CIS263AA Programming Assignment 14

Assignment Goal:

In this chapter you learned about lambda expressions and method references.

Assignment Specifications:

Create a program call InvoiceSummary that allows the user to enter the following data:

* Customer name
* Customer number
* Invoice subtotal

You will calculate a discount, tax, and total and display the summary on the monitor.

Create three static class level lambda expressions using the predefined functional interface Supplier <T> for the following where <T> is a float:

* Discount limit of 500
* Discount rate of 5%
* Tax rate of 8.25%

Create a lambda expression using the predefined functional interface Function <T,R> to calculate the amount of the discount using the class level lambdas for your comparison and calculation. Subtotals greater than or equal to $500 will receive a discount, otherwise the discount is 0.

Create a lambda expression using the predefined functional interface Function <T,R> to calculate the amount of tax using the class level lambda expression.

Create a functional interface that accepts three values (subtotal, discount, tax) and returns the total. Create a lambda expression to calculate the total.

Create a lambda expression to display the customer name, number, subtotal, discount, tax, and total on the monitor.

Use the lambda expressions to calculate the discount, tax, and total and display the invoice summary.

Deliverables (what you are to submit):

1. Planning document for InvoiceSummary.
   1. Program Outline
   2. Methods
   3. Data Items
   4. Sample Output
   5. Test Data
2. Your completed project folder in zip format.

CIS263AA Programming Assignment 13 (InvoiceSummary)

Name: \_\_\_\_\_Daniel Cender\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Program Outline:** < This is an outline of what your program is to do. Be detailed. >

Take customer input to populate customer name, number, and invoice subtotal.

Pass input to lambda expressions

Display invoice summary

**Methods:** < This is a list of methods you will define in your program. >

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Function** | **Access Modifier(1)** | **Method Name** | **Parameters (dataType identifier)** | **Return Type (2)** |
| Main method: | public static | main() |  | void |
| Calculate | public | calculate() | float s, float d, float t | float |
| Invoice | public | display() | String n, int nbr, float sub, float dis, float tax, float t | void |

1. Access Modifier: local, public, private, protected

2. Return Type: void, string, char, byte, short, integer, long, double, float, boolean, object, etc.

**Data Items:** < This is a list of fields (variables, constants, and objects you will need. >

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Data Item** | **Source (1)** | **Access Modifier (2)** | **Data Type (3)** | **Identifier** | **Notes** |
| Customer Name | input | local | string | customerName |  |
| Customer Number | input | local | string | customerNbr |  |
| Invoice Subtotal | input | local | float | invoiceSubtotal |  |
| Discount Total | calculated | local | float | discountTotal |  |
| Total Tax | calculated | local | float | totalTax |  |
| Total | calculated | local | float | total |  |
| param i | parameter | local | float | i |  |
| param t | parameter | local | float | t |  |
| param s | parameter | local | float | s |  |
| param dis | parameter | local | float | d |  |
| param name | parameter | local | String | n |  |
| param nbr | parameter | local | int | nbr |  |
| param subtotal | parameter | local | float | sub |  |
| param tax | parameter | local | float | tax |  |
| param total | parameter | local | float | t |  |
| Discount Limit | lambda | local | Supplier | discountLimit |  |
| Discount Rate | lambda | local | Supplier | discountRate |  |
| Tax Rate | lambda | local | Supplier | taxRate |  |
| Calculate Discount | lambda | local | Function | calculateDiscount |  |
| Calculate Tax | lambda | local | Function | calculateTax |  |
| Calculate Total | lambda | local | Calculate | calculateTotal |  |
| Display Data | lambda | local | Invoice | displayData |  |

1. Source (where the data comes from): calculated, input, constant, parameter, instance, object

2. Access Modifier: local, public, private, protected

3. Data Type: string, char, byte, short, integer, long, double, float, boolean, object, etc.

**Sample Output:** < What will the user see? >

Please input a customer’s name:

XXXXXX

Please input a customer number:

99999

Please input an invoice subtotal:

9999.99

Customer Name: XXXXX XXXXX

Customer Number: 99999

Invoice Subtotal: 999.99

Total Discount: 99.99

Tax: 9.99

Invoice Total: 9999.99

**Test Data:** < How will you prove your program works? >

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identifier** | Case 1 | Case 2 | Case 3 | Case 4 |
| customerName | Dan C | John Doe | Sally | Sally |
| customerNbr | 1004 | 111 | 604 | three |
| invoiceSubtotal | 34.66 | 650 | 1520.50 | error |
| discountTotal | 0 | 32.5 | 76.02 | error |
| totalTax | 2.85 | 50.94 | 119.1696 | error |
| total | 37.51 | 668.44 | 1563.64 | error |

Note: You made more or fewer test cases depending on your application.1444.48