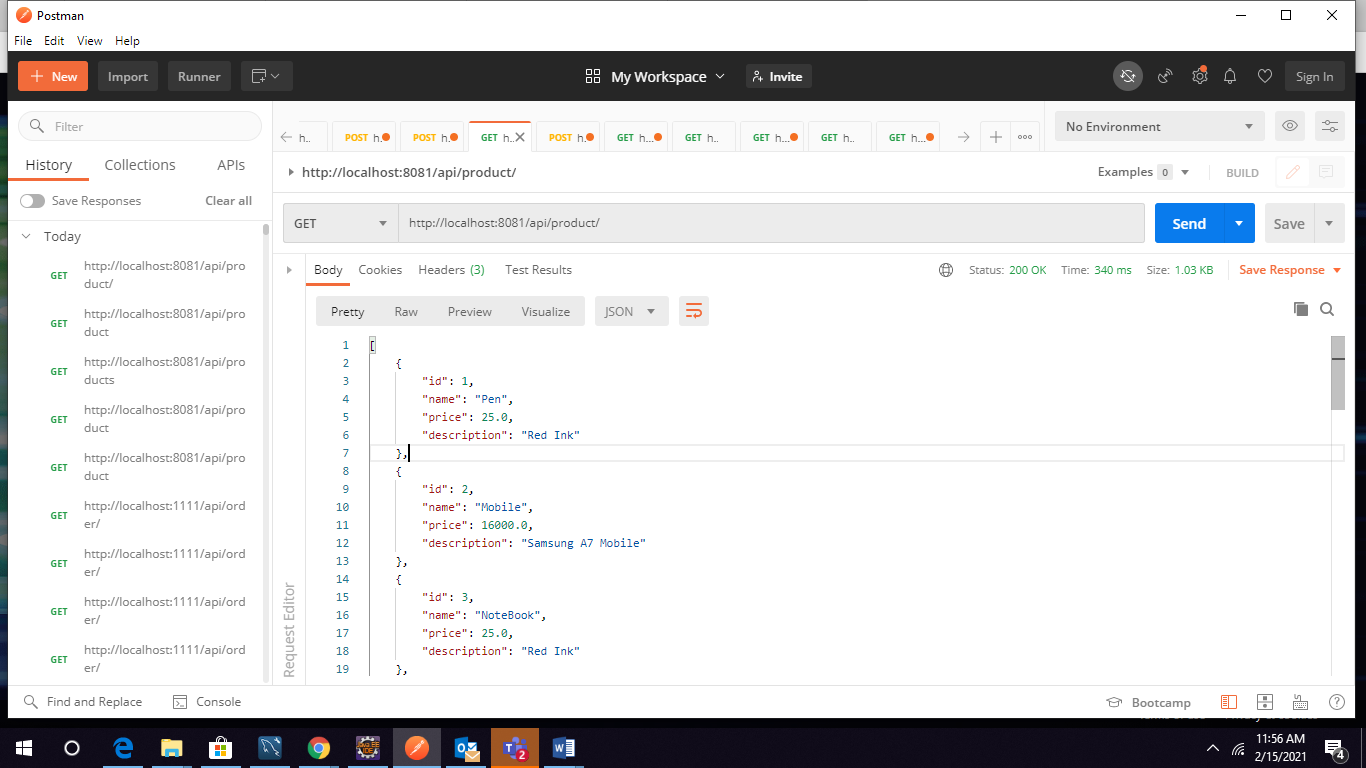
#### A) Product-Catalog-Service :

1. **GET request: http://localhost:8081/api/products**

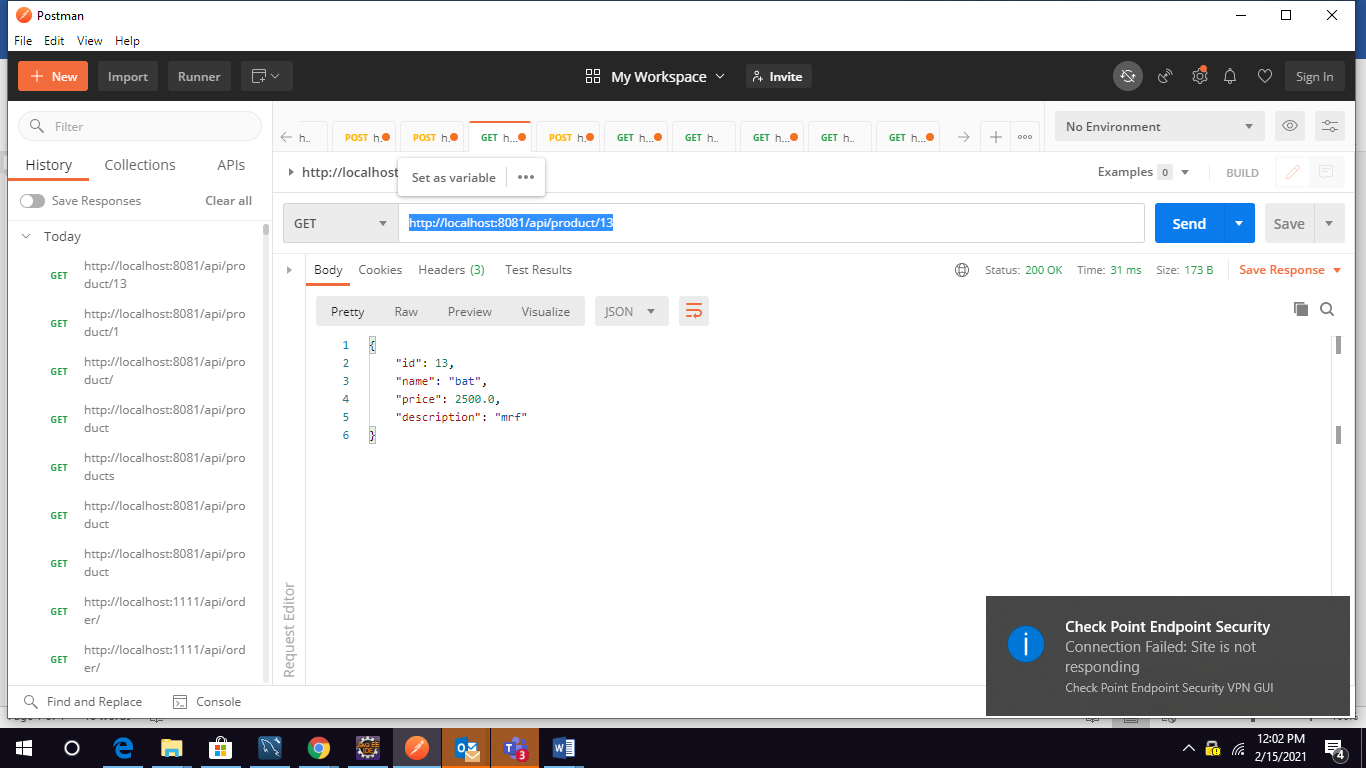
Finding all products



1. **GET request:** [**http://localhost:8081/api/products/1**](http://localhost:8081/api/products/1)

Finding specified product.

