

CNT 4714 – Project 1 – Fall 2022

Title: “Program Assignment 1: Event-Driven Programming”

Points: 100 points

Due Date: **Sunday September 11, 2022 by 11:59 pm (WebCourses time)**

Objectives: To practice event-driven programming using Java-based GUIs. To refresh your basic Java skills. To simulate (albeit at a very high-level) an enterprise application.

Description: Develop a Java program that creates a standalone GUI application that simulates an e-store (we’ll call our store Nile Dot Com... we’re not quite as big as Amazon.com!)¹ which allows the user to add in stock items to a shopping cart and once all items are included, total all costs (including tax), produces an invoice, and append a transaction log file.

Your program development must include the following steps:

1. Create a main GUI containing the following components:
 - a. An area that allows the user to input data into the application along with the descriptive text that describes each input area.
 - b. A total of six buttons as shown below with functionality as described below.
 - c. As illustrated below, the various buttons on the interface are only accessible at certain points during a user’s interaction with the e-store.
2. An input file named “`inventory.txt`”. This is a comma separated file which contains the data that will be read by the application when the user makes a selection. Each line in this file contains four entries: an item id (a string), a quoted string containing the description of the item, an in stock status (a string), and the unit price for one of the item (a double). A sample file is provided for you on WebCourses. Feel free to create your own input file for testing purposes or augment the provided input file.
3. An output file (append only) named “`transactions.txt`” must be created that uniquely logs each user transaction with the e-store. The unique transaction id will be generated as a permutation of the date/time string when the transaction occurred (see below).

Restrictions:

Your source file shall begin with comments containing the following information:

```
/* Name: <your name goes here>  
   Course: CNT 4714 – Fall 2022  
   Assignment title: Project 1 – Event-driven Enterprise Simulation  
   Date: Sunday September 11, 2022  
*/
```

Input Specification: The file “`inventory.txt`” as described above (see example below as well).

Output Specification: Output is to appear in the specified components of the GUI and various message boxes that appear, plus the contents of the “`transactions.txt`” log file that will be generated.

¹ Source = <https://www.wonderslist.com/top-10-largest-rivers/>

Deliverables:

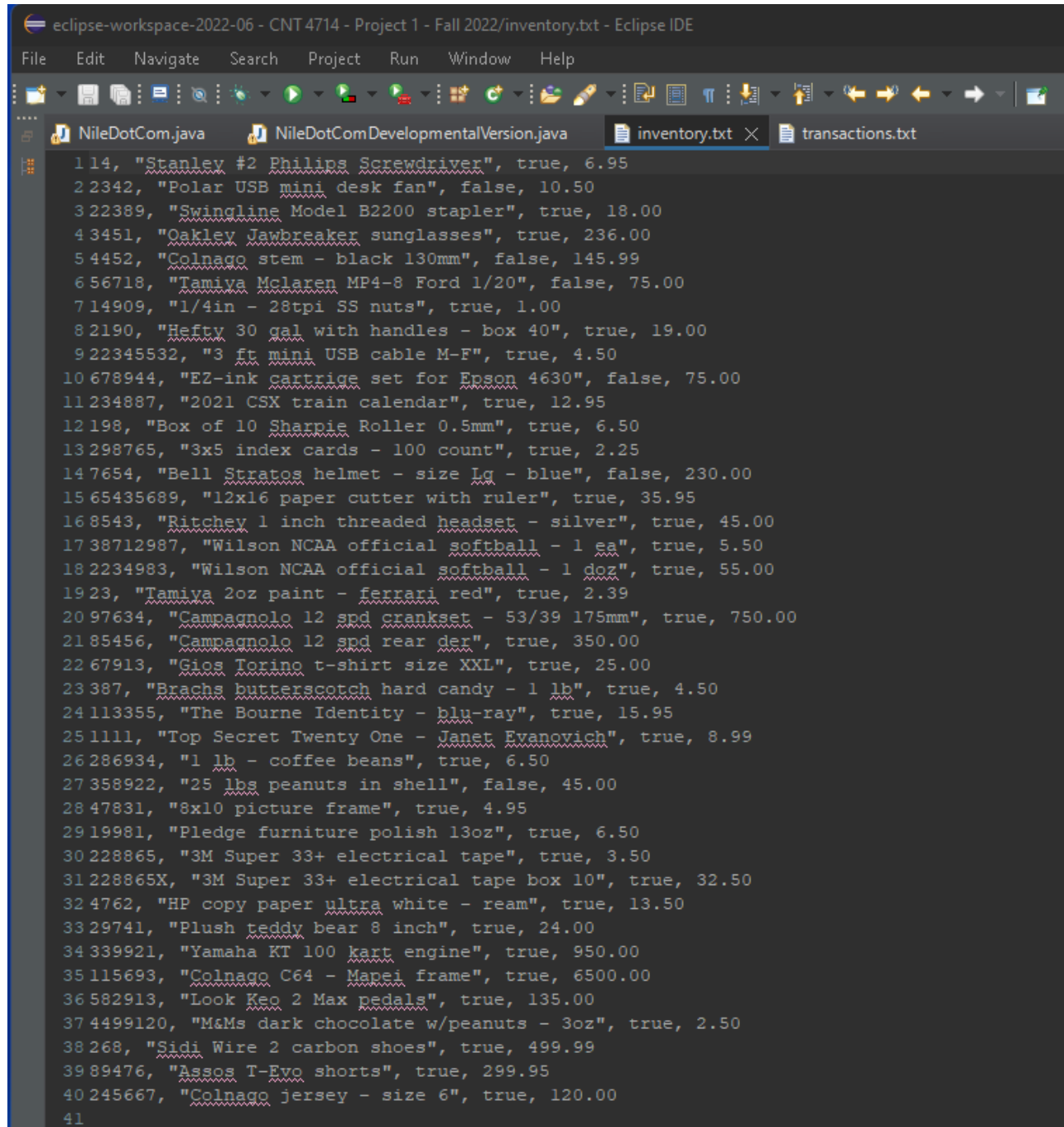
Submit the following via WebCourses no later than 11:59pm Sunday September 11, 2022.

