# Overview

The purpose of this assignment is to better your understanding of CU’s data access routines while implementing a listing in your program.

# 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)
* [Installing Traditional Synergy in Visual Studio Templates - CU Wiki](http://echo.cu.net/cuwiki/Installing_Traditional_Synergy_in_Visual_Studio_Templates)
* [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))
* [CU Toolkit Manual](http://jobfunc2.cu.net/Job%20Functions/Programmer/Programmer%20Handbook/Synergy%20Reference%20-%20CUToolkit%20Reference%20Manual.doc)
* [Getting Started With Repository](https://www.synergex.com/docs/versions/v111/index.htm#rps/1_WELCOME_RPS.htm)
* [Synergy Data Language](https://www.synergex.com/docs/versions/v111/index.htm#rps/6_SDL.htm)

# Exercise

Create a listing for your program from which a user could specify a range of records to list.

1. Using Visual Studio, create a DemoMntListing stub subroutine in “DemoMnt.dbv”:
   * Simply display a message for now.
2. Modify the DemoMnt\_Menu routine:
   * Within this routine, trap only the menu entry “**CU\_LISTING**”.
   * Disable the program’s toolbar and menu columns with **CUTBM\_UPDATE()**.
   * Call your listing routine (DemoMntListing).
   * Enable the toolbar and menu columns with another call to **CUTBM\_UPDATE()**.
3. Compile your code and verify that the message in your listing stub is displayed and the toolbars are disabled when you select “Listing” from the menu.
4. Using Synergy Composer, create an input window for the filter screen.
   * Allow the user to specify a Demo Code range for the listing by adding two DemoCode fields to the window:
     + Name the fields **DemoCode$01** and **DemoCode$02**.
     + This naming convention allows you to get the range processing for free.
   * Add a user string of "\R00" to the **DemoCode$01** field.
   * Add the change method **CM\_RZLSTRNG** to both fields
   * Add the display method **DSP\_BLANKALL** to the **DemoCode$01** field.
     + This method will display “All” when the user does not enter any value.
   * Add “OK” (**I\_OK**), and “Cancel” (**CU\_PANIC**) buttons to the input window.
   * You will need to add text to your DemoMnt\_TXT window that you can retrieve in your listing for building the headings.
   * Compile the script using: “**…Synergy\BuildSynergyWindowsLibrary.bat**”
5. Using Visual Studio, modify DemoMntListing in “DemoMnt.dbv” to displays all the key fields and the first 50 characters of the description:
   * You first need to load the filter input window, create a simple program control, and process the filter screen.
   * When the user presses OK, **G\_ENTNAM** will be returned as “**CU\_SAVE**”.
   * At this point you know the user has completed the filter screen and wishes to print a listing.
   * The routine **CUI\_LISTCOUNT()** is called to determine how many DemoCode values the user entered.
   * Be sure to close the data access cursor and the print channel and destroy the program control when finished with them.
   * Rebuild and run the solution.

# Discussion

You will be using your DemoMas table for this exercise. Be sure you have several rows in your DemoMas table.