

به نام خدا



دانشگاه صنعتی شریف

دانشکده مهندسی برق

## برنامه سازی شی گرا

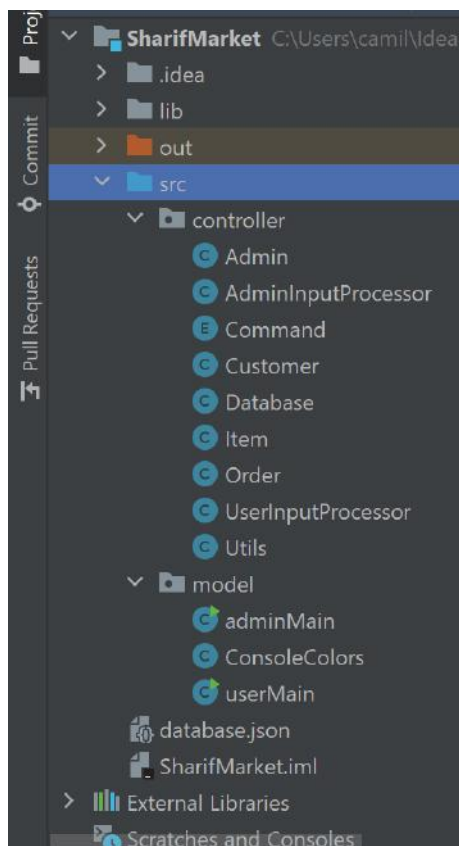
گزارش تمرین سوم – شریف مارکت

کاملیا کربلائی ۹۹۱۰۹۶۶۹

۱۴۰۰/۳/۴

کلاس‌ها:

ساختار برنامه به صورت زیر است:



برنامه دارای دو Main جداگانه برای برای فروشنده (Admin) و مشتری (Customer/User) می‌باشد.

**کلاس Admin و کلاس Customer:**

هر ادمین و همچنین هر مشتری دارای یک ID و یک password است که ID به صورت int و password به صورت استرینگ می‌باشد.

**Command Enum:**

Regex های دستورها در این enum قرار دارند.

## کلاس Database:

کلاس Database کلاسی است که امور مربوط به خواندن از و نوشتن به فایل را انجام می‌دهد. برای ذخیره اطلاعات، از کتابخانه‌ی Gson استفاده شده است. فایل database.json در ابتدا به صورت زیر وجود دارد و با استفاده از برنامه، موارد درون آن بروزرسانی می‌شود.

```
database.json × AdminInputProcessor.java × Database.java ×
1 {"customers":[],"items":[],"deletedItems":[],"orders":[],"orderHistory":[],"admins":[]}
```

آنچه در فایل ذخیره می‌شود، ArrayList های customers, items, deletedItems, orders, orderHistory, admins و currentAdmin از جنس Admin و currentCustomer از جنس Customer است.

در کلاس Database، توابعی نیز برای اضافه کردن، حذف کردن و تغییر دادن اعضای ArrayList های مذکور وجود دارد.

## تابع‌های find:

تابع‌های find در کلاس‌های Admin, Customer, Item و Order با گرفتن ID یا در مواردی نام object، object منسوب به آن ID یا نام را return می‌کنند و در صورتی که objectی از آن کلاس با ID/نام مورد نظر وجود نداشته باشد null بر می‌گردانند.

## کلاس Item:

هر Item دارای ویژگی های زیر می باشد:

```
public class Item {  
    private String name;  
    private int inStock;  
    private int buyingPrice;  
    private int sellingPrice;  
    private int ID;  
    private int ordersIn;  
    private int numberSold;  
    private int moneyMadeFrom;  
    private int itemProfit;  
}
```

- name: نام کالا
- inStock: تعداد موجود
- buyingPrice: قیمت خرید
- sellingPrice: قیمت فروش
- ID: شناسه ی کالا
- ordersIn: تعداد سفارش های این کالا
- numberSold: تعداد فروخته شده
- moneyMadeFrom: پول حاصل از فروش کالا
- itemProfit: سود حاصل از فروش کالا

کلاس Item علاوه بر تابع های find، دارای تابع های updateSales و چاپ کالاها نیز می باشد.

تابع updateSales در دستور checkout فرا خوانده می شود و اعمال زیر را انجام می دهد:

- تعداد orderهایی که از item خریداری شده وجود دارد را به علاوه ی ۱ می کند.
- تعداد فروخته شده در سفارش را به تعداد کل فروخته شده از آن کالا اضافه می کند.
- درآمد حاصل از آن سفارش را به درآمد کل کسب شده از فروش آن کالا اضافه می کند.
- سود کسب شده در سفارش را به سود کل حاصل از فروش آن کالا اضافه می کند.

