**Excel Assignment - 18**

***1. What are comments and what is the importance if commenting in any code?***

**VBA Comments**

Comments are the lines in the code that are ignored while executing the code. These are represented as green text in the code. The comments help describe the written code. Knowing the correct use of comments is very important because while working with long and complex code, comments help us identify which part of code does what. It is very helpful for development purposes.

We can add comments in multiple lines. We use multi-line comments when we have to add points in our description or the description is long.

Commenting in a VBA code is one of the best practices and there are a few benefits that come with it.

**Helps you to Document your Work:** We can use a comment to describe how code works, which can help us in the future to recall it easily or any other user.

**Track the Changes:** If some codes need to change them frequently we can use comments to track or record changes within the code.

**Describe a Function Procedure:** When we write a procedure we can add a comment at the starting to describe the purpose of this procedure and how it works.

**Describe a Variable:** Variables are one of the most important things that we need to use while writing a VBA code and we can use a comment to describe a variable.

**Debug Your Code:** we can use VBA comments to debug the code by converting code lines into comments for testing.

***2. What is Call Statement and when do you use this statement?***

**Call statement :**

Call statement are used to transfers control to a Sub procedure, Function procedure, or dynamic-link library (DLL) procedure.

The Call statement syntax has these parts:

**Call :** Optional; keyword. If specified, you must enclose argumentlist in parentheses.

For example: Call MyProc(0)

**Name :** Name of the procedure to call.

**Argumentlist :** Comma-delimited list of variables, arrays, or expressions to pass to the procedure. Components of argumentlist may include the keywords ByVal or ByRef to describe how the arguments are treated by the called procedure.

**Call** keyword is not required when calling a procedure.

However, if Call keyword is used to call a procedure that requires arguments, argumentlist must be enclosed in parentheses. If we omit the Call keyword, we also must omit the parentheses around argumentlist. If we use either Call syntax to call any intrinsic or user-defined function, the function's return value is discarded. To pass a whole array to a procedure, we use the array name followed by empty parentheses.

**When Call Is Required**

The Call statement can be used to call a dynamic link library procedure.

***3. How do you compile a code in VBA? What are some of the problem that you might face when you don’t compile a code?***

Below 10 steps will compile VBA code:

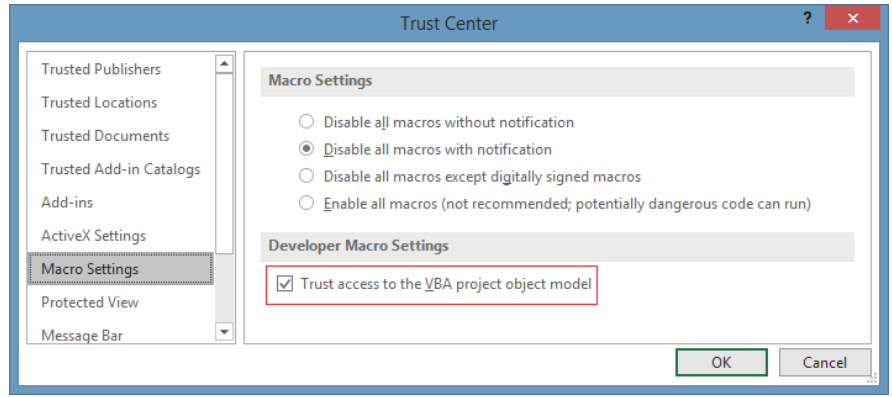
1. Download the VbaCompiler for Excel on our computer.

2. Install the VbaCompiler for Excel.

To install the product we need to start VbaCompiler4Excel.msi file and follow the installer steps.

3. Enable “Trust access to the VBA project object model” MS Excel option by following the path:

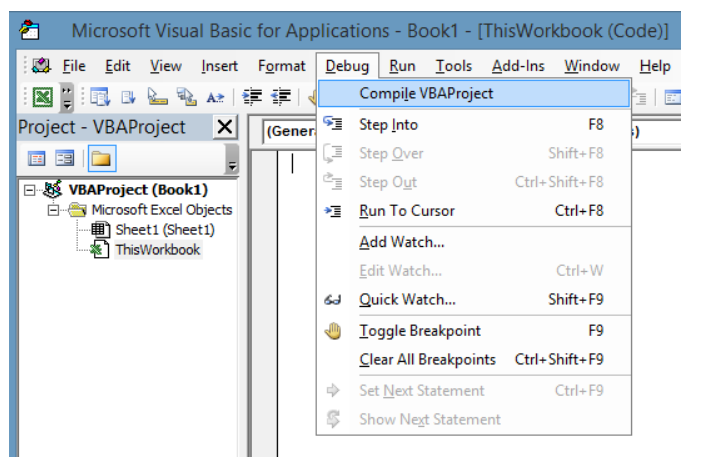
Excel Options >> Trust Center >> click button “Trust Center Settings…” >> Macro Settings >> enable check box “Trust access to the VBA project object model”



4. Examine your VBA code for syntax errors.

To do that we need to run the embedded Visual Basic code analyzer, located by following this path:

VBA Project window (Alt + F11) >> Main menu >> Debug >> Compile VBA project



if after running “Compile VBA Project” Excel found a syntax error, then we need to fix it and repeat this step again until the “Compile VBA Project” menu item is greyed out.

