

# **FXTrading user-guide**

# FxTrading Application Backend(SpringBoot)

To see list of options available call this Api

Api link: <http://localhost:8082/index>

The screenshot shows the Postman application interface. At the top, there are tabs for Home, Workspaces, API Network, and Explore. A search bar is at the top right. Below the header, a list of requests is shown, with the first one being a GET request to <http://localhost:8082/index>. The request details panel shows the method (GET), URL, and various configuration tabs like Params, Authorization, Headers, Body, Pre-request Script, Tests, and Settings. The Body tab is selected, displaying the following JSON response:

```
1 To book Trade click this link http://localhost:8082/booktrade
2 To Print Trade click this link http://localhost:8082/printtrade
3 To Exit click this link http://localhost:8082/exit
4
```

At the bottom of the interface, there are various status indicators and a footer with icons and text.

# To Book a Trade Call this API

(note : this API only sets status as processing you need to confirm or cancel the trade)

Api link: <http://localhost:8082/booktrade> (method as POST)

The screenshot shows the Postman interface with a POST request to <http://localhost:8082/booktrade>. The request body is set to JSON and contains the following data:

```
1 {"customerName": "Hameed",  
2 "currencyPair": "USDINR",  
3 "amount": 100}
```

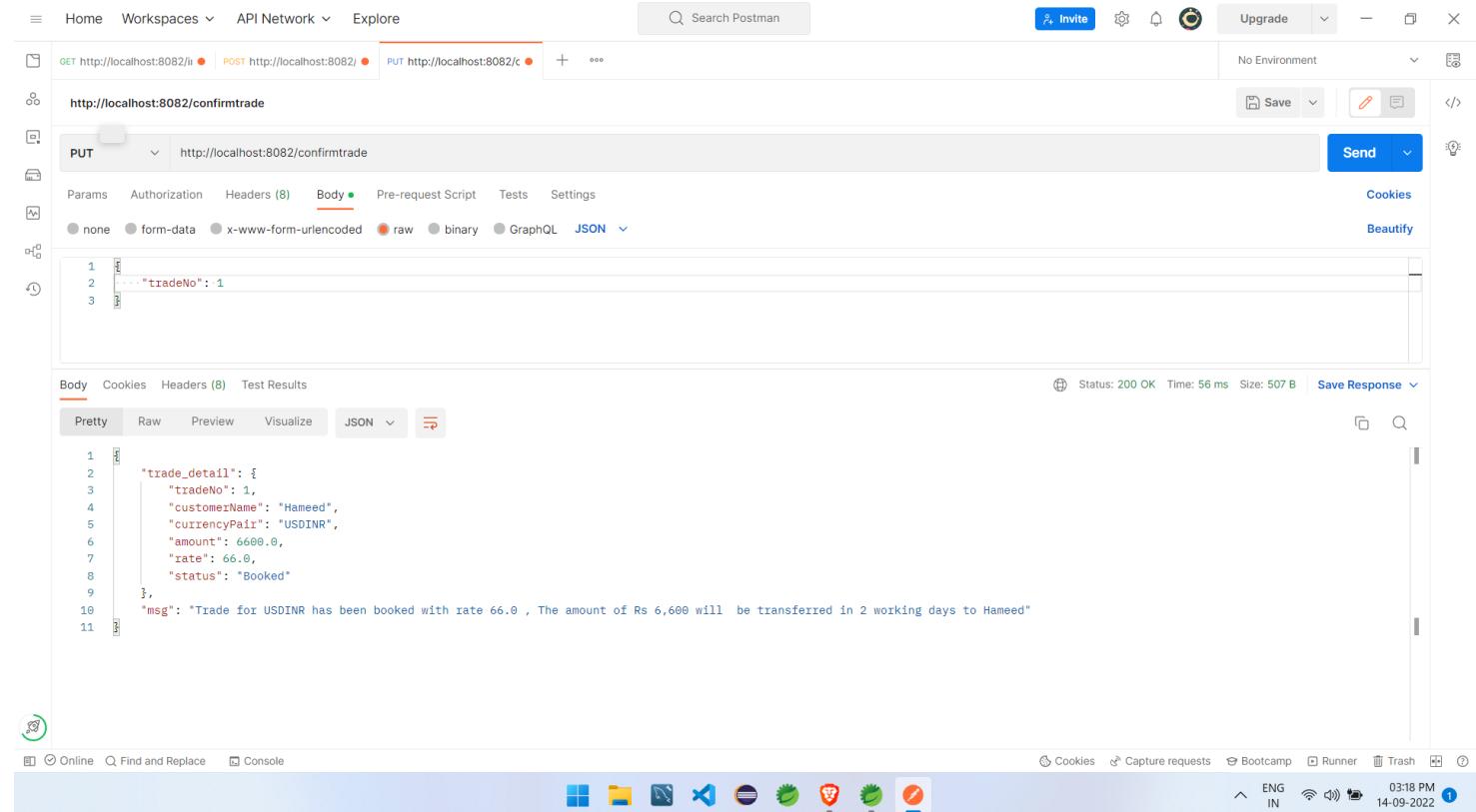
The response status is 202 Accepted with a time of 318 ms and a size of 561 B. The response body is also in JSON format:

```
1 {  
2   "trade": {  
3     "tradeNo": 1,  
4     "customerName": "Hameed",  
5     "currencyPair": "USDINR",  
6     "amount": 6600.0,  
7     "rate": 66.0,  
8     "status": "processing"  
9   },  
10  "Print Rate Click here": "http://localhost:8082/printrate",  
11  "Book Trade Click here": "http://localhost:8082/confirmtrade",  
12  "Cancel Trade Click here": "http://localhost:8082/canceltrade"  
13 }
```

