# Overview

During this exercise you will create a Synergy UI Toolkit application. The application will display text in the screen header, information line and footer, and will then wait for the user to press RETURN before exiting.

# Resources

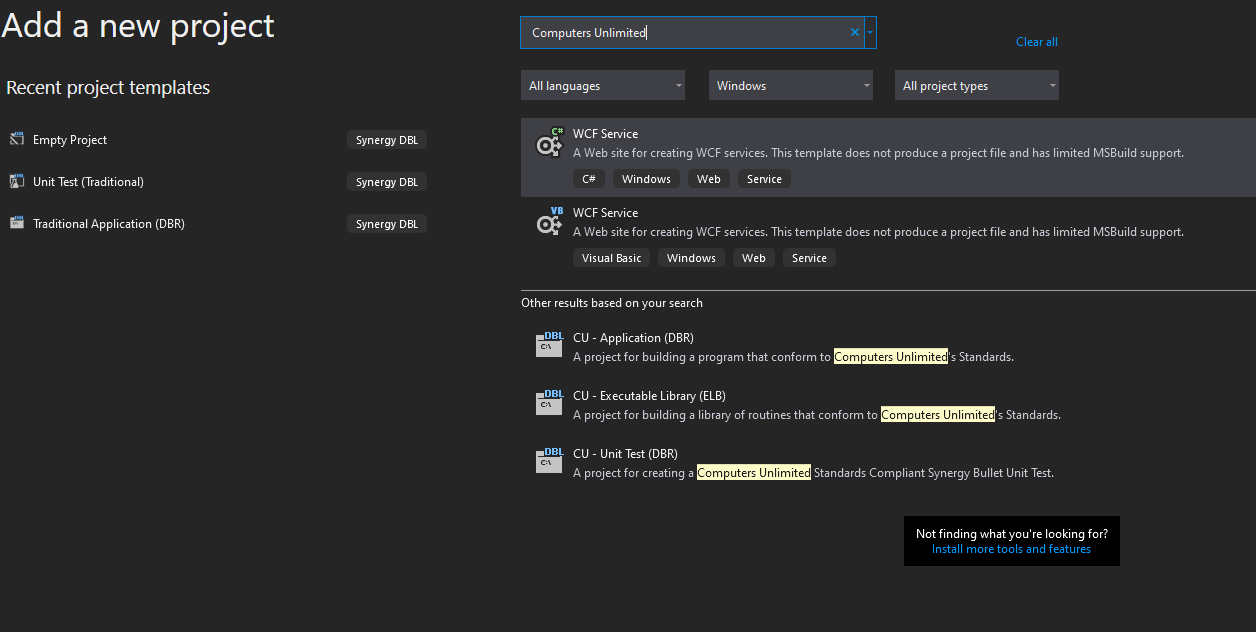
* [Synergy DBL Language Reference](https://www.synergex.com/docs/index.htm)
* [Synergy Best Practices - Coding Standards](http://jobfunc2.cu.net/Job%20Functions/Programmer/Programmer%20Handbook/Tims%20Best%20Practices%20-%20Standards/Synergy%20Best%20Practices%20-%20Coding%20Standards.docx)
* [Traditional Synergy in Visual Studio - CU Wiki](http://echo.cu.net/cuwiki/Traditional_Synergy_in_Visual_Studio)
* [Traditional Synergy in Visual Studio Common Terminology - CU Wiki](http://echo.cu.net/cuwiki/Traditional_Synergy_in_Visual_Studio_Common_Terminology)
* [Creating a New DBR Project (TSVS) - CU Wiki](http://echo.cu.net/cuwiki/Creating_a_New_DBR_Project_(TSVS))
* [Debugging (TSVS) - CU Wiki](http://echo.cu.net/cuwiki/Debugging_(TSVS))

# Exercise

For this exercise you should complete the following steps in the order shown:

1. Create a “Training” folder under the “Synergy” folder of your local source tree.
2. Create a “UIToolkit” folder under the “Training” folder.
3. Open Synergy\TIMS.ELBs.sln.
4. Add a new [CU – Application (DBR)](http://echo.cu.net/cuwiki/Creating_a_New_DBR_Project_(TSVS)) project “Orders” in \Synergy\Training\UIToolkit”.

**Note:**  You will need to search for “Computers Unlimited” in the “Add a new project” dialog:



1. Due to current limitations in Visual Studio, the name of the project will always be Program. To work around this you need to select the project and Hit F2 to rename the project.
2. Rename the project to “Demo”.
3. Perform these same steps for the Program.dbv file which was generated for you.
4. Rename the file to “Demo.dbv”.
5. Now open the Project Properties (Hotkey Alt+Enter).
6. Change the Output name under the Application Tab to “Demo”.
7. Set “Demo” as the startup project.
8. In “Demo.dbv” identify the beginning of the procedure division with the “**proc**” compiler directive, and end the program with the “**endmain**” compiler directive. This is important for the successful completion of future exercises.
9. In the data division, add the following code:
   * **.include “tims.def” library “cu\_records”**

**Note:** This file must be included in any program or routine, which uses UI Toolkit routines.

1. In the procedure division, add the code required to achieve the following:
   * Start the UI Toolkit, specifying one screen header line and one screen footer line.
   * Display the text “My First UI Toolkit Application” in the screen header.
     + Specify that the text should be centered.
   * Display your own name, right justified, in the screen footer.
   * Display a pop-up message box containing the text “Press OK to exit”.
   * Close down the UI Toolkit.
2. Stop program execution.
3. Rebuild the solution.
4. Run the program.

# Discussion

Displaying a pop-up message box is essential in this exercise, as it is the only thing that performs any kind of keyboard input. Without this, there would be no screen update, so the program would not display anything to the screen and would simply run through and terminate, without displaying anything to the screen.