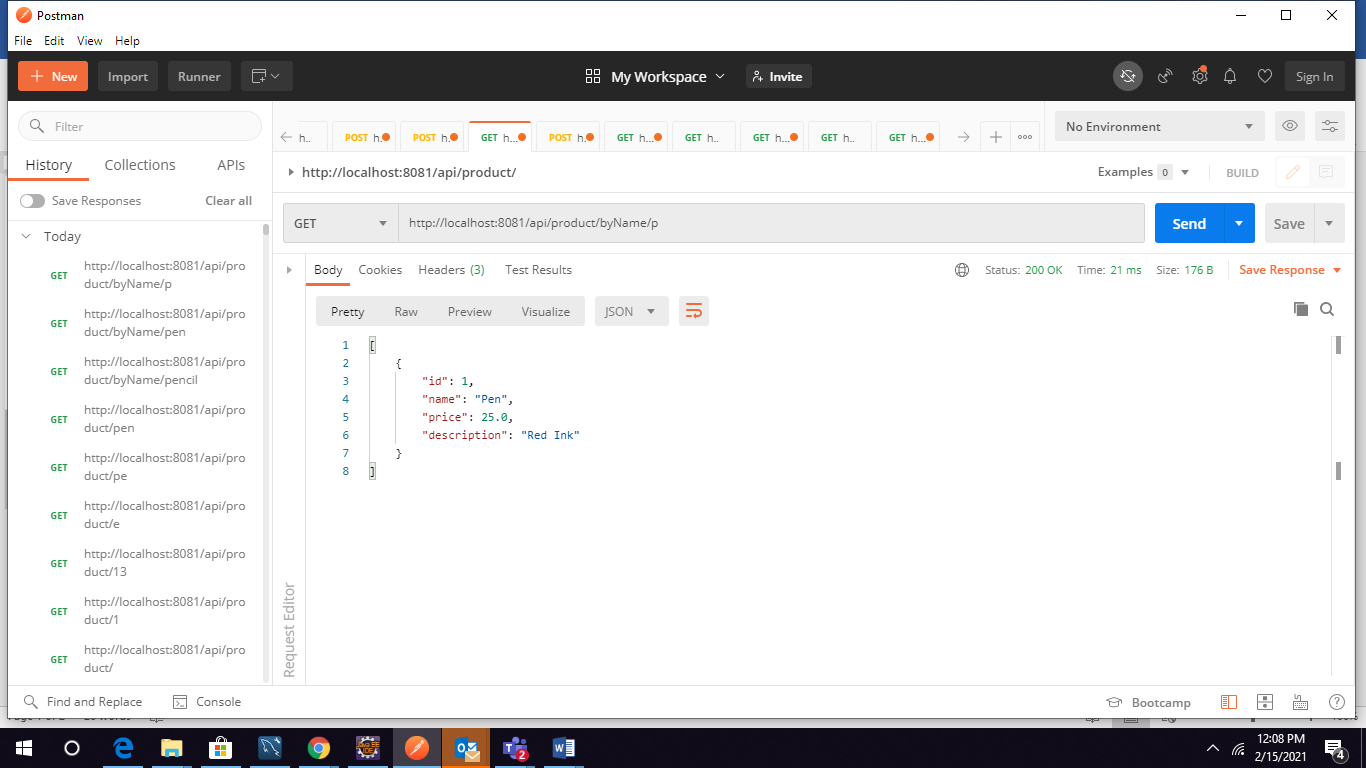
(ex: finding a product whose id is 13)



1. **GET request: http://localhost:8081/api/products/byName/e**

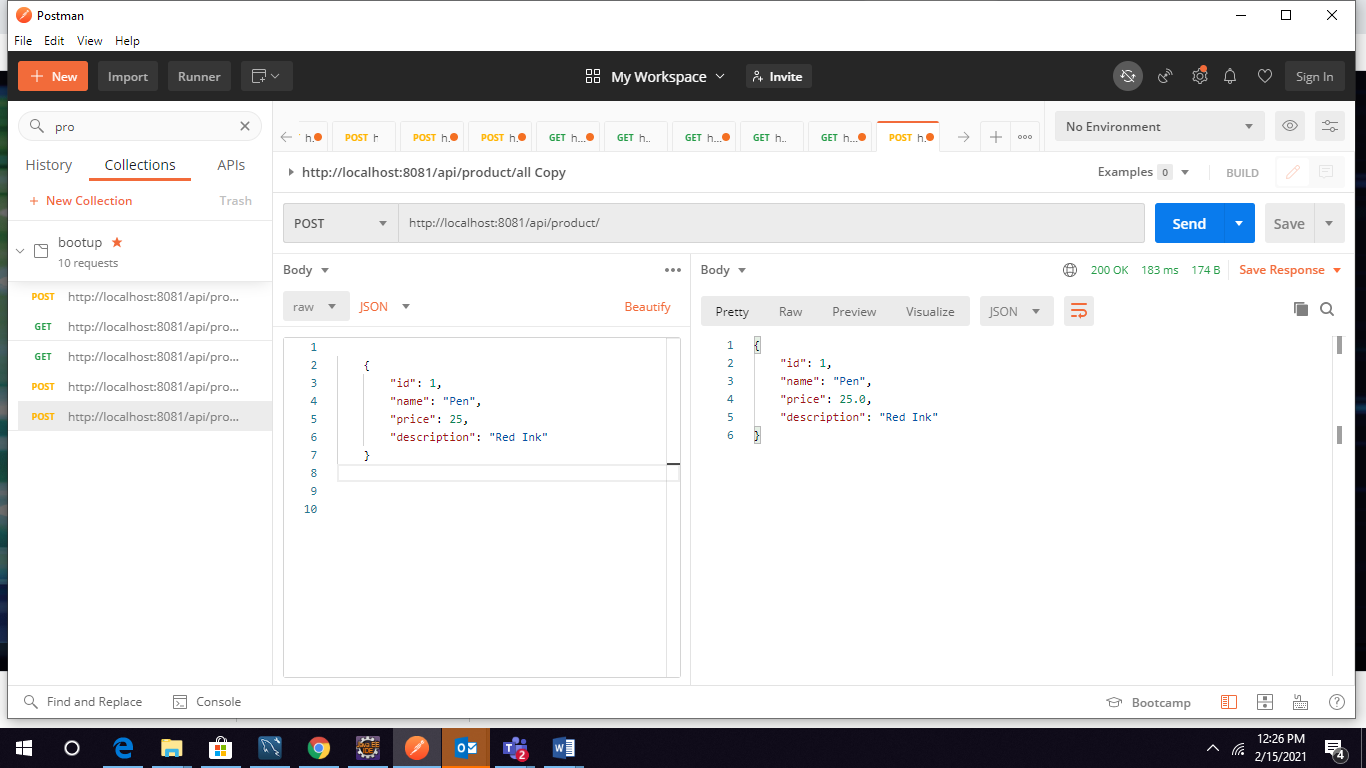
Finding all products by product name.