## To Confirm the trade call this API

API link: <http://localhost:8082/confirmtrade> (Method as PUT)

For confirm the trade we need to enter the tradeno (Enter the tradeNo displayed in bookTrade API)



The screenshot shows the Postman interface with a PUT request to `http://localhost:8082/confirmtrade`. The request body is a JSON object with a single key-value pair: `"tradeNo": 1`. The response status is 200 OK, and the response body is a JSON object:

```
1 "trade_detail": {  
2   "tradeNo": 1,  
3   "customerName": "Hameed",  
4   "currencyPair": "USDINR",  
5   "amount": 6600.0,  
6   "rate": 66.0,  
7   "status": "Booked"  
8 },  
9 "  
10 "msg": "Trade for USDINR has been booked with rate 66.0 , The amount of Rs 6,600 will be transferred in 2 working days to Hameed"  
11 "
```

## To cancel the trade call this API

API link: <http://localhost:8082/canceltrade> (Method as DELETE)

For cancelling the trade we also need to enter the TradeNo.

The screenshot shows the Postman application interface. At the top, there are tabs for Home, Workspaces, API Network, and Explore. A search bar is located at the top right. Below the tabs, a list of API endpoints is shown: GET http://localhost:8082/ii, POST http://localhost:8082/j, PUT http://localhost:8082/c, and a selected DELETE endpoint: http://localhost:8082/c. The main workspace shows a DELETE request to http://localhost:8082/canceltrade. The 'Body' tab is selected, showing a JSON payload: 

```
1 {"tradeNo": 2}
```

. Other tabs like Params, Authorization, Headers, and Settings are visible but not selected. Below the body, the response status is shown as 200 OK with a size of 272 B. The bottom of the screen shows the Windows taskbar with various pinned icons and system status indicators.

## To see the rate call this API

API link: <http://localhost:8082/printrate>

The screenshot shows the Postman application interface. At the top, there are tabs for Home, Workspaces, API Network, and Explore. A search bar says "Search Postman". On the right, there are buttons for Invite, Upgrade, and various settings. Below the header, a sidebar lists "POST http://localhost:8082/" and "GET http://localhost:8082/f". The main area shows a "GET" request to "http://localhost:8082/printrate". Under "Params", there is a "Query Params" table:

	KEY	VALUE	DESCRIPTION	...	Bulk Edit
	Key	Value	Description		

Below the table, there are tabs for Body, Cookies, Headers (8), and Test Results. The "Test Results" tab is selected, showing a status of 200 OK, time of 14 ms, and size of 295 B. It contains the response body: "1 You are transferring INR 435,600 to finlly". At the bottom, there are various icons for integration with other tools like Jenkins, GitHub, and Docker.

# This application only accepts USDINR and the amount must be above 0 rs

## Testing

The screenshot shows the Postman application interface. At the top, there are navigation tabs: Home, Workspaces, API Network, Explore, and a search bar labeled "Search Postman". On the right side, there are buttons for "Invite", "Upgrade", and "Learn More".

In the main workspace, a request is being made to `http://localhost:8082/booktrade`. The method is set to POST. The "Body" tab is selected, showing the following JSON payload:

```
1 {"customerName": "Hameed",  
2   "currencyPair": "USDINR",  
3   "amount": -100  
4 }  
5 
```

Below the body, the status of the request is shown as "Status: 500 Internal Server Error Time: 15 ms Size: 5.62 KB". The "Pretty" tab is selected in the preview section.

