

Assignment 5 Grade & Feedback

23+13 +2 for on time work = 38/40 =96%

Great work Mos. I didn't expect a method inside the switch but that was creative and clean. And you're getting a lot of appreciation from methods I see as your main is simply a loop that goes through the rest of the methods.

I really like how you've formatted the printout very carefully and with a user's experience in mind. Keep that up as it really stands out.

Take a look at passing in variables into your methods, which would make your variables safer and less risk for other methods to modify them. As they are static, they can be modified by any class, or methods. We'll talk more about this in Objects classes.

Assignment 4	
Grading Rubric	Score

<p>Part 1:</p> <p>CoffeeShopAccount</p> <p>[25 points]</p> <ol style="list-style-type: none"> 1. Choose from the menu what action to perform (5 points) 2. Add money into the balance, and see their ending balance. Can be positive/ negative (5 points) 3. Extract money from the balance, and see their ending balance. Can be positive/ negative. (5 points) 4. Set up a 4-digit PIN (5 points) 5. Exit immediately when the option to exit is chosen (5 points) 	<p>5 + 5 + 5 + 5 + 5 = 25 -2</p> <p>-2</p> <p>Good job using static variables. However I had asked for variables to be declared in the main method. We'll talk more about static values and the advantages / disadvantages of using them. You can declare balance in the main then use it as a variable inside your method like this checkBalance(int bal) the return it to the main method then in the main you would capture it with your existing balance var -> balance = checkBalance(balance)</p>
<p>Part 2: Reflect with a buddy (13 points)</p>	<p>13</p>

Rules & Guidelines fr Assignment (-2 pts if not included)

1. If the user enters any other number not in the menu, simply print a message that it is invalid and loop through the menu again.

2. Create at least 4 methods
3. Use a switch case to call the method that performs the operation.
 - E.g. If the user enters 1 to check balance then call that method
4. Create any additional methods you like
5. Name your methods as camelCased, and action-oriented, i.e. actionValue, getBalance, setBalance, spendBalance, etc.
6. Use while loop to display the bank operations and terminate the loop on Exit (i.e. operation 5)
7. Variables to declare in the main method: balance, pin