نکته ی شایان توجه در مورد تابع های چاپ کالاها این است که عرض ستون ها بر اساس طول طولانی ترین عبارتی که قرار است در ستون نوشته شود تنظیم می شود.

نمونه چاپ تمام کالاها، کالاهای موجود و کالاهای ناموجود:

```
Enter command: ls -r
ALL ITEMS
ITEM      IN STOCK PRICE
-----
Water     500      1000
CocaCola  200      1500
Sugar     100      3000
Salt      100      3000
Pepper    0         4000
Tomato    0         5000
Potato    0         2000

Enter command: ls -i
ITEMS IN STOCK
ITEM      IN STOCK PRICE
-----
Water     500      1000
CocaCola  200      1500
Sugar     100      3000
Salt      100      3000

Enter command: ls -n
UNAVAILABLE ITEMS
ITEM      PRICE
-----
Pepper    4000
Tomato    5000
Potato    2000

Enter command:
```

## کلاس Order:

هر Order دارای ویژگی‌های زیر است:

```
public class Order {
    private int userID;
    private String date;
    private int itemID;
    private int number;
    private int orderID;
    private String itemName;
```

- userID: شناسه‌ی خریدار
- date: تاریخ خرید
- itemID: شناسه‌ی کالای خریداری شده
- number: تعداد سفارش داده شده
- orderID: شناسه‌ی سفارش

○ itemName: نام کالای خریداری شده

کلاس Order علاوه بر تابع‌های find، دارای تابع‌های چاپ سفارش‌های بررسی نشده و چاپ تاریخچه می‌باشد.

در تابع‌های چاپ سفارش‌ها و تاریخچه نیز مانند تابع‌های چاپ کالاها، عرض ستون‌ها بر اساس طول طولانی‌ترین عبارتی که قرار است در ستون نوشته شود تنظیم می‌شود.

## کلاس Utils:

در کلاس Utils تعدادی تابع مفید قرار داده شده است که در دستورها و توابع مختلف برنامه، مورد استفاده قرار می‌گیرند.

## کلاس ConsoleColors:

این کلاس قابلیت رنگی چاپ کردن متن در cmd را فراهم می‌آورد.

## دستور لاگین، لاگوت و احراز هویت:

دستورهای login و logout برای Admin و Customer مشابه می‌باشد.

**شناسه باید به صورت عدد صحیح وارد شود.** سپس اگر شناسه قبلاً ثبت شده بود، برنامه از کاربر رمز درخواست می‌کند. کاربر ۳ بار اجازه‌ی وارد کردن رمز دارد و اگر هر سه بار غلط وارد کند، برنامه از دستور login خارج می‌شود و برای لاگین کردن بایستی مجدداً دستور login را وارد کند.

اگر شناسه قبلاً ثبت نشده بود (کاربر جدید)، برنامه از کاربر درخواست انتخاب رمز می‌کند. رمز به صورت String است. پس از وارد کردن رمز پیشنهادی توسط کاربر، برنامه از او درخواست می‌کند تا دوباره همان رشته را وارد کند (Confirm کردن رمز) و در صورتی که دو رشته برابر نبودند، کاربر باید مجدداً رمز پیشنهادی خود را وارد کند و تا وقتی رشته‌های اول و دوم برابر نشوند، این فرایند ادامه دارد.

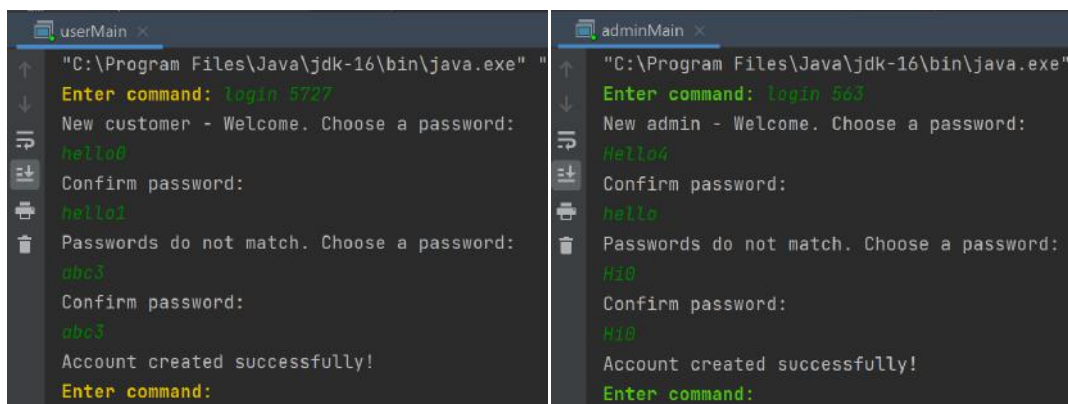
متغیرهای currentCustomer/currentAdmin بعد از login، برابر کاربر وارد شده می‌شود.

