

CSE1322L Assignment 2 - Fall 2024

Introduction:

In this assignment, we'll write a simple program capable of generating bills for some kind of business. The bills will have items in them, which themselves can contain subitems. The items and subitems will all have individual descriptions and dollar amounts. Once a bill is done, we can print it to the screen, with an option to generate another one or quit. Each generated bill needs to have their own unique identifying number.

Our program won't have the capability of storing or loading bills from disk (we'll learn how to do this in a later module). For now, we'll just assume that the business owners have to save bills manually, or said bills will be lost.

Note that this assignment deals with a data type which can be expanded at runtime called ArrayList in Java and List in C#. The assignment will only refer to it as "ArrayList", but C# coders should be aware that they must use a List when the assignment asks for an arraylist.

Requirements

The features described below must be in your program.

- A total of four classes: the driver, Bill, BillingItem, and BillingSubItem.
- Bill class:
 - Must have 4 fields:
 1. An ArrayList which holds BillingItems, called "items"
 2. A static integer called "nextId", initialized at 0
 3. An integer called "id"
 4. A static double called "taxPercentage", initialized at 0
 - It must only feature the default constructor, which initializes the arraylist, sets id with the value of nextId, and then increments nextId by 1.
 - It must feature the following methods:
 1. setTax(), which is static, takes in a double, and returns nothing. The double in the parameter updates taxPercentage only if the parameter is greater than or equal to 0. Otherwise, this method does nothing.
 2. A getter for taxPercentage, which is static.
 3. addItem(), which takes in a BillingItem and returns nothing. It adds said item to items.
 4. removeItem(), which takes in a BillingItem and returns nothing. It removes said item from items.
 5. getItem(), which takes in an integer and returns a BillingItem. If the integer in the parameter is within the bound of items, getItem() returns the item in that position. Otherwise, return null.

6. `seeltems()`, which takes in nothing and returns a string. If `items` is empty, this method returns an empty string. Otherwise, it will return a list of all the items in `items` in the following format:

POSITION. ITEM-DESCRIPTION: \$ITEM-AMOUNT

e.g.: The bill below with 3 items, when calling `seeltems()`, would return the following string:

```
0. Item 1: $100.00
1. Item 4: $30.00
2. Item 5: $120.00
```

7. `calculateSubTotal()`, which takes in nothing and returns a double. This method adds the amount of all `BillingItems` stored inside of `items` and then returns this summation.
8. `calculateTotal()`, which takes in nothing and returns a double. This method adds the amount of all `BillingItems` stored inside of `items` and then returns this summation multiplied by the tax percentage.
 - NOTE: The tax percentage is stored as a percentage (e.g.: 10 for 10%). This means you will have to convert it to the appropriate decimal number before you can use it to calculate your total. In the example above, if `taxPercentage` is 10, then you would multiply your subtotal by 1.1 to get the total.
9. An override of `toString()`. This override must contain the following information:
 - The Bill's id, in its own line
 - Every `BillingItem` in the Bill (which you can retrieve by calling every `BillingItem`'s `toString()`), each in its own line
 - The subtotal, in its own line
 - The total, in its own line

- **BillingItem class**

- Must have 3 fields:
 1. An `ArrayList` which holds `BillingSubItems`, called "subitems"
 2. A double called "amount"
 3. A string called "description"
- It must have 3 constructors:
 1. The default constructor, which initializes all fields with 0 or an empty string, as appropriate.
 2. An overloaded constructor, which takes in a double to initialize amount, and initializes description with an empty string.
 3. An overloaded constructor, which takes in a double to initialize amount and a string to initialize description.
 4. All constructors should initialize subitems.

- It must have the following methods:
 1. `getAmount()`, which takes in nothing and returns a double. If subitems is empty, it returns amount. Otherwise, this method adds up the amount of all of the `BillingSubItems` in subitems, returning that summation. If subitems has elements in it, do not add the BillingItem's amount to the summation.
 2. `setAmount()`, which takes in a double and returns nothing. It updates the value in amount with the value in the parameter. Note that no guard for the input is necessary, as we want to allow for negative values in case we want to offer a discount on something.
 3. `addSubItem()`, which takes in a `BillingSubItem` and returns nothing, adding it to subitems.
 4. `removeSubItem()`, which takes in a `BillingSubItem` and returns nothing, removing it from subitems.
 5. `getSubItem()`, which takes in an integer and returns a `BillingSubItem`. If the integer is in bounds of subitems, return the `BillingSubItem` at that position. Otherwise, return null.
 6. `seeSubItems()`, which takes in nothing and returns a string. This method works exactly as `seeItems()` in `Bill`, but it uses the `BillingSubItems` stored inside subitems.
 7. A getter and a setter for description.
 8. An overload of `toString()`. If the `BillingItem` has no `BillingSubItems`, it will simply return its description and its amount. If the `BillingItem` does have `BillingSubItems`, it must return:
 - Its own description, in its own line
 - Every `BillingSubItem` in the `BillingItem` (which you can retrieve by calling every `BillingSubItem`'s `toString()`), each in its own line
 - Every `BillingSubItem` must be preceded by a tab character (“\t”).

