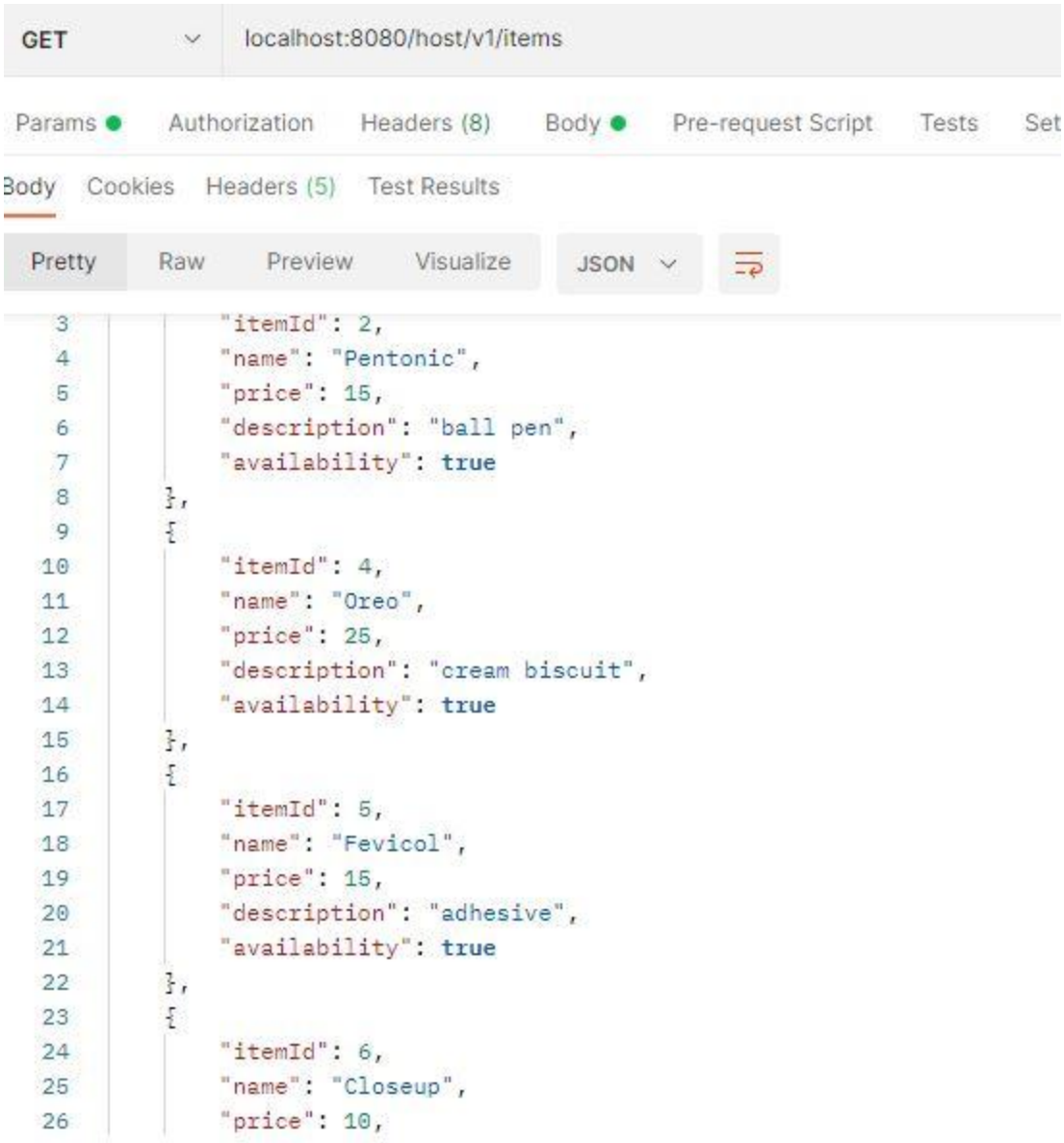
G2G Backend Task

Description

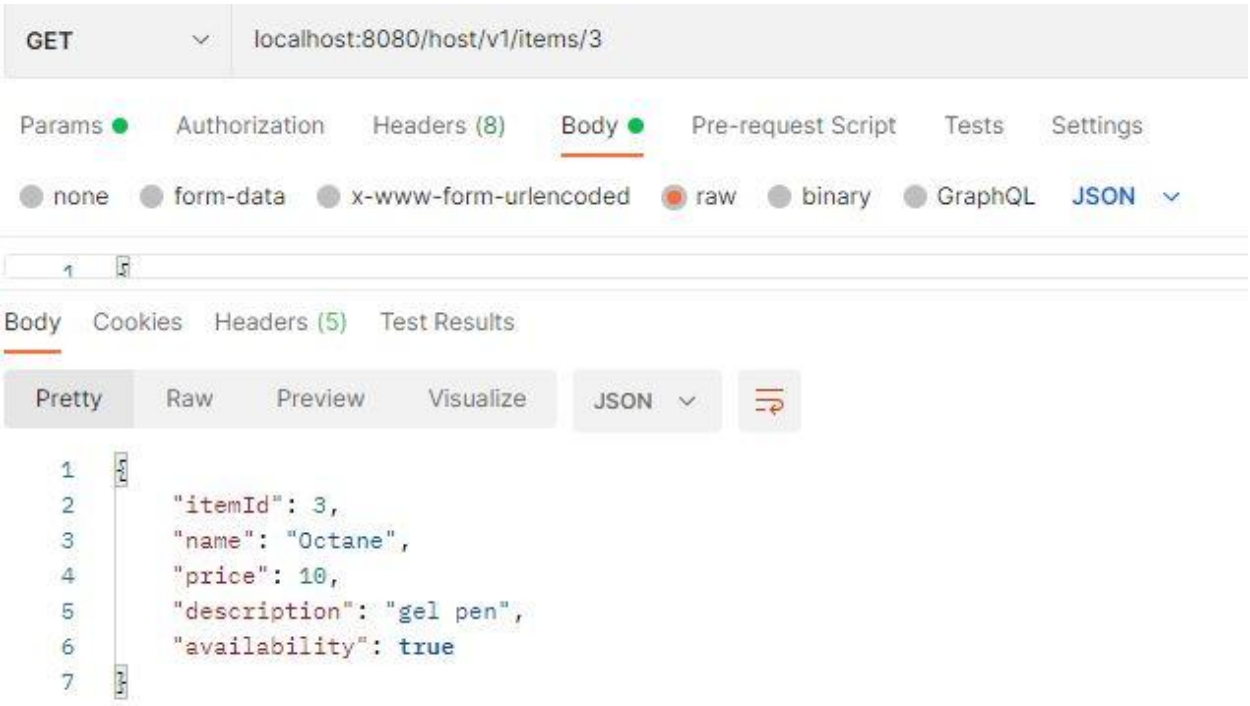
Spring Boot project with postgreSQL. It is an api for fetching, sorting, filtering, modifying and deleting items. It also has a pagination feature.
The api is tested using Postman and the screenshots are attached.