در صورت وارد کردن دستور `logout`، اگر متغیرهای مذکور `null` نباشند، آنها برابر `null` می‌شوند.

برای ایجاد هر گونه تغییر در اطلاعات، **Admin** باید لاگین شده باشد.

نمونه ورودی و خروجی:

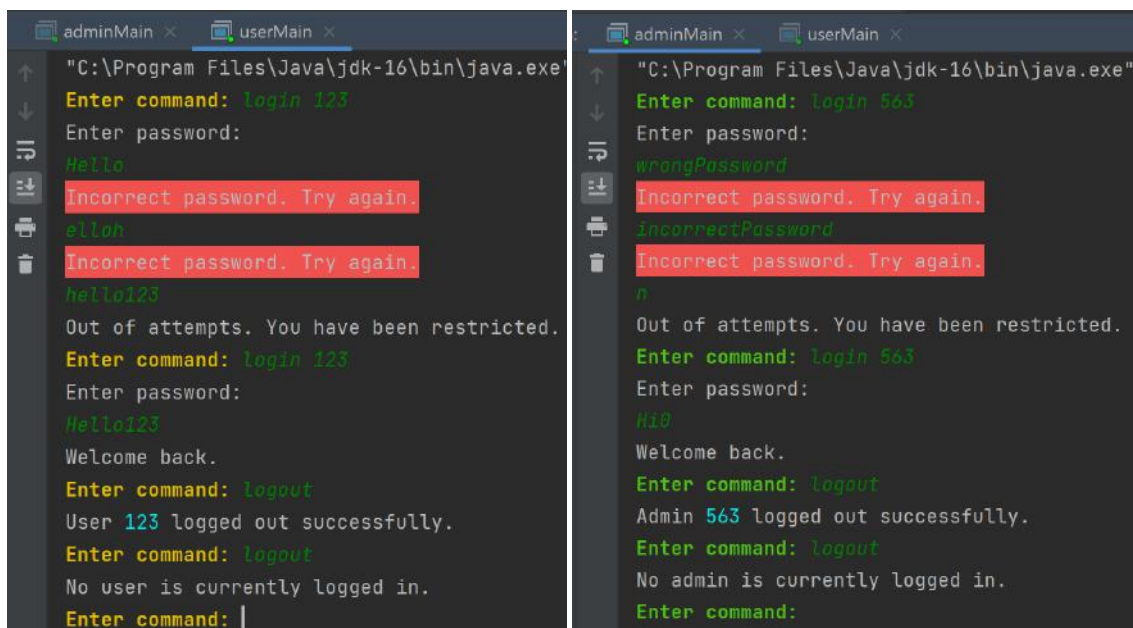
ایجاد حساب:



```
userMain x "C:\Program Files\Java\jdk-16\bin\java.exe"
Enter command: login 5727
New customer - Welcome. Choose a password:
hello0
Confirm password:
hello1
Passwords do not match. Choose a password:
abc3
Confirm password:
abc3
Account created successfully!
Enter command:

adminMain x "C:\Program Files\Java\jdk-16\bin\java.exe"
Enter command: login 563
New admin - Welcome. Choose a password:
Hello4
Confirm password:
hello
Passwords do not match. Choose a password:
H10
Confirm password:
H10
Account created successfully!
Enter command:
```

login و logout:



```
adminMain x userMain x "C:\Program Files\Java\jdk-16\bin\java.exe"
Enter command: login 123
Enter password:
Hello
Incorrect password. Try again.
elloh
Incorrect password. Try again.
hello123
Out of attempts. You have been restricted.
Enter command: login 123
Enter password:
Hello123
Welcome back.
Enter command: logout
User 123 logged out successfully.
Enter command: logout
No user is currently logged in.
Enter command:

adminMain x userMain x "C:\Program Files\Java\jdk-16\bin\java.exe"
Enter command: login 563
Enter password:
wrongPassword
Incorrect password. Try again.
IncorrectPassword
Incorrect password. Try again.
n
Out of attempts. You have been restricted.
Enter command: login 563
Enter password:
H10
Welcome back.
Enter command: logout
Admin 563 logged out successfully.
Enter command: logout
No admin is currently logged in.
Enter command:
```

