# CSE1341 - Lab 3 Programming Assignment
## *Algorithms with multiple methods*
## Assignment Due Saturday, September 29th, 2018 at 6:00 AM

This assignment will require a Pre-Lab and a Lab.

**Pre-Lab (10 POINTS)**: Write out the pseudocode for Part I of this program.

**Lab**: Write the code based on your design.

**Part I (45 POINTS)**

Our third Lab has us in the role of a DVD Store Owner...let's just pretend this is before Netflix ☺. We're in this business to make some money so we buy DVDs from our supplier for $9 each but sell each DVD to our customers for $10/each. This program is to simulate our inventory system where we can facilitate buying, selling and querying for the current number of DVDs, as well as the cash on hand, we have.

**When the program starts it**:
- Sets the number of DVDs we currently have to 10 and sets our cash on hand to $100.
- Displays a menu saying:
  Welcome to DVDs R Us. Please choose from the options below:
  1 – Buy DVDs
  2 – Sell DVDs
  3 – Check how many DVDs we have in stock
  4 – Check how much cash we have
  9 – Exit the program
  Enter Option:

**Requirements**:
- Within a loop, display the menu.   If the user enters an option that is not 1, 2, 3, 4 or 9, display the message "This option is not acceptable" and loop back around to redisplay the menu.   Continue in this manner until a correct option is input.
- If option 1 is selected, prompt the user for the number of DVDs to buy, and once we have this number, check to see if we can afford the DVDs (based on the formula: number of DVDs to be bought * 9 <= total cash we have)
  - o If we can, then we add the newly purchased DVDs to our current DVD count and reduce our money to whatever is left. We also

output the number of DVDs we have (with the prefix text "Number of DVDs:") and the cash we have (with the prefix text "Cash available:") <u>after the transaction</u>, and re-display the main menu.
- o If we cannot, then we display a message saying – "You do not have enough money for this transaction".
- If option 2 is selected, prompt the user for the number of DVDs to sell, and once we have this number, check to see if we have enough to sell
    - o If we can, then we reduce the DVD count by the amount sold and increase the money we have (by number of DVDs sold * 10). We also output the number of DVDs we have left (with the prefix text "Number of DVDs:") and the cash we have (with the prefix text "Cash available:") <u>after the transaction</u>, and re-display the main menu.
    - o If we cannot, then we display a message saying – "You do not have enough DVDs for this transaction".
- If option 3 is selected, we output the number of DVDs we have left (with the prefix text "Number of DVDs:") and re-display the main menu.
- If option 4 is selected, we output the cash we have (with the prefix text "Cash available:") and re-display the main menu.
- If option 9 is selected, then the program will end. Display a message saying "Thank you for shopping with us!  Please return soon!".

Sample Output:

```
Welcome to DVDs R Us. Please choose from the options below:
1 – Buy DVDs
2 – Sell DVDs
3 – Check how many DVDs we have in stock
4 – Check how much cash we have
9 – Exit the program
Enter Option: 1

Enter the numbers of DVDs to buy:3
Number of DVDs: 13
Cash available: $73.00

Welcome to DVDs R Us. Please choose from the options below:
1 – Buy DVDs
2 – Sell DVDs
3 – Check how many DVDs we have in stock
4 – Check how much cash we have
9 – Exit the program
```

```
Enter Option: 2

How many Dvds to sell? 8
Number of DVDs: 5
Cash available: $153.00

Welcome to DVDs R Us. Please choose from the options below:
1 - Buy DVDs
2 - Sell DVDs
3 - Check how many DVDs we have in stock
4 - Check how much cash we have
9 - Exit the program
Enter Option: 3

Number of DVDs: 5

Welcome to DVDs R Us. Please choose from the options below:
1 - Buy DVDs
2 - Sell DVDs
3 - Check how many DVDs we have in stock
4 - Check how much cash we have
9 - Exit the program
Enter Option: 4

Cash available: $153.00

Welcome to DVDs R Us. Please choose from the options below:
1 - Buy DVDs
2 - Sell DVDs
3 - Check how many DVDs we have in stock
4 - Check how much cash we have
9 - Exit the program
Enter Option: 9

Thank you for shopping with us! Please return soon!.
```

## Part II (45 POINTS)

Based on Part I, our business has been booming!  For this reason, we must enhance our system program so it becomes more manageable and expandable.  We can do this by changing our single method program into multi methods. Our inventory system still facilitates buying, selling and querying for the current number of DVDs, as well as cash on hand, we have.

First, declare the number of DVDs and cash on hand to be class member variables. Here's a declaration for numOfDvds as an example:

**static int numOfDvds = 10;**

*Note:* A variable declared within a class but outside any method is called a *class member variable* or **field**, in contrast to a local variable defined inside a method. A field's scope extends from the class's opening brace to the class's closing brace and reaches into methods regardless of where the field is declared within the class.

**We now want the following methods included:**

**main** – We always need main, but don't want it to do all the work

*Note:* If a method needs to read user input, a good practice is to create a single Scanner object in main() and pass that Scanner object to the method.

**displayMenu** – Will display the menu and return the number the user selects.

**buyDVD** – Move the logic you created in Part I for option 1 to this method. The number of DVDs and amount of cash on hand will be displayed by calling the method getInventory.

**getInventory** – Will receive and display the number of DVDs we have and the cash amount we have.

**sellDVD** – Move the logic you have created in Part I for option 2 to this method. The number of DVDs and amount of cash on hand will be displayed by calling the method getInventory.

**getNumberOfDVDs** – This method will take in the number of DVDs and will display the number of DVDs in stock.

**getCashOnHand** – This method will take in the cash on hand we have and will display this amount.

If/When main receives the number 9 from the displayMenu method, the looping structure within main should stop and the program will end. Display a message saying "Thank you for shopping with us! Please return soon!"

Your output should remain the same as in Part I.

NOTES: Comment your program to explain your steps. Each program should compile without errors and should run to produce outputs described for each exercise. The following points will be discounted if the related element is missing or incorrect:

- Output formatting monetary amounts (dollar sign and 2 places after the decimal point) [2 points each]

- Proper names for classes, variables, and methods [1 point each]

- No Comments [5 points]

- Program doesn't compile [5 points for each minor error up to 5 errors provided that after fixing the errors the program compiles. If the program does not compiler after the 5 errors are fixed, partial credit will be given not to exceed 50 points]

- Source code (java file) missing [ 15 points]

- Executable (class file) missing [15 points]

- Both java file and class file missing [100 points]

- Missing method where required [5 points each]