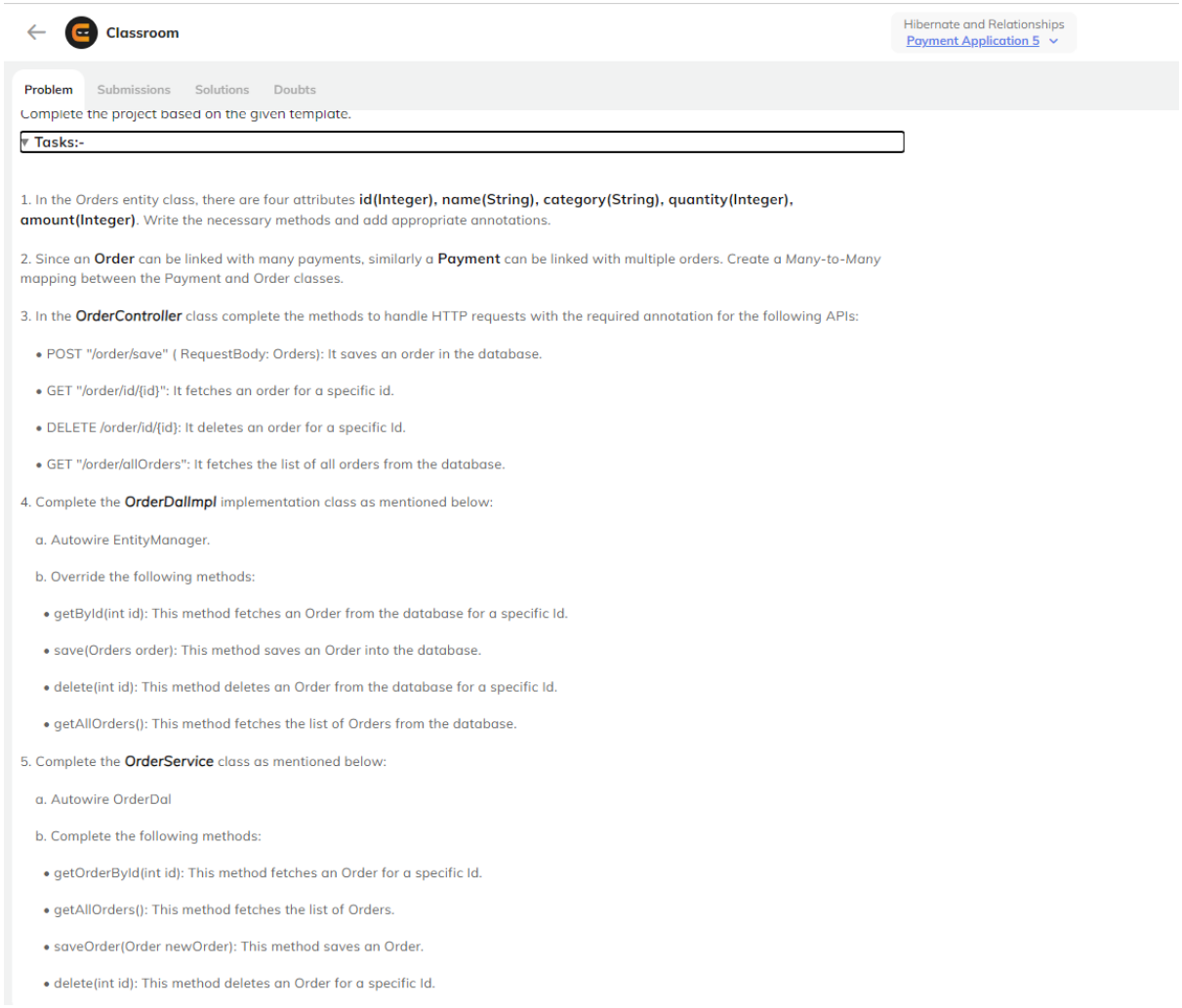
# Payment Application V

**Problem Statement:**



Classroom

Hibernate and Relationships
Payment Application 5

Problem | Submissions | Solutions | Doubts

Complete the project based on the given template.

**Tasks:-**

1. In the Orders entity class, there are four attributes **id(Integer), name(String), category(String), quantity(Integer), amount(Integer)**. Write the necessary methods and add appropriate annotations.

