# Overview

In this exercise you will modify the **Orders.dbv** program that you created in [Exercise – 1](http://jobfunc2.cu.net/Job%20Functions/Programmer/Programmer%20Handbook/Synergy%20Training%20-%20Synergy%20Language%20Essentials/Exercise%20-%201.docx). You will add the code necessary to respond to the “Add an order” item on the main menu. For this exercise you will be provided with a new external subroutine called DataInput (**DataInput.dbl**), which will be used to perform terminal input in a window. The reason that this routine has been provided to you is that there are some complex issues surrounding terminal input of different data types. Having added this routine into your program, whenever the user selects the “Add an order” item from the main menu you will create a new window, into which you will display appropriate prompts for the fields in the HEADER record. Having displayed the prompts, you will use the DataInput subroutine to perform data entry for each of the fields.

The arguments of the DataInput subroutine are as follows:

* **Window ID** The window in which input should be performed (n).
* **Row** The window row at which to perform input (n).
* **Column** The window column at which to perform input (n).
* **Variable** The variable in which the data should be returned (a/n).
* **Allow Negative** For numeric fields, allow negative values (n).

# Resources

* [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)

# Exercise

1. Using Visual Studio, open the previously created “Orders” project.
2. Using Visual Studio, open “Orders.dbv”.
3. Create a new internal subroutine called pAddOrders.
4. Add code to the pAddOrders routine to create a window for the order header information. In the window, display the appropriate prompts for all fields in the Header record.
5. Use the supplied DataInput routine to enter the required data for each field into the fields of the Header record.
6. Code the pAddOrders routine to repeatedly accept new orders, until the user enters a blank (or rather zero) order number.
7. Change the code for the main menu to call the pAddOrders routine when the user selects the “Add an order” option.
8. Compile, link, and run the program.