

ENGR 1213 VBA programming HelpSheet

Input to a VBA Subroutine

See the code on the next page on one way to get a range as input for a VBA program – this uses the Selection Property of a Range. So that a user just selects the range they would like to input to the program.

Output from a VBA Subroutine

See the code on the next page to see one way to output a range to a worksheet by using a predefined starting cell for the output. Note that if you want to run this subroutine you will need to go into the VBA editor and change the variable *outrange* to the upper-left cell of the range where you would like the output to go. You should make sure that the cell you specify and the cells the right and down from it are unoccupied.

Alternative Input to a VBA Subroutine

Instead of using a selected range you may imitate the method for outputting from the subroutine described above. Just create string variables that have the name of the worksheet you are getting the input from (e.g. "sheet1") and the name of the cells that contain the input. For example if the the range is in cells defined by the box "f3:x15" on a sheet named "input_sheet" then you could define the variables:

```
insheet = "input_sheet"  
inrange = "f3:x15"
```

Now you can define a range called `input_range` in the subroutine by

```
Set input_range = Worksheets(insheet).Range(inrange)
```

now you have a range that may be used in any way you see fit.

How to Associate a Subroutine with a Control Key Combo

How to Assign a Subroutine to a Command Button

Sample input output subroutine for VBA

```

Sub input_output()
    Dim newrange_array()    'you will see this again later

    'First let's create a range that is input
    'One way to do this is to have the user select the range
    Dim rangein As Range
    Set rangein = Selection
    'That's all you have to do... now whatever the user had selected
    'before starting the program will be in the variable rangein

    'I now have a range in the variable rangein
    'I have no idea whether that range is a column
    'row, square, or rectangle...
    'but I can find out as follows

    Dim rows As Integer, cols As Integer
    'set up two variables to hold number of rows and columns

    'Now actually load up rows and cols with numerical values
    rows = rangein.Rows.Count
    cols = rangein.Columns.Count

    'Now let's change the values in rows and columns somewhat
    'First we need a two-dimensional array to hold the values in
    ReDim newrange_array(rows, cols)
    'Notice it is redim... which means redimension... this forces a
    'dynamic array ... the only way this would work

    For i = 1 To rows
        For j = 1 To cols
            newrange_array(i, j) = rangein(i, j) / 2
        Next
    Next

    'now I would like to output the results somewhere
    Dim outsheet As String, outrange As String
    outsheet = "input_output" 'the name of the sheet where I want output
    outrange = "f18"
    'the name of the starting cell where I want output
    Set rangeout = Worksheets(outsheet).Range(outrange)
    'last line just creates the actual range

    For i = 1 To rows
        For j = 1 To cols
            rangeout(i, j) = newrange_array(i, j)
        Next
    Next

    'I'm done
End Sub

```