e.g.: There are 2 Billing Items below. The first one has no `BillingSubItems`, while the second one has 2.

```
Item 1: $100.00
Item 4
    SubItem 1: $10.00
    SubItem 2: $20.00
```

- `BillingSubItem` class:
 - Must have 2 fields:
 1. A double called “amount”
 2. A string called “description”
 - It only has a single overloaded constructor, which takes in a double and a string, setting them to the appropriate fields.
 - It has getters and setters for both fields.

- It has an overloaded of the toString() method, which returns both fields in the BillingSubItem in the following format:

DESCRIPTION: \$AMOUNT

SubItem 1: \$10.00

- Driver class
 - Creates a Bill object
 - In a loop, prompts the user for the following options:
 1. **Add item:** Prompts the user for the BillingItems's description and amount, adding it to the Bill
 2. **Remove item:** Shows a list of all available BillingItems and prompts the user for which item they want to delete. If the user picks a valid one, remove it from Bill. Otherwise, print an error message.
 3. **Add subitem:** Shows a list of all available BillingItems and prompts the user for a BillingItem. If the user picks a valid one, prompt them for the BillingSubItem's description and amount, adding this new BillingSubItem to the selected BillingItem. Otherwise, print error messages as appropriate.
 4. **Remove subitem:** Shows a list of all available BillingItems and prompts the user for a BillingItem. If the user picks a valid one, show a list of all available BillingSubItems in that BillingItem and prompt them for a BillingSubItem. If the user picks a valid one, remove it from the BillingItem. Otherwise, print error messages as appropriate.
 5. **See tax:** prints the current tax percentage
 6. **Set tax:** prompts the user for a new tax percentage, updating it on Bill.
 7. **Preview bill:** Prints out the Bill's toString().
 8. **Finish:** prints out the Bill's toString(). It will then ask the user if they would like to create another Bill. If they say yes, re-initialize the Bill object and start the loop again. Otherwise, terminate the program.

Considerations

- This assignment may seem intimidating, but that's just because of the number of things you have to do; the assignment itself isn't very hard, so don't be discouraged.
- Remember that you will get partial credit for partial work. Try to deliver as much of the assignment as you can.
- You may have noticed that there is some overlap between what you are being asked to write for BillingItem and BillingSubItem. In the future, we'll learn how to avoid some of this repetition.
- You may add any other helper methods you believe are necessary, but they won't count towards your grade.

Example: [User input in red]

[Bill Generator]

New bill created.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **2**

There are no items to remove!

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **4**

There are no items from which you can remove subitems!

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **1**

Enter the item's description: **Lab work (Blood)**

Enter the item's amount: **370.45**

Item added to bill.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem

- 5. See tax
- 6. Set tax
- 7. Preview bill
- 8. Finish

Select option: **1**

Enter the item's description: **GP Office Visit**

Enter the item's amount: **212.30**

Item added to bill.

- 1. Add item
- 2. Remove item
- 3. Add subitem
- 4. Remove subitem
- 5. See tax
- 6. Set tax
- 7. Preview bill
- 8. Finish

Select option: **1**

Enter the item's description: **Insurance coverage**

Enter the item's amount: **-250**

Item added to bill.

- 1. Add item
- 2. Remove item
- 3. Add subitem
- 4. Remove subitem
- 5. See tax
- 6. Set tax
- 7. Preview bill
- 8. Finish

Select option: **8**

=====

0

Lab work (Blood): \$370.45

GP Office Visit: \$212.30

Insurance coverage: \$-250.00

Subtotal: \$332.75

Total + Tax: \$332.75

=====

Would you like to create a new bill? (say "no" to terminate) **yes**
New bill created.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **1**

Enter the item's description: **Knockya cellphone**

Enter the item's amount: **200**

Item added to bill.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **1**

Enter the item's description: **AyePhone**

Enter the item's amount: **650**

Item added to bill.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **1**

Enter the item's description: **EIVello laptop**

Enter the item's amount: **1200**

Item added to bill.

