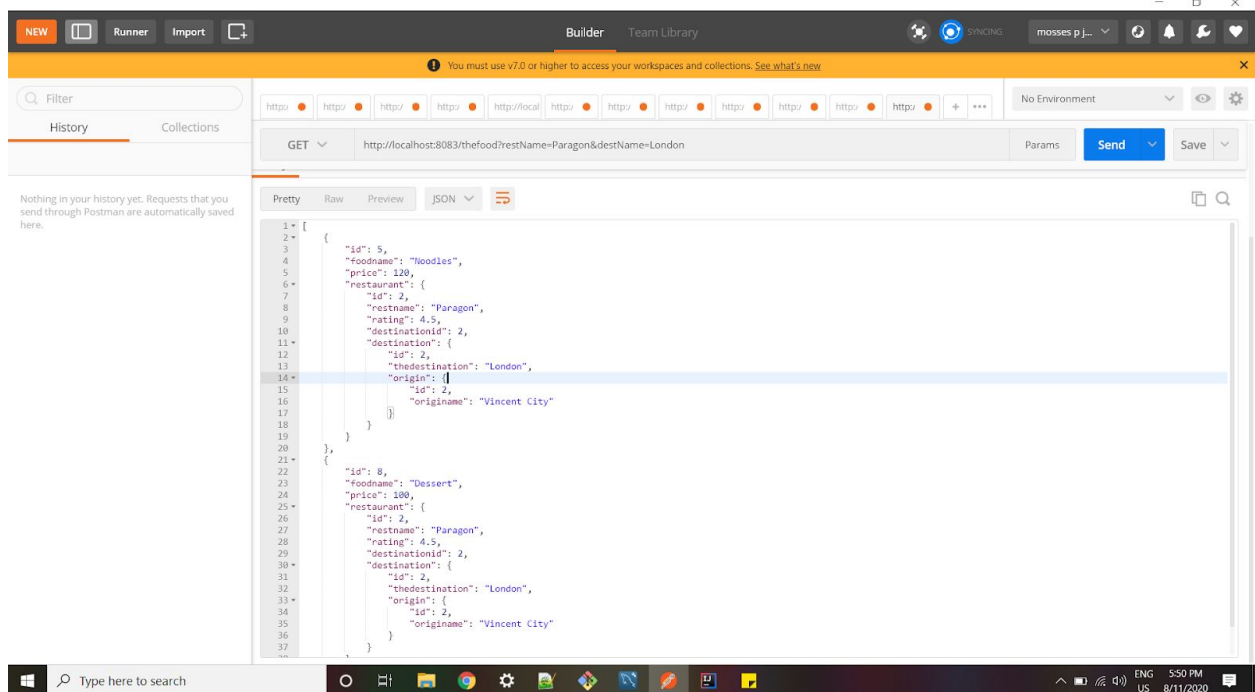


Food Lover APIs

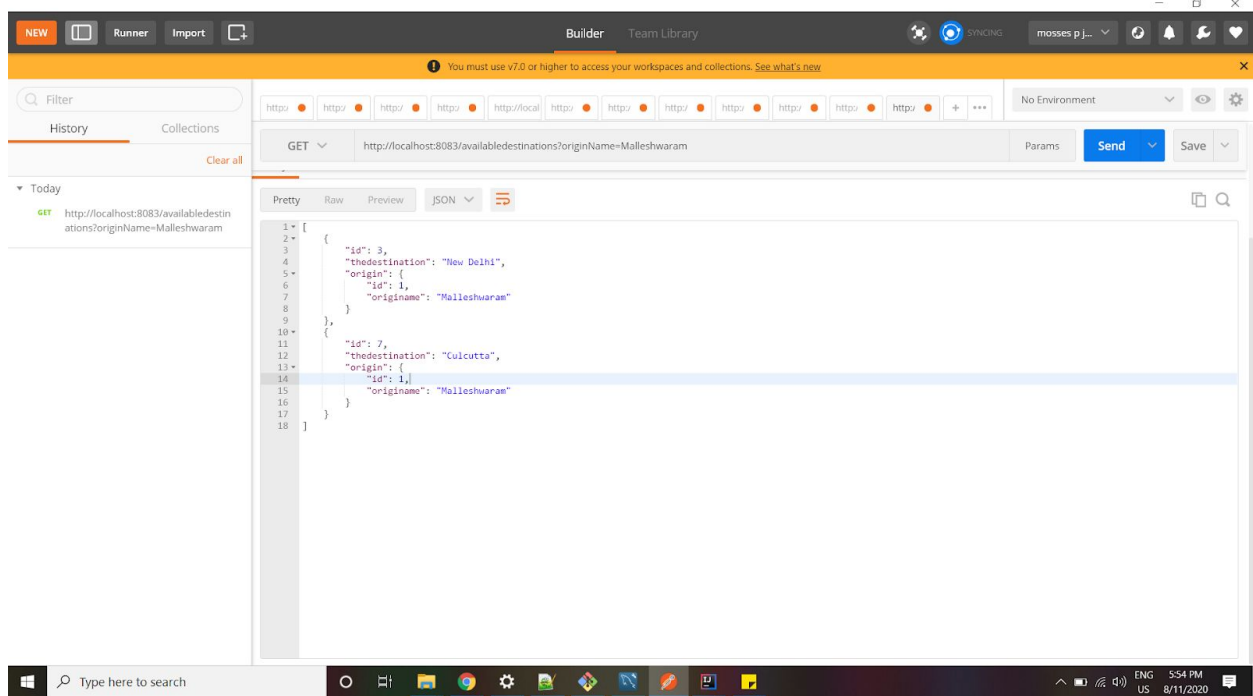
The test data have added with the .sql file

1. Users should be able to browse the food available at the given restaurants and the desired destination, prepopulate the data for the search to be done.