GET api

1. localhost:8080/host/v1/items – lists all items

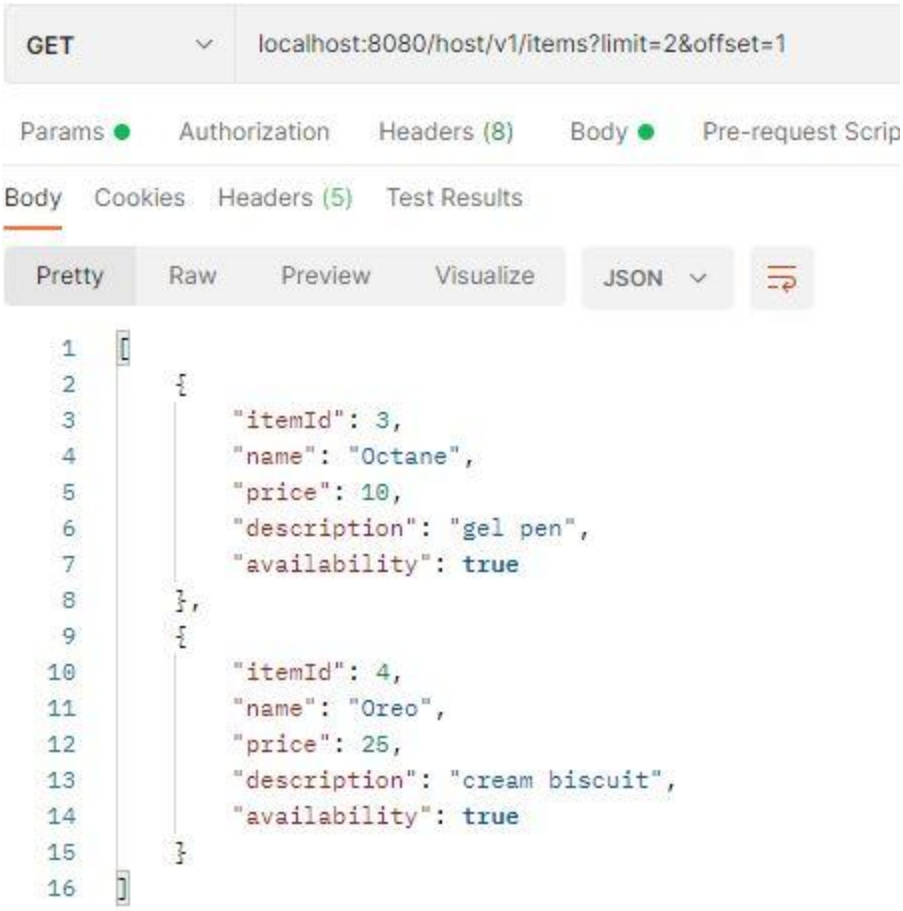


2. localhost:8080/host/v1/items/id/{id} – gets item by id



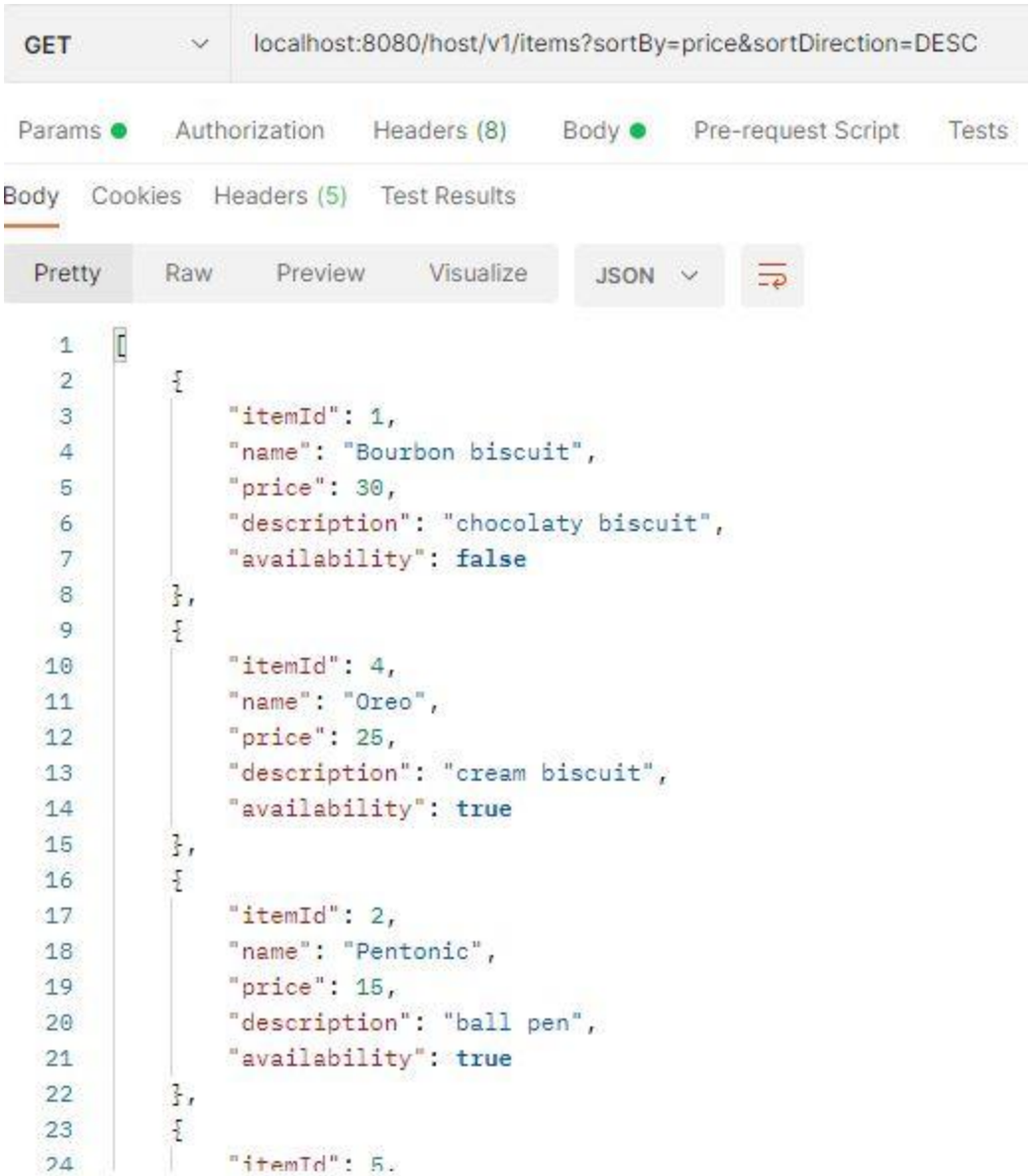
3. localhost:8080/host/v1/items?limit = x & offset = y – pagination with limit x and offset y

