IOE 373 Fall 2024 – Lab 5

Creating a Userform

Objective

- Learn how to create a userform and perform simple commands

Notes

- Userforms **do not** work natively on Mac. You *can* hack a solution to display them, but it is fairly complex and buggy. **Therefore, for the VBA labs in this course, we highly recommend you use Excel on Windows through a physical windows PC or a remote CAEN connection**
- PDFs and PowerPoints use different quotations than VBA and it is difficult to see the slight changes in the quotations. So, if you directly copy code from lecture with quotes it will not run. If you do this (which is okay), make sure to replace the quotation marks

```
'code copied directly from powerpoint won't run
Entry1 = InputBox(“Enter the name”)

'manually changing quotations works
Entry1 = InputBox("Enter the name")
```

Example of Issue with Copying Quotations from PowerPoint or PDF

- Before you submit the assignment, double-check your code runs and functions as expected. You should re-open your submitted file and make sure the form runs.
- **Hint**: For finding the next empty row, we learned helpful syntax in Lecture 10

Task

Following the lab video, we will be creating a userform that uses textboxes, command buttons, and labels to allow a user to input data, as well as copy the inputted data to an Excel sheet.

The Userform takes the following inputs using textboxes:

- First Name
- Last Name
- Uniqname

The userform has two command buttons: 1) “Add” and 2) “Cancel”.

- “Add” will copy the user-entered data to the worksheet with column headers corresponding to the textboxes (First Name, Last Name, Uniqname)
    - Additionally, when the “Add” button is clicked, the columns should be resized, and the data currently entered in the textboxes should reset
- “Cancel” will close/hide the form.

When the userform loads create the column headings in bold (“First Name”, “Last Name”, and “Uniqname”) and resize the columns.

When you are submitting your lab, **make it so that every time that the ADD button is clicked, the new student is entered in the next available row** instead of changing the values of the second row as it is done in the lab video.

Submission

Submit your macro-enabled Excel File (.xlsm)

- Note that **the extension needs to be .xlsm** so your macros can be viewed. Files not saved as the proper type will not be able to be graded and will receive a 0.

Use the file name format: <uniqname>_Lab5.xlsm

Additional References

Information on Userforms:

Userforms combine the capabilities of InputBox and MsgBox to create a more efficient way of interacting with the user. For example, rather than have the user fill out personal information on a sheet, you can create a userform that prompts for the required data (see Figure 1).

Figure 1 Create a custom userform to get more information from the user.

Insert a userform in the VB Editor by selecting Insert, UserForm from the main menu. When a UserForm module is added to the Project Explorer, a blank form appears in the window where your code usually is, and the Controls toolbox appears.

You can resize the form by grabbing and dragging the handles on the right side, bottom edge, or lower-right corner of the userform. To add controls to the form, click the desired control in the toolbox and draw it on the form. Controls can be moved and resized at any time.

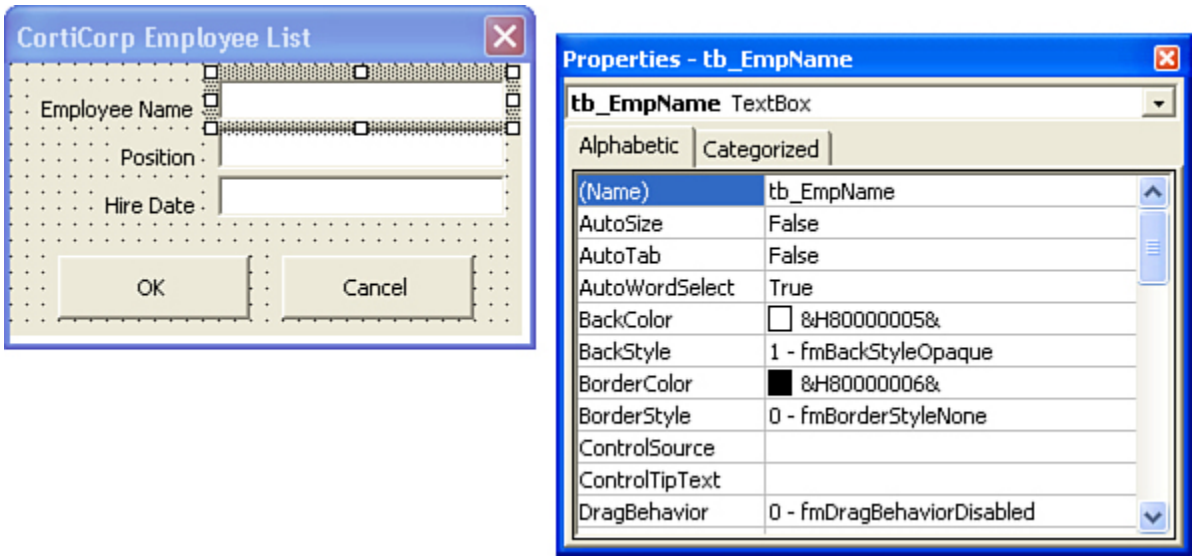
Note

By default, the toolbox displays the most common controls. To access more controls, right-click the toolbox and select Additional Controls. However, be careful; other users may not have the same additional controls as you do. If you send these users a form with a control they do not have installed, the program will generate an error.

After a control is added to a form, its properties can be changed from the Properties window. These properties can be set manually now or set later programmatically. If the Properties

window is not visible, you can bring it up by selecting View, Properties Window. Figure 2 shows the Properties window for a text box.

Figure 2. Use the Properties window to change the properties of a control.



Calling and Hiding a Userform

- A userform can be called from any module. FormName.Show pops up a form for the user:

  frm_AddEmp.Show

- The Load method can also be used to call a userform. This allows a form to be loaded but remain hidden:

  Load frm_AddEmp

- To hide a userform, use the Hide method. The form is still active but hidden from the user. However, the controls on the form can still be accessed programmatically:

  Frm_AddEmp.Hide

- The Unload method unloads the form from memory and removes it from the user's view, which means the form cannot be accessed by the user or programmatically:

  Unload Me

The Me Keyword

- Me is a keyword that can be used to refer to the userform itself. It can be used in the code of any control to refer to itself.
- For example, we can hide a userform using the following code:

  ```
  Me.Hide
  ```