`http://localhost:8083/thefood?restName=Paragon&destName=London`



2. API to read the data from the database to display for the desired destination from a given origin



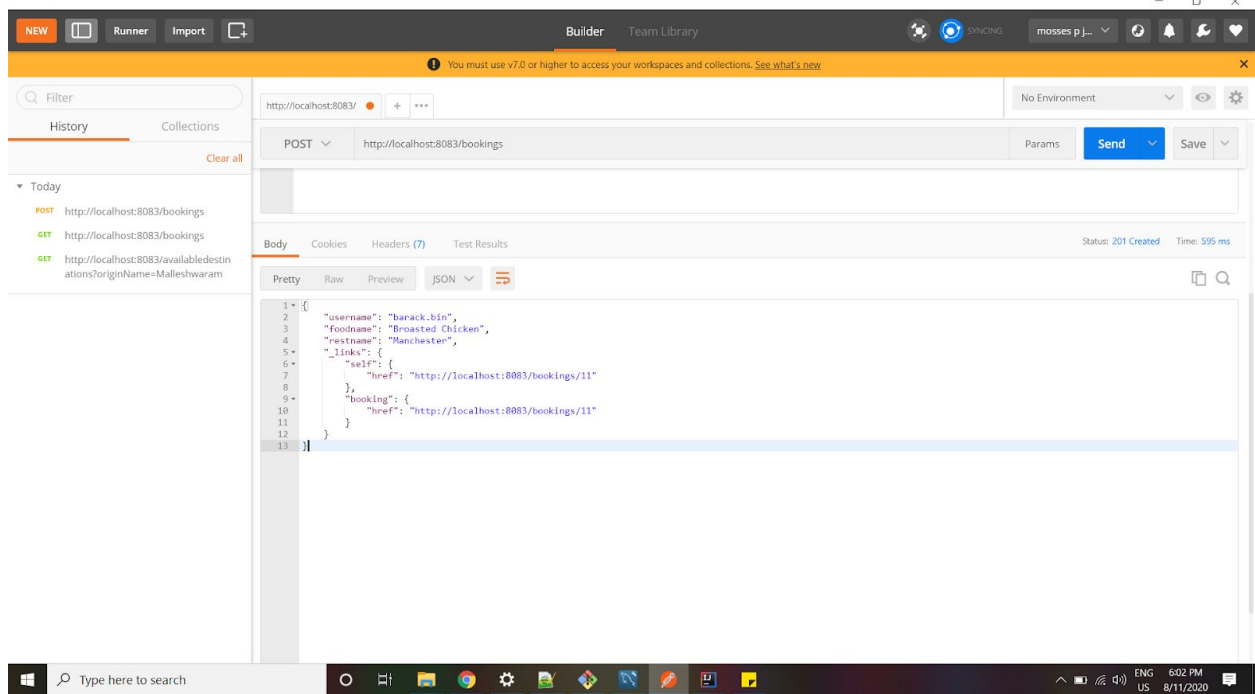
3. CURD operation to do the food bookings

