Abstract Classes Exercise

Answer the following exercises.

1. Define an Abstract class named Drink which defines the following methods:
   a. Constructor with two parameters: serialNo (drink’s serial number) and description (brief description about the drink). A drink has a sugarContent which is by default is 0.
   b. toString method that prints the serial number followed by the description and the sugar content of the drink.
   c. An increaseSugarContent method that allow us to increase the sugar content by a particular integer amount. For example, if p is a concrete class that extends Drink, then we should be able to call increaseSugarContent as follows: p.increaseSugarContent(10).increaseSugarContent(20);
   d. An abstract method named getIngredients which returns a string with the drink’s ingredients.

2. Define a Concrete class named Popsi (yes, o not e 😎) which extends the Drink class above. A Popsi has as ingredients sugar and caffeine. The class constructor will initialize the Drink with a serial number equal to 10 and “Great carbonated soda” as a description.

3. Define an finalize() method for the Popsi class. In order to test the finalize() method use the System.gc() method which suggests that the Java Virtual Machine expend effort toward recycling unused objects in order to make the memory they currently occupy available for quick reuse. Hint: set references to null.

4. To submit your work:
   a. Put the name of all the members of your group at the top of the Popsi class.
   b. Export the folder representing your project, zip it, and upload to the submit server (the project name in the submit server is AbsClassExerciseLab). Only one member of your group needs to upload the exercise.
   c. Notice you don’t need to complete the whole exercise to receive full credit. We will be grading on effort according to the following criteria:
      i. Around 25% of work completed → (25 pts)
      ii. Around 50% of work completed → (50 pts)
      iii. At least 75% of work completed → (100 pts)