You can zip everything into a single folder if you would like.

- (1) A working copy of your source code (all .java files) and any necessary supporting libraries.
- (2) Include a file that contains (at least) the following nine screen shots.
 - a. The initial GUI (like number 2 in Additional Information below).
 - b. The GUI after the user has entered an item number (ID) and quantity and clicked the “Process Item #” button (like number 4 below).
 - c. The pop-up message box when the user has confirmed the selected item (like number 5 below).
 - d. The pop-up message box when the user has entered an item number (ID) that was not found in the inventory file (like number 5 below).
 - e. The GUI after the user has confirmed the selected item (like number 6 below).
 - f. The GUI after the user has confirmed an item and has entered another item and processed it, but not yet confirmed it (like number 7 below).
 - g. The message dialog box when the user clicks the “View Order” button with multiple items in the order (like number 9 below).
 - h. The message dialog box when the user clicks the “Finish Order” button for an order containing multiple items (like number 10 below).
 - i. The message dialog box when the user has entered an item number (ID) for an item that is out of stock (like number 12 below).
- (3) Include your “transactions.txt” file as it stands after several different transactions have been recorded. You should have at least 5 different transactions in the file with at least three of these transactions having four or more items in the transaction. (similar to the one shown on the last page of this document).

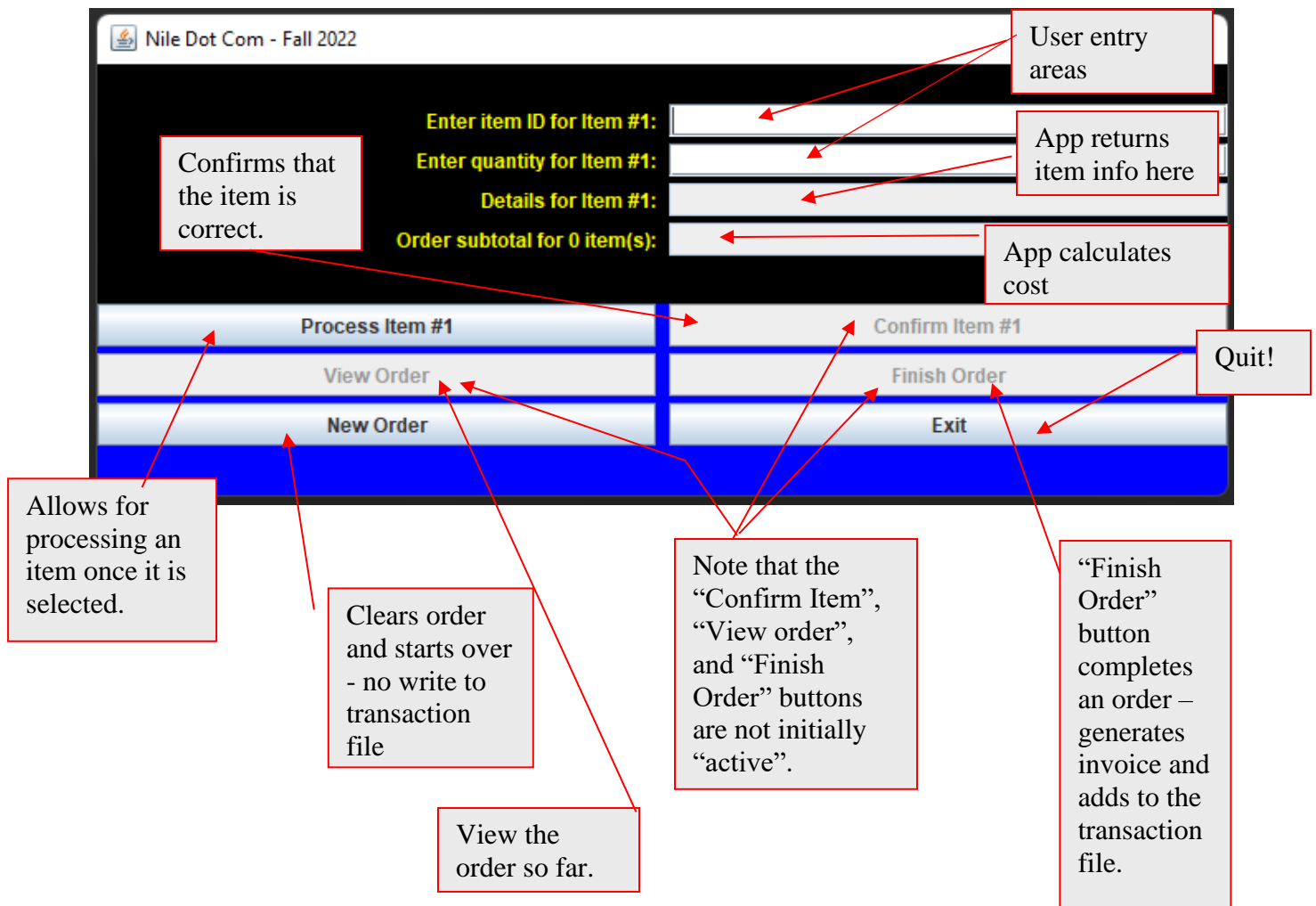
Additional Information:

