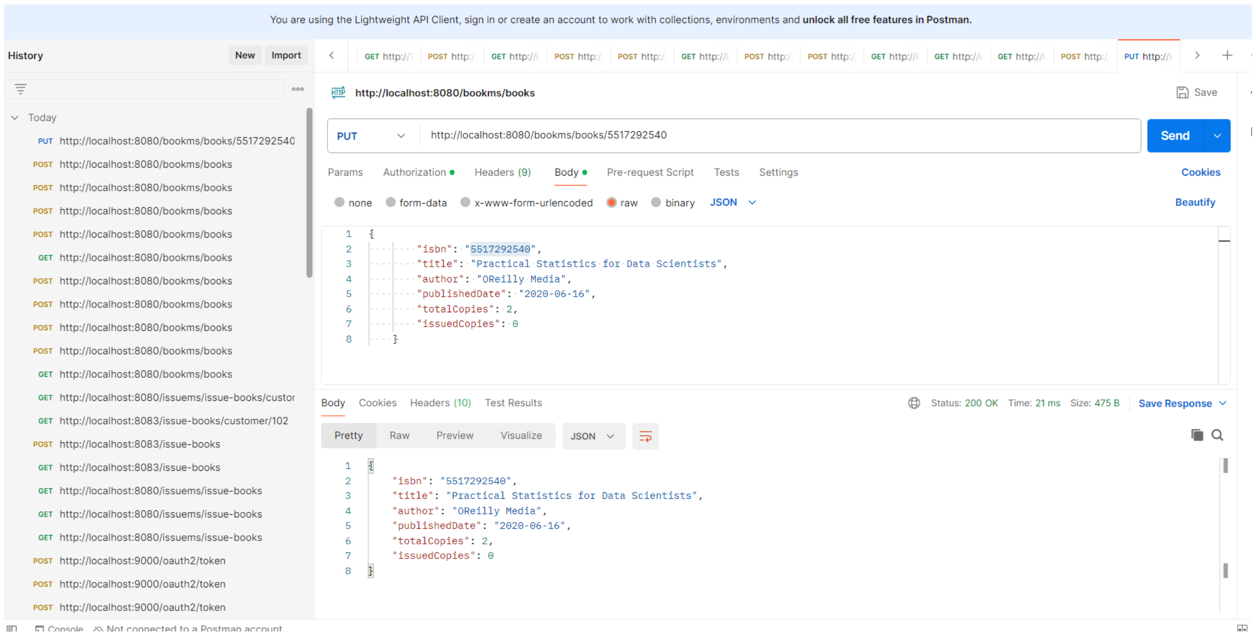
```
{
    "isbn": "0596009208",
    "customerId": 102,
    "noOfCopies": 1
}
```

Status: 200 OK Time: 32 ms Size: 440 B

```
[
    {
        "id": 1,
        "isbn": "13376139241",
        "customerId": 102,
        "noOfCopies": 1
    },
    {
        "id": 3,
        "isbn": "0596009208",
        "customerId": 102,
        "noOfCopies": 1
    }
]
```

PUT

Editing a Book details


You are using the Lightweight API Client, sign in or create an account to work with collections, environments and **unlock all free features in Postman.**

History

PUT http://localhost:8080/bookms/books/5517292540
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
GET http://localhost:8080/issuems/issue-books/custor
GET http://localhost:8083/issue-books/customer/102
POST http://localhost:8083/issue-books
GET http://localhost:8083/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token

http://localhost:8080/bookms/books

PUT http://localhost:8080/bookms/books/5517292540

```
{
    "isbn": "5517292540",
    "title": "Practical Statistics for Data Scientists",
    "author": "OReilly Media",
    "publishedDate": "2020-06-16",
    "totalCopies": 2,
    "issuedCopies": 0
  }
```