(ex: any product having a letter ‘p’)



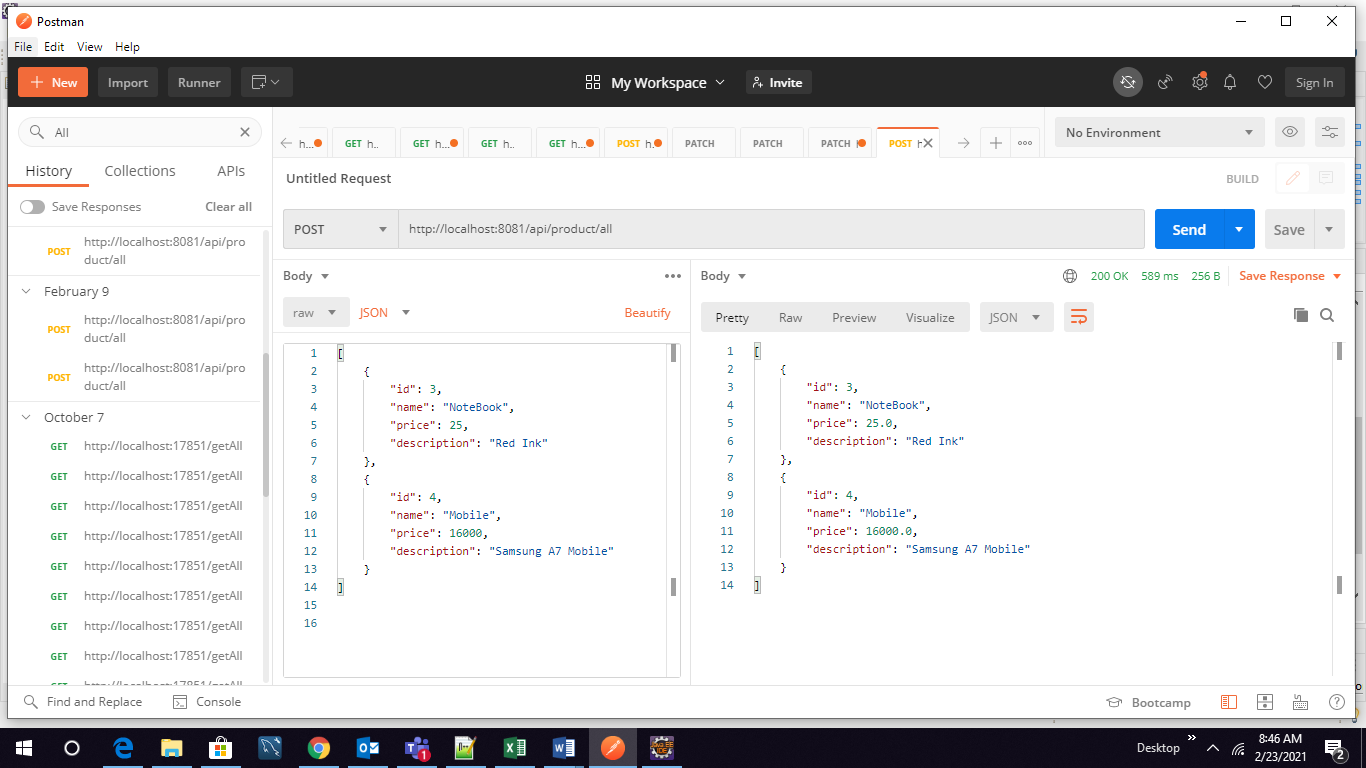
1. **POST request :** [**http://localhost:8081/api/products**](http://localhost:8081/api/products)