a. Create bookings

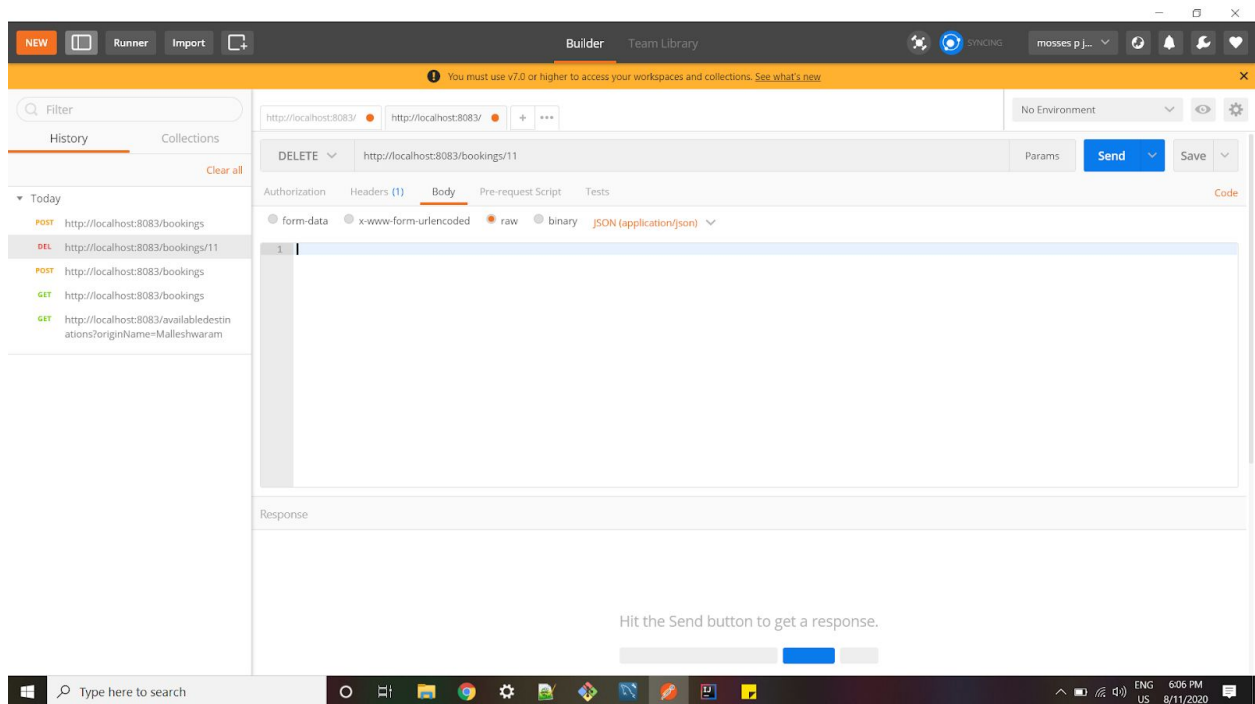
API: Post Method: `http://localhost:8083/bookings`

