AP Computer Science A@Beijing National Day School

Problem Set 15: `Bakery Shop`

Due date: Monday, April 1, 2019
Instructor: Mr. Alwin Tareen
Total Points: 15

Task Overview

- Place an information box at the beginning of your `Java` program, **which includes your name**. This formally confers your ownership of the code.
- Implement a program that keeps track of the baked goods being purchased at a bakery shop. Your program should compute the total cost and tax for these goods.

Background

- A bakery shop is responsible for cooking various treats and pastries. However, due to the nature of the items that it sells, the shop must have different ways of calculating the price of its goods.
- Cakes are sold by weight. The bakery shop will quote the cost of a cake as being a certain **price per kilogram**. For example, the bakery shop may sell a Chocolate Cake for `$6.00` per kg. This means that if a customer requests 500g of Chocolate Cake, then they must pay `$3.00`.
- Cookies are priced in bundles of 12, otherwise known as a dozen. Customers may request however many single cookies they desire. For example, if Chocolate Chip Cookies are `$6.00` per dozen, and the customer requests 2 cookies, then they must pay: `2 * $6.00/12.0`, which evaluates as `$1.00`.

Specification

The Information Box Which Includes Your Name [5 points]

- **You are responsible** for typing your own name into the `Author` field, where it says: `YOUR NAME HERE`. Include both your English and Pinyin names.
- If the `Author` field is left blank, or if it contains **someone else's name**, then these 5 points will not be awarded to you.

Calculating a Sales Receipt for a Bakery Shop [10 points]

- **Note:** You are expected to write three separate `Java` programs for this problem set. However, the only file that must be uploaded to `Web-CAT` is `Checkout.java`.
- Write a `Java` program in the file `Cake.java` that encapsulates the properties of the bakery shop's cakes. Specifically, a cake should have a **weight** associated with it, and a **price per kilogram**, which are used to determine cost. Note that `Cake` is a subclass of the abstract class `Confection`. The `Confection` class has an abstract method called `getCost()`, so `Cake` must provide an implementation for this.
- Write a `Java` program in the file `Cookie.java` that encapsulates the properties of the bakery shop's cookies. Specifically, a cookie should have a **quantity** associated with it, and a **price per dozen**, which are used to determine cost. Note that `Cookie` is a subclass of the abstract class `Confection`. The `Confection` class has an abstract method called `getCost()`, so `Cookie` must provide an implementation for this.

- Write a `Java` program in the file `Checkout.java` that performs the actions of the bakery shop's cash register. Specifically, you are required to implement the following methods:
  - `public int numberOfItems()`
    This method returns the quantity of goods that the customer has purchased.
  - `public double totalCost()`
    This method returns the total cost of the goods that the customer has purchased.
  - `public double totalTax()`
    This method returns the amount of tax that has been imposed on the total cost of the goods that have been purchased. **Note that the tax rate is `6.5%`.**

Testing

- In order to run the `JUnit` test bench, simply click on the `Run Tests` button. If you are missing this button, then right-click on the module `CheckoutJUnitTest` and select `All Tests`. Then, you should see a `BlueJ: Test Results` window appear.
- If your `JUnit` test is successful, you should see a green bar appear. Also, each of the specific testing functions should have a green checkmark in front of them. If your test is unsuccessful, then a red bar will appear, and you will need to correct the errors.

Submission

- Submit your `Java` program by uploading it to the `Web-CAT` automated grading platform. Click on the following link:
  `http://ec2-54-65-207-33.ap-northeast-1.compute.amazonaws.com:8080/Web-CAT/WebObjects/Web-CAT.woa`