Posting a single product



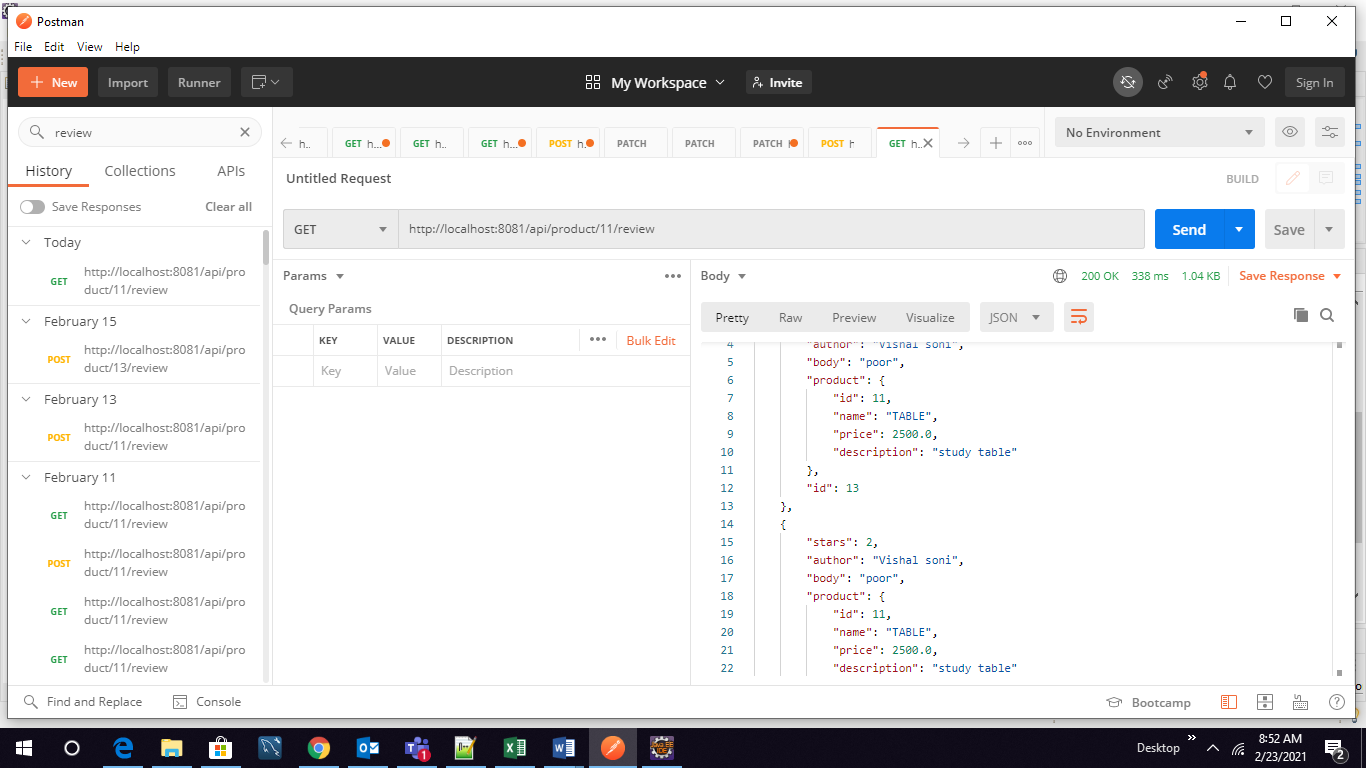
1. **POST request:** [**http://localhost:8081/api/products/all**](http://localhost:8081/api/products/all)

Posting more than a product



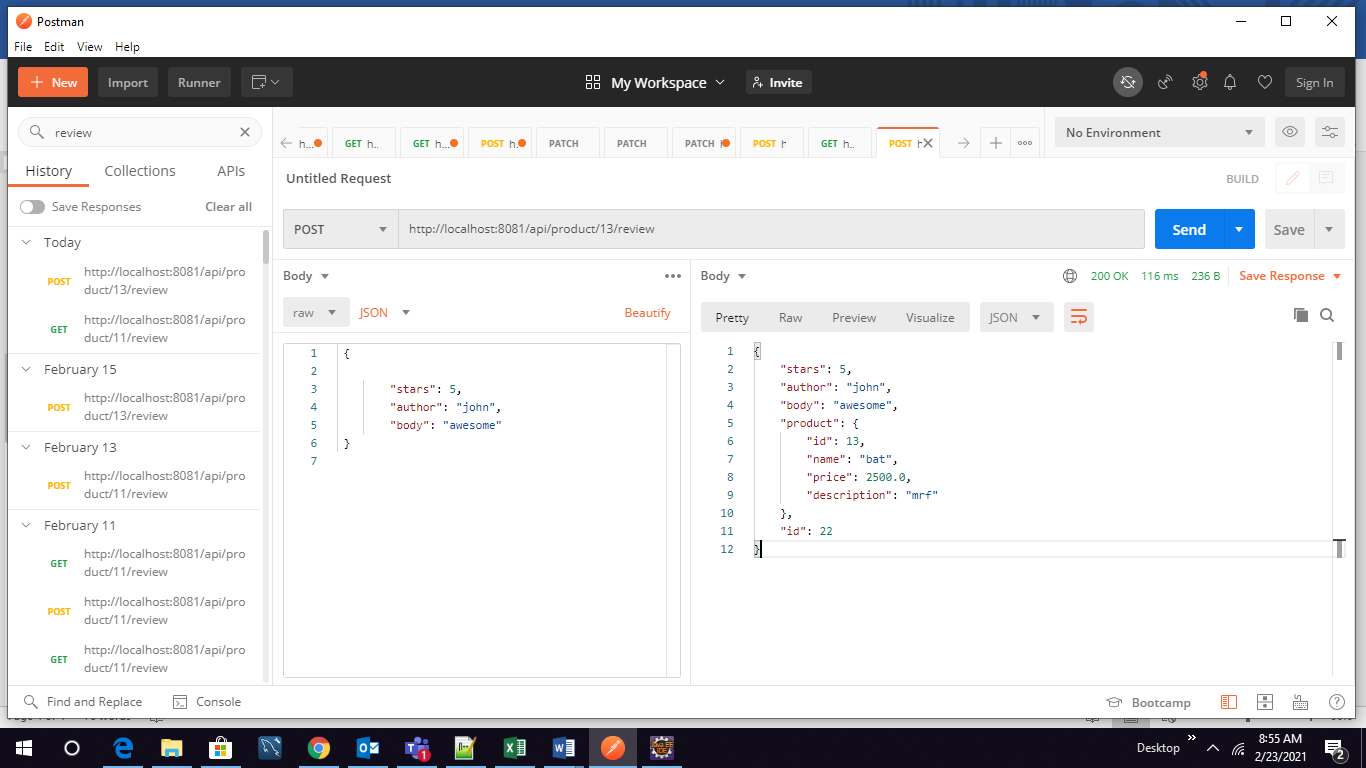
1. **GET request: http://localhost:8081/api/products/2/reviews**

