CS 3331: Advanced Object-Oriented Programming

Programming Assignment 4

Due: April 27, 2019 by 9:00am (**No late assignment will be accepted**. **The assignment must be uploaded to Connections Cloud by 9:00am.**) This homework shall be done individually.

In this homework, you are to develop the third iteration of the Price Watcher application as shown in Figure 1 using JavaFX [4]. Your application shall support multiple items.

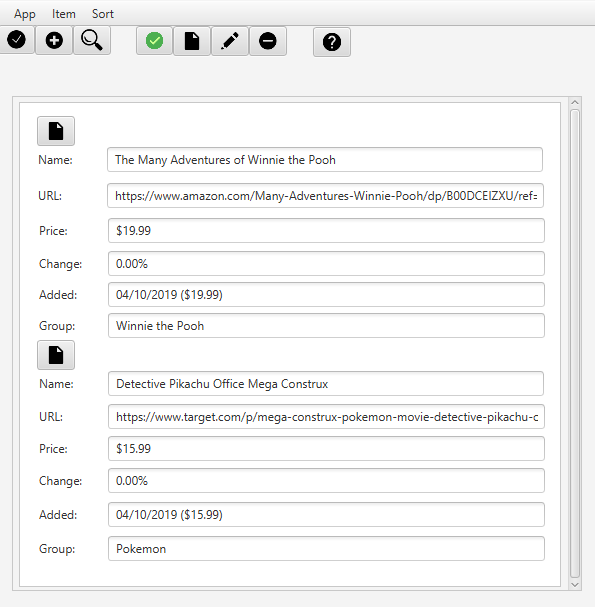


Figure : Price Watcher Application GUI

Extend your code from programming assignment 3 to meet the following requirements:

R1. The application shall provide a way to manage the list of items whose prices are to be watched over.

R2. The application shall provide an option to find current price of a single item or current price of all items.

R3. The application shall provide a menu bar with the following menus:

1. App
2. Item
3. Sort.

R4. The App menu shall include the following menu items:

1. About
2. Exit.

R5. When the user clicks on the About menu item, the application shall display a popup window displaying the name of the application and your name.

R6. When the user clicks on the Exit menu item, the application shall terminate the application.

R7. The Item menu shall include the following menu items:

1. Check prices Alt-C
2. Add item Alt-A
3. Separator
4. Search Alt-S
5. Selected menu.

R8. When the user clicks on Check Prices menu item or Alt-C, the application shall update pricing information of **all** items.

R9. When the user clicks on Add item, the application shall display a popup window with the following:

1. A text field labeled as Name.
2. A text field labeled as URL.
3. A URL button.
4. A text field labeled as Price.
5. An editable dropdown box labeled as Group.
6. A button labeled as Add.
7. A button labeled as Cancel.

R10. When the user clicks on the URL button, the application shall open a browser window and go the specified URL.

R11. When the user clicks on the URL button without a URL in the URL text field, the application shall display an error message.

R12. When the user clicks on the Add button, the application shall add the item to the list of items.

R13. When the user clicks on the Cancel button, the application shall clear the entered information and dismiss the popup window.

R14. When the user clicks on the Search menu item or Alt-S, the application shall display a popup window with the following:

1. A text field labeled as Name.
2. A dropdown box labeled as Store. The list of stores displayed in the dropdown box is dynamically generated based on the list of items.
3. A dropdown box labeled as Group. The list of groups displayed in the dropdown box is dynamically generated based on the list of items.
4. A button labeled as Search.
5. A button labeled as Cancel.

R15. The user shall be able to search item(s) by name, store, or group.

R16. The Selected menu shall include the following menu items:

1. Price Alt-P
2. View Alt-V
3. Edit Alt-E
4. Remove Alt-R
5. Separator
6. Copy name
7. Copy URL

R17. When the user clicks on Price menu item or Alt-P with a selected item, the application shall update the pricing information of the selected item.

R18. When the user clicks on Price menu item or Alt-P without a selected item, the application shall display an error message.

R19. When the user clicks on the View menu item or Alt-V with a selected item, the application shall open a browser window and go the specified URL.

R20. When the user clicks on View menu item or Alt-V without a selected item, the application shall display an error message.

