

**Excel Assignment - 18**

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

Answer:

A **comment** is text in a program's code, script, or another file that is not meant to be seen by the user running the program. However, is seen when viewing the source code.

Comments help make code easier to understand by explaining what is happening and help prevent portions of a program from executing.

When writing or editing code to help fix a problem or debug, the programmer may comment out one or more lines. After a line is commented out, it's skipped the next time the code is executed. If, after commenting out code the problem is fixed, it helps give the programmer an idea of where the problem in the code exists.

Hidden text in Microsoft Excel, Google Sheets, and other spreadsheet software that enables users to make **comments** about cells and their values. Cells with comments have a color corner in the top-right corner of the cell.

Generally, a **comment** is a remark, suggestion, or feedback about a product or service. To give Computer Hope a comment, contact customer service.

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

Answer:

Call statement is used to transfer control to a Sub procedure, an intrinsic function, and a dynamic-link library (DLL) procedure. DLLs are not used on the Macintosh.

VB

Copy

' Call a Sub procedure.

Call PrintToDebugWindow("Hello World")

' The above statement causes control to be passed to the following

' Sub procedure.

Sub PrintToDebugWindow(AnyString)

Debug.Print AnyString ' Print to the Immediate window.

End Sub

' Call an intrinsic function. The return value of the function is

' discarded.

Call Shell(AppName, 1) ' AppName contains the path of the

' executable file.

' Call a Microsoft Windows DLL procedure. The Declare statement must be

' Private in a Class Module, but not in a standard Module.

Private Declare Sub MessageBeep Lib "User" (ByVal N As Integer)

Sub CallMyDll()

Call MessageBeep(0) ' Call Windows DLL procedure.

MessageBeep 0 ' Call again without Call keyword.

End Sub

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?

Answer: Compile errors refer to a wider group of VBA errors, which include syntax errors. Compile errors also identify problems with your code when considered as a whole. The syntax of each individual line may be correct, but when put together, the lines of your code don’t make sense. Compile errors are highlighted when you compile or run your code.

#### How to Identify a Compile Error

When you run your VBA code, compile errors will be presented in a VBA dialog box as per the examples below.

Alternatively, if your project is long or complex and involves multiple routines, it can be helpful to compile your code before you run it. It prevents situations where half of your code runs successfully and then an error pops up. You can compile your VBA code by clicking Debug Menu → Compile VBA Project.

**Example 1**

A compile error for a missing “End if” part of an IF statement. Every individual line in the code is correct, but together, they don’t represent a complete IF statement.

Compile Error - Example 1

**Example 2**

In this example, the VBA compile process has detected a syntax error, highlighted in red. VBA indicates that the code below is missing a closing bracket on the function.

Compile Error - Example 2

**Example 3**

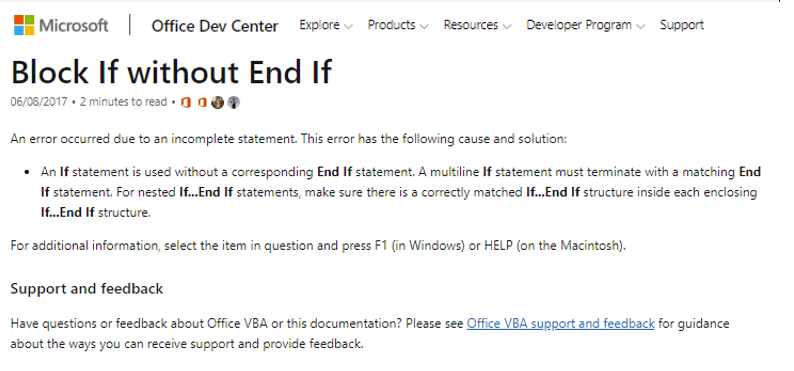
Compile errors are common when using Option Explicit and occur when a variable has not been explicitly defined. With Option Explicit activated, a Dim statement is required to declare all variables before they can be used in your code.