Payload:

```
{
  "username": "barack.bin",
  "foodname": "Broasted Chicken",
  "restname": "Manchester"
}
```



API: DeleteMethod: <http://localhost:8083/bookings/{id}>



	id	username	foodname	restname
▶	1	mosses.john	Noodles	Paragon
	2	Tony.louis	Chilli Chicken	Manchester
	3	mama.mia	Dosa	WaterFall
	4	mama.mia	Veg Biryani	WaterFall
	5	mosses.john	Broasted Chicken	Manchester
	6	mosses.john	Dragon Chicken	China Town
	7	Tony.louis	Desi Biryani	WaterFall
	8	mama.mia	Dessert	Paragon
	10	mosses.john	Noodles	Paragon
	12	barack.bin	Broasted Chicken	Manchester
✱	NULL	NULL	NULL	NULL

API: putMethod: <http://localhost:8083/bookings/{id}>

Payload:

Payload:

```
{
  "username": "hello.there",
  "foodname": "Broasted Chicken",
  "restname": "Manchester"
}
```

PUT http://localhost:8083/bookings/12

Params Send

```
1 {
2   "username": "hello.there",
3   "foodname": "Broasted Chicken",
4   "restname": "Manchester"
5 }
6
```

Body Cookies Headers (7) Test Results Status: 200 OK Time

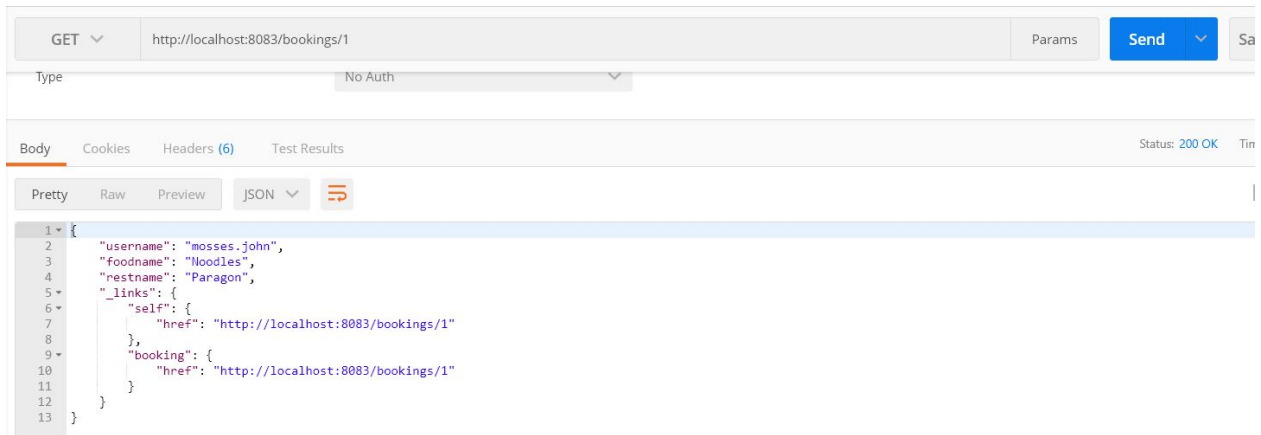
Pretty Raw Preview JSON

```
1 {
2   "username": "hello.there",
3   "foodname": "Broasted Chicken",
4   "restname": "Manchester",
5   "_links": {
6     "self": {
7       "href": "http://localhost:8083/bookings/12"
8     },
9     "booking": {
10      "href": "http://localhost:8083/bookings/12"
11    }
12  }
13 }
```

Result Grid Filter Rows: Edit:

