

McGill COMP 303 – Software Development

Mini Assignment 6

Due: November 15, 2018 at 23:55 on MyCourses

This mini assignment practices the use of the Composite, Decorator, Iterator and Singleton design patterns.

Question: implement in Java the following system using the appropriate design patterns (as well as all other good OO design principles).

System:

You are implementing a restaurant’s ordering system. This system should allow customers to browse the menu items, select some of them to create an order, see the full order, and pay it.

The system should include the following details:

- Each item can be either an appetizer, a main course, a dessert, a side, or a drink. Drinks can be alcoholized or not.
- The list of all available items are stored in a database.
- Purchasing an appetizer, a main course, and a dessert together constitute a full meal, and results in the dessert being half-priced.
- Each day, a different “Item-of-the-day” appears on the menu. This item is a regular item, but with a discount for the day.
- When ordering items, the customer should be able to list all items in their order, with their respective prices, in addition to the total. For Items-of-the-day and full meals, the regular price of the items should be indicated in addition to the discounted price.
- When ordering at least one alcoholized drink, the system should show a warning, saying that the customer must be at least 18 years old.

Your system should contain at least the following elements:

- Item: has a name and a price.
- Menu: allows other classes to retrieve the list of all available items.
- OrderingSystem: has the following functionalities: start a new order, list all items, add an item to the order, create a full meal, finalize (pay and close) the order.

You DO NOT need to implement the following:

- Connection to a database of items. For this assignment, the list of items can be hardcoded in this the Menu class (or interface). Include 2-3 examples of each type of item.
- Graphical visualization. You can simply print information in the console.
- Item-of-the-day selection. Assume the Item-of-the-day is already selected for today. Make sure it is easy to change it.

For this assignment, you do not need to make your program interactive, but provide a simple Driver class with a main method that shows a customer that lists all items on the menu, orders a full meal, a beer, and the Item-of-the-day (assume it's a side dish), looks at the full order, and pays. This Driver class is only to show how to use the system. The system should not rely on it.

What to hand in?

Submit all your Java files for this assignment, including the Driver.

How it will be graded?

- +10 – Program works as expected, and all constraints are respected
- +10 – Correct use of design patterns where appropriate
- +10 – The solution follows good OO practices