

Network & Thread questions

1. Usage of thread

Implement a movie ticket selling application.

Assume there are 100 tickets with unique id (range 0 ~ 99) and 3 ticket booths (Booth 1, Booth 2, Booth 3), it takes 0.1s to finish a single ticket transaction.

The application will print out the id of the ticket and the sale booth name when a transaction is done.

You may use inheritance of Thread class or implement Runnable interface to achieve multi-threading.

Sample output:

```
Window 3 sold the 87th ticket
Window 2 sold the 88th ticket
Window 3 sold the 89th ticket
Window 1 sold the 90th ticket
Window 2 sold the 91th ticket
Window 1 sold the 92th ticket
Window 3 sold the 93th ticket
Window 2 sold the 94th ticket
Window 1 sold the 95th ticket
Window 3 sold the 96th ticket
Window 2 sold the 97th ticket
Window 3 sold the 98th ticket
Window 1 sold the 99th ticket
Window 2 sold the 100th ticket
```

2. Simple server/client

Implement a simple stock data api that the server (StockPriceServer) stores and returns stock prices upon client (StockPriceClient) requests.

Client

- 1a. Create a socket that connect to localhost at port 59898
- 1b. use PrintWriter to handle socket output stream and BufferedReader
- 1c. allow user to send stock code to the server and receive response
- 1d. implement logic to terminate the connection. (eg. when receiving "exit" msg)

Server

Threading

- 1a. create a server socket listening at port 59898

- 1b. create ExecutorService object of 20 threads (Check static method Executors.newFixedThreadPool)
- 1c. have the server keep listening the coming socket requests

Interaction

- 2a. Use data structure to store and map the stock code and the corresponding prices.
- 2b. use BufferedReader, InputStreamReader to handle socket input stream (output to clients)
- 2c. Inside the while loop, read the stock code from the clients and return the corresponding price to the client
- 2d. Implement logic to terminate the connection. (eg. when receiving "exit" msg)

Sample output:

Server

```
The stock price server is running...
Connected: Socket[addr=/127.0.0.1,port=65063,localport=59898]
Server: Waiting for input..
FB: $194.320000
Waiting for the next input...
BABA: $182.000000
Waiting for the next input...
NVDA: $209.610000
Waiting for the next input...
MB: is not a valid stock code.
Waiting for the next input...
```

Client

```
Use 127.0.0.1 as server IP address
Enter lines of text then enter 'exit' to quit
FB
Client - FB
Server - FB: $194.320000
BABA
Client - BABA
Server - BABA: $182.000000
NVDA
Client - NVDA
Server - NVDA: $209.610000
MB
Client - MB
Server - Error: 'MB' is not a valid stock code.
```