1. Screen shot of the contents of an example “inventory.txt” file.

A screenshot of the Eclipse IDE interface. The title bar reads "eclipse-workspace-2022-06 - CNT 4714 - Project 1 - Fall 2022/inventory.txt - Eclipse IDE". The menu bar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development. The project explorer on the left shows two files: NileDotCom.java and NileDotComDevelopmentalVersion.java. The editor window displays the contents of inventory.txt, which is a list of 40 items, each with a unique ID, a description, a boolean value (true or false), and a price. The items include tools like screwdrivers, staplers, and power tools, as well as household items like paper cutters, index cards, and toys like teddy bears and karts. The list ends with a line number 41.

```
1 114, "Stanley #2 Philips Screwdriver", true, 6.95
2 2342, "Polar USB mini desk fan", false, 10.50
3 22389, "Swingline Model B2200 stapler", true, 18.00
4 3451, "Oakley Jawbreaker sunglasses", true, 236.00
5 4452, "Colnago stem - black 130mm", false, 145.99
6 56718, "Tamiya McLaren MP4-8 Ford 1/20", false, 75.00
7 14909, "1/4in - 28tpi SS nuts", true, 1.00
8 2190, "Hefty 30 gal with handles - box 40", true, 19.00
9 22345532, "3 ft mini USB cable M-F", true, 4.50
10 678944, "EZ-ink cartridge set for Epson 4630", false, 75.00
11 234887, "2021 CSX train calendar", true, 12.95
12 198, "Box of 10 Sharpie Roller 0.5mm", true, 6.50
13 298765, "3x5 index cards - 100 count", true, 2.25
14 7654, "Bell Stratos helmet - size Lg - blue", false, 230.00
15 65435689, "12x16 paper cutter with ruler", true, 35.95
16 8543, "Ritchey 1 inch threaded headset - silver", true, 45.00
17 38712987, "Wilson NCAA official softball - 1 ea", true, 5.50
18 2234983, "Wilson NCAA official softball - 1 doz", true, 55.00
19 23, "Tamiya 2oz paint - ferrari red", true, 2.39
20 97634, "Campagnolo 12 spd crankset - 53/39 175mm", true, 750.00
21 85456, "Campagnolo 12 spd rear der", true, 350.00
22 67913, "Gios Torino t-shirt size XXL", true, 25.00
23 387, "Brachs butterscotch hard candy - 1 lb", true, 4.50
24 113355, "The Bourne Identity - blu-ray", true, 15.95
25 1111, "Top Secret Twenty One - Janet Evanovich", true, 8.99
26 286934, "1 lb - coffee beans", true, 6.50
27 358922, "25 lbs peanuts in shell", false, 45.00
28 47831, "8x10 picture frame", true, 4.95
29 19981, "Pledge furniture polish 13oz", true, 6.50
30 228865, "3M Super 33+ electrical tape", true, 3.50
31 228865X, "3M Super 33+ electrical tape box 10", true, 32.50
32 4762, "HP copy paper ultra white - ream", true, 13.50
33 29741, "Plush teddy bear 8 inch", true, 24.00
34 339921, "Yamaha KT 100 kart engine", true, 950.00
35 115693, "Colnago C64 - Mapei frame", true, 6500.00
36 582913, "Look Keo 2 Max pedals", true, 135.00
37 4499120, "M&Ms dark chocolate w/peanuts - 3oz", true, 2.50
38 268, "Sidi Wire 2 carbon shoes", true, 499.99
39 89476, "Assos T-Evo shorts", true, 299.95
40 245667, "Colnago jersey - size 6", true, 120.00
41
```

2. Initial GUI:



3. GUI after user has entered first item number and the quantity of the item they want to purchase, but before clicking the “Process Item” button.

