CSCI-101 Programming 2

Lab 4, Part a

INSTRUCTIONS

Log into cs.bridgewater.edu. Change your working directory to your **labs** directory in your repository and create a directory named **lab4**. Inside the **lab4** directory create the following files, compile, and test.

1. Create an enumerated class named **OrderItem** that satisfies the following requirements.
   - ☐ The class has a field named **cost** what holds a decimal value.
   - ☐ The class has a method named **getCost** that returns the **cost** of the named instance.
   - ☐ The class defines the following named instances:
     - ☐ SODA_SM - cost: $1.50
     - ☐ SODA_LG - cost: $2.00
     - ☐ PIZZA_SM - cost: $8.00
     - ☐ PIZZA_LG - cost: $10.50

2. Create an abstract class named **AbstractOrder** that satisfies the following requirements.
   - ☐ The class declares a method named **getCustomerName** which returns a String.
   - ☐ The class declares a method named **getOrderItems** which returns an array of **OrderItems**.
   - ☐ The class declares a method named **getTotalCost** which returns a double.

3. Create a class named **Order** that satisfies the following requirements.
   - ☐ The class extends the **AbstractOrder** class
   - ☐ The class has fields for all of the data required by the abstract class: **customerName**, **orderItems**, and **totalCost**.
   - ☐ The **orderItems** array can hold 10 items.
   - ☐ The class has a constructor that takes a customer's name (String) as an argument.
   - ☐ The class has a method named **addItem** which takes an **OrderItem** as an argument and stores the item in the array named **orderItems** if there is room for the item.

4. Create a program named **PizzaPizza** that satisfies the following requirements.
   - ☐ When run, the program an instance of the **Order** class using the customer name "**Mookie Blaylock**".
   - ☐ A small soda and large pizza are added to the order.
   - ☐ The name of the customer is printed by calling **getCustomerName**.
   - ☐ The items of the order are printed to the screen by calling **getOrderItems**.
   - ☐ Print the cost of the order to the screen by calling **getTotalCost**.