CIS263AA Programming Assignment 1

Assignment Goal:

For you first program, we will put to use skills you should have mastered in Java I.

Assignment Specifications:

You will create a report of sales data. First create an object class called **Sales** containing the following data:

item number 8 digits, 0 decimals place

description 40 characters

quantity 9 digits, 0 decimal places

sales amount 11 digits, 2 decimal places

Include a constructor that accepts all variables. All data is to be defined with a private access modifier.

Write a program called **SalesReport** that allows the user to enter sales data for up to 100 items. Each entry is to be stored in an instance of the Sales class. After the data has been entered, compute a total of the quantity and sales amount.

Produce a formatted report on the monitor showing each of the 4 instance variables of the Sales class and the percent of sales to the total sales amount. At the end of the report display the number of items entered, total quantity and total sales amount.

Deliverables (what you are to submit):

1. **Sales** planning document.
   1. Methods
   2. Data items
   3. Sample output
   4. Testing criteria
2. **SalesReport** planning document.
   1. Program Outline
   2. Methods
   3. Data items
   4. Sample output
   5. Testing criteria
3. Your completed project folder in zip format.

CIS263AA Programming Assignment 1 (Sales Class)

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

**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)** |
| Constructor | public | Sales() | int itemNum, String description, int quantity, long salesAmt |  |
| Get Item Num | public | getItemNum() |  | int |
| Set Item Num | public | setItemNum() | int num | int |
| Get Description | public | getDescription() |  | String |
| Set Description | public | setDescription() | String description |  |
| Get Quantity | public | getQuantity() |  | int |
| Set Quantity | public | setQuantity() | int num |  |
| Get Sales Amt | public | getSalesAmt() | none | float |
| Set Sales Amt | public | setSalesAmt() | float sale |  |
| Display Data | public | displayData() | none | none |

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** |
| Item Number | Instance | Private | Integer | itemNum |  |
| Description | Instance | Private | String | description |  |
| Quantity | Instance | Private | Integer | quantity |  |
| Sales Amount | instance | Private | Long | salesAmt |  |

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? >

Item Number: XXXXXXXX

Description: XXXXXXXX XXXXX

Quantity: 9,999

Sales Amount: 9,999,999.99

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identifier** | Item 1 | Item 2 | Item 3 | Item 4 |
| itemNum | 12345 | 54321 | 9999 | 6785 |
| description | An Item | Another item | A friend’s toy | A little widget |
| quantity | 4 | 67 | 243 | 5777 |
| salesAmt | 34.65 | 50 | 3255.60 | 65000 |
|  |  |  |  |  |
| totalItems | 6091 | 6091 | 6091 | 6091 |
| totalSales | 68340.25 | 68340.25 | 68340.25 | 68340.25 |
| percentSale | .05% | .073% | 4.76% | 95.1% |

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

CIS263AA Programming Assignment 1 (SalesReport)

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

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

Create instance of Sales object upon user input

Display report of items entered

**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 |
|  |  |  |  |  |

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** |
| Sales | Instance | public | Object | Sales |  |
| Total Items Sold | calculated | public | integer | totalItems |  |
| User Entries | calculated | public | integer | entries |  |
| Total Sales | calculated | public | float | totalSales |  |
| Sales’ Item Array | object | public | object | items[] |  |

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? >

Item Number: XXXXXXXX

Description: XXXXXXXX XXXXX

Quantity: 9,999

Sales Amount: 9,999,999.99

Percent of Total Sales: 33.33%

(above report repeats for each item)

Number of Items Entered: 9

Total Quantity: 9999

Total Sales Amount: 99,999,999

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Identifier** | Item 1 | Item 2 | Item 3 | Item 4 |
| itemNum | 12345 | 54321 | 9999 | 6785 |
| description | An Item | Another item | A friend’s toy | A little widget |
| quantity | 4 | 67 | 243 | 5777 |
| salesAmt | 34.65 | 50 | 3255.60 | 65000 |
| entries | 1 | 2 | 3 | 4 |
| totalItems | 6091 | 6091 | 6091 | 6091 |
| totalSales | 68340.25 | 68340.25 | 68340.25 | 68340.25 |
| percentSale | .05% | .073% | 4.76% | 95.1% |

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