Code is present in <https://github.com/KruthikaSwaminathan/CustRewardCode.git>

Code consists of Controller layer(Rest API),Service layer(Business logic),Repository layer for data access and Entities.

**Controller Layer**: This layer handles Restful APIs and also handles HTTP request and response.It maps endpoints to service methods and return appropriate responses.

RewardsController class manages endpoints to get total rewards for three months. We mention the class as @RestController.Since we want to retrieve the data we use @Getmapping.

It also contains post, put and delete mapping for adding,editing and deleting the transaction respectively.

**Service Layer**: This layer contains business logic and communicates with repository layer to take the data from DB and applying it on the logic

RewardsService class contains the business logic to calculate the rewards and total of last three months rewards.

It contains logic to add/edit and delete the transaction

**Repository Layer**: This layer handles all the database interactions.This layer helps in storage, retrieval, update, delete and search operation on objects

CustomerRepository gives the storage and retrieval operations for Customer

CustomerTransactionRespository gives the storage and retrieval operations for CustomerTransaction entity

**Entity Layer:**This layer represents the structure of Database table.It provides mapping between columns and java objects

Customer and CustomerTransaction entity represent details of customer and the transaction and it specifies the columns and tables.

In Customer. java Model class we have object for customer and total rewards

Pom.xml has dependencies for the code like web dependency,Jpa dependency

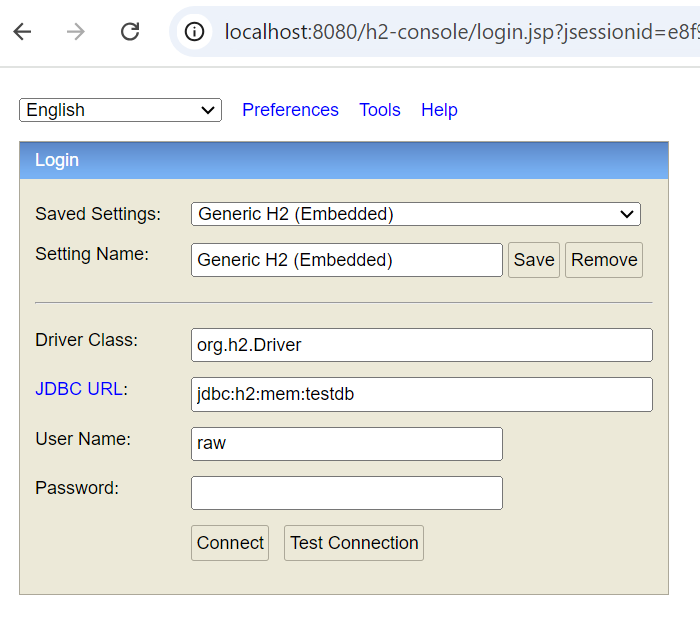
In application.properties username password and database details of it.

There are two database tables CUST(customer) and CUST\_TRANS(Customer Transaction)

CUST table consist of customer id,name and address

CUST\_TRANS table consist of customer id,transction id,spent details, date and amount

Once we run the spring boot application ,login into <http://localhost:8080/h2-console>