The response body contains a large stack trace from the Tomcat server, indicating that the application is rejecting the negative amount value. The error message in the stack trace is:

```
6     "message": "Cannot accept value -100.0 Enter positive value",  
7     "path": "/booktrade"  
8 }
```

At the bottom of the interface, there are various toolbars and status indicators, including "Online", "Find and Replace", "Console", "Cookies", "Capture requests", "Bootcamp", "Runner", "Trash", and system status like ENG US, 08:42 AM, and 12-09-2022.

# Data Stored in Database

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Navigator

SCHEMAS

- cricket
- employee\_db
- employee\_management\_system
- ems
- finaly\_developerscourse
- fxtradingdb
  - Tables
    - tradedata
  - Views
  - Stored Procedures
  - Functions
- hibernatepractice
- sys
  - Tables
  - Views
  - Stored Procedures
  - Functions

Administration Schemas

Information

Table: tradedata

Columns:

trade_no	bigint	P
amount	double	
currency_pair	varchar	
customer_name	varchar	
rate	float	
status	varchar	

Object Info Session

Query 1 tradedata

```
1 • SELECT * FROM fxtradingdb.tradedata;
```

Result Grid | Filter Rows: | Edit: | Export/Imports: | Wrap Cell Content: |

trade_no	amount	currency_pair	customer_name	rate	status
1	6600	USDINR	Hameed	66	Booked
NULL	NULL	NULL	NULL	NULL	NULL

