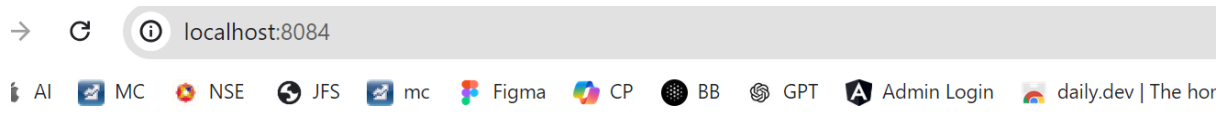


Outputs :

## 1)JSP FULL STACK:

\*This index.jsp page is to enter task details



### Create Your Task

Enter the Task Title:

Enter the Task Description:

Enter the Due Date:

Priority:

\*This is to check our task details after submitting



### Task Created Successfully

Task Details:

Title: Task

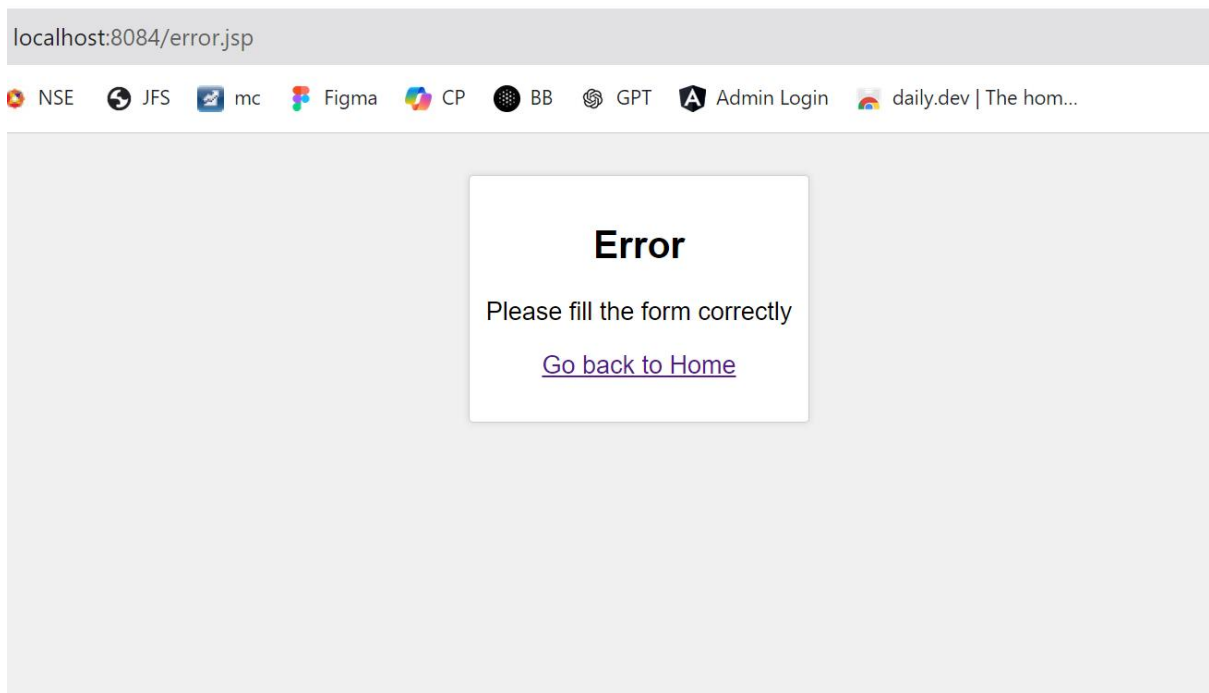
Description: This is Task Description

Due Date: 2024-07-20

Priority: Medium

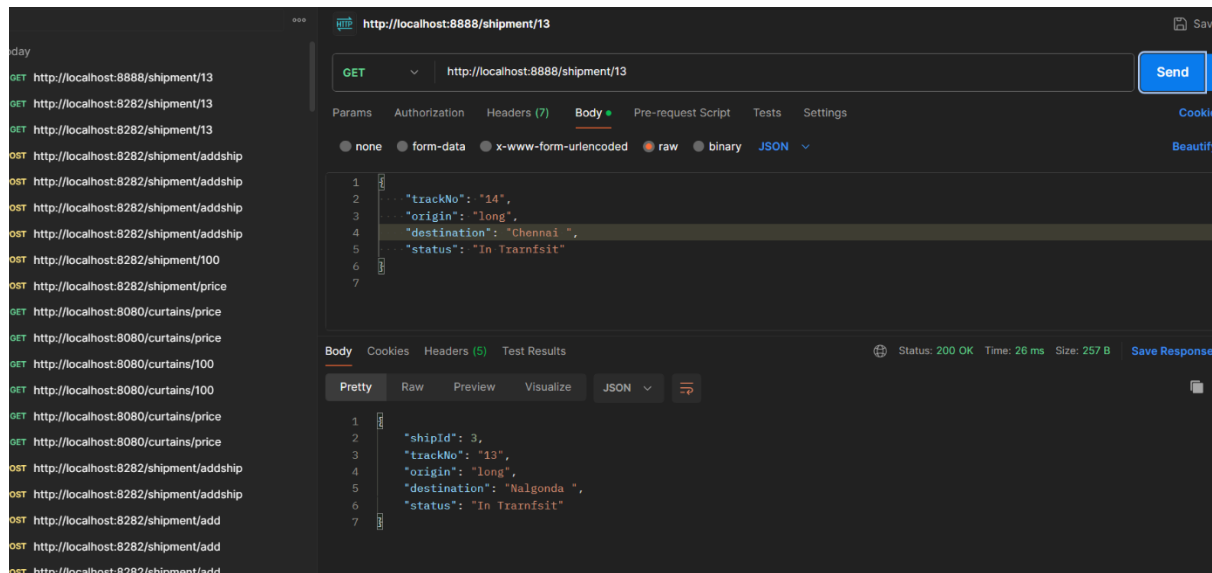
[Back to Create Task](#)