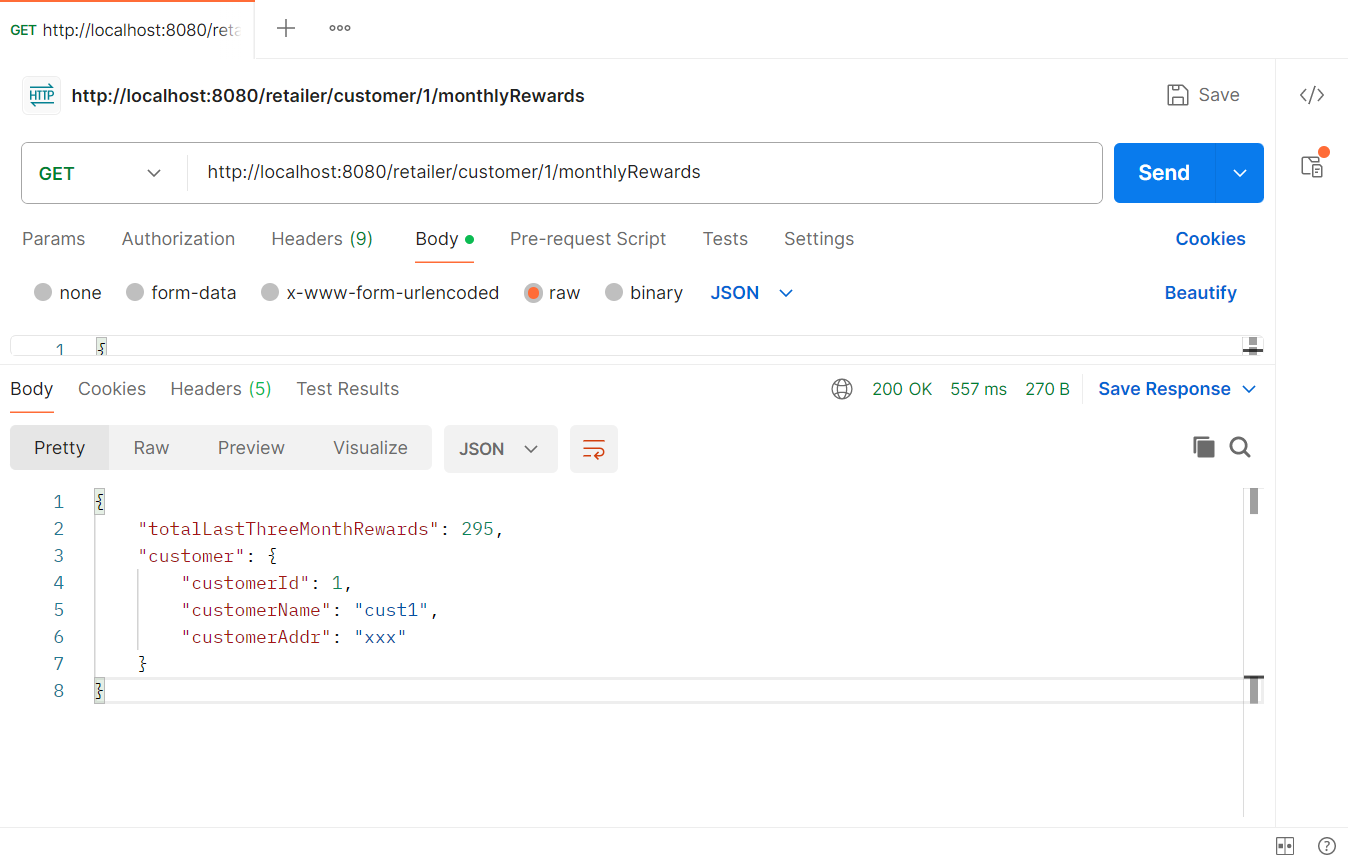
Here mention Driver Class, JDBC URL ,Username and password from application.properties

Once login is done check whether tables are present and execute the insert script from script.sql

Then go to postman and hit the URL

**to get the total reward points for last three months for particular customer id**

URL: <http://localhost:8080/retailer/customer/1/monthlyRewards>



**To add a transaction**

<http://localhost:8080/retailer/customer/addTransaction/1> (Post method)

