

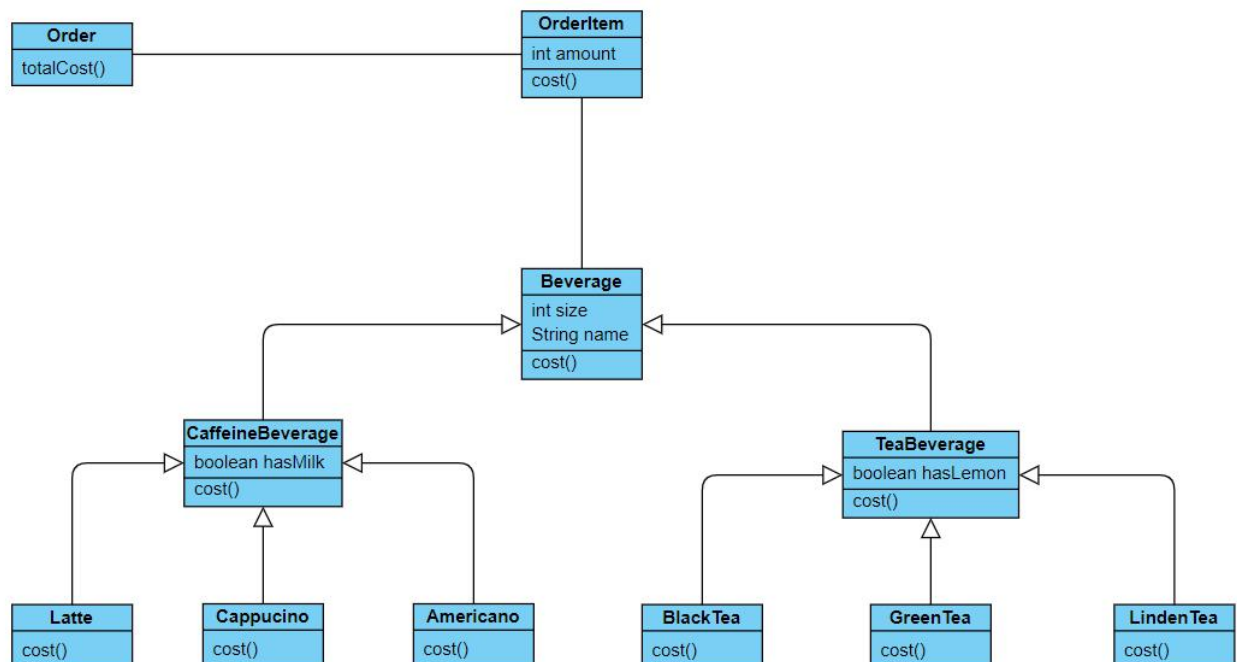
CENG 1004 Spring 2020 Final Homework

CODE

Mugla Cafe needs an application to get orders from customers and print and calculate the price of these orders. Mugla Cafe Beverage prices are shown in the following table, note that price of beverage varies depending on its size. Furthermore, milk can be added to Caffeine Beverages and lemon can be added to Tea Beverages with an extra cost of 1 Turkish Liras.

	Beverage Name	SMALL	MEDIUM	LARGE
	Latte	5 TL	6 TL	7 TL
	Capucino	6 TL	7 TL	8 TL
	Americano	7 TL	8 TL	9 TL
	Black Tea	3 TL	4 TL	5 TL
	Green Tea	4 TL	5 TL	6 TL
	Linden Tea	5 TL	6 TL	7 TL

Following Class Diagram shows the relations between the classes. Order can contain one or more order items, order item contain a beverage and specifies the amount (how many) of beverage. Beverage has an inheritance hierarchy, important methods and attributes are shown.



Following Test class is given. Create the required classes and implement the necessary functionalities to make the test class compilable so that it can generate the intended output. Your classes should reflect the design shown in the class diagram.

```

public class TestCafe {

    public static void main(String args[]){

        /**
         * A group of fiends ordered
         * 2 Small-sized Latte with Milk
         * 2 Medium-sized Cappucino without Milk
         * 1 Large-sized Americano with Milk
         * 3 Small-sized Black Tea without Lemon
         * 1 Medium-sized Green Tea with Lemon
         * 1 Small-sized Linden Tea with Lemon
         */
        Order order = new Order();
        CaffeineBeverage cBeverage = new Latte(true, Beverage.SMALL);
        order.add(new OrderItem(cBeverage, 2));
        cBeverage = new Cappucino(false, Beverage.MEDIUM);
        order.add(new OrderItem(cBeverage, 2));
        cBeverage = new Americano(true, Beverage.LARGE);
        order.add(new OrderItem(new Americano(), 1));

        TeaBeverage tBeverage = new BlackTea(false, Beverage.SMALL);
        order.add(new OrderItem(tBeverage, 3));
        tBeverage = new GreenTea(true, Beverage.MEDIUM);
        order.add(new OrderItem(tBeverage, 1));
        tBeverage = new LindenTea(true, Beverage.SMALL);
        order.add(new OrderItem(tBeverage, 1));

        //print the order
        System.out.println(order);
        //Expected output :
        // 2 Small Latte with Milk 12 TL
        // 2 Medium Cappucino without Milk 12 TL
        // 1 Large Americano with Milk 10 TL
        // 3 Small Black Tea without Lemon 9 TL
        // 1 Medium Green Tea with Lemon 6 TL
        // 1 Small Linden Tea with Lemon 6 TL
        // TOTAL : 55 TL

    }
}

```

Documentation

Give answers to the following questions in your documentation.

- A. What is an abstract class? Which class(es) can be defined as abstract class in your code?
- B. What is an abstract method? Which methods can be defined as abstract in your code?
- C. What is the difference between an Interface and Class? Can you use an Interface instead of a Class in your implementation? Explain why or why not?
- D. What is the difference between a static and non-static variable. Have you used any static variables in your implementation. If yes, give the name of the variables?
- E. Have you benefited from polymorphism in your implementation. If yes, copy the code segment where you use polymorphism to your report and give the name of the polymorphic variable.

Grading Policy

Code

- A. Defining Classes, Attributes & Methods : **40 Points** (Objective - 3)
- B. Printing the order output : **20 Points** (Objective - 1)

Documentation

- A. **5 Points** (Objective - 4)
- B. **5 Points** (Objective - 4)
- C. **10 Points** (Objective - 4)
- D. **10 Points** (Objective - 3)
- E. **10 Points** (Objective - 4)

Submission

Zip your source folder as yourid.rar. Also save your documentation as yourid.pdf. If your id is "12345678" you will submit the following files:

- ✓ 12345678.rar : archive file containing your source code
- ✓ 12345678.pdf: documentation file containing explanations request in the Documentation Section

Submit your file through DYS system. You should be able to upload files in the homework announcement page.

Important Note: All work should be your own work.

Students will get ZERO (0) if they use others' work in their homework or if they share their work with others .

Students will get ZERO (0) if their code cannot compile or generates error during runtime.