

VBA INPUT BOX

What is an Input Box?

The InputBox function helps the user to get the values from the user. After entering the values, if the user clicks the OK button or presses ENTER on the keyboard, the InputBox function will return the text in the text box. If the user clicks on the Cancel button, the function will return an empty string "".

Syntax

```
InputBox(prompt[, title][, default][, xpos][, ypos][, helpfile, context])
```

Parameter Description :

- Prompt - A Required Parameter. A String that is displayed as a message in the dialog box. The maximum length of prompt is approximately 1024 characters. If the message extends to more than a line, then we can separate the lines using a carriage return character *Chr(13)* or a linefeed character *Chr(10)* between each line.
- Title - An Optional Parameter. A String expression displayed in the title bar of the dialog box. If the title is left blank, the application name is placed in the title bar.
- Default - An Optional Parameter. A default text in the text box that the user would like to be displayed.
- XPos - An Optional Parameter. The Position of X axis which represents the prompt distance from left side of the screen horizontally. If left blank, the input box is horizontally centered.
- YPos - An Optional Parameter. The Position of Y axis which represents the prompt distance from left side of the screen Vertically. If left blank, the input box is Vertically centered.
- helpfile - An Optional Parameter. A String expression that identifies the Help file to use to provide context-sensitive Help for the dialog box.
- context - An Optional Parameter. A Numeric expression that identifies the Help context number assigned by the Help author to the appropriate Help topic. If context is provided, helpfile must also be provided.

Example

We will calculate the area of a rectangle by getting values from the user at run time with the help of two input boxes *one for length and one for width*

```
Function findArea()
    Dim Length As Double
    Dim Width As Double

    Length = InputBox("Enter Length ", "Enter a Number")
    Width = InputBox("Enter Width", "Enter a Number")
    findArea = Length * Width
End Function
```

Output

1. To Execute the same, we will need to call using the function name and press Enter as shown below.



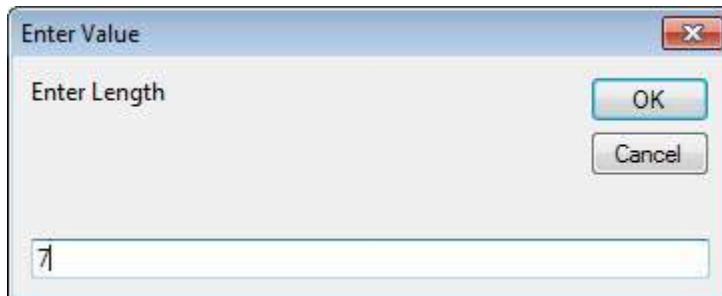
The screenshot shows the Microsoft Excel interface. In the top left, the Project Explorer displays two projects: 'VBAProject (All_Info.xlsx)' and 'VBAProject (Book2.xlsx)'. Under 'Book2.xlsx', there are 'Microsoft Excel Objects' (Sheet1, Sheet2, Sheet3, ThisWorkbook) and 'Modules' (Module1). The code for Module1 is visible in the main window:

```
Function findArea()
    Dim Length As Double
    Dim Width As Double

    Length = InputBox("Enter Length ", "Enter Value")
    Width = InputBox("Enter Width", "Enter Value")
    findArea = Length * Width
End Function
```

The ribbon tabs are 'File', 'Home', 'Insert', 'Page Layout', 'Formulas', 'Data', 'Review', 'View', 'Developer', 'Acrobat', 'Team'. The 'Home' tab is selected. The formula bar shows '=findArea()'. The spreadsheet area has rows 3 through 7. Cell D4 contains the formula '=findArea()'.

2. Upon Execution, The First Input box *Length* is displayed and user has to enter a value into the input box.



3. After entering the first value, the second input box *width* is displayed to the user.



4. Upon entering the second number and clicking OK button, the area is displayed to the user as shown below.

A screenshot of the Microsoft Excel spreadsheet. Cell D4 contains the formula '=findArea()'. The result, '28', is displayed in the cell and is circled with a red oval. The rest of the spreadsheet is empty with columns B, C, E visible.