Coding Assignment 1

CSE 1325

Create a pencil machine with the behaviors described below and as shown in the video that is part of the Canvas assignment.

`final int` constant

Pencil price must be a `final int` constant and not hardcoded throughout the program. Changing the price of a pencil should only require changing the constant and recompiling and not anything else in the program. Start with a cost of sixty cents. You must use an integer value - not a decimal value – 60 is correct and 0.60 is wrong.

Enumeration

Create an enumeration named `ACTION` inside the `public class` of your program but outside of `main()`. Set the values as follows…

`DISPENSECHANGE, INSUFFICIENTCHANGE, INSUFFICIENTFUNDS, EXACTPAYMENT`

`displayMoney()`

Create a function to display any money amounts as a string containing `$xx.yy` where `xx` is the number of dollars and `yy` is the number of cents. Anywhere a money amount needs to be displayed by the program, this function must be called.

`PencilMenu()`

Create a function to handle the menu. Your menu should look like the menu in the video. It should print the menu and handle getting the user's input. The call to the menu function should be in a conditional loop in `main()` so that the program will keep running until the user chooses to exit. The validation loop inside the function should use the user input to determine the loop continues or ends. The `Scanner` used in the menu function should be declared locally. A globally declared `Scanner` will result in a 0 for this assignment.

`buyPencils()`

Create a function to buy pencils. This function should do the following…

a) Does not print anything or accept anything – everything is passed in/passed back
b) Check if exact payment was entered or not. If the payment exceeds the cost, then change must be given.
c) Check if enough change is already in the machine to give back change to the user. Money from the payment itself may not be used for providing change for the payment during the current transaction. For example, if the required payment is $6.00 and the user gives a payment of $10.00 and there is $2.00 of change in the machine, then the machine should not accept the payment since $4.00 of change is required and the machine only contains $2.00 of change. Payment amount should be added to change level after change has been given.
d) Check if insufficient payment was given
e) Decrement inventory level and increment change level as needed.
f) This function will have a return type of `ACTION` and that tells `main()` what action/information should be printed out in main(). A `switch` statement should be used in `main()` where the `switch` is based on the `action` that was returned by the buy a pencil function.

`DISPENSECHANGE` - Pencils were sold and change was given.

`EXACTPAYMENT` - Pencils were sold and exact payment was given and no change was needed

`INSUFFICIENTCHANGE` - not enough change was available to complete the purchase

`INSUFFICIENTFUNDS` - the provided payment was insufficient (no sale took place)

`default` - something unknown happened

The values returned in the `ACTION` variable must be enumerated to make your program more readable. For example, set your `ACTION` variable to an enumeration of `OK` rather than a value of `0`.

Tracking inventory level and change level

The variables for tracking the inventory level and change level will be declared in `main()` and will be passed to your function to buy pencils using an integer array so that the increments/decrements made in your buy pencils function continue to exist outside of the function. Variables in Java are passed by value so to actually change the inventory level and the change level in the function, you must add the values to an integer array and pass the array rather than the individual variables. To facilitate uniform testing for the GTAs, please initialize inventory level to 100 and change level to 500 (500 cents which is $5.00 – we will not use floats to represent money – we will be using the number of cents).

`main()`

Create a switch statement in `main()` that acts upon the menu choice returned by your menu handling function. The default case prints "`Invalid menu option`". The `switch` will reside inside a `while` or `do-while` that controls redisplaying the menu until the menu option to exit the program is chosen. The loop must be the menu choice as the condition for continuing or not.

Menu option 1

When menu option 1 is chosen, `main()` should check if the inventory level is 0 or not. See video for what to display when inventory level is 0. If the inventory level is not 0, then print out how much a pencil costs using the `final int` constant created earlier and function `displayMoney()`. Prompt user for quantity as shown in the video. A while or do while loop should be used to verify that the quantity is greater than 0 and less than the inventory level. The condition of the loop must use the entered quantity to decide whether to continue or stop. Once an in range quantity is input, use the pencil price and the quantity along with `displayMoney()` to print the total cost. Payment should be stored in an integer (there should not be **any** decimal/floating numbers in this assignment). Call your buy pencil function and pass in the necessary values.

Menu option 2 and 3

Menu option 2 will display the value of the inventory level variable. Menu option 3 will display the value of the level of change – function `displayMoney()` **must** be called to display the integer change level value – it will return a string that will display the change level using `$xx.yy`.

`action`

Based on the `action` returned by your buy a pencil function, `main()` should print the messages to the user shown in the video. Use a `switch` statement based on the value of `action`. Case statements should use the enumerated values.

Test Cases

Test	Test Case Description	Expected Result
1	0 is entered for menu choice	program completes without any further printing or prompts
2	1 is entered for menu choice 3 pencils are requested for purchase Exact payment is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is thanked for exact payment. Menu is redisplayed.
3	1 is entered for menu choice 3 pencils are requested for purchase Over payment not exceeding available change is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is given the correct change. Menu is redisplayed.
4	1 is entered for menu choice 3 pencils are requested for purchase Over payment exceeding available change is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is informed that the pencil machine does not have enough change and cannot accept the payment. Menu is redisplayed.
5	1 is entered for menu choice 3 pencils are requested for purchase Under payment of greater than 0 is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is informed the payment was insufficient and pencils will not be dispensed. Menu is redisplayed.
6	1 is entered for menu choice 3 pencils are requested for purchase Under payment of 0 is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is informed the payment was insufficient and pencils will not be dispensed. Menu is redisplayed.
7	1 is entered for menu choice 3 pencils are requested for purchase Under payment of less than 0 is given	Pencil purchase price is properly displayed as 3 times the pencil price. User is informed the payment was insufficient and pencils will not be dispensed. Menu is redisplayed.
8	2 is entered for menu choice	Inventory level is properly displayed. Option should be run before and after test case 2 to show proper inventory level decreases
9	3 is entered for menu choice	Change level is properly displayed. Option should be run before and after test case 2 to show proper change level increases.
10	9 is entered for menu choice	Invalid menu option message is displayed and menu is redisplayed.
11	1 is entered for menu choice -3 pencils are requested for purchase	User is told “Cannot sell that quantity of pencils. Please reenter quantity”
12	1 is entered for menu choice 0 pencils are requested for purchase	User is told “Cannot sell that quantity of pencils. Please reenter quantity”
13	1 is entered for menu choice 200 pencils are requested for purchase (inventory should start at 100)	User is told “Cannot sell that quantity of pencils. Please reenter quantity”
14	2 is entered for menu choice to retrieve current inventory level. 1 is entered for menu choice All pencils in inventory are purchased. 1 is entered for menu choice	After purchasing all available pencils, the second run of menu choice 1 should display a message stating that the pencil dispenser is out of pencils.
15	A letter is entered at the menu prompt	Exception is thrown and program ends
16	1 is entered for menu choice A letter is entered for the number of pencils to purchase	Exception is thrown and program ends
17	1 is entered for menu choice 3 pencils are requested for purchase A letter is entered for the payment.	Exception is thrown and program ends