R21. When the user clicks on the Edit menu item or Alt-E with a selected item, the application shall display an Edit popup with the following and have the ability to edit the editable content:

1. Name of the selected item (editable).
2. URL of the selected item (editable).
3. URL button.
4. Price of the selected item.
5. Price checking button.
6. Group name of the selected item (editable).

R22. When the user clicks on Edit menu item or Alt-E without a selected item, the application shall display an error message.

R23. When the user clicks on Remove menu item or Alt-R with a selected item, the application shall remove the selected item from the list of items.

R24. When the user clicks on Remove menu item or Alt-R without a selected item, the application shall display an error message.

R25. When the user clicks on the Copy Name menu item with a selected item, the application shall copy the name of the selected item.

R26. When the user clicks on the Copy URL menu item with a selected item, the application shall copy the URL of the selected item.

R25. When the user clicks on the Copy Name menu item without a selected item, the application shall display an error message.

R26. When the user clicks on the Copy URL menu item without a selected item, the application shall display an error message.

R27. The Sort menu shall include the following menu items:

1. Added oldest
2. Added newest
3. Separator
4. Name ascending
5. Name descending
6. Separator
7. Price change (%)
8. Price low ($)
9. Price high ($).

R28. When the user clicks on a menu item under the Sort menu, the application shall sort the list of items according to the selected sorting method.

R29. The application shall display a list of menu options as buttons:

1. Check all prices
2. Add an item
3. Search items
4. Check price of a selected item
5. View webpage of a selected item
6. Edit selected item
7. Delete selected item
8. Show app information.

R30. When the user clicks on the menu option buttons, the application shall respond according to the selected button.

R31. The application shall support a context menu with the following options:

1. All
2. Search
3. Separator
4. List of URLs
5. Separator
6. Groups.

R32. When the user clicks on an option in the context menu, the application shall filter the list of items according to the selected option.

R33. When the user clicks on the Search button in the Search menu item popup window, the application shall search for the list of items that matches the searching criteria.

R34. When the user clicks on the Cancel button in the Search menu item popup window, the application shall clear the entered information and dismiss the popup window.

Please refer to the screenshots shown in class.

INSTRUCTION

The purpose of this homework is to be able to reuse existing code and learn the graphics capability of JavaFX. Reuse your design and code from previous programming assignments as much as possible. All model classes should be reusable, and some portion of control classes may be adapted and reused too.

1. Design the Price Watcher application and document it by drawing a UML class diagram. (20 points)
   1. Your class diagram should show the main components (classes) of your application, their roles and their relationships.
   2. Your model (business logic) classes should be clearly separated from the view/control (UI) classes with no dependency, use the Model-View-Control design pattern [3].
   3. For each class in your diagram, define operations to show its roles or responsibilities in your application.
   4. For each association, include at least a label, multiplicities and navigation directions.
   5. You should also need to provide a short, textual description of each class appearing in your class diagram.
2. Code your design by making your code conform to your design. (80 points)

TESTING

Your code should compile and run correctly under Java 8 or later versions.

WHAT AND HOW TO TURN IN

Submit your program and model (model must be drawn via a UML tool and not by hand) to your sub-community on Connections Cloud:

1. Your program submission should include the following in a single zip file:
   1. src directory of source code files in .exe form.
   2. Your zip file should include only a single directory named YourFirstNameLastName containing all your source code files and other support files needed to compile and run your program. DO NOT INCLUDE BYTECODE (.class) FILES.
2. Design.doc (UML class diagram along with the descriptions)

GRADING

You will be graded on the quality of the design, how clear your code is, and presentation of your diagrams and the correctness of notations used. Excessively long code will be penalized: don't repeat code in multiple places. Your code should be reasonably documented and sensibly indented so it is easy to read and understand.

Be sure your name is in the comments in your code.

REFERENCES

[1] Wikipedia, Strategy pattern, <https://en.wikipedia.org/wiki/Strategy_pattern>.

[2] Martina Seidl, et al., UML@Classroom: An Introduction to Object-Oriented Modeling, Springer, 2015. Free ebook through UTEP library.

[3] Holger Gast, How to Use Objects, Addison-Wesley, 2016. Sections 9.1 and 9.2. Ebook available from UTEP library.

[4] Cay S. Horstmann, Core Java Volume I - Fundamentals, 11th edition, Prentice Hall, 2018