The screenshot shows a window titled "Nile Dot Com - Fall 2022". The interface has a black background with yellow text for labels. There are four input fields: "Enter item ID for Item #1:" with the value "14", "Enter quantity for Item #1:" with the value "1", "Details for Item #1:" which is empty, and "Order subtotal for 0 item(s):" which is empty. Below these fields is a grid of buttons. The first row contains "Process Item #1" and "Confirm Item #1". The second row contains "View Order" and "Finish Order". The third row contains "New Order" and "Exit". The buttons are arranged in a 3x2 grid with a blue border around the entire grid.

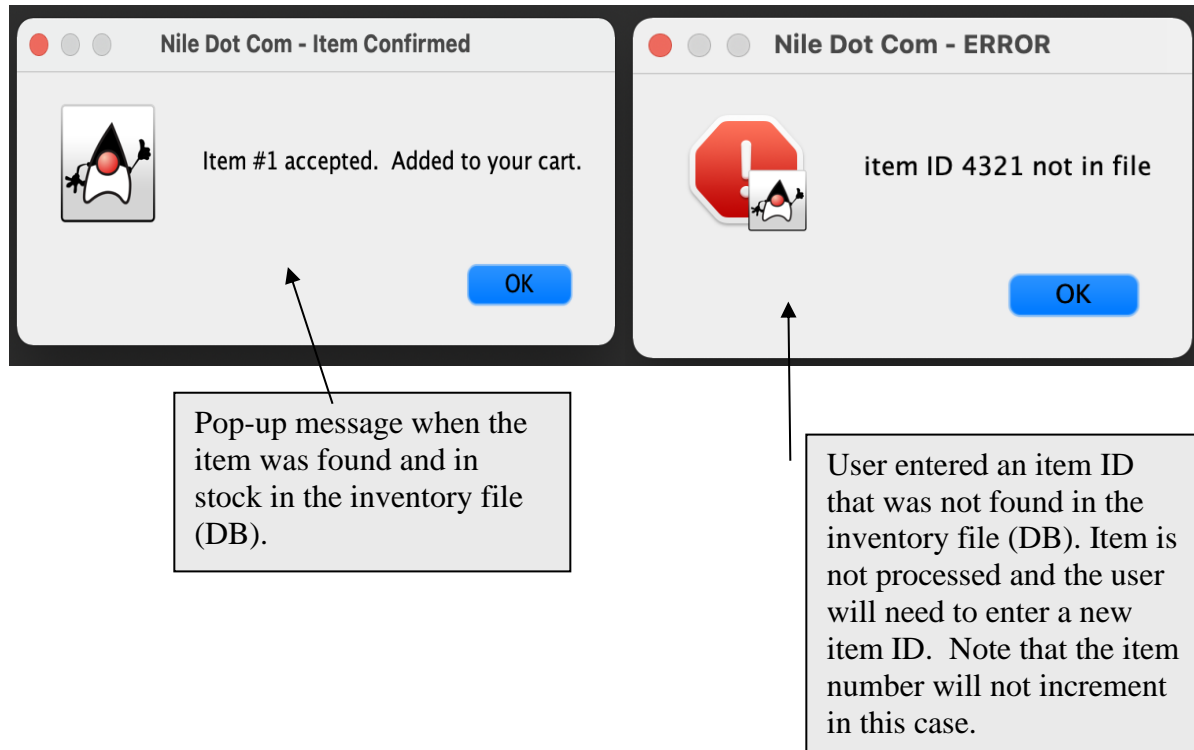
4. GUI after user has selected the first item and clicked the “Process Item #1” button.

The screenshot shows the same window as before, but with updated data. The "Details for Item #1:" field now contains the text "14 \"Stanley #2 Philips Screwdriver\" \$6.95 1 0% \$6.95". The "Order subtotal for 0 item(s):" field is still empty. The buttons are the same, but the "Confirm Item #1" button is now highlighted with a blue background, indicating it is active. The "Process Item #1" button is no longer highlighted. Red arrows point from the active button and the item details to the explanatory text boxes below.

Note that the “Confirm Item #1” button is now active and the “Process Item #1” button is inactive.

Information on the item selected/purchased has been extracted from the inventory file and written into the window on the line for item #1. Note proper currency formatting.

5. When the user clicks on the “Confirm Item #1” button, a confirmation information message appears on the screen.



6. The main GUI after the user has confirmed the first item.

Discount is based on the quantity of an item. If:
1-4 = 0%
5-9 = 10%
10-14 = 15%
≥15 = 20%

Note that the buttons change number.