5. Make backup copy of the file we are going to compile.

The simplest way of doing this is to copy the file into another folder on your computer.

However it is a good practice to have a backup copy of your original file.

6. Remove the VBA Project password in the file we are going to compile.

we can apply it later, on to the VBA Project of the file with the compiled VBA code.

This step is required because the VbaCompiler has to get full access to the VBA code.

7. Run VbaCompiler for Excel by clicking the shortcut on the Windows desktop.

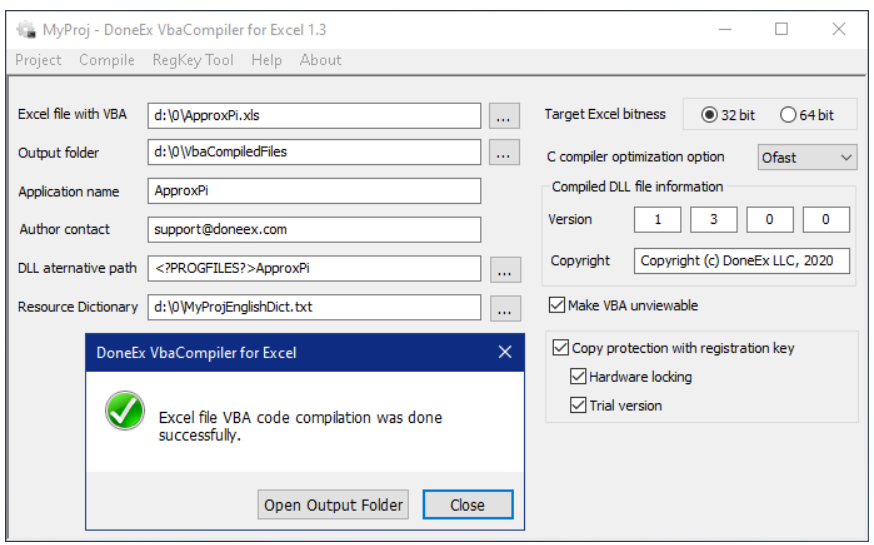
8. Enter the name of file we want to compile.

The easiest way is to select the file by clicking the […] button which is next to edit field and selecting the file in the dialogue window.

9. Enter the output folder where resulting files/modified excel file and Windows DLL with compiled VBA code will be located after compilation.

The easiest way is to select a folder by clicking the […] button which is next to edit field and selecting the folder in the dialogue window.You can find each compilation option meaning at the VBA Compiler options page.

10. Click the ‘Compile’ main menu item and wait until VBA code compilation is finished.



**Compile Errors :-**

It comes when we write code to perform an activity, but that activity is not valid or can’t be performed by VBA. The best example is where we have a code using the IF statement but missed to add END IF at the end of the statement and now when we run this VBA will show we a compilation error message.

*Sub VBA\_CompileError\_Example1 ()*

*If Range ("A1").Value = 1, Then*

*MsgBox "Keep it up! Working well!"*

*End Sub*

some other examples of compile errors:

* If statement in your macro without End If statement
* Using the For loop without the Next statement
* Select statement in your macro without End Select statement
* Defining a Sub or Function in your program that does not exist
* Calling a Sub or Function with the wrong parameters
* Option Explicit present at the beginning of your code and variables not declared at the top

***4. What are hot keys in VBA? How can you create your own hot keys?***

A hotkey is a key or a combination of keys on a computer keyboard that, when pressed at one time, performs a task (such as starting an application) more quickly than using a mouse or other input device.

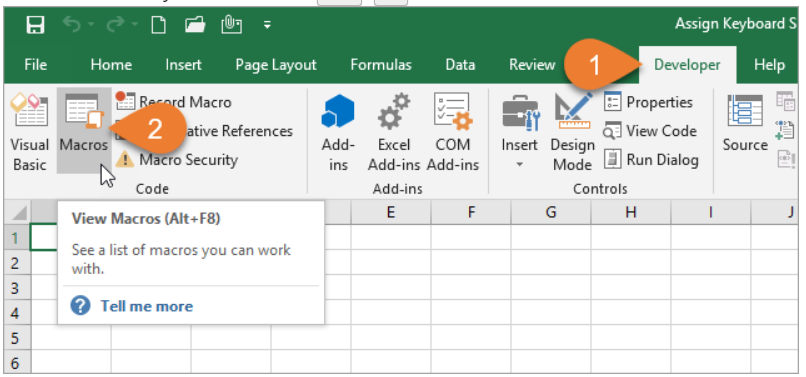
Hotkeys are sometimes called shortcut keys. Hotkeys are supported by many operating systems and applications.Below are some examples of Hot keys in VBA :-

