MS3111 Data Analytics with Excel VBA

Assignment

Instructions:

- Save the workbook containing your work as *nnnnnnnn*.xlsm where *nnnnnnnn* is your CityU EID name (CityU email name). You are reminded to save the workbook as an Excel Macro-Enabled Workbook. Submit your Excel file via the designated link under the Assignments section in Canvas before 06:00 am on 26 Feb 2024. Only the last submitted file will be marked if you submit your file more than once. Late submissions will not be accepted.
- You must not change the appearance of the provided Userform and the properties of any controls on the Userform at design time or runtime unless you are told explicitly to do so.
- You may enter your code into the included subs, but you must not change the code that is already contained in these subs.
- You must specify all utilised properties of Userform controls properly in your code. For example, the Value property of TextBox1 must be specified as TextBox1.Value. Missing the property specification will be considered an error.
- Unless the assigned value of a variable is needed again by any subs while the application is still running, it must be declared inside the sub that uses the variable. If the value of a module-level variable is not to be reused by more than one sub, it will be considered an error.
- Redundant statements (including irrelevant statements or statements not associated with the listed tasks) will be considered errors.
- All code must be processed sequentially according to the order imposed by the task list. That is, Task 1ai must be executed before Task 1aii. Code not executed in the required task order will be considered an error.
- Marks will be deducted for the poor layout of the code. That includes non-properly named or declared variables, poor flow of the program, lack of comment statements, low readability, etc.
- The assignment is not a team assignment. You are expected to complete the assignment by yourself. Discussions between classmates are encouraged, but you must not cross the line between discussion and plagiarism (such as sharing the answer in any way).

Tasks:

UserForm frmNotesChange in workbook Assignment1_2023B.xlsm is designed to change a given pile of $10 notes into an equivalent value of notes in higher face value, namely $1,000 notes, $500 notes, $100 notes, $50 notes, and $20 notes. The application always changes the pile of $10 notes into the highest face value notes whenever applicable. For example, a pile of 1,051 $10 notes will be changed into ten $1,000 notes, one $500 note and 1 $10 note. All

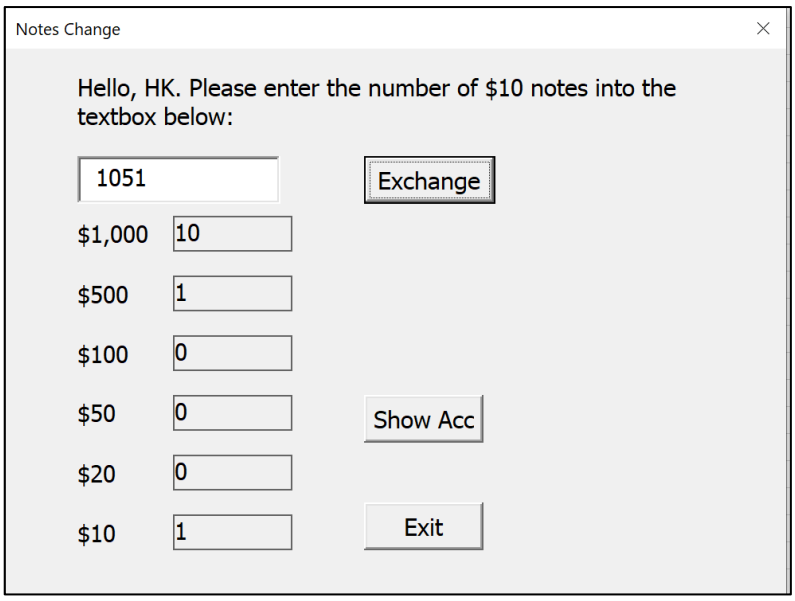
control objects on frmNotesChange have already been named. You must use these names in your code. You are asked to provide code in the application for the following tasks.

Task 1: Before frmNotesChange is drawn to the screen at runtime of the application:

a) Ask for the user's name in an input dialog box.
b) Set the caption of lblYourName as "Hello, xxx. Please enter the number of $10 into the textbox below.", where xxx is the name obtained in Task 1a above.

Task 2: After the user clicks the CommandButton btnExchange on frmNotesChange :

a) Display the number of $1,000 notes, $500 notes, $100 notes, $50 notes, $20 notes, and $10 notes in each caption of Label controls lblNote1000, lblNote500, lblNote100, lblNote50, lblNote20, and lblNote10 respectively on frmNotesChange. The total value of these notes must equal the value of the $10 notes specified by the user in Task 1b. A "0" should be displayed for any notes not involved in the change. The following diagram shows a runtime example of changing a pile of 1,051 $10 notes. {Note: You may assume the user will always enter a proper value into the Textbox txtNumber10. No further checking is required. However, you must ensure the application can process more than 32,767 $10 notes in one go.}



b) Accumulate the numbers of $10 notes specified by the user so that the accumulated number can be recalled in the current runtime session.

Task 3: When the CommandButton btnShowAcc is clicked at runtime, display the message in a message dialog box: "The accumulated number of $10 notes is XXX." The XXX is the total number of $10 notes specified by the user so far in the current runtime session of the application. For example, if the user entered 1051 and 500 in two trials, the message would be "The accumulated number of $10 notes is 1551." {Note: To display the correct number of

$10 notes in the message, frmNotesChange must not be unloaded before clicking btnShowAcc.}

Task 4: Insert an ActiveX CommandButton control onto the 'Sheet1' of the workbook containing the application. Launch the application when the button is clicked.

-End-

Save the workbook containing your work as *nnnnnnnn*.xlsm where *nnnnnnnn* is your CityU EID name (CityU email name). You are reminded to save the workbook as an Excel Macro-Enabled Workbook.