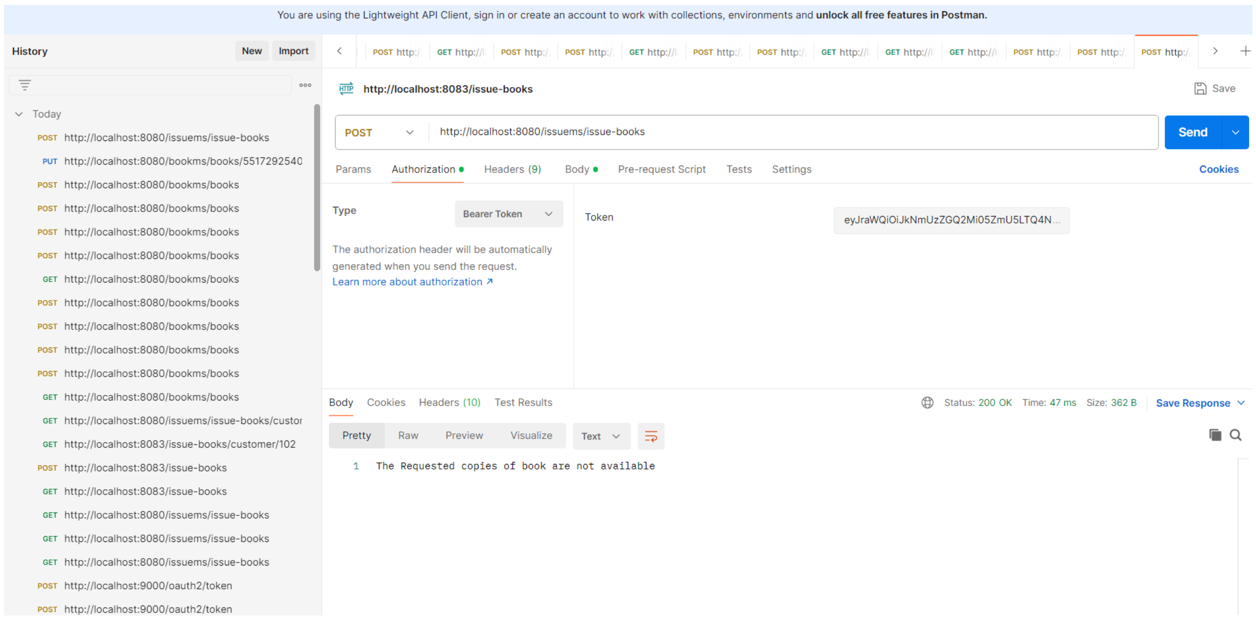
Status: 200 OK Time: 21 ms Size: 475 B

```
{
  "isbn": "5517292540",
  "title": "Practical Statistics for Data Scientists",
  "author": "OReilly Media",
  "publishedDate": "2020-06-16",
  "totalCopies": 2,
  "issuedCopies": 0
}
```

Console Not connected to a Postman account

Trying to issue more copies of books which are not available

If the requested copies of books are not available. Then It will display the copies are not available


You are using the Lightweight API Client, sign in or create an account to work with collections, environments and **unlock all free features in Postman.**

History

POST http://localhost:8080/issuems/issue-books
PUT http://localhost:8080/bookms/books/5517292540
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
POST http://localhost:8080/bookms/books
GET http://localhost:8080/bookms/books
GET http://localhost:8080/issuems/issue-books/custor
GET http://localhost:8083/issue-books/customer/102
POST http://localhost:8083/issue-books
GET http://localhost:8083/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
GET http://localhost:8080/issuems/issue-books
POST http://localhost:9000/oauth2/token
POST http://localhost:9000/oauth2/token