Enter item ID for Item #2:

Enter quantity for Item #2:

Details for Item #1: 14 "Stanley #2 Philips Screwdriver" \$6.95 1 0% \$6.95

Order subtotal for 1 item(s): \$6.95

Process Item #2	Confirm Item #2
View Order	Finish Order
New Order	Exit

Note that the text areas for entering the second item information are cleared and ready for input.

Subtotal should reflect current total of all confirmed items.

Information for previous item remains on display.

7. User enters and processes (but has not yet confirmed) the next item in the order (which we'll make the last one for this order).

Nile Dot Com - Fall 2022

Enter item ID for Item #2: 3451

Enter quantity for Item #2: 2

Details for Item #2: 451 "Oakley Jawbreaker sunglasses" \$236.00 2 0% \$472.00

Order subtotal for 1 item(s): \$6.95

Process Item #2	Confirm Item #2
View Order	Finish Order
New Order	Exit

8. The GUI after the user has entered the information for all the items and confirmed the last item they wish to order.

The screenshot shows a window titled "Nile Dot Com - Fall 2022". The interface has a black background with yellow text for labels. There are three input fields for item information. Below the inputs, a summary line shows "Details for Item #2: 451 'Oakley Jawbreaker sunglasses' \$236.00 2 0% \$472.00". Another line shows "Order subtotal for 2 item(s): \$478.95". At the bottom, there is a grid of six buttons: "Process Item #3", "Confirm Item #3", "View Order", "Finish Order", "New Order", and "Exit".

Enter item ID for Item #3:	<input type="text"/>
Enter quantity for Item #3:	<input type="text"/>
Details for Item #2:	451 "Oakley Jawbreaker sunglasses" \$236.00 2 0% \$472.00
Order subtotal for 2 item(s):	\$478.95

Process Item #3	Confirm Item #3
View Order	Finish Order
New Order	Exit

9. When the user clicks the “View Order” button, the following message box should appear.


The message box is titled "Nile Dot Com - Current Shopping Cart Status" and has a close button (X) in the top right corner. It contains an information icon (i) on the left. The text inside lists two items in the cart:

- 1. 14 "Stanley #2 Philips Screwdriver" \$6.95 1 0% \$6.95
- 2. 3451 "Oakley Jawbreaker sunglasses" \$236.00 2 0% \$472.00

At the bottom of the message box is an "OK" button.

10. When the user clicks the “Finish Order” button, the invoice message should be generated and displayed.

Nile Dot Com - FINAL INVOICE



Date: 8/23/22, 7:31:28 PM EDT

Number of line items: 2

Item# / ID / Title / Price / Qty / Disc % / Subtotal:

1. 14 "Stanley #2 Philips Screwdriver" \$6.95 1 0% \$6.95
2. 3451 "Oakley Jawbreaker sunglasses" \$236.00 2 0% \$472.00