|  |  |
| --- | --- |
| **hortcut Key** | **Action Performed** |
| Alt + A | Add-Ins Menu |
| Alt + D | Debug Menu |
| Alt + E | Edit Menu |
| Alt + F | File Menu |
| Alt + H | Help Menu |
| Alt + I | Insert Menu |
| Alt + O | Format Menu |
| Alt + Q | Closes the Visual Basic Editor and Return to Application |
| Alt + R | Run Menu |
| Alt + T | Tools Menu |
| Alt + V | View Menu |
| Alt + W | Window Menu |
| F1 | Microsoft Visual Basic for Applications Help/td> |
| F2 | Object Browser |
| F3 | Find |
| F4 | Properties Window |
| F5 | Runs the current procedure |
| F6 | Split between two code Windows |
| F7 | Code window |
| F8 | Step into code line by line |
| F9 | Breakpoint |
| F10 | Activates Menu Bar |

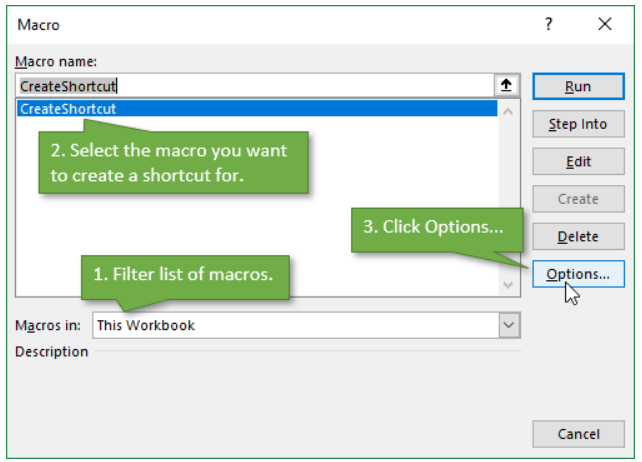
**1. The Macro Options Window: Shortcut Key :**

We can use the Macro Options window in Excel to create a shortcut key to call the macro.  Here are the instructions on how to set it up.

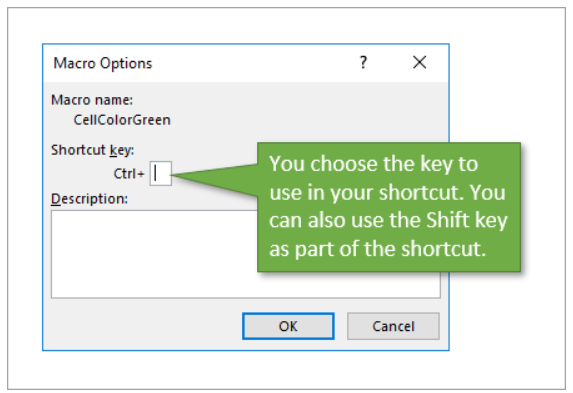
i) Start by going to the Developer tab and clicking on the Macros button.



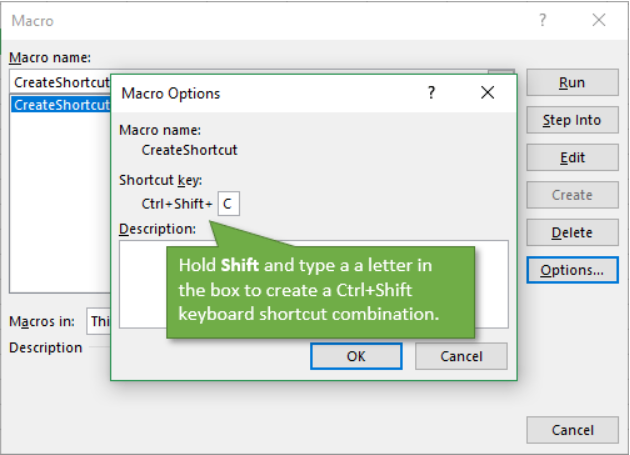
ii) After selecting the macro that you want to assign the shortcut to, click the Options button.



iii) In the Macro Options Window, you can create the shortcut you want by adding a letter, number, or symbol.



iv) Be careful **not to override an existing shortcut** that you frequently use, such as Ctrl+C to copy. One way to avoid doing this is by adding Shift to the shortcut to make it a bit more complex. In my example, I used Ctrl+Shift+C.