Finding a product with Reviews

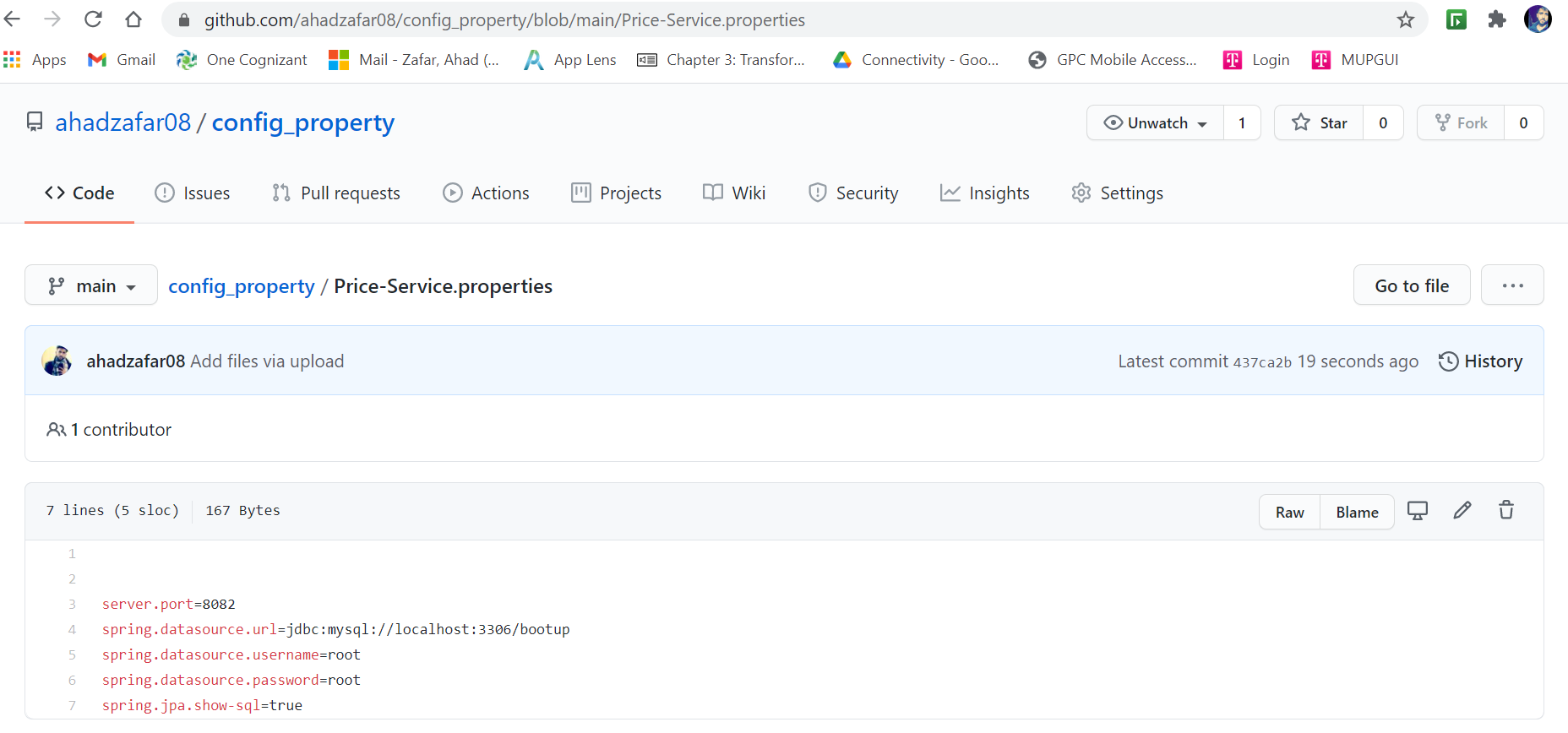


1. **POST request :** [**http://localhost:8081/api/products/6/reviews**](http://localhost:8081/api/products/6/reviews)

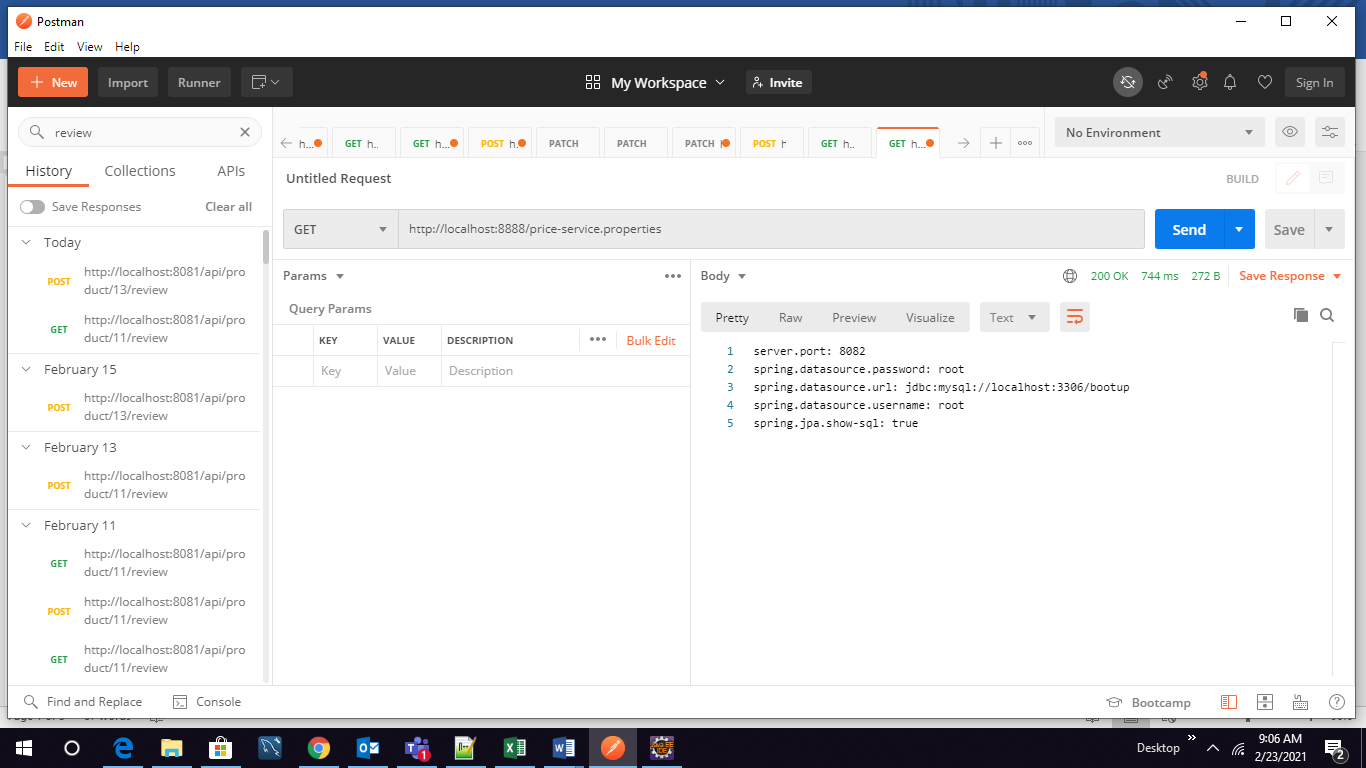
Posting a review for a specified product