Order subtotal: \$478.95

Tax rate: 6%

Tax amount: \$28.74

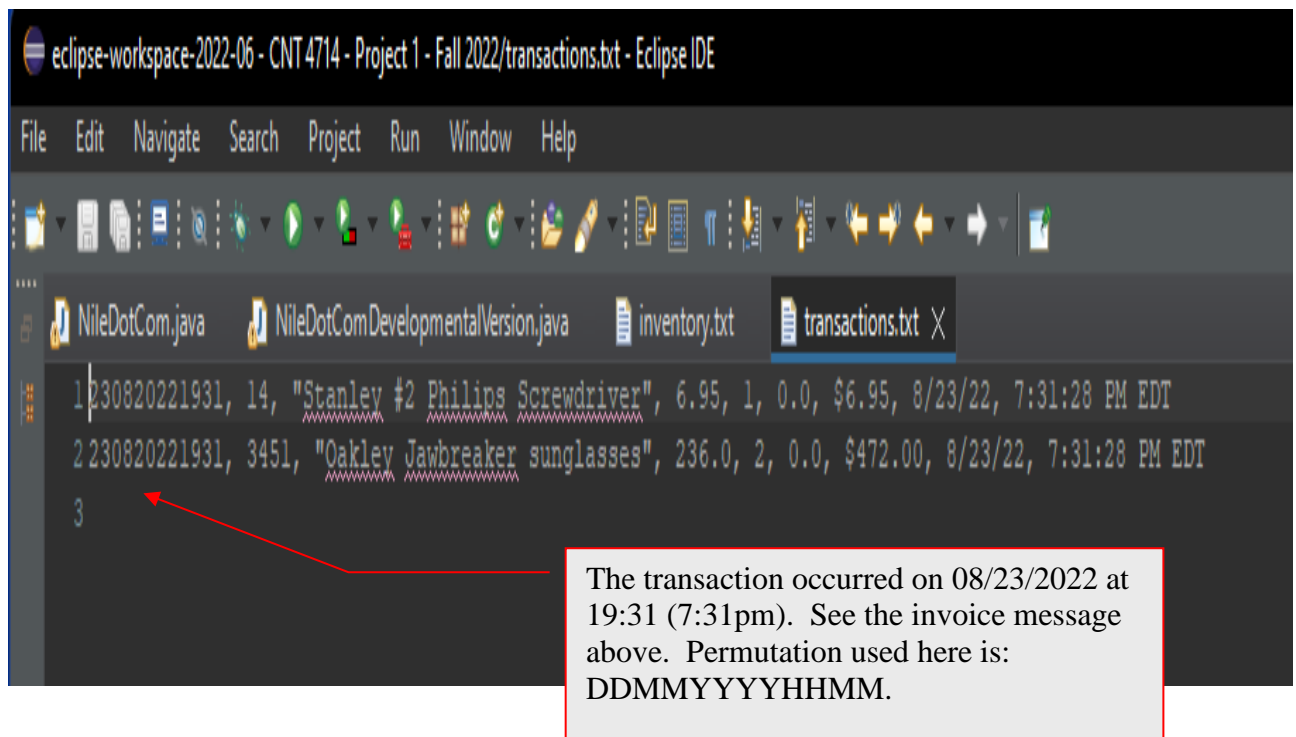
ORDER TOTAL: \$507.69

Thanks for shopping at Nile Dot Com!

OK

The date and time are used to create the unique transaction id in the “transactions.txt” file. For the invoice the format is “DD/MM/YY HH:MM:SS TMZ”

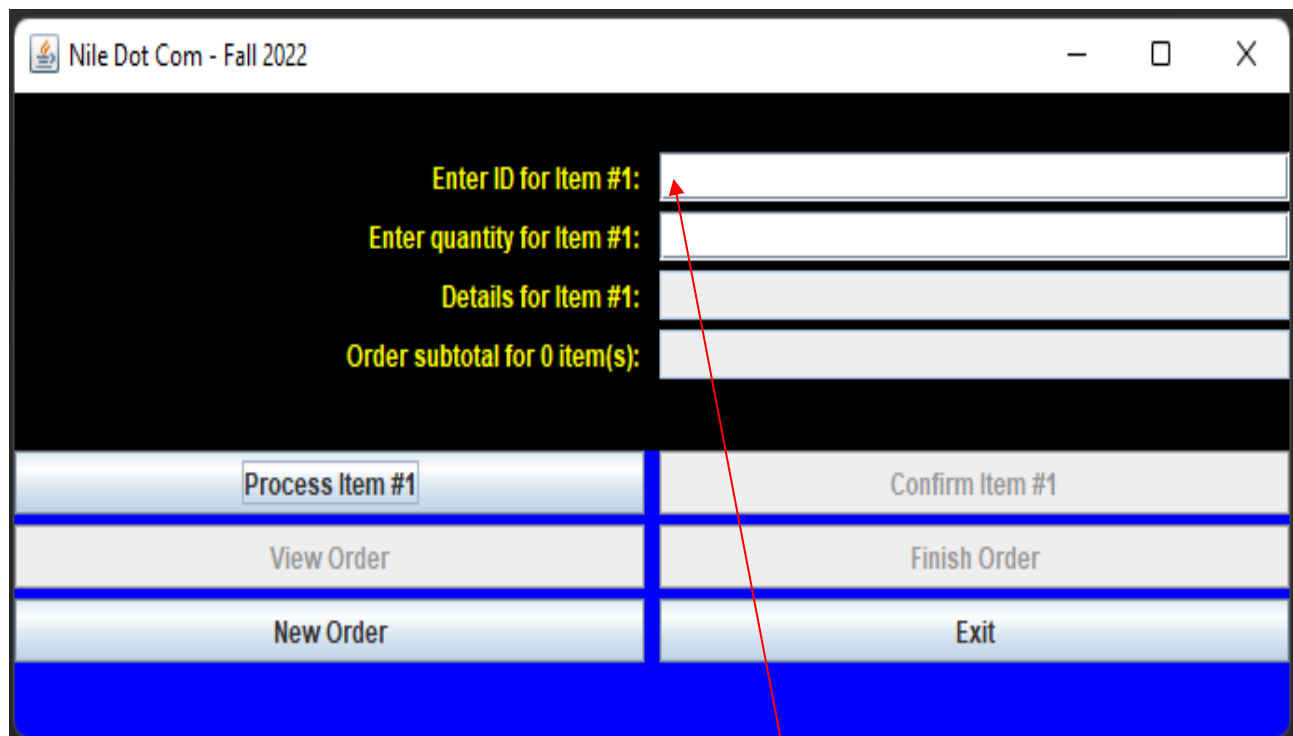
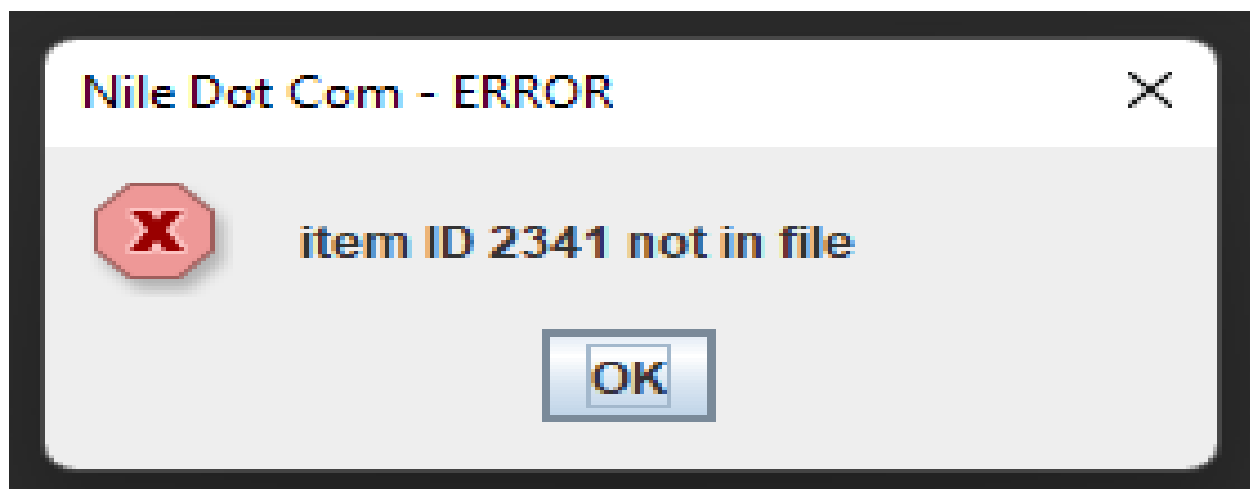
11. The transaction file after order shown above was finished. Note the unique transaction ids based on the date and time.



