



The University of the West Indies, St. Augustine
COMP 2603 Object Oriented Programming 1
OPTIONAL ASSIGNMENT
2023/2024 Semester 2

Due Date: April 20, 2024 at 11:50 p.m.

Overview: This assignment requires you to create and document an object-oriented program that calculates and prints the price of various coffee drinks. There are 5 basic ingredients that may occur in a coffee drink: ground coffee, water, milk, steamed milk and foamy milk. Each coffee drink has a different combination of ingredients as shown in Figure 1. The combination of ingredients determines the price of a coffee drink.

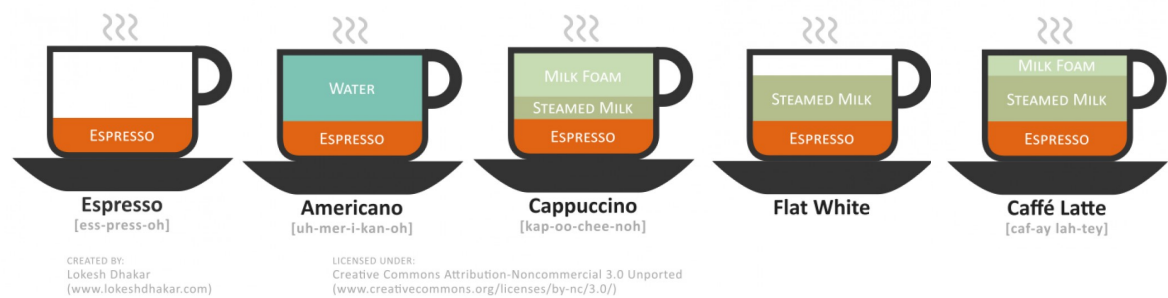


Figure 1. Coffee Drink Makeup

An Espresso has one part ground coffee and one part water. All other coffee drinks build on the Espresso. An Americano adds on two parts of water. A Cappuccino has one part of steamed milk compared to a Flat White which has two parts of steamed milk. A Caffe Latte is essentially a Flat White coffee with one part of foamy milk. Capuccinos however have two parts of foamy milk.

One part of ground coffee costs \$5.00 and one part of milk costs \$0.50. One part of steamed milk costs \$1.50. One part of foamy milk costs \$4.50. There is no charge for water.

UML Diagram of Domain Classes

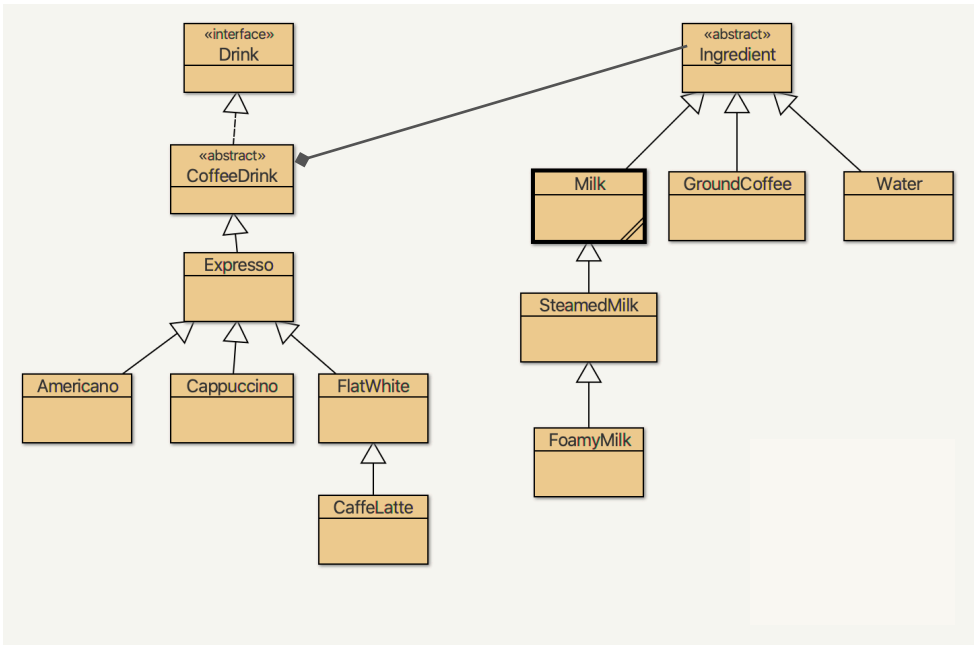


Figure 2: UML Diagram of Domain Classes in the Coffee Drink Application

Figure 2 shows the classes in the application and the relationships between the classes. Details of the individual classes and interface follow next.