Compile Error - Example 3

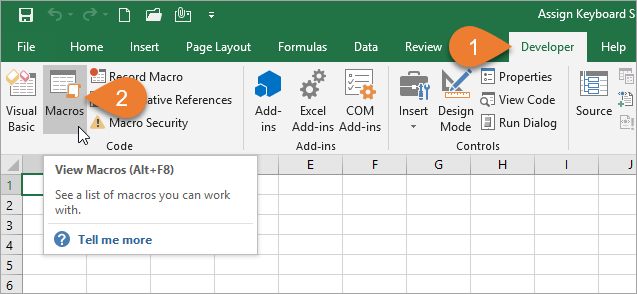
### How to Resolve Compile Errors

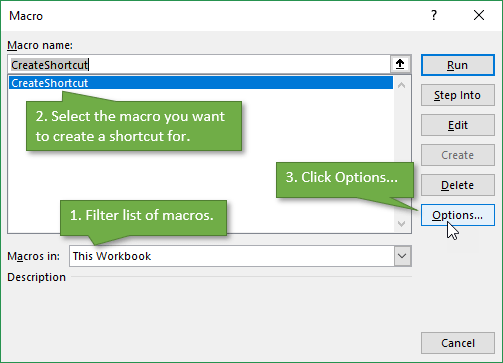
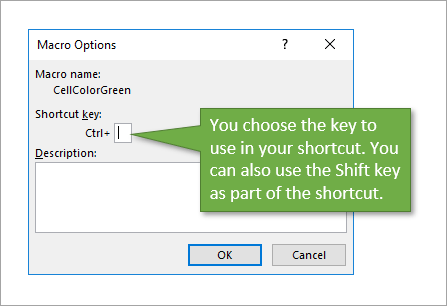
VBA compile errors will prevent affected routines from running. Until you fix them, your code cannot be interpreted correctly by VBA. The error message boxes often provide more helpful advice than with syntax errors.

If it’s not immediately obvious to you what the issue is from the error message, try usingMicrosoft’s Help Center to get an idea of exactly how your statement or function should be written. Here’s an example of the “Block if without end if” error message on the Microsoft website.

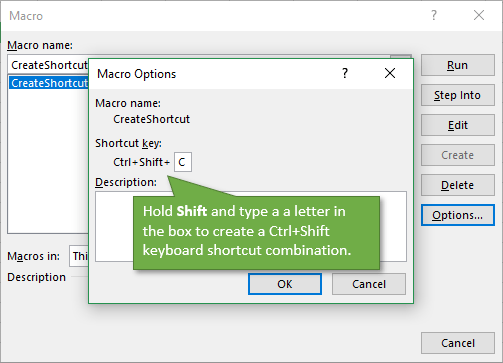


4. What are hot keys in VBA? How can you create your own hot keys?

Answer: Developer tab and clicking on the Macros button. (If you don't see the Developer tab on your ribbon, you can add it using [these instructions](https://www.excelcampus.com/vba/enable-developer-tab/).) Alternatively, you can use the keyboard shortcut **Alt**+**F8**.  