12. If the user enters an item number for an item that is not in stock (it is in the inventory – just not in stock), the following message dialog should appear. Once the user clicks OK in the dialog box, then they would enter a new item number for another item and continue processing their current transaction.
- 13.

The screenshot shows the Nile Dot Com - Fall 2022 application window. The form contains the following fields and buttons:

- Enter ID for Item #1:
- Enter quantity for Item #1:
- Details for Item #1:
- Order subtotal for 0 item(s):
- Buttons: Process Item #1, Confirm Item #1, View Order, Finish Order, New Order, Exit



Notice that after the "Not In Stock" error has occurred, the GUI has cleared the user entries in the two input fields and has not incremented the item count.

14. The GUI after the user has clicked the “Finish Order” button and also clicked “OK” on the invoice dialog box.

Enter item ID for Item #3:

Enter quantity for Item #3:

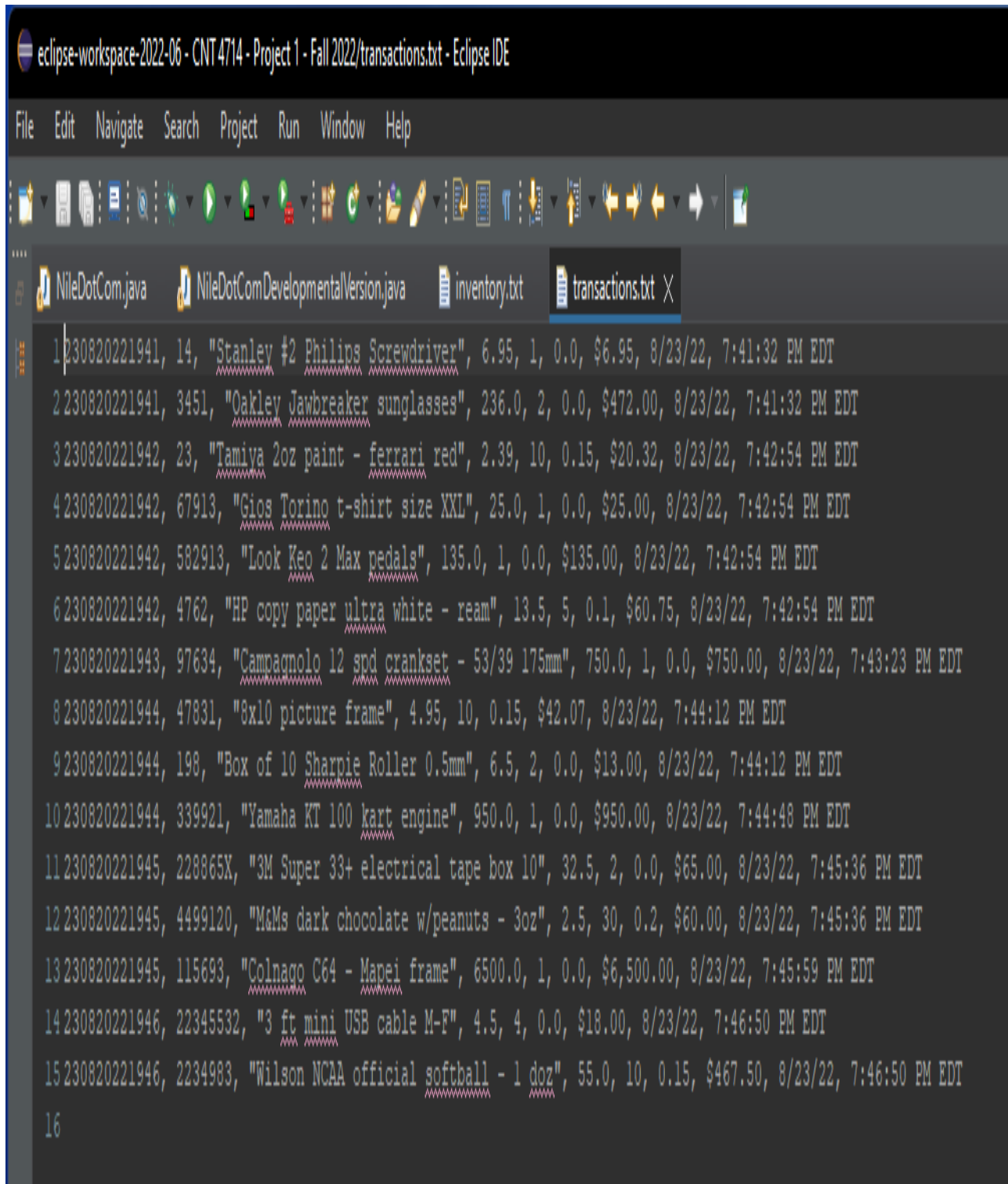
Details for Item #2: 451 "Oakley Jawbreaker sunglasses" \$236.00 2 0% \$472.00