http://localhost:8083/issue-books

POST http://localhost:8080/issuems/issue-books

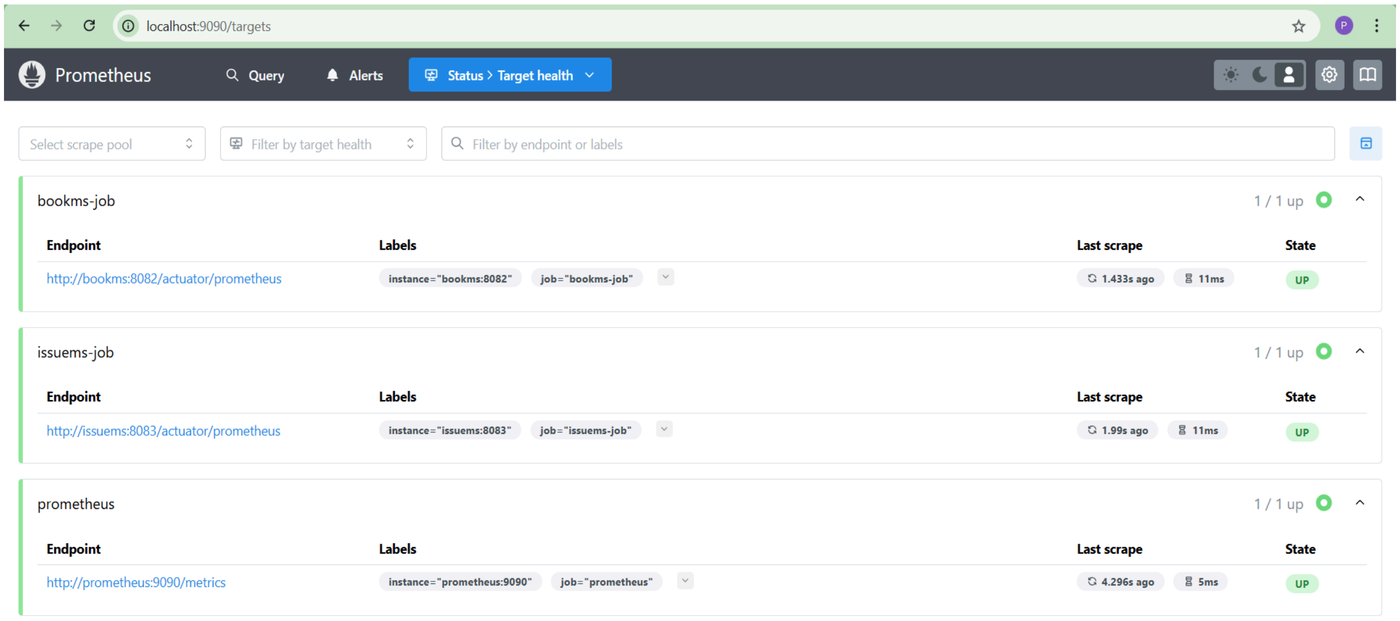
Authorization — Type: Bearer Token — Token: eyJraWQiOiJkNmUzZGQ2Mi05ZmU5LTQ4N...

The authorization header will be automatically generated when you send the request. Learn more about authorization ↗

Status: 200 OK Time: 47 ms Size: 362 B

```
The Requested copies of book are not available
```

Prometheus Targets:



bookms-job 1 / 1 up

Endpoint	Labels	Last scrape	State
http://bookms:8082/actuator/prometheus	instance="bookms:8082" job="bookms-job"	1.433s ago 11ms	UP