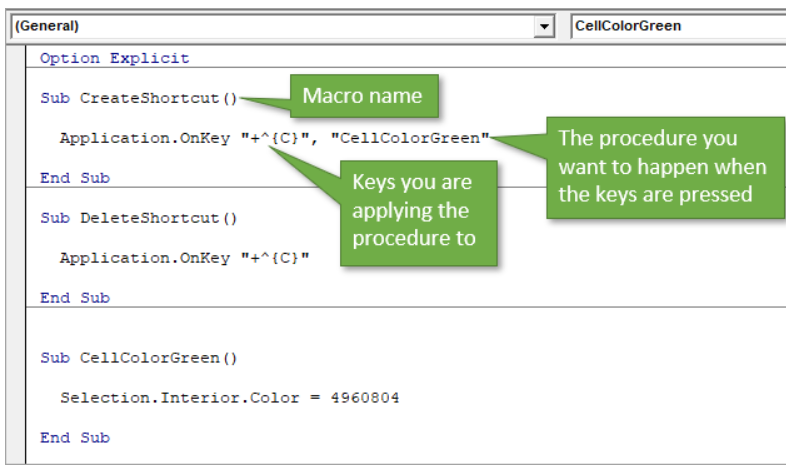


To delete the shortcut, simply repeat the process for accessing the Macro Options Window and then delete the character that you entered to create the shortcut.

**Create Shortcuts with OnKey**

In the VB editor, we are going to write some simple code to assign a macro to a keyboard shortcut.

1. Create a new macro and name it CreateShortcut (or whatever you choose to name the procedure),
2. Add a new line and start it with the command Application.OnKey followed by a space.
3. The Application.OnKey method has two parameters for the Key and Procedure. The Key is the keyboard shortcut combination represented by key codes. The Procedure is the name of the macro that will be called when the key combination is pressed.
4. Both parameters are enclosed in quotation marks.
5. In my example I use “+^{C}” for the Key parameter. The + is the code for Ctrl, the ^ is code for Shift, and the C key is enclosed in curly brackets (or braces). How are you supposed to know the code for each key? Microsoft has this helpful document, which contains a complete list.
6. Following this code, you are going to name the procedure that you want to assign to that combination of keys. In this case, we want the key combination to run the macro called “CellColorGreen”.



***5. Create a macro and shortcut key to ﬁnd the square root of the following numbers 665, 89, 72, 86, 48, 32, 569, 7521***

*Sub Square()*

*Dim given\_number As Range*

*Dim sqr\_root As Double*

*Dim square\_root As Range*

*Set given\_number = Range("A2:A9")*

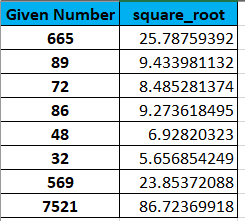
*Offset = 1*

*For Each square\_root In given\_number*

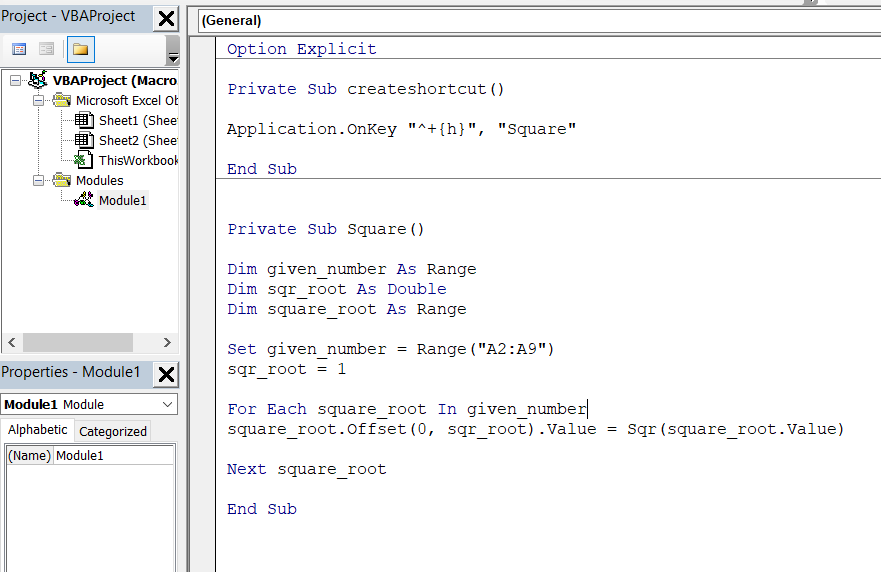
*square\_root.Offset(0, Offset).Value = Sqr(square\_root.Value)*

*Next square\_root*

*End Sub*

**

***Adding Short cut key to Macro :-***



***6. What are the shortcut keys used to***

***a. Run the code :* F5**

***b. Step into the code:-* F8**

***c. Step out of code :-* Shift + F8**

***d. Reset the code :-* Alt,R,R**

**All above short cuts are for windows OS.**