In body of postman {

"customerId":1,

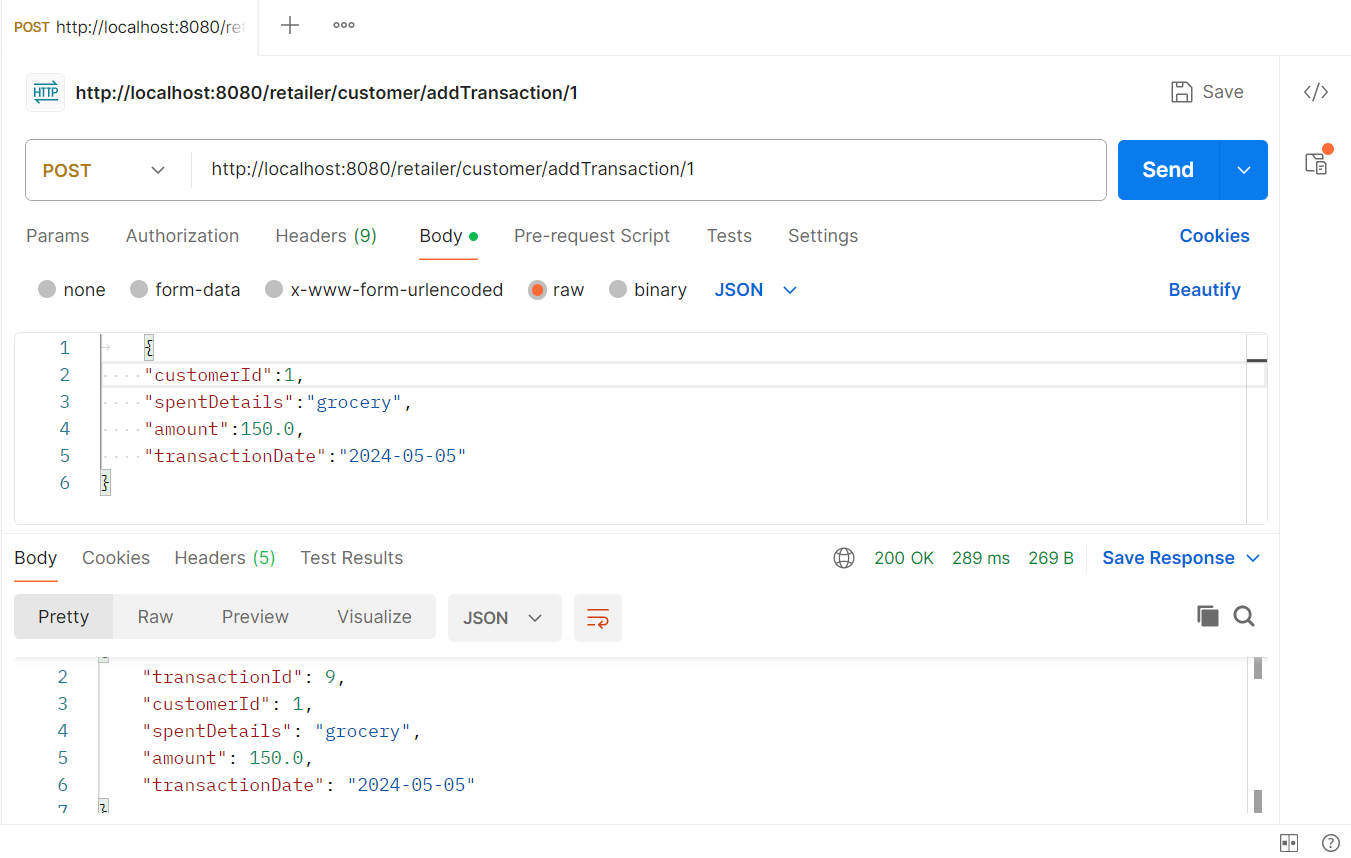
"spentDetails":"grocery",

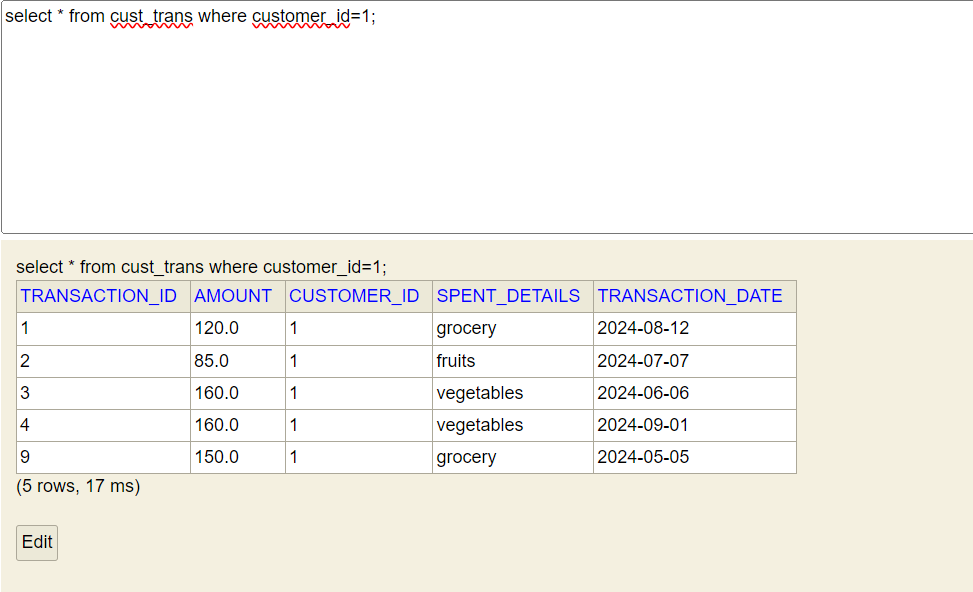
"amount":150.0,

"transactionDate":"2024-05-05"