SQLAdditions

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Result Grid

Form Editor

Field Types

Query Stats

tradedata 1

Output

Action Output

#	Time	Action
1	15:25:04	SELECT * FROM fxtradingdb.tradedata LIMIT 0, 1000

Message

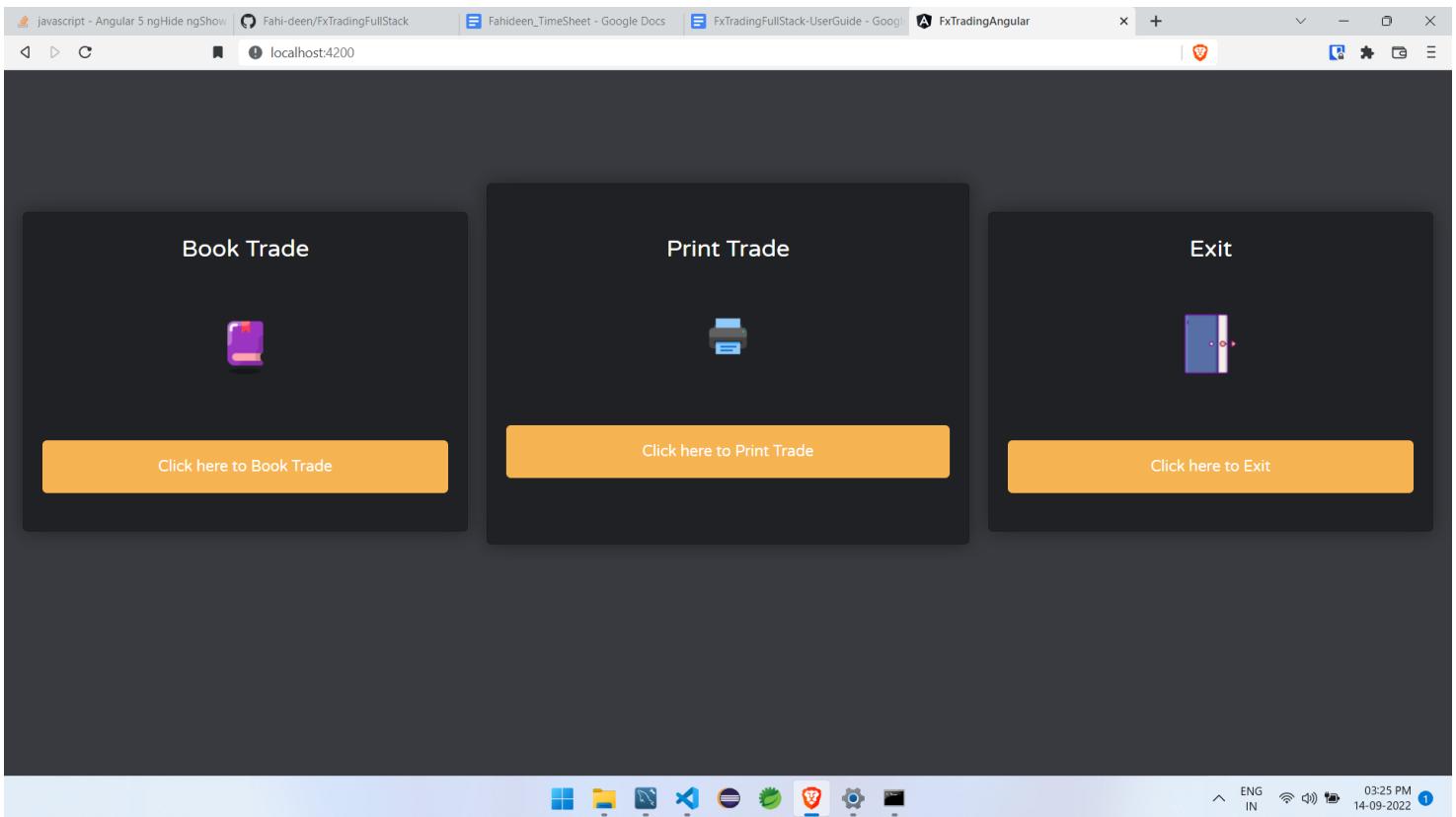
1 row(s) returned

Duration / Fetch

0.000 sec / 0.000 sec