1. Add item
2. Remove item

3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$200.00
1. AyePhone: \$650.00
2. ElVello laptop: \$1200.00

Select an item: **0**

Enter subitem's description: **Monthly instalments**

Enter subitem's amount: **20**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$20.00
1. AyePhone: \$650.00
2. ElVello laptop: \$1200.00

Select an item: **0**

Enter subitem's description: **Standard data plan**

Enter subitem's amount: **55**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$75.00

1. AyePhone: \$650.00
2. ElVello laptop: \$1200.00

Select an item: **0**

Enter subitem's description: **Standard damage warranty**

Enter subitem's amount: **25**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$100.00

1. AyePhone: \$650.00

2. ElVello laptop: \$1200.00

Select an item: **1**

Enter subitem's description: **Monthly instalments**

Enter subitem's amount: **30**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$100.00

1. AyePhone: \$30.00

2. ElVello laptop: \$1200.00

Select an item: **1**

Enter subitem's description: **Advanced data plan**

Enter subitem's amount: **80**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$100.00

1. AyePhone: \$110.00

2. ElVello laptop: \$1200.00

Select an item: **1**

Enter subitem's description: **Total damage warranty**

Enter subitem's amount: **200**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$100.00

1. AyePhone: \$310.00

2. ElVello laptop: \$1200.00

Select an item: **2**

Enter subitem's description: **Monthly instalments**

Enter subitem's amount: **120**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **7**

The current contents of the bill are:

=====

1

Knockya cellphone

Monthly instalments: \$20.00

Standard data plan: \$55.00

Standard damage warranty: \$25.00

AyePhone

Monthly instalments: \$30.00

Advanced data plan: \$80.00

Total damage warranty: \$200.00

EIVello laptop

Monthly instalments: \$120.00

Subtotal: \$530.00

Total + Tax: \$530.00

=====

1. Add item
 2. Remove item
 3. Add subitem
 4. Remove subitem
 5. See tax
 6. Set tax
 7. Preview bill
 8. Finish
- Select option: **5**
Current tax is 0.00%

1. Add item
 2. Remove item
 3. Add subitem
 4. Remove subitem
 5. See tax
 6. Set tax
 7. Preview bill
 8. Finish
- Select option: **6**
Enter new tax %: **10**
Tax set.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **7**

The current contents of the bill are:

=====

1

Knockya cellphone

Monthly instalments: \$20.00

Standard data plan: \$55.00

Standard damage warranty: \$25.00

AyePhone

Monthly instalments: \$30.00

Advanced data plan: \$80.00

Total damage warranty: \$200.00

EIVello laptop

Monthly instalments: \$120.00

Subtotal: \$530.00

Total + Tax: \$583.00

=====

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **2**

0. Knockya cellphone: \$100.00

1. AyePhone: \$310.00

2. EIVello laptop: \$120.00

Select item to remove: **1**

Item removed from bill.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **4**

0. Knockya cellphone: \$100.00
1. ElVello laptop: \$120.00

Select an item: **1**

0. Monthly instalments: \$120.00

Select subitem to remove: **0**

Subitem removed from item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **4**

0. Knockya cellphone: \$100.00
1. ElVello laptop: \$1200.00

Select an item: **0**

0. Monthly instalments: \$20.00
1. Standard data plan: \$55.00
2. Standard damage warranty: \$25.00

Select subitem to remove: **1**

Subitem removed from item.

1. Add item
2. Remove item
3. Add subitem

4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **3**

0. Knockya cellphone: \$45.00
1. ElVello laptop: \$1200.00

Select an item: **0**

Enter subitem's description: **Advanced data plan**

Enter subitem's amount: **85.50**

Subitem added to item.

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **2**

0. Knockya cellphone: \$130.50
1. ElVello laptop: \$1200.00

Select item to remove: **2**

Invalid item to remove!

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **4**

0. Knockya cellphone: \$130.50
1. ElVello laptop: \$1200.00

Select an item: **1**

That item has no subitems!

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **4**

- 0. Knockya cellphone: \$130.50
- 1. EIVello laptop: \$1200.00

Select an item: **0**

- 0. Monthly instalments: \$20.00
- 1. Standard damage warranty: \$25.00
- 2. Advanced data plan: \$85.50

Select subitem to remove: **3**

Invalid subitem to remove!

1. Add item
2. Remove item
3. Add subitem
4. Remove subitem
5. See tax
6. Set tax
7. Preview bill
8. Finish

Select option: **8**

=====

1

Knockya cellphone

Monthly instalments: \$20.00

Standard damage warranty: \$25.00

Advanced data plan: \$85.50

EIVello laptop: \$1200.00

Subtotal: \$1330.50

Total + Tax: \$1463.55

=====

Would you like to create a new bill? (say "no" to terminate) **no**

Shutting off...

Submitting your answer:

Please follow the posted submission guidelines here:

<https://ccse.kennesaw.edu/fye/submissionguidelines.php>

Ensure you submit before the deadline listed on the lab schedule for CSE1322L here:

<https://www.kennesaw.edu/ccse/first-year-experience/cse-1322-lab.php>