}

The transaction is inserted in table





**To edit the transaction**

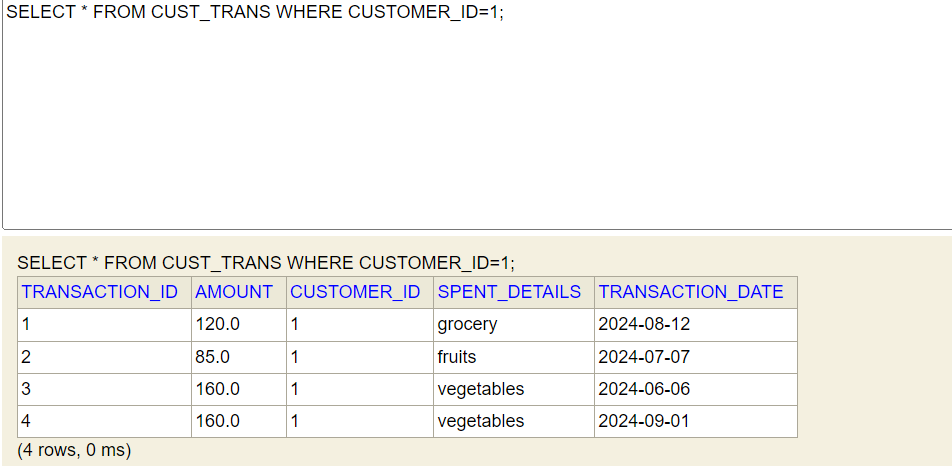
<http://localhost:8080/retailer/customer/editTransaction/1> (PUT method)