ENG IN 03:25 PM 14-09-2022

# Angular FrontEnd Screenshots



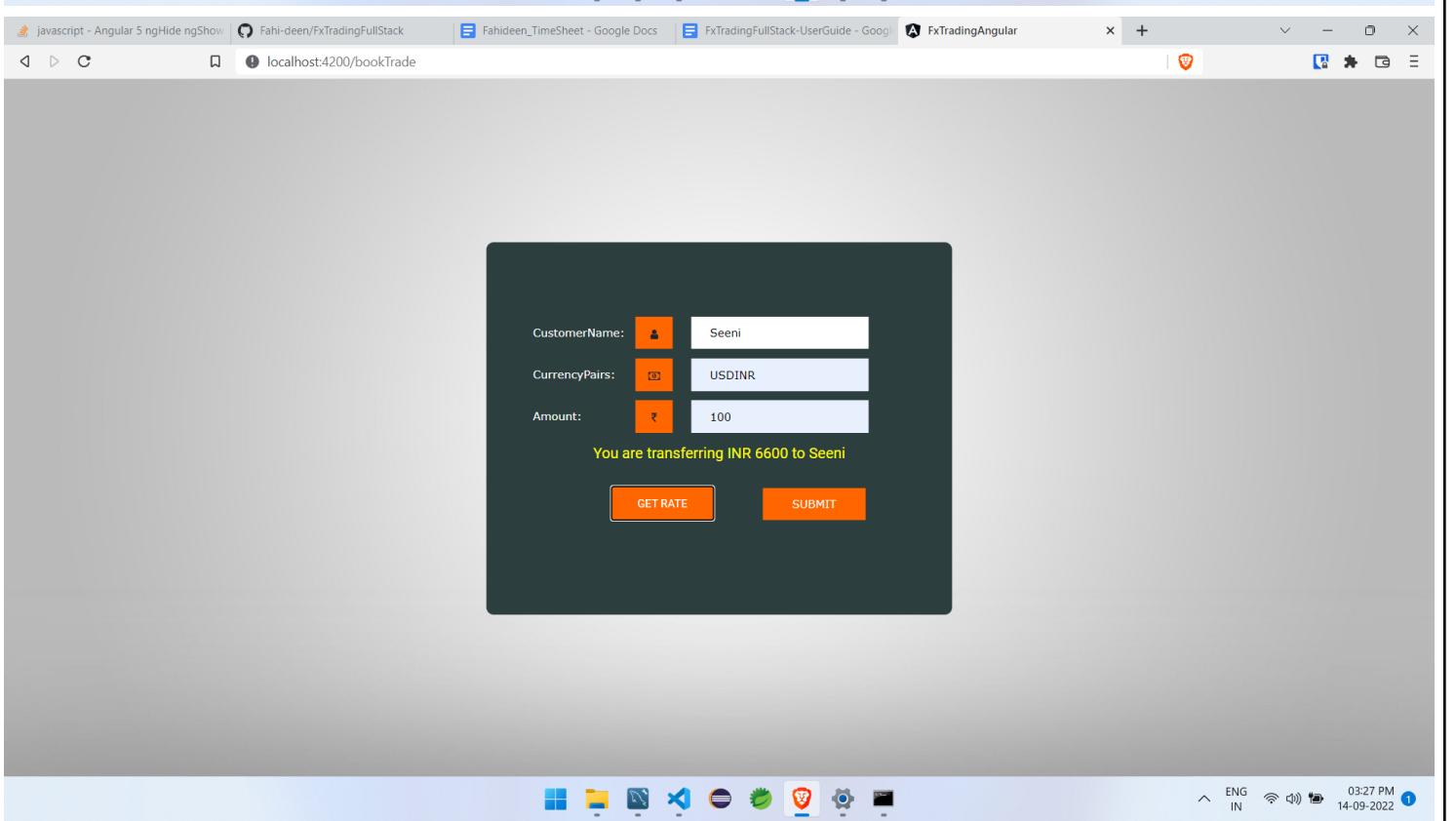
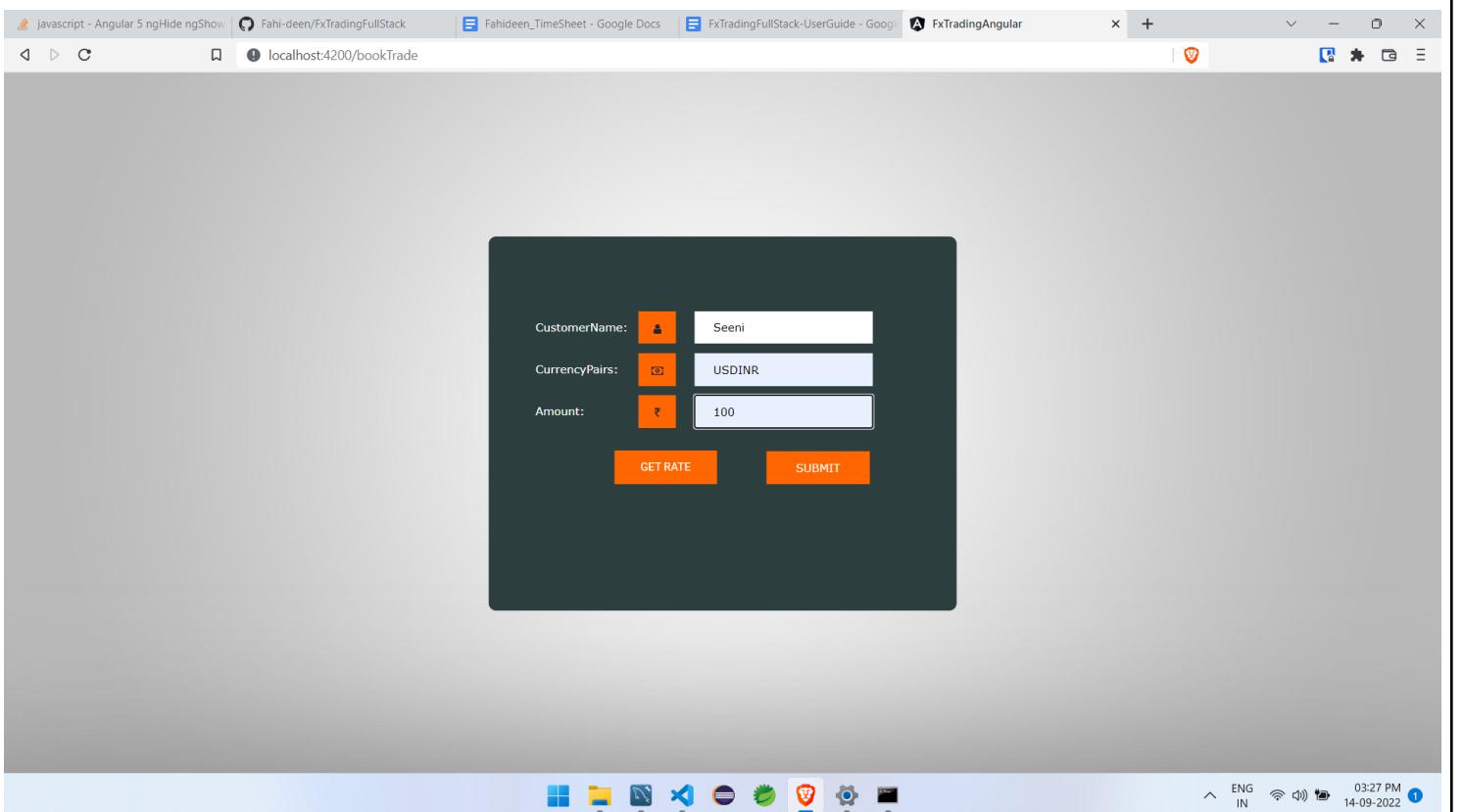
A screenshot of a web browser window displaying a table of trading data. The browser tabs show various documents and a Google search result. The main content area is titled "Trading Data".

