# Unit 5 - Exercises

Create a new Visual Studio solution called Unit5Exercises containing a Windows Form Application project called FirstWindowsApp.

## Windows Form Applications

**EX1**. Rename the code file Form1.cs to HelloWorldForm.cs.

1. Right-click on the Form1.cs node in Solution Explorer. Choose Rename from the context menu.
2. Edit the file name.
3. Press the key to accept the change.
4. With the file selected in Solution Explorer, open it in code view.
5. Verify that the class name changed to HelloWorldForm.

**EX2**. Update the initial state of the form

1. Open the HelloWorldForm class in Design View.
2. Select the form in design view by single left clicking the form.
3. In the Properties panel, update the Text property to “Hello World - {your\_name}”, substituting your name into the String.

**EX3**. Add a Label control to the form.

1. Open the HelloWorldForm class in Design View.
2. Locate the Label control in the Toolbox panel.
3. Drag-and-drop an instance of Label to the form.
4. With the Label on the form selected, update the Name property of the control to lblMessage in the Properties panel. Also update the Text property to “Hello World”.
5. Re-locate the Label to be in the top-left corner of the form.

**EX4**. Add a button Control to the form.

1. Open the HelloWorldForm class in Design View.
2. Locate the Button control in the Toolbox panel.
3. Drag-and-drop an instance of Button to the form.
4. Update the Name property of the Button to btnFrench.
5. Re-locate the Button to the bottom-right corner of the form.

**EX5**. Handle the form’s Load event.

1. Open the HelloWorldForm class in Code View.
2. Locate the constructor method. It will include a call to the InitializeComponent() method.
3. After the method called, subscribe to the Load event of the Form.

**EX6**. Handle the Button’s Click event.

1. In the form’s Load event handler method, subscribe to the btnFrench Button’s Click event.
2. Add implementation to the Button’s Click event handler so that when the Button is clicked the lblMessage Label will update to “Bonjour le monde”.

**EX7**. Test the application.

## List Controls

The following exercises will requires additional files. Download the files [here](https://thethirstycoder.gitlab.io/adev-2005-learn/unit_5/downloads/adev-2005_unit_5_exercise_files.zip).

Add the files to the FirstWindowsApp project.

**EX8**. Update the start form for the application.

1. Open the Program.cs file in Code View.
2. Update the code
3. Application.Run(new HelloWorldForm());

to

Application.Run(new ListControlsForm());

1. Test the application to verify the correct form displays at startup.

**EX9**. Add items to a ComboBox control

1. Open the ListControlsForm.cs file in Code View.
2. In the Form’s Load event handler add the following items to the cboSlurpeeFlavours ComboBox:
   * “Banana”
   * “Coke”
   * “Grape”
   * “Root Beer”
   * “Pepsi”
   * “Sprite”
3. Test the application to verify the items were added to the ComboBox.
4. If the items were added correctly, click the View button to the right of the ComboBox. This will open a dialog window reporting on the state of the cboSlurpeeFlavours ComboBox. Note the results.
5. Update the state of the ComboBox by selecting a new item. Click the View button again and note the results.
6. Close the application.

**EX10**. Change the style of the ComboBox.

1. In Design View, select the cboSlurpeeFlavours ComboBox.
2. Locate the DropDownStyle property in the Properties panel.
3. Update the DropDownStyle property to Simple.
4. Run the application and click the View button; note the results.
5. Choose a different item, click the View button and note the results.
6. Type the text “Strawberry” in the ComboBox, click the View button and note the results.

**EX11**. Test the DropDownList ComboBox style.

1. Repeat exercise 10 using the DropDownStyle DropDownList.

**EX12**. Populate a ListBox.

1. Add implementation to the btnAdd\_Click event handler. The implementation will add a random integer number to the ListBox.
2. Test the application to verify that random numbers are added to the ListBox.
3. Click the View Button to the right of the ListBox. Note the results.
4. Add two numbers to the ListBox by clicking the Add button twice. Click the View button and note the results.
5. Select one of the items in the ListBox. Click the View button and note the results.

**EX13**. Remove an item from a ListBox.

1. Add implementation to the btnRemove\_Click event handler. The implementation will remove the selected item from the ListBox. Use the Remove() method.
2. Test the remove functionality.
3. Repeat setup 1 using RemoveAt().
4. Test again.

**EX14**. Remove all items from a ListBox.

1. Add implementation to the btnRemoveAll\_Click event handler. The implementation will remove all the items from the ListBox.
2. Test the remove functionality.