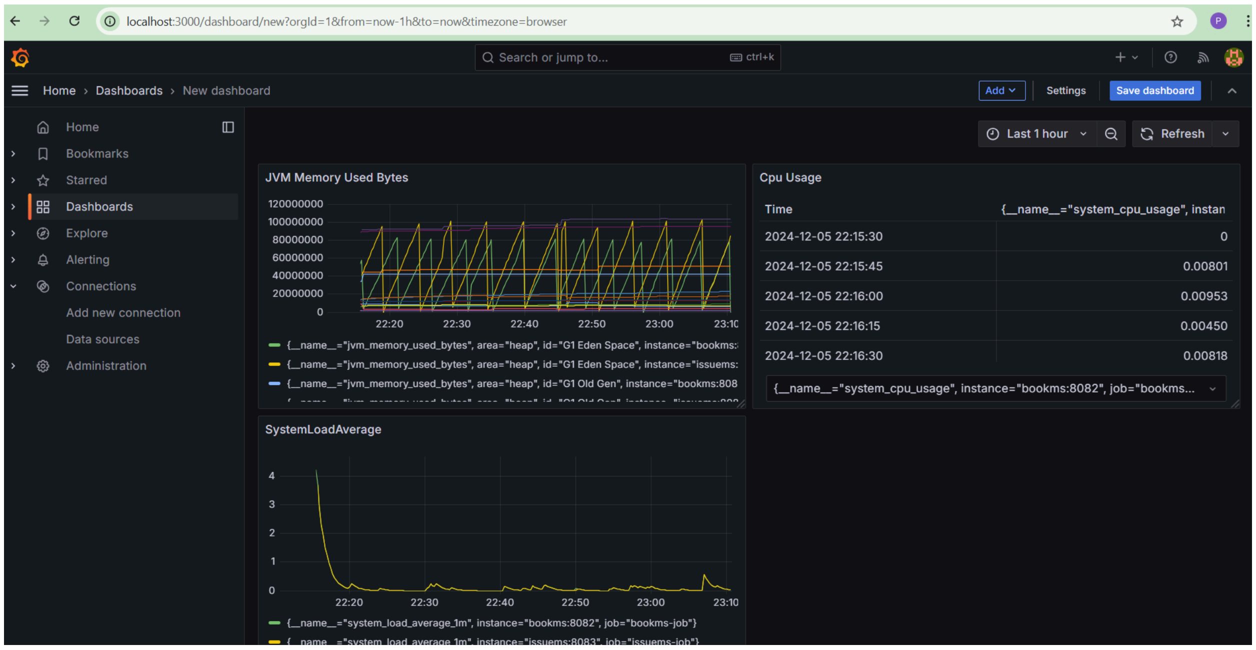
issuems-job 1 / 1 up

Endpoint	Labels	Last scrape	State
http://issuems:8083/actuator/prometheus	instance="issuems:8083" job="issuems-job"	1.99s ago 11ms	UP

prometheus 1 / 1 up

Endpoint	Labels	Last scrape	State
http://prometheus:9090/metrics	instance="prometheus:9090" job="prometheus"	4.296s ago 5ms	UP

Grafana Metrics


localhost:3000/dashboard/new?orgId=1&from=now-1h&to=now&timezone=browser
Search or jump to...
ctrl+k
Home › Dashboards › New dashboard
Add
Settings
Save dashboard
Home
Bookmarks
Starred
Dashboards
Explore
Alerting
Connections
Add new connection
Data sources
Administration
Last 1 hour
Refresh
JVM Memory Used Bytes
120000000
100000000
80000000
60000000
40000000
20000000
0
22:20
22:30
22:40
22:50
23:00
23:10
{__name__="jvm_memory_used_bytes", area="heap", id="G1 Eden Space", instance="bookms:
{__name__="jvm_memory_used_bytes", area="heap", id="G1 Eden Space", instance="issuems:
{__name__="jvm_memory_used_bytes", area="heap", id="G1 Old Gen", instance="bookms:808
SystemLoadAverage
4
3
2
1
0
22:20
22:30
22:40
22:50
23:00
23:10
{__name__="system_load_average_1m", instance="bookms:8082", job="bookms-job"}
{__name__="system_load_average_1m", instance="issuems:8083", job="issuems-job"}

Cpu Usage

Time	{__name__="system_cpu_usage", instan
2024-12-05 22:15:30	0
2024-12-05 22:15:45	0.00801
2024-12-05 22:16:00	0.00953
2024-12-05 22:16:15	0.00450
2024-12-05 22:16:30	0.00818

{__name__="system_cpu_usage", instance="bookms:8082", job="bookms...