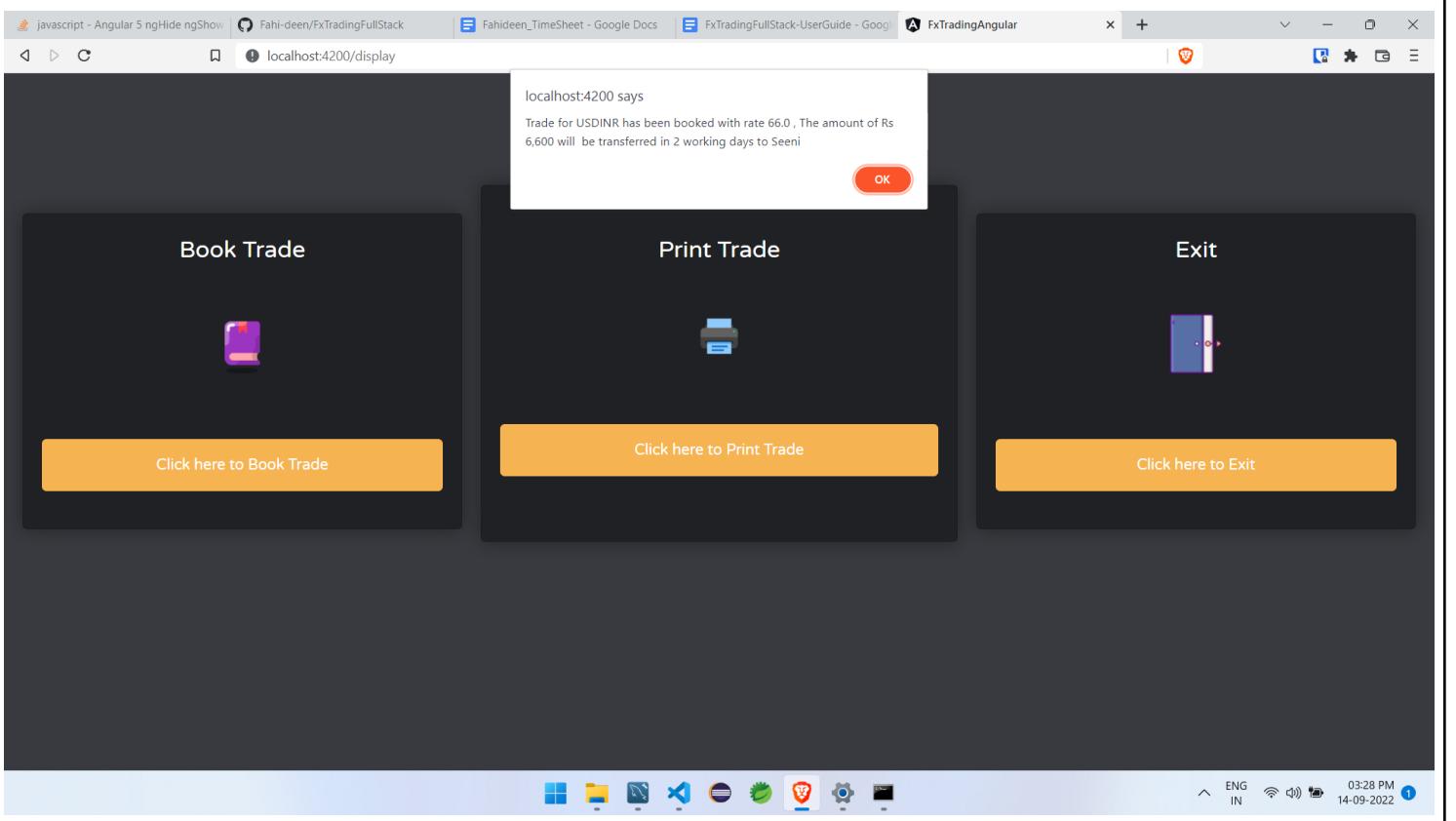
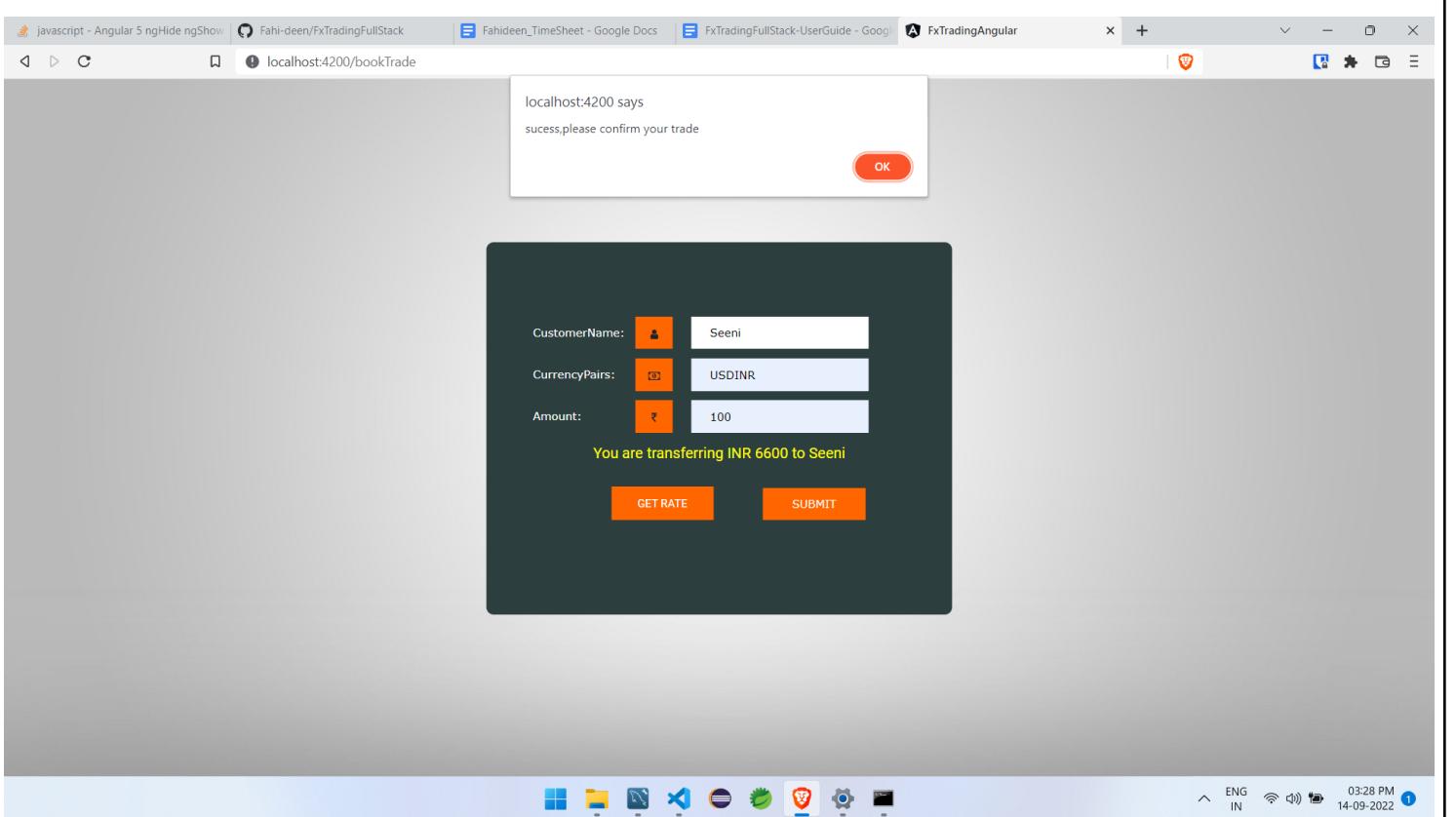
Trade no	customerName	currencyPair	Amount	Rate
1	Hameed	USDINR	6600	66