2. Since an **Order** can be linked with many payments, similarly a **Payment** can be linked with multiple orders. Create a *Many-to-Many* mapping between the Payment and Order classes.

3. In the **OrderController** class complete the methods to handle HTTP requests with the required annotation for the following APIs:

- POST "/order/save" ( RequestBody: Orders): It saves an order in the database.
- GET "/order/id/{id}": It fetches an order for a specific id.
- DELETE /order/id/{id}: It deletes an order for a specific Id.
- GET "/order/allOrders": It fetches the list of all orders from the database.

4. Complete the **OrderDaoImpl** implementation class as mentioned below:

a. Autowire EntityManager.

b. Override the following methods:

- getById(int id): This method fetches an Order from the database for a specific Id.
- save(Orders order): This method saves an Order into the database.
- delete(int id): This method deletes an Order from the database for a specific Id.
- getAllOrders(): This method fetches the list of Orders from the database.

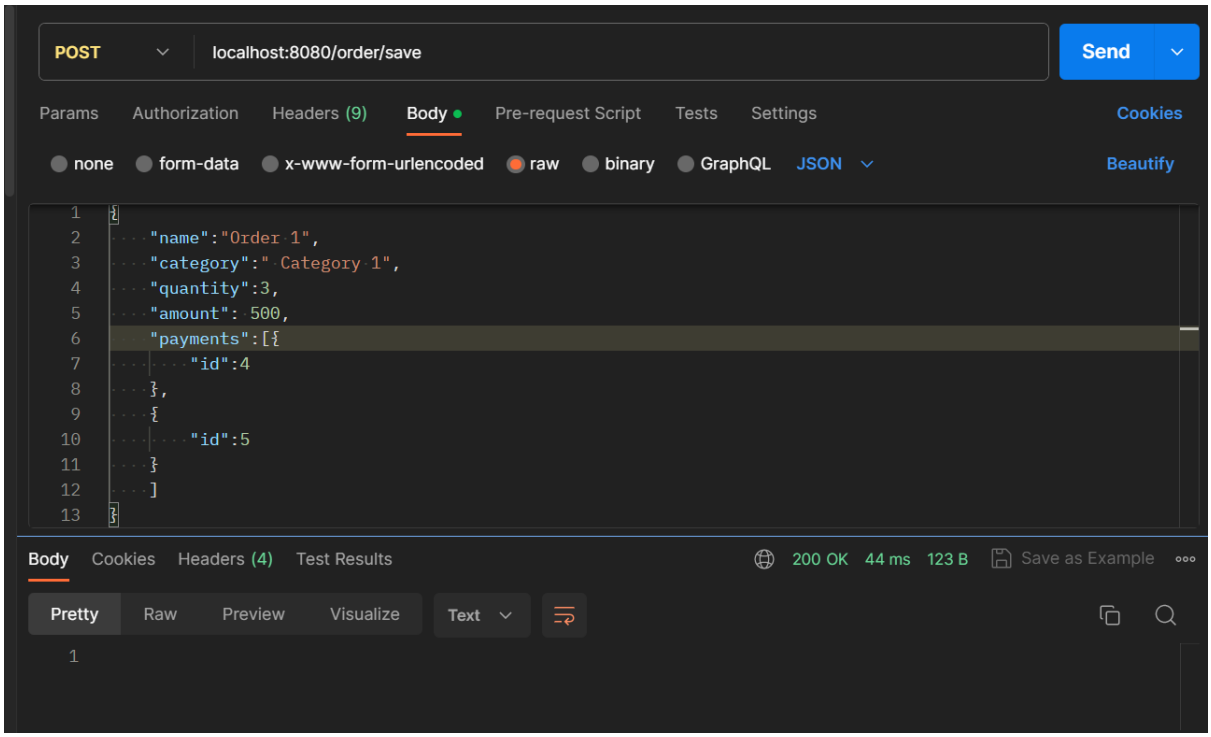
5. Complete the **OrderService** class as mentioned below:

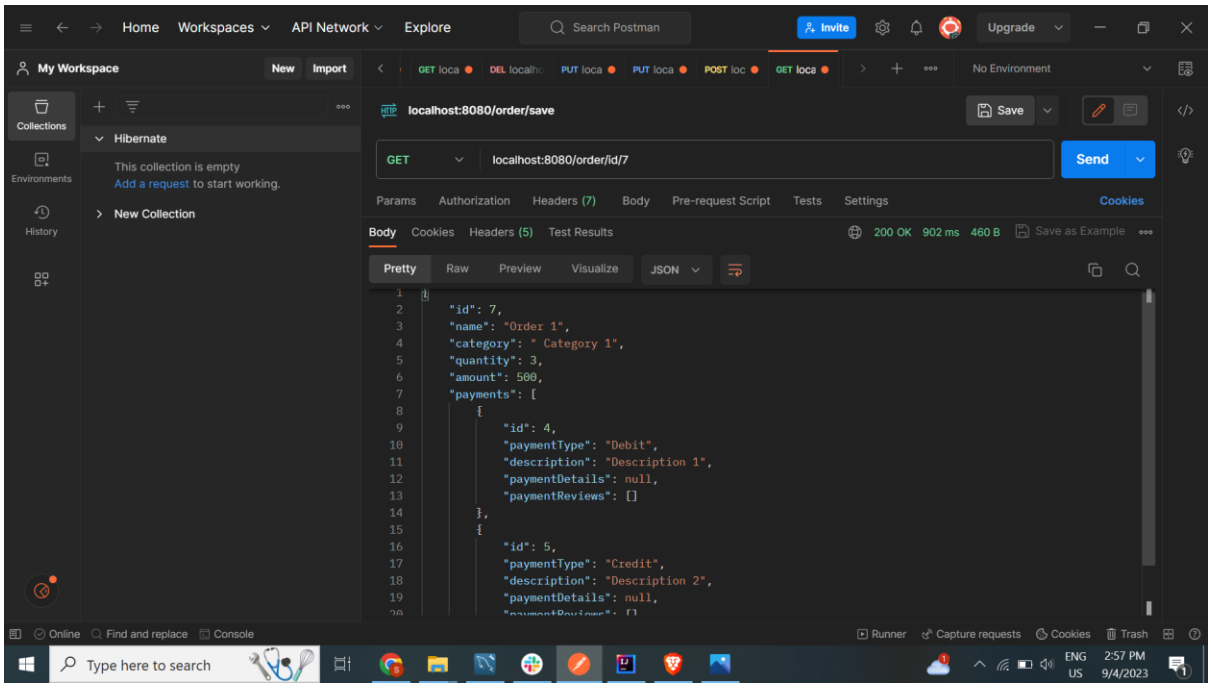
a. Autowire OrderDao

b. Complete the following methods:

- getOrderById(int id): This method fetches an Order for a specific Id.
- getAllOrders(): This method fetches the list of Orders.
- saveOrder(Order newOrder): This method saves an Order.
- delete(int id): This method deletes an Order for a specific Id.

**Output:**


POST
localhost:8080/order/save
Send
Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
Settings
Cookies
none
form-data
x-www-form-urlencoded
raw
binary
GraphQL
JSON
Beautify
{
"name":"Order 1",
"category":" Category 1",
"quantity":3,
"amount": 500,
"payments":[{
"id":4
},
{
"id":5
}
]
}
Body
Cookies
Headers (4)
Test Results
200 OK
44 ms
123 B
Save as Example
Pretty
Raw
Preview
Visualize
Text
1


Home
Workspaces
API Network
Explore
Search Postman
Invite
Upgrade
My Workspace
New
Import
Collections
Environments
History
Hibernate
This collection is empty
Add a request to start working.
New Collection
localhost:8080/order/save
Save
GET
localhost:8080/order/id/7
Send
Params
Authorization
Headers (7)
Body
Pre-request Script
Tests
Settings
Cookies
Body
Cookies
Headers (5)
Test Results
200 OK
902 ms
460 B
Save as Example
Pretty
Raw
Preview
Visualize
JSON
{
"id": 7,
"name": "Order 1",
"category": " Category 1",
"quantity": 3,
"amount": 500,
"payments": [
{
"id": 4,
"paymentType": "Debit",
"description": "Description 1",
"paymentDetails": null,
"paymentReviews": []
},
{
"id": 5,
"paymentType": "Credit",
"description": "Description 2",
"paymentDetails": null,
Online
Find and replace
Console
Runner
Capture requests
Cookies
Trash
Type here to search
ENG
US
2:57 PM
9/4/2023