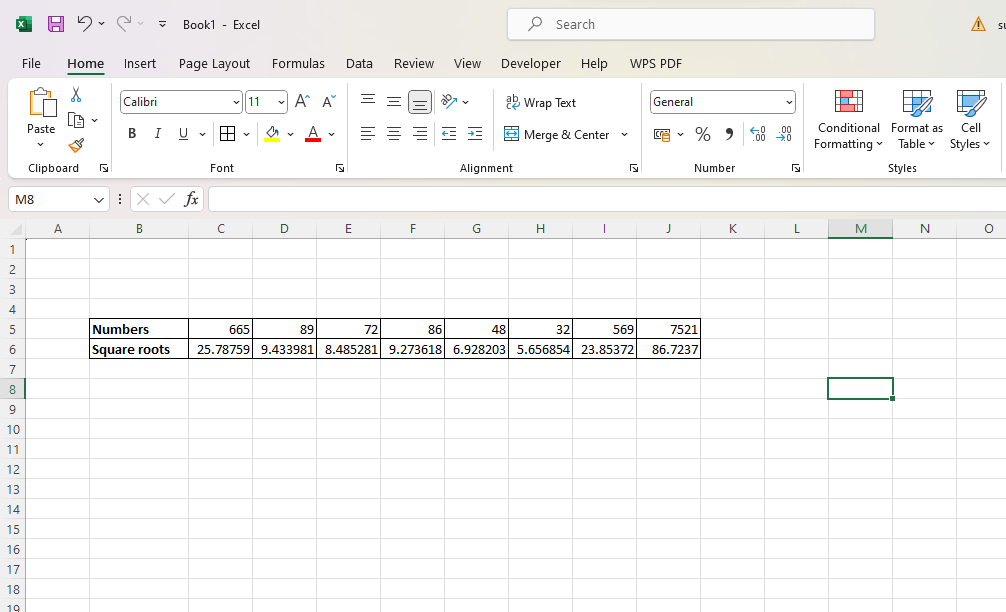
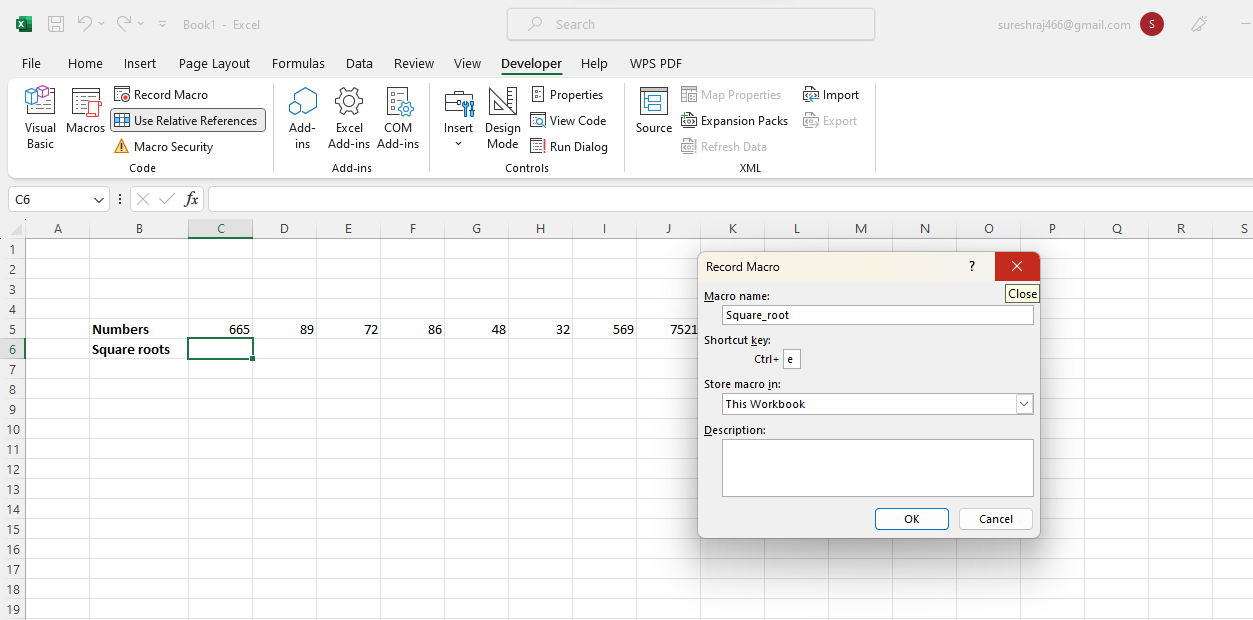
1. After selecting the macro that you want to assign the shortcut to, click the Options button.  
   
2. In the Macro Options Window, you can create the shortcut you want by adding a letter, number, or symbol.  
   

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.

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

Answer: 

6. What are the shortcut keys used to

a. Run the code

b. Step into the code

c. Step out of code

d. Reset the code

Answer:

| a. Run the code | ALT + F11 |
| --- | --- |
| b. Step into the code | F11 |
| c. Step out of code | Shift + F11 |
| d. Reset the code | Ctrl + Shift + F10 |