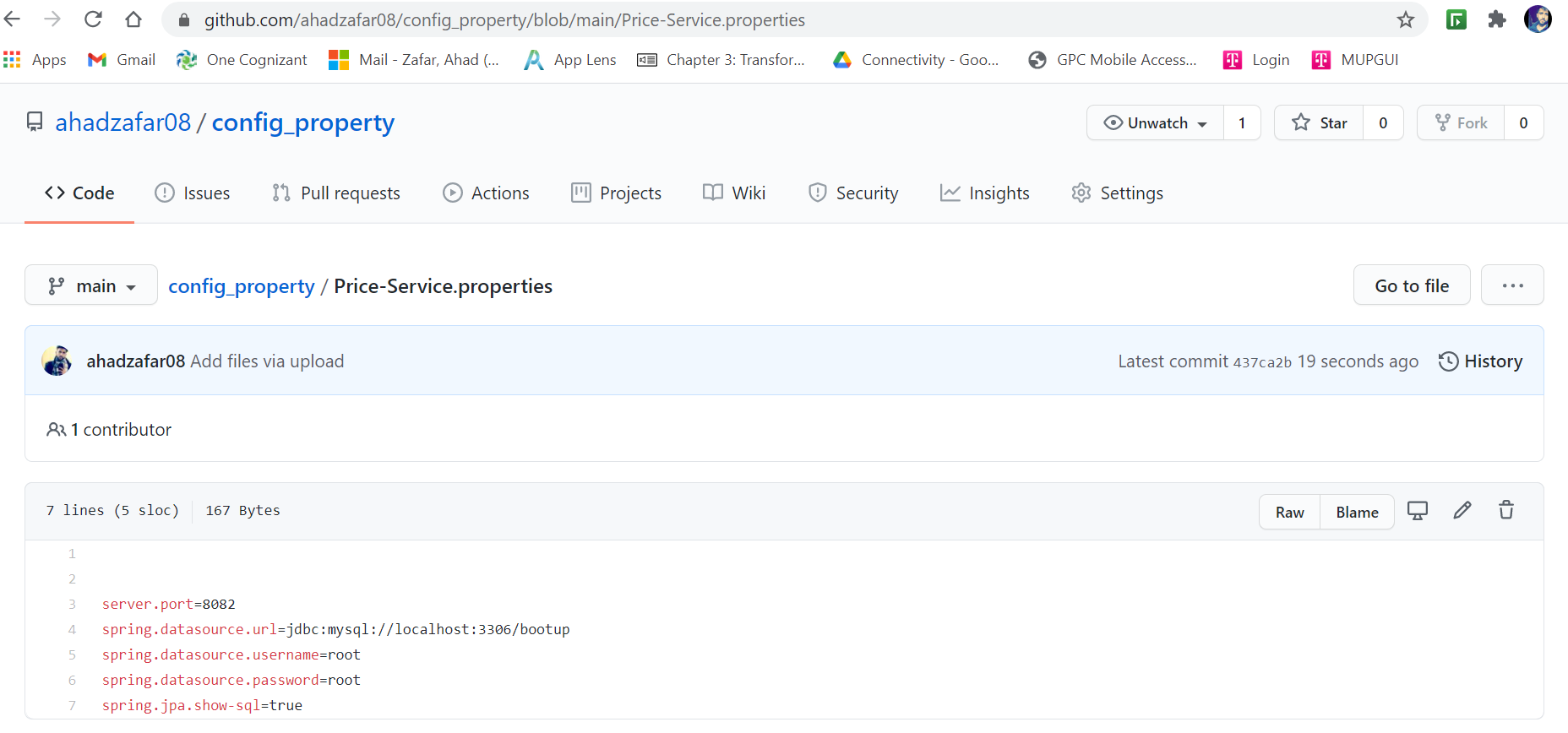
**Part-2 – Config Server, Microservices communication**



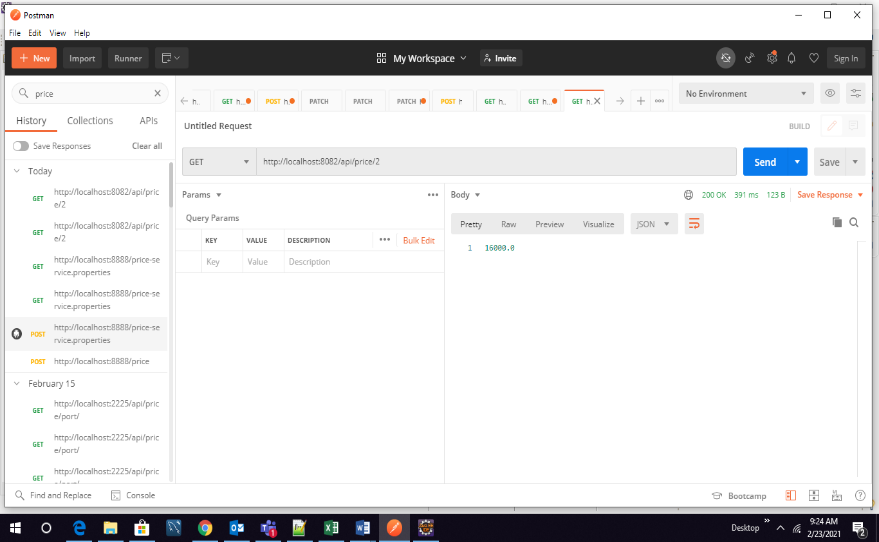
**http://localhost:8888/price-service.properties**