Order subtotal for 2 item(s): \$478.95

Process Item #3	Confirm Item #3
View Order	Finish Order
New Order	Exit

Notice that ID and quantity fields are no longer editable, since the order has been completed. Thus a new order must be started.(A little hard to see here, but the fields are greyed out and not editable.

15. The transaction file after several orders have been completed. Note the unique transaction ids based on the date and time for every separate transaction. This file shows eight separate transactions.



The screenshot shows the Eclipse IDE interface. The title bar reads "eclipse-workspace-2022-06 - CNT 4714 - Project 1 - Fall 2022/transactions.txt - Eclipse IDE". The menu bar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development tools. The editor window shows the "transactions.txt" file with 15 transactions. Each transaction is a line of text containing a unique ID, a date and time, a description of the item, and its price. The transactions are numbered 1 through 15. The text is as follows:

```
1 230820221941, 14, "Stanley #2 Philips Screwdriver", 6.95, 1, 0.0, $6.95, 8/23/22, 7:41:32 PM EDT
2 230820221941, 3451, "Oakley Jawbreaker sunglasses", 236.0, 2, 0.0, $472.00, 8/23/22, 7:41:32 PM EDT
3 230820221942, 23, "Tamiya 2oz paint - ferrari red", 2.39, 10, 0.15, $20.32, 8/23/22, 7:42:54 PM EDT
4 230820221942, 67913, "Gios Torino t-shirt size XXL", 25.0, 1, 0.0, $25.00, 8/23/22, 7:42:54 PM EDT
5 230820221942, 582913, "Look Keo 2 Max pedals", 135.0, 1, 0.0, $135.00, 8/23/22, 7:42:54 PM EDT
6 230820221942, 4762, "HP copy paper ultra white - ream", 13.5, 5, 0.1, $60.75, 8/23/22, 7:42:54 PM EDT
7 230820221943, 97634, "Campagnolo 12 spd crankset - 53/39 175mm", 750.0, 1, 0.0, $750.00, 8/23/22, 7:43:23 PM EDT
8 230820221944, 47831, "8x10 picture frame", 4.95, 10, 0.15, $42.07, 8/23/22, 7:44:12 PM EDT
9 230820221944, 198, "Box of 10 Sharpie Roller 0.5mm", 6.5, 2, 0.0, $13.00, 8/23/22, 7:44:12 PM EDT
10 230820221944, 339921, "Yamaha KT 100 kart engine", 950.0, 1, 0.0, $950.00, 8/23/22, 7:44:48 PM EDT
11 230820221945, 228865X, "3M Super 33+ electrical tape box 10", 32.5, 2, 0.0, $65.00, 8/23/22, 7:45:36 PM EDT
12 230820221945, 4499120, "M&Ms dark chocolate w/peanuts - 3oz", 2.5, 30, 0.2, $60.00, 8/23/22, 7:45:36 PM EDT
13 230820221945, 115693, "Colnago C64 - Mapei frame", 6500.0, 1, 0.0, $6,500.00, 8/23/22, 7:45:59 PM EDT
14 230820221946, 22345532, "3 ft mini USB cable M-F", 4.5, 4, 0.0, $18.00, 8/23/22, 7:46:50 PM EDT
15 230820221946, 2234983, "Wilson NCAA official softball - 1 doz", 55.0, 10, 0.15, $467.50, 8/23/22, 7:46:50 PM EDT
```

16