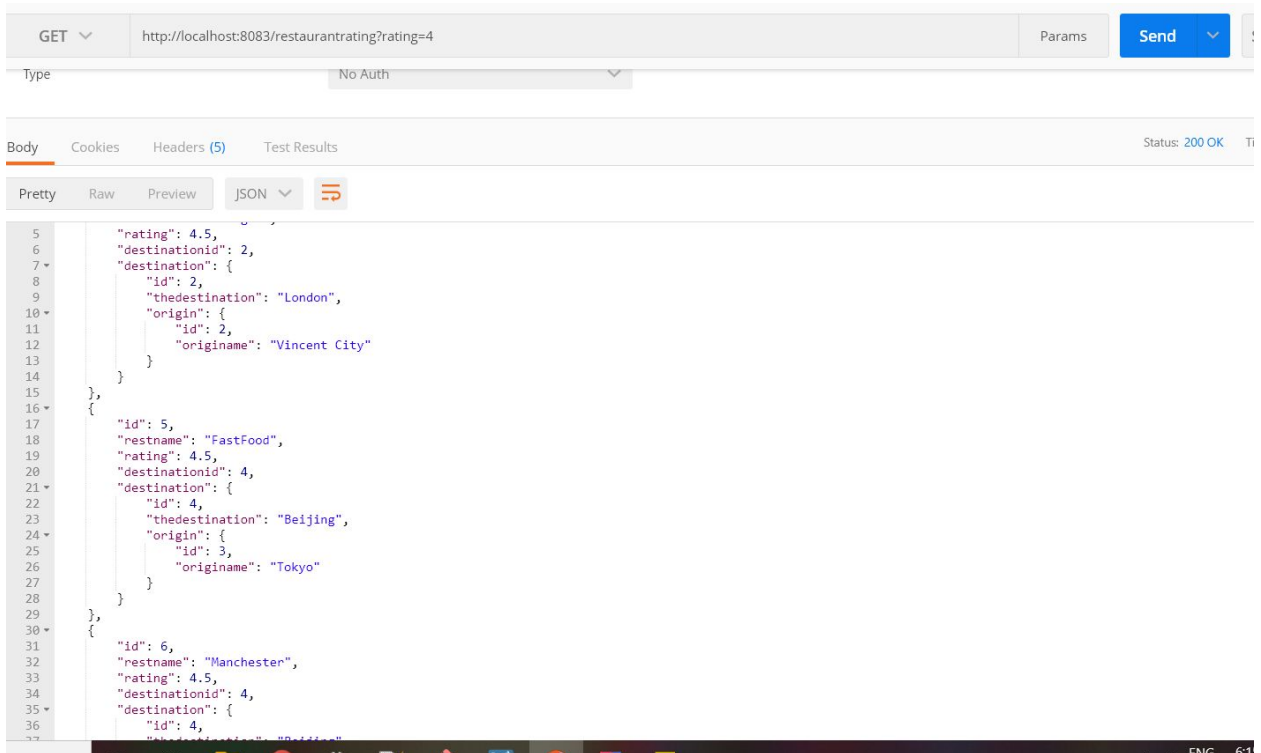
	id	username	foodname	restname
▶	1	mosses.john	Noodles	Paragon
	2	Tony.louis	Chilli Chicken	Manchester
	3	mama.mia	Dosa	WaterFall
	4	mama.mia	Veg Biryani	WaterFall
	5	mosses.john	Broasted Chicken	Manchester
	6	mosses.john	Dragon Chicken	China Town
	7	Tony.louis	Desi Biryani	WaterFall
	8	mama.mia	Dessert	Paragon
	10	mosses.john	Noodles	Paragon
	12	hello.there	Broasted Chicken	Manchester
*	NULL	NULL	NULL	NULL

API: getMethod: <http://localhost:8083/bookings/{id}>



4. Include search criteria to have the restaurant rating and price

API: getMethod: <http://localhost:8083/restaurantrating?rating=4>



5. Include search according to the price

API GetMethod : <http://localhost:8083/foods/search/findByPriceGreaterThan?price=100>