**B) Price-Service**



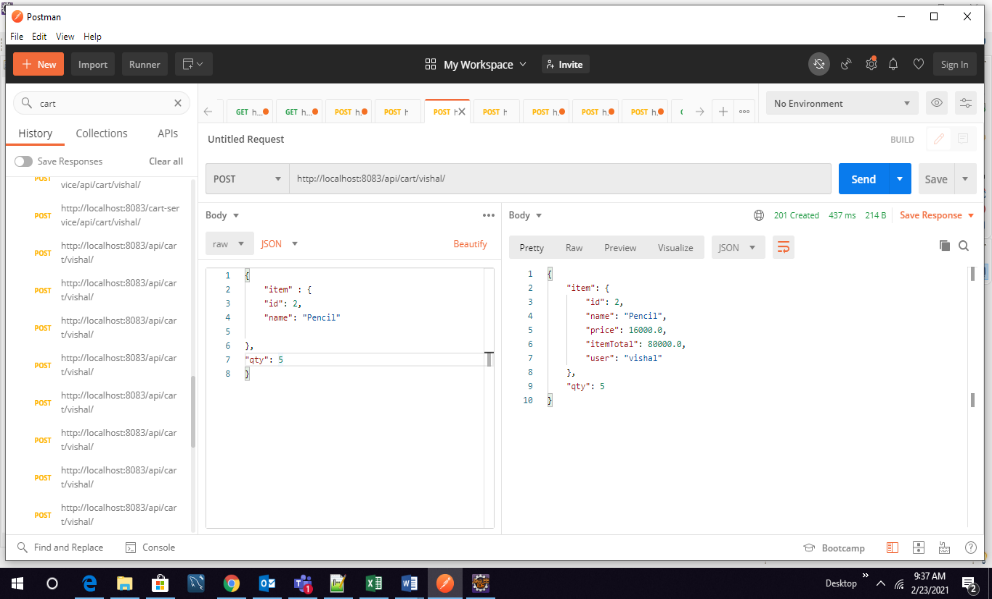
1. **GET** request: <http://localhost:8082/api/price/2> :Should return price of product id 2



**C) Cart-Service**

1. **POST request** [**http://localhost:8083/api/cart/ahad**](http://localhost:8083/api/cart/ahad)

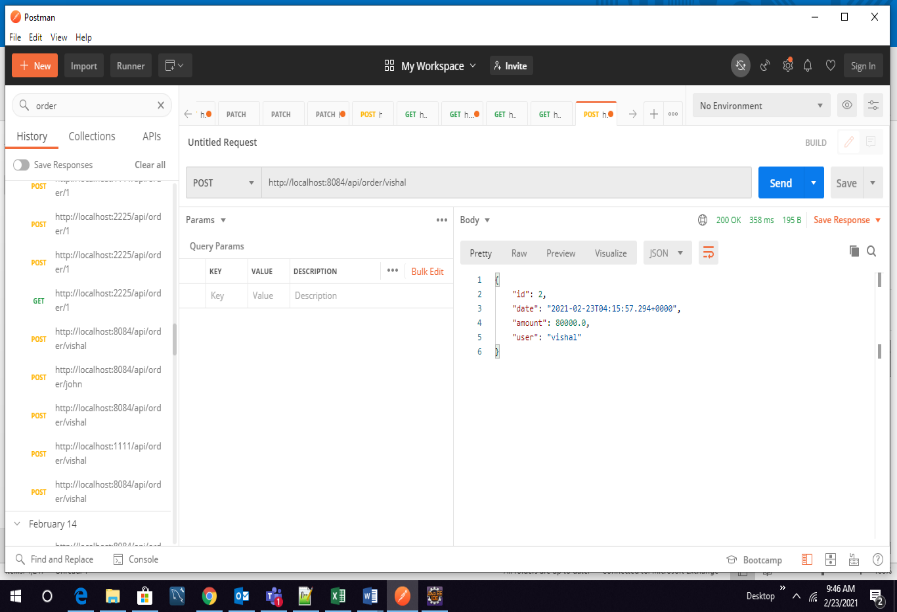
* The endpoint will add the product in the cart of logged in user
* Passing price in the body of the request is optional
* Pathvariable is the string which is the name of the user.