Drink Interface

Method Signature	Return Type	Description/Purpose
calculatePrice()	double	Calculates the price of the drink based on the ingredients

CoffeeDrink Class

Attribute	Type	Description/Purpose
ingredients	ArrayList<Ingredient>	Stores the Ingredient objects that make up a Coffee Drink object

Method Signature	Return Type	Description/Purpose
getIngredientsList()	ArrayList<Ingredient>	Accessor for the ingredients list
calculatePrice()	double	Calculates the price of the drink based on the ingredients
add(Ingredient ingredient)	void	Adds an Ingredient to the list of ingredients
toString()	String	Returns the name of the Coffee Drink and a list of the ingredients with prices

Subclasses of CoffeeDrink

Method Signature	Return Type	Description/Purpose
toString()	String	Refines the parent method by appending “PRICE: “ and the price of the drink

Ingredient Class

Attribute	Type	Description/Purpose
price	double	The cost of the ingredient

Method Signature	Return Type	Description/Purpose
Ingredient()		No-argument constructor that initialises the price to 0
Ingredient(double price)		Overloaded constructor that sets the price to the value of the input parameter
getPrice()	double	Accessor for the price attribute
toString()	String	Returns the name of the ingredient and the price formatted with \$ symbol to two decimal places.
equals()	boolean	Checks equality based on the class name



Sample output expected for the toString() method of the Espresso class

Espresso [Water \$0.00, GroundCoffee \$5.00]
PRICE: \$5.00

Sample output expected for the toString() method of the FoamyMilk class

FoamyMilk \$4.50

Coding Instructions

Write the Java code for each class in the application shown in Figure 2 using BlueJ and set up the associations shown in Figure 1. Ensure your student ID is documented in each file.
Tip: try to limit the number of hardcoded values that you use and opt for calculations/method calls instead.

WRITTEN PARAGRAPH - submit in the comments section of the assignment box

You must write a 150 word paragraph explaining why you are in an UNCLE state and include this in the comments of the submission area. If this is omitted, your score will be 0.

Test Classes

Some test classes will be provided for you to test whether your code adheres to the assignment specification and works properly. These tests are not exhaustive. To run a test, include the .class file in your BlueJ source folder, right click and run from the main window.

Submission Instructions

- Document your student ID at the top of each file within a comment block.
- Upload a ZIP file of your compiled project source and class files to the myElearning course page by the deadline. Submissions that do not compile will receive 0 marks. Empty submission files or faulty zip files will receive 0 marks. Check your submitted files for completeness and correctness.
- Name your ZIP file as follows: **FirstName_LastName_ID_UNCLE.zip**. Marks will be deducted for submissions that do not conform to this naming convention. Ensure that the unzipped folder has the same name as the zipped file. No nested folders must be present in your zipped file.
- Sign and submit the University Plagiarism declaration confirming that you are submitting your own original work and that you have not copied or collaborated with other students.

Important Information Regarding Academic Conduct

- This is an individual assignment. You must attempt this assignment by yourself without any help from others, aside from the course lecturer, tutor or marker.
- You may use legitimate resources on the Internet, in books, or from the course notes to assist (unless prohibited by a question). Copying or modifying such content does not make it yours. Indicate on your submission any or all sources used in your answers within comments at the end of your files. Identify code snippets that you did not write yourself but reused from some source.
- You are not allowed to communicate, share or disclose any aspect of your solutions/work in any form to anyone except for the course lecturer, tutors, markers, and examiners.
- You are not allowed to assist others with this assignment.
- University plagiarism and academic misconduct policies apply fully.
- No part of your submission should be made publicly available even after the due date without permission from the course lecturer.
- The use of Generative AI Tools, Chegg.com, CourseHero.com or any tutoring or homework assistance services, online or not, are prohibited.
- If you are faced with extenuating circumstances, contact your lecturer right away. Concessions may be arranged on a case-by-case basis. Be honest with your claims and provide evidence where possible.