The screenshot displays a web browser's developer tools interface. The top section shows a GET request to the URL `http://localhost:8083/foods/search/findByPriceGreaterThan?price=100`. The status is 200 OK and the time taken is 127ms. The response body is shown in JSON format, displaying a list of food items with their names, prices, and links to their details.

```
1 {
2   "_embedded": {
3     "foods": [
4       {
5         "foodname": "Chilli Chicken",
6         "price": 345,
7         "_links": {
8           "self": {
9             "href": "http://localhost:8083/foods/1"
10          },
11          "food": {
12            "href": "http://localhost:8083/foods/1"
13          },
14          "restaurant": {
15            "href": "http://localhost:8083/foods/1/restaurant"
16          }
17        }
18      },
19      {
20        "foodname": "Noodles",
21        "price": 120,
22        "_links": {
23          "self": {
24            "href": "http://localhost:8083/foods/5"
25          },
26          "food": {
27            "href": "http://localhost:8083/foods/5"
28          }
29        }
30      }
31    ]
32  }
```

6. Include search for destination wise search

API GetMethod <http://localhost:8083/restaurantdestination?destName=New Delhi>

The screenshot shows a REST client interface with the following details:

- Method:** GET
- URL:** <http://localhost:8083/restaurantdestination?destName=New Delhi>
- Params:** (empty)
- Send:** (button)
- Save:** (button)
- Type:** No Auth
- Body:** (selected tab)
- Headers (5):** (tab)
- Test Results:** (tab)
- Status:** 200 OK
- Time:** 63 ms
- Format:** JSON
- Response Body (Pretty):**

```
6   "destinationid": 3,
7   "destination": {
8     "id": 3,
9     "thdestination": "New Delhi",
10    "origin": {
11      "id": 1,
12      "originname": "Malleshwaram"
13    }
14  },
15 },
16 {
17   "id": 4,
18   "restname": "Liverpool",
19   "rating": 2.5,
20   "destinationid": 3,
21   "destination": {
22     "id": 3,
23     "thdestination": "New Delhi",
24     "origin": {
25       "id": 1,
26       "originname": "Malleshwaram"
27     }
28   }
29 },
30 {
31   "id": 7,
32   "restname": "Paragon",
33   "rating": 4.5,
34   "destinationid": 3,
35   "destination": {
36     "id": 3,
37     "thdestination": "New Delhi",
38     "origin": {
39       "id": 1,
40       "originname": "Malleshwaram"
41     }
42   }
43 }
```

7. API to summarize the data of their bookings with prize

Prize Distribution:

If the time is between 2 pm to 4 pm and the total purchase price is greater than 250, 35% discount is given

If the time is between 2 pm to 4 pm 20% discount is given

If the time is not between 2 pm to 4 pm no prizes or discount has been given

API: **getMethod:** <http://localhost:8083/bookingdetails?username=Tony.louis>

A total summary is displayed as per the users

GET

http://localhost:8083/restaurantrating?rating=4

Params

Send

TypeNo Auth

BodyCookiesHeaders (5)Test ResultsStatus: 200 OK

PrettyRawPreviewJSON

5

6

7

8

9

10

11

12

13

14

15

16

17

18

19

20

21

22

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

```
"rating": 4.5,
"destinationid": 2,
"destination": {
  "id": 2,
  "thdestination": "London",
  "origin": {
    "id": 2,
    "originname": "Vincent City"
  }
},
{
  "id": 5,
  "restname": "FastFood",
  "rating": 4.5,
  "destinationid": 4,
  "destination": {
    "id": 4,
    "thdestination": "Beijing",
    "origin": {
      "id": 3,
      "originname": "Tokyo"
    }
  }
},
{
  "id": 6,
  "restname": "Manchester",
  "rating": 4.5,
  "destinationid": 4,
  "destination": {
    "id": 4,
```

ENG64