Below the table are navigation links: "« Prev", a blue highlighted number "1", and "Next »".



^ ENG  
IN 03:26 PM  
14-09-2022





javascript - Angular 5 ngHide ngShow | Fahideen\_FxTradingFullStack | Fahideen\_TimeSheet - Google Docs | FxTradingFullStack-UserGuide - Google | FxTradingAngular

localhost:4200/printtrade

## Trading Data

Trade no	customerName	currencyPair	Amount	Rate
1	Hameed	USDINR	6600	66
3	Seeni	USDINR	6600	66

« Prev 1 Next »

ENG IN 03:28 PM 14-09-2022

javascript - Angular 5 ngHide ngShow | Fahideen\_FxTradingFullStack | Fahideen\_TimeSheet - Google Docs | FxTradingFullStack-UserGuide - Google | FxTradingAngular

localhost:4200/bookTrade

CustomerName:

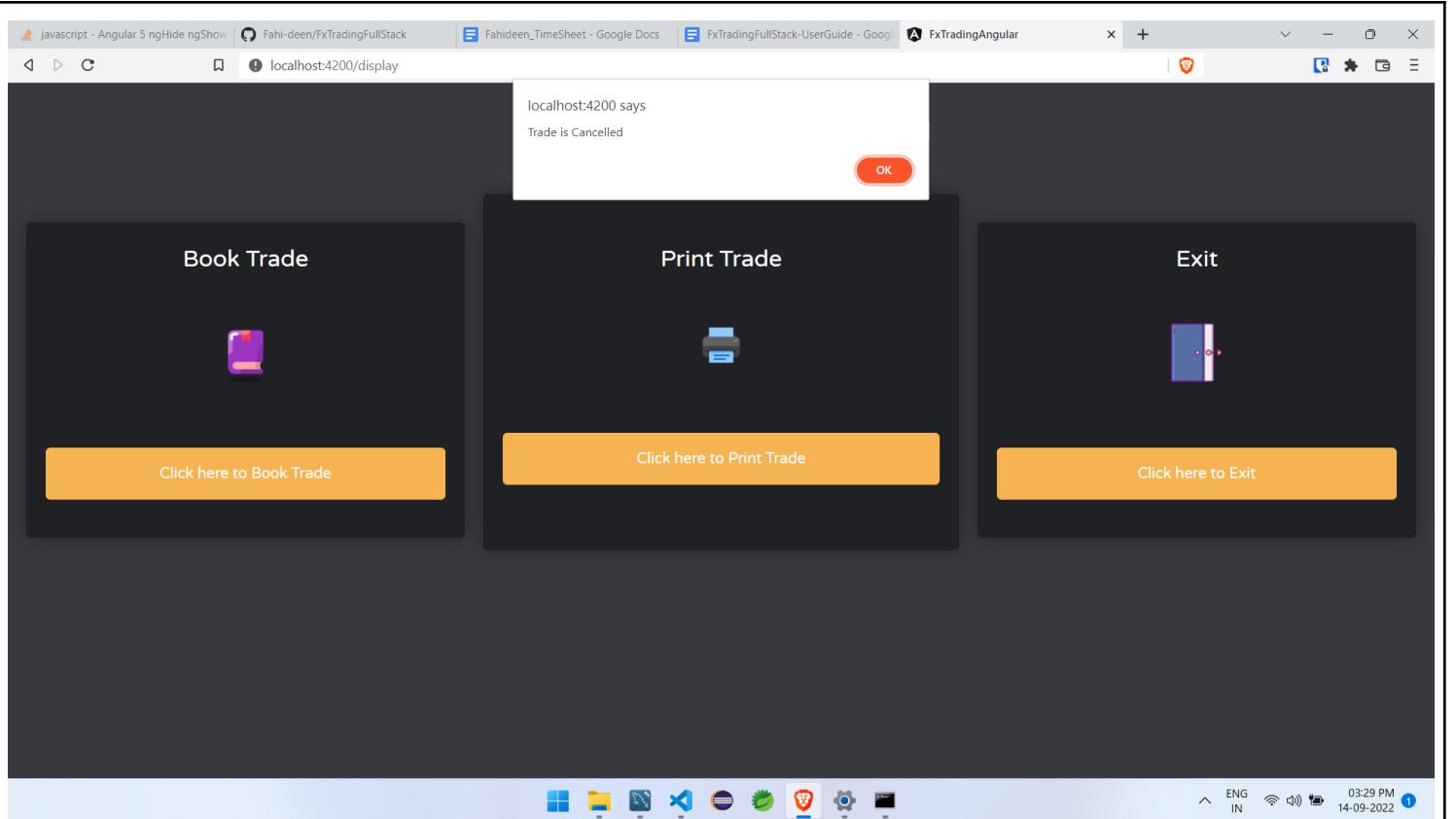
CurrencyPairs:

Amount:

CONFIRM TRADE CANCEL TRADE

ENG IN 03:29 PM 14-09-2022



Trading Data					
Trade no	customerName	currencyPair	Amount	Rate	
1	Hameed	USDINR	6600	66	
3	Seeni	USDINR	6600	66	
« Prev <span style="background-color: blue; color: white; padding: 2px;">1</span> Next »					

## Test cases in Angular