   Param : limit field and offset field (Integer)



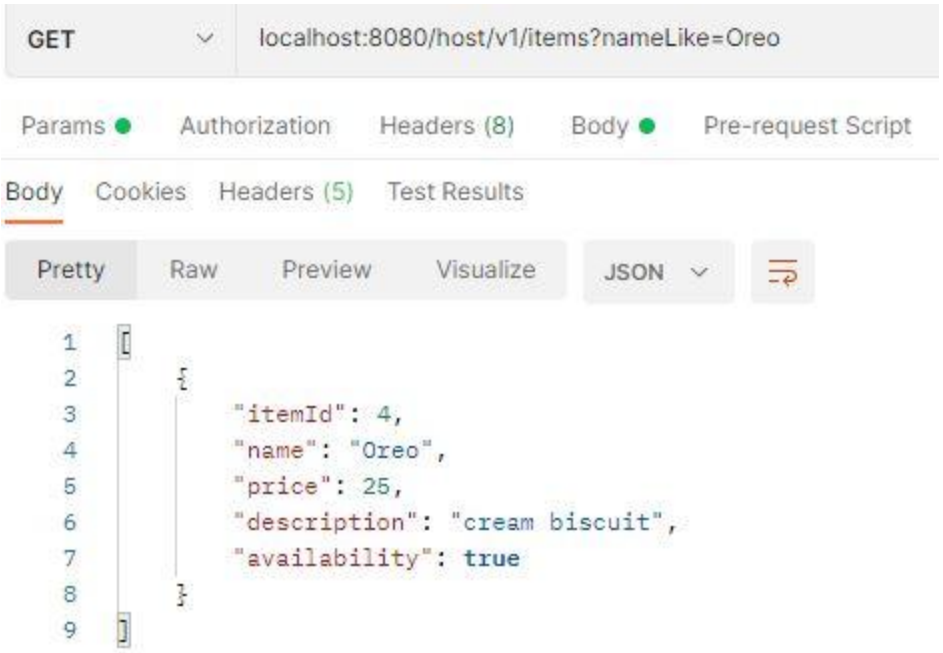
4. localhost:8080/host/v1/items?sortBy = x & sortByDirection = y — sorts field x in direction y (ASC or DESC)

   Param : sortBy field and sortByDirection field (strings, sortByDirection can only take ASC or DESC values)



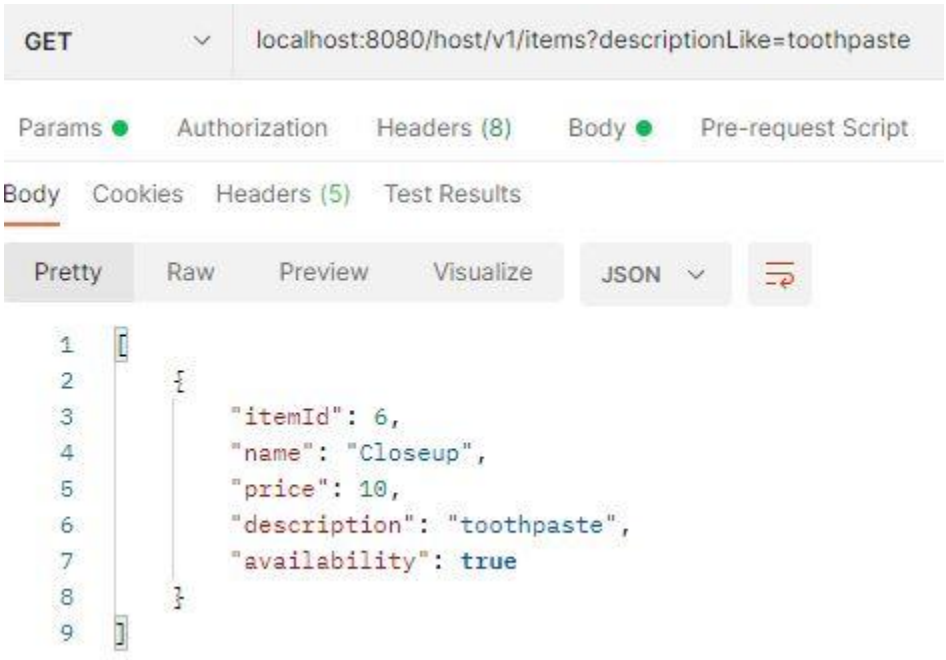
5. localhost:8080/host/v1/items?nameLike = x — filter name entries by x