## دستور افزودن کالای جدید:

کالای جدید نمی‌تواند نامی برابر نام یکی از کالاهای موجود داشته باشد و همچنین قیمت فروش آن نباید کمتر از قیمت خرید آن باشد. در صورتی که این گونه `input` وارد شود، برنامه پیام خطا می‌دهد و کالای جدید افزوده نمی‌شود.

```
Enter command: add -n Milk -c 50 -sp 4000 -bp 1000
Item added successfully. Item ID: 725449
Enter command: add -n Milk -c 10 -sp 4000 -bp 1000
Item with this name already exists. Item was not added.
Enter command: add -n Yogurt -c 40 -sp 1000 -bp 3000
Selling price cannot be less than buying price. Item was not added.
Enter command: |
```

## تابع حذف کالا و `deletedItems ArrayList`:

در صورت حذف کالا توسط ادمین، کالای حذف شده به `deletedItems ArrayList` اضافه می‌شود و از `items ArrayList` پاک می‌شود. ما به `deletedItems` نیاز داریم به این دلیل که گاهی می‌خواهیم اطلاعات و آمار کالاهای حذف شده در اختیارمان باشد؛ مانند هنگامی که سود یا درآمد را می‌خواهیم محاسبه کنیم.

نمونه ورودی و خروجی دستور حذف کالا:

```
"C:\Program Files\Java\jdk-16\bin
Enter command: login 563
Enter password:
Hi0
Welcome back.
Enter command: remove -c 602953
Item Cheese removed successfully
Enter command:
```

## دستورهای ویرایش مشخصات کالا:



تابع‌های ویرایش مشخصات کالا به طرزى نوشته شده‌اند که نیازی به قابل قبول بودن تمام ورودی‌ها نمی‌باشد. برای مثال، اگر در تابع ویرایش نام و تعداد، تعداد وارد شده عدد قابل قبولی نباشد، بدون تغییر تعداد موجود، فقط نام آن تغییر می‌کند. این کار با استفاده از Boolean ها انجام شده است. به صورتی که برای مثال، یک changeCount Boolean داریم که در صورت نقض شدن شرایط عدد وارد شده برای موجودی برابر false می‌شود و عدد موجودی تغییر نمی‌کند.

نمونه‌ی ورودی خروجی دستورهای ویرایش مشخصات کالا:

```
adminMain x userMain x
Enter command: edit 629077 -n Bubblegum -c 900
Item Potato renamed "Bubblegum".
900 Bubblegum currently in stock.
Enter command: edit 725449 -n Pepsi
Item Milk renamed "Pepsi".
Enter command: edit 725449 -sp 1500 -bp 950 -c 920
920 Pepsi currently in stock.
Pepsi is now being sold for 1500.
Pepsi is now being bought for 950.
Enter command: edit 254212 -n Sugar
This item already has this name. Nothing was changed.
Enter command: edit 254212 -sp 2000 -bp 3500
Command does not exist.
Enter command: edit 254212 -sp 2000 -bp 3500 -c 30
Invalid values. (New selling price cannot be less than new buying price.)
30 Sugar currently in stock.
Enter command: edit 254212 -sp 3000 -bp 900 -c 30
30 Sugar in stock already.
Sell Price of Sugar is already 3000.
Sugar is now being bought for 900.
Enter command:
```

دستور سفارش کالا:

برای سفارش کالا، مشتری باید لاگین شده باشد.

درون تابع سفارش کالا، یک Boolean به نام setOrder تعریف می‌شود که با نقض شدن شرایط ورودی، برابر false می‌شود.

دستور لغو سفارش:

برای لغو سفارش، همان مشتری‌ای که سفارش منسوب به اوست باید لاگین شده باشد. در هنگام لغو سفارش، تعداد کالاهایی که سفارش داده شده بود به موجودی کالا برگردانده می‌شود.

