ASSESSMENT 1

OVERVIEW

Create a Java console application that utilizes loops. The application simulates the idea of a customer ordering from a restaurant that serves only burgers and soda. It takes the customer's order one item at a time, then prints the subtotal and total with tax.

BUILD SPECIFICATIONS

The assignment is worth ten points, one for each of the test cases below. **Pay special attention to the spelling and capitalization of the items in bold**.

For this challenge, you will need to create a new Java Project named **Assessment1b**. You will also need to create a class named **Assessment** that lives inside a package named **co.grandcircus**.

1. 1pt - In your **main** method, write a loop that repeatedly presents the user with a menu of the following numbered options. The output must be formatted with each option on its own line exactly like this.
   ```
   1. Burger
   2. Soda
   3. Exit
   ```
2. 1pt - Continue looping until the user enters **3**. (3 for Exit)
3. Each time through the loop...
   a. If the selection is 1, add a burger to the order.
   b. If the selection is 2, add a soda to the order.
   c. If the selection is 3, the loop ends.
   d. 1pt - If any other number is entered, print "**Sorry I didn't understand, let's try again.**". The loop then continues to the next run
4. In the same class, create an additional **public static** method named **calculateSubtotal( )** that accepts two **int** parameters corresponding to the number of burgers and sodas ordered. Call this method and print the subtotal of the user's order. It **MUST** be printed on a line along with the word "**Subtotal**". Each burger costs **$2.50**. Each soda costs **$1.25**. (Note: the number formatting is not important as long as the value is correct. For example, an order of 3 burgers and 0 sodas can simply print 7.5 for the subtotal.) It is possible to get partial credit based on the following cases, up to 3pts:
   a. 1pt - The subtotal is calculated correctly when order contains only burgers.
   b. 1pt - The subtotal is calculated correctly when order contains only sodas.
   c. 1pt - The subtotal is calculated correctly for an order of several burgers and sodas.
5. 1pt - Next create another method that accepts the subtotal as a parameter and returns nothing. This method named **addTaxDisplayTotal( )** should print the total by adding 7% to the subtotal. It **MUST** be printed on a line along with the word "**Total**".

6. 2pt - Next print a line with one asterisk for each burger purchased. It **MUST** be printed on a line along with the word "**Burger**". For example, if 4 burgers were added to the order:
   `Burgers: ****`
7. 1pt - Next print a line with one asterisk for each soda purchased. It **MUST** be printed on a line along with the word "**Soda**". For example, if 2 sodas were added to the order:
   `Sodas: **`

EXAMPLE RUN

```
1. Burger
2. Soda
3. Exit
Select an option: {User enters 1}
1. Burger
2. Soda
3. Exit
Select an option: {User enters 1}
1. Burger
2. Soda
3. Exit
Select an option: {User enters 2}
1. Burger
2. Soda
3. Exit
Select an option: {User enters 7}
Sorry I didn't understand, let's try again.
1. Burger
2. Soda
3. Exit
Select an option: {User enters 3}
Subtotal: $6.25
Total: $6.6875
Burgers: **
Sodas: *
```