   Param : nameLike (string)



6. localhost:8080/host/v1/items?descriptionLike = x — filter description entries by x

   Param : descriptionLike (string)

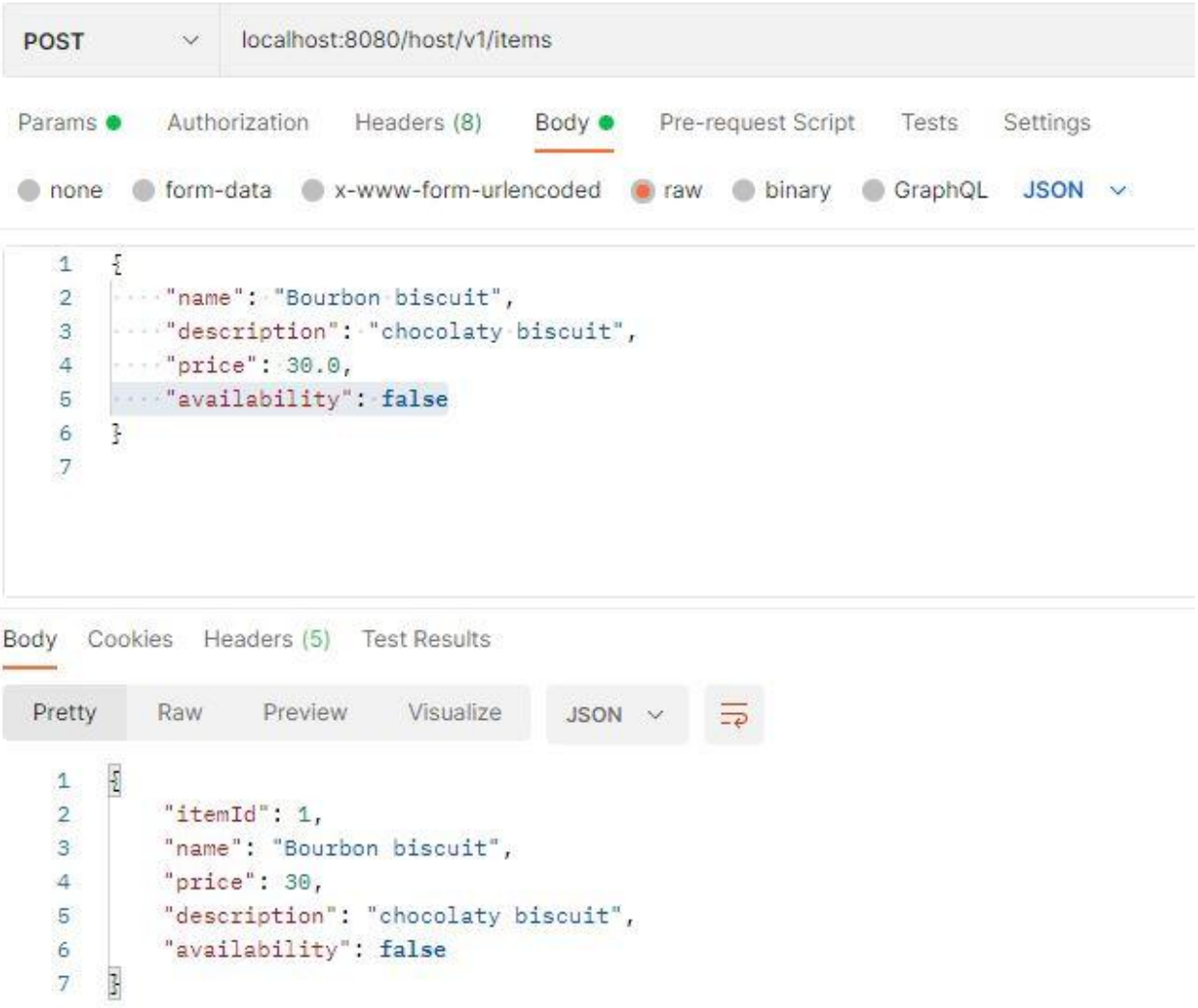


POST api

1. localhost:8080/host/v1/items — posts a new payload

   Param : payload

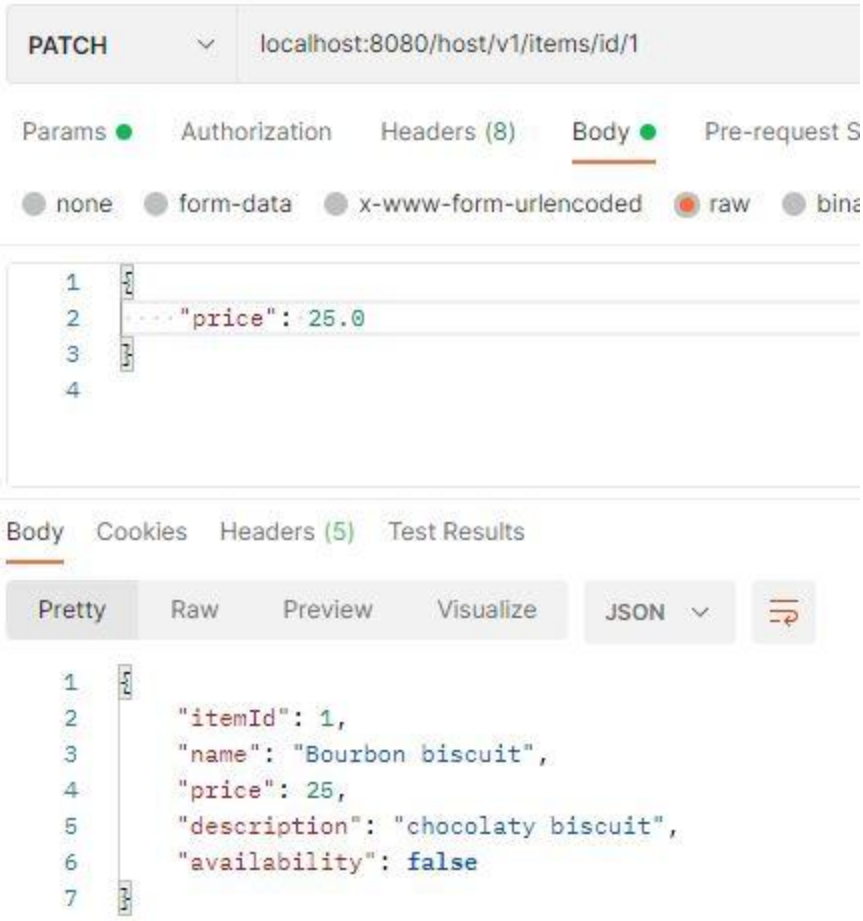
   Payload format :
   {
   “name” : String,
   “description” : String,
   “price”: long,
   “availability” : boolean
   }