نمونه ورودی و خروجی سفارش و لغو سفارش کالا:

```
adminMain x userMain x
"C:\Program Files\Java\jdk-16\bin\java.exe" "-java
Enter command: order 438624 -c 3
You must be logged in to place an order.
Enter command: login 6
Enter password:
hey
Welcome back.
Enter command: order 438624 -c 3
Item does not exist. Order not placed.
Enter command: order 254471 -c 1
Not enough Pepper in stock.
Order was not placed.
Enter command: order 451315 -c 5
You have ordered 5 Salt. Your order ID is: 423415
Enter command: order -d 423415
Order No. 423415 cancelled successfully.
Enter command: order -d 346189
No order with such order ID exists.
Enter command: |
```

```
adminMain x userMain x
Enter command: order -d 346189
No order with such order ID exists.
Enter command: login 5727
Enter password:
abc
Incorrect password. Try again
abc3
Welcome back.
Enter command: order -d 989260 -c 5
Command does not exist.
Enter command: order 989260 -c 5
You have ordered 5 Water. Your order ID is: 346501
Enter command: login 6
Enter password:
hey
Welcome back.
Enter command: order -d 346501
You can only cancel orders you have made. Order was not cancelled.
Enter command: |
```

## دستور checkout:

این دستور، تابع `updateSales` در کلاس `Item` را فرا می‌خواند. این تابع هم در `ArrayList` کالاهای پاک نشده هم پاک شده دنبال `Item` مورد خریداری می‌گردد تا تغییرات لازم بر اطلاعات کالا حتی در صورتی که توسط فروشنده `remove` شده باشد انجام گیرد. در نهایت نیز سفارش را به `orderHistory` اضافه می‌کند و از `orders` پاک می‌کند.

دستور چاپ سفارش‌های بررسی نشده و تاریخچه‌ی سفارشات:

این دستورها توابع `printOrders` و `printHistory` را از کلاس `Order` فرا می‌خوانند.

نمونه ورودی و خروجی دستورهای چاپ سفارش‌های بررسی نشده، چاپ تاریخچه سفارشات و checkout:

```
adminMain x userMain x
"C:\Program Files\Java\jdk-16\bin\java.exe" "-javaa
Enter command: ls -o
NEW ORDERS:
ORDER ID USER ID DATE ITEM COUNT
-----
346501 5727 2021/05/25 14:06:08 Water 5
285150 5727 2021/05/25 14:09:13 Milk 20
671163 5727 2021/05/25 14:09:43 Sugar 15
Enter command: checkout 671163
Order checked out.
Enter command: checkout 346501
Order checked out.
Enter command: ls -ho
ORDER HISTORY:
ORDER ID USER ID DATE ITEM COUNT
-----
671163 5727 2021/05/25 14:09:43 Sugar 15
346501 5727 2021/05/25 14:06:08 Water 5
Enter command:
```

دستورهای محاسبه‌ی فروش و سود:

- دستور محاسبه‌ی سود کل: سود تک تک کالاهای موجود و حذف شده را جمع می‌کند.
- دستور محاسبه‌ی سود کالا: متغیر `itemProfit` کالای مورد نظر را چاپ می‌کند.

```
adminMain x userMain x
Enter command: calc -p
Total profit: 36500
Enter command: calc -p -c 254212
Profit from Sugar: 33000
```

○ دستور محاسبه‌ی آمار فروش کلی: آمار کالاهای موجود و حذف شده را چاپ می‌کند.

```
adminMain x userMain x
Enter command: calc -s
Water Sales:
Orders: 1 Number Sold: 5
Money Received: 5000 Profit: 3500
CocaCola Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Sugar Sales:
Orders: 1 Number Sold: 15
Money Received: 45000 Profit: 33000
Salt Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Pepper Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Tomato Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Bubblegum Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Pepsi Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Cheese Sales:
Orders: 0 Number Sold: 0
Money Received: 0 Profit: 0
Enter command: |
```

○ دستور محاسبه‌ی آمار فروش یک کالا: آمار یک کالا را چاپ می‌کند.

```
adminMain x userMain x
Enter command: calc -s -c 989260
Water Sales:
Orders: 1 Number Sold: 5
Money Received: 5000 Profit: 3500
Enter command: |
```

**دستور exit:**

با وارد کردن کلمه‌ی “exit” برنامه به اتمام می‌رسد و currentCustomer/ currentAdmin برابر null قرار داده می‌شود.

در صورتیکه ورودی کاربر با هیچ یک از دستورات مطابقت نداشت پیام Command does not exist چاپ می‌شود:

```
Enter command: login
Command does not exist.
Enter command:
```