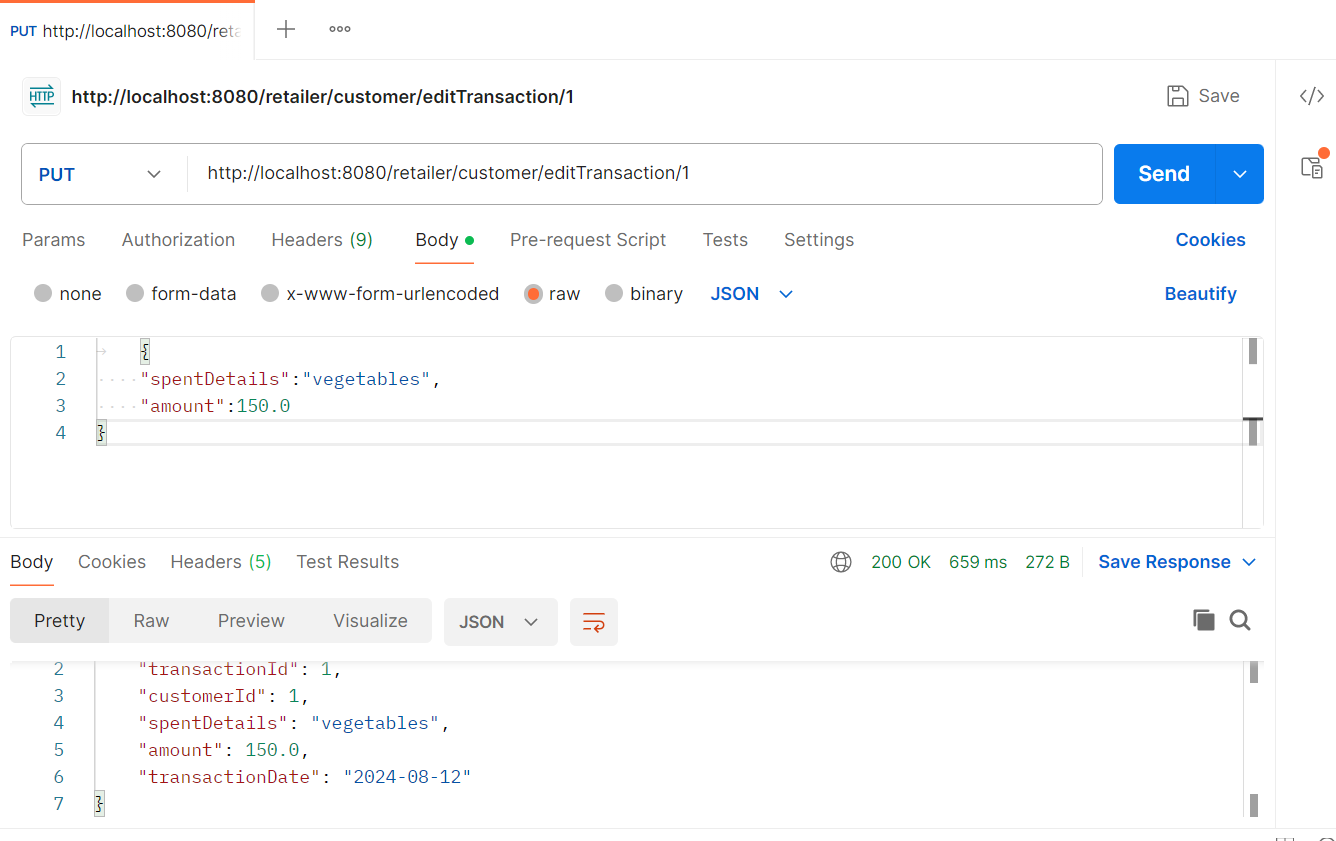
In body of postman {

"spentDetails":"vegetables",

"amount":150.0 }

Before editing the database entries are:





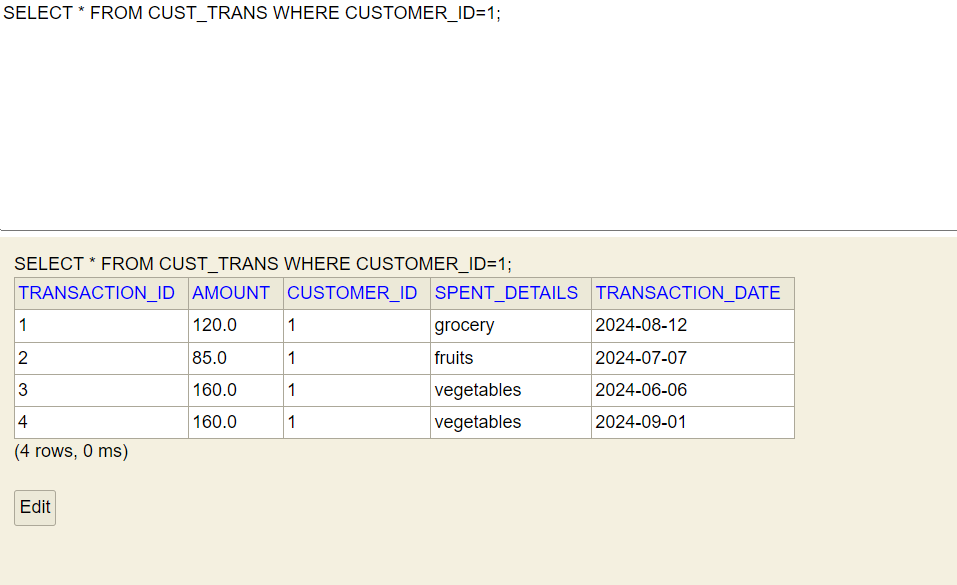
After editing the database entries:

A screenshot of a computer

Description automatically generated

**To delete the transaction**

<http://localhost:8080/retailer/customer/deleteTransaction/1>



A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

If the customer is not found the below message will be displayed.