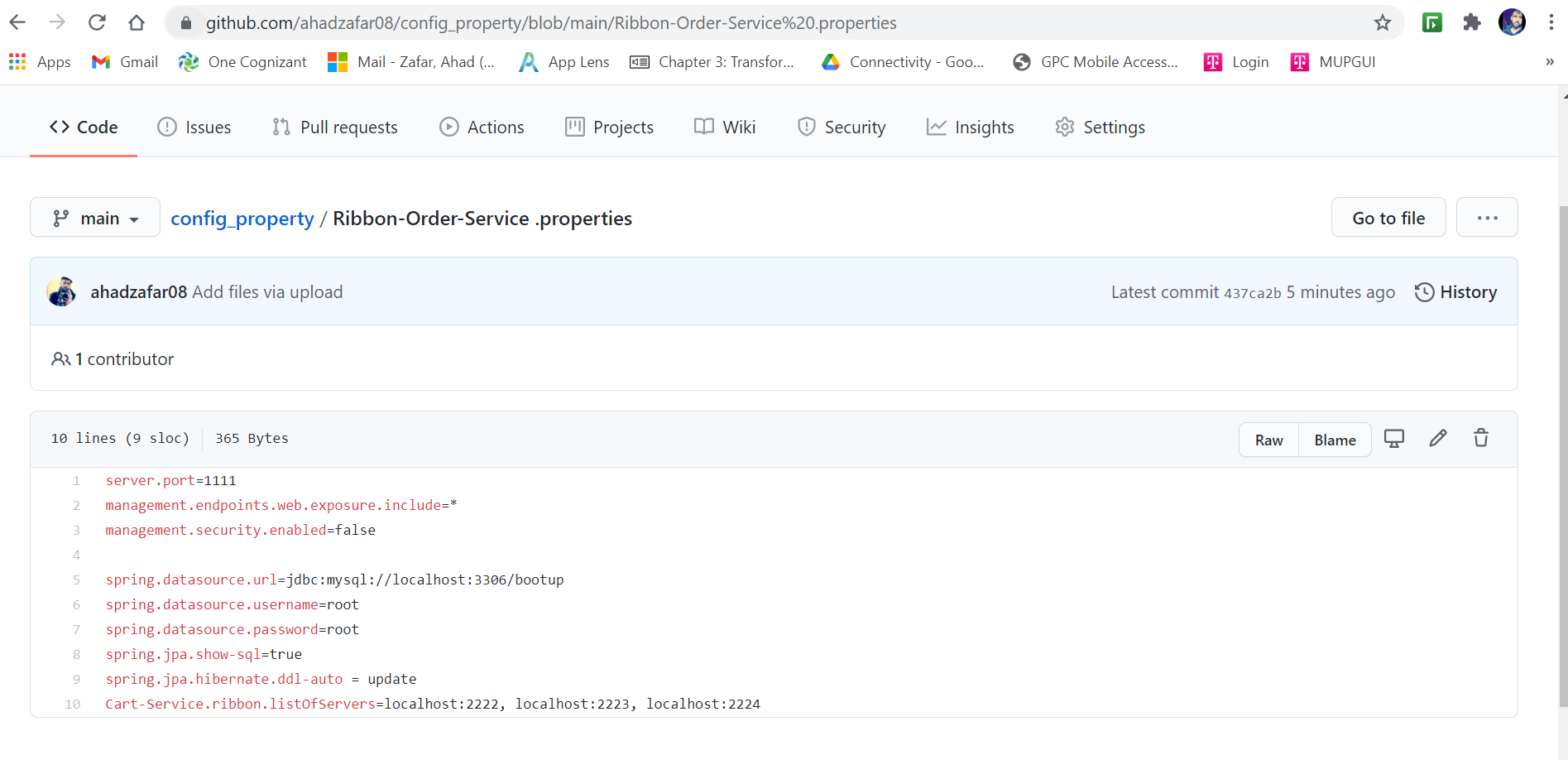
**D) Order-Service**

1. **POST request.** [**http://localhost:8084/api/orders/ahad**](http://localhost:8084/api/orders/ahad)

* This Api will pull the items from the cart of the user
* After pulling the item from cart order will be places
* Item will be deleted from the cart once order is done**.**

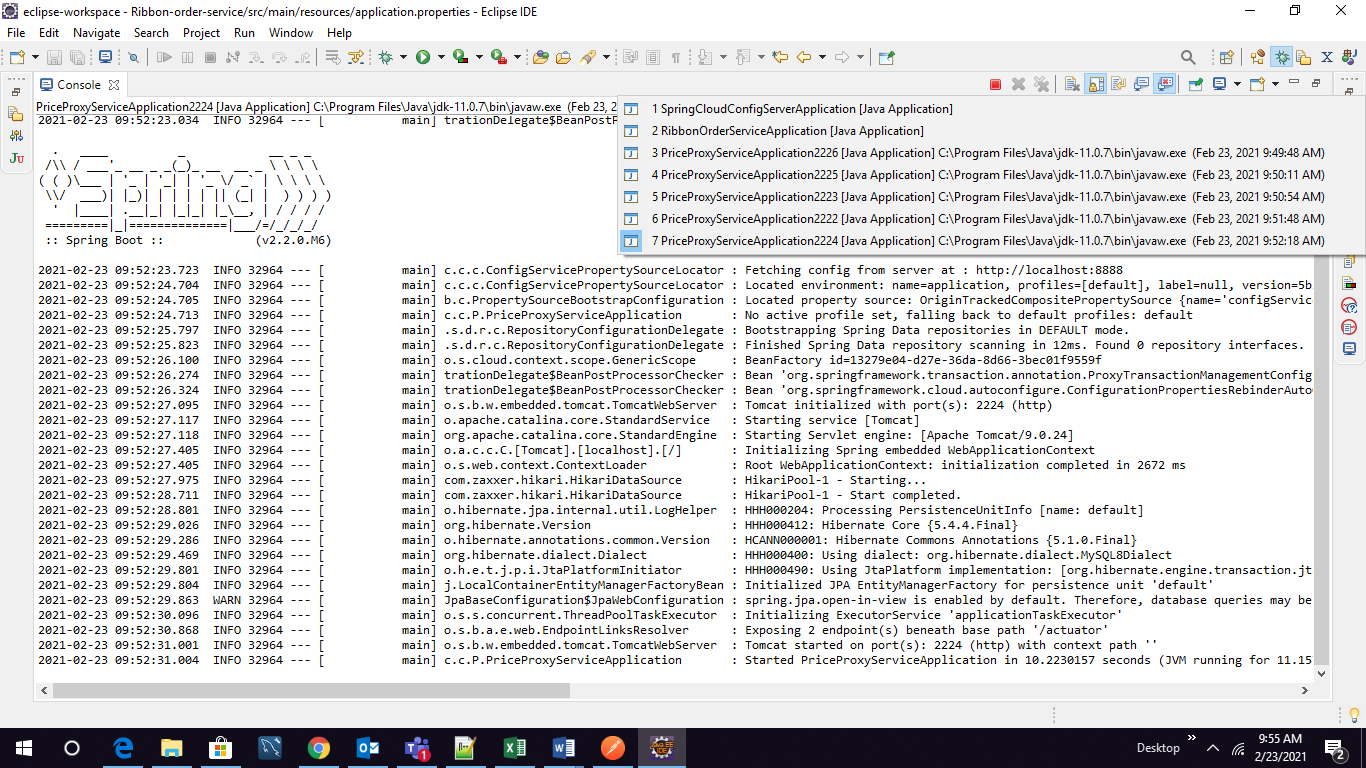


**Part-3 – Load Balancing with Ribbon**



* All the instances of prices service are running.
* Order service can call any of the running instances which means the load is divided among the all running instance rather than the singleinstance**.**

**Below is the attached screenshot for all the running instances.**



**Database Screenshot**