\*If any details are not entered it will redirect to error.jsp

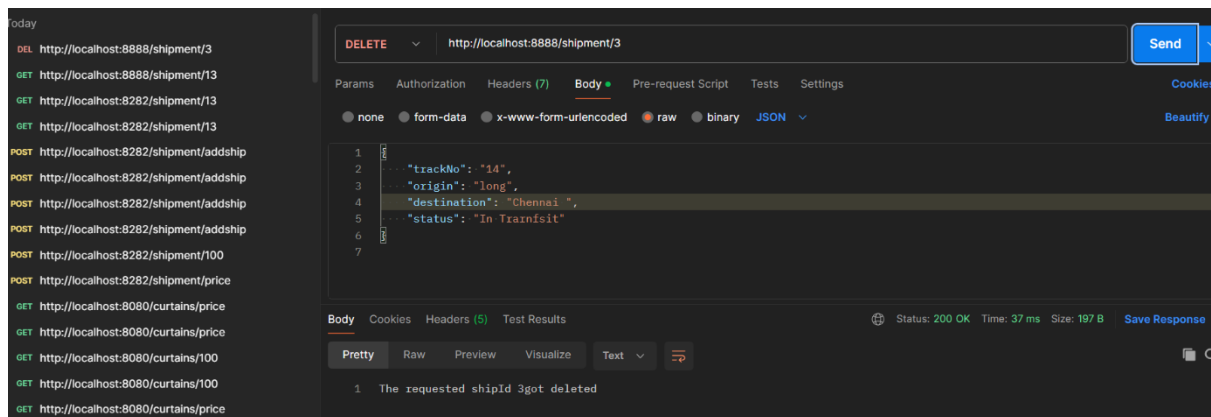


## 2)SpringBoot Full Stack:

\*getting shipment details with trackno :



\*Deleting shipment details with shipId



3)Spring JPA: Description Implement the JPQL statement to fetch the list of curtains whose price is greater than the given price and the list of curtains whose brand is equal to the given brand name.

```
Database Changed
mysql> select * from curtain;
+----+-----+-----+-----+-----+
| id | brand | color | material | price |
+----+-----+-----+-----+-----+
| 1  | Brand1 | Red   | Leather  | 120   |
| 2  | Brand2 | Blue  | Synthetic | 100   |
| 3  | Brand3 | Black | Canvas   | 80    |
| 4  | Brand4 | White | Mesh     | 90    |
| 5  | Brand5 | Green | Leather  | 110   |
+----+-----+-----+-----+-----+
5 rows in set (0.00 sec)
```

GET http://localhost:8080/curtains/price?price=50

Query Params

| Key   | Value |
|-------|-------|
| price | 50    |

Body

Pretty

```
{
  "id": 1,
  "brand": "Brand1",
  "price": 120,
  "color": "Red",
  "material": "Leather"
},
{
  "id": 2,
  "brand": "Brand2",
  "price": 100,
  "color": "Blue",
  "material": "Synthetic"
}
```

Status: 200 OK Time: 27 ms Size: 533 B

GET http://localhost:8080/curtains/brand?brand=Brand1

Query Params

| Key   | Value  |
|-------|--------|
| brand | Brand1 |

Body

Pretty

```
{
  "id": 1,
  "brand": "Brand1",
  "price": 120,
  "color": "Red",
  "material": "Leather"
}
```

Status: 200 OK Time: 15 ms Size: 238 B

EVENT

| EVENT                 | TIME     |
|-----------------------|----------|
| Prepare               | 3.01 ms  |
| Socket Initialization | 0.87 ms  |
| DNS Lookup            | 0.22 ms  |
| TCP Handshake         | 0.35 ms  |
| Transfer Start        | 11.8 ms  |
| Download              | 1.04 ms  |
| Process               | 0.05 ms  |
| Total                 | 17.31 ms |

```

<terminated> SelectEx [Java Application] C:\Users\saina\A
Price that are greater than 150 are
2      BrandB
3      BrandC

Brands that are
1      BrandA      Red      Cotton

```

## 4)Seasonal Activity Organizer

demo [boot] [devtools]  
 demo-22 [boot] [devtools]  
 JSPFullStack [boot] [devtools]  
 Movie-Booking [boot] [devtools]  
 SeasonalActivities [boot] [devtools]  
 SeasonalActivityOrganizer  
 Security [boot] [devtools]  
 src/main/java  
 src/main/resources  
 src/test/java  
 JRE System Library [JavaSE-17]  
 Maven Dependencies  
 src  
 target  
 HELP.md  
 mvnw  
 mvnw.cmd  
 pom.xml  
 shipment [boot] [devtools]  
 Shipping [boot] [devtools]

```

7 public enum Season {
8     SPRING, SUMMER, AUTUMN, WINTER
9 }
10
11 // Enum for Activities
12 public enum Activity {

```

Console x

```

<terminated> SeasonalActivityOrganizer [Java Application] C:\Users\saina\AppData\Local\Programs\E
[HIKING, SWIMMING, SKIING]
[HIKING]
[SWIMMING]
[SKIING]
[]

```