PATCH api

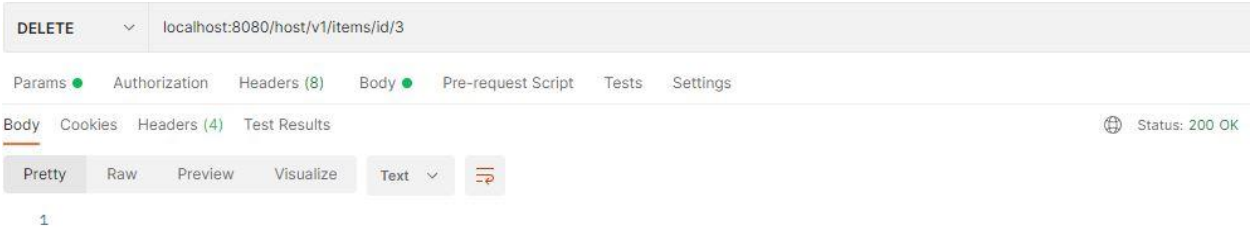
1. localhost:8080/host/v1/items/id/{id} — modifies specific field according to id

   Param : field to modify (String - name, price, description; long - price; availability - boolean)



Delete api

1. localhost:8080/host/v1/items/id/{id} — deletes entry according to id



Features

1. All apis work and return 200 OK for all responses.
2. Data persists in DB using JPA.
3. Meaningfully named variables