**Advance Excel Assignment 18**

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

Ans. 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. The image is an example of an HTML comment. See our no executable statement page for a full definition and further examples of comments in programming.

The importance of commenting in code cannot be overstated. Here are some of the reasons why commenting is important:

* Clarity: Comments help to make the code more understandable and easier to read by providing additional information about the code's purpose and functionality.
* Debugging: Comments can help to locate and fix errors in the code more quickly by providing a context and explanation for the code.
* Collaboration: Comments can help to facilitate collaboration between team members who are working on the same code by providing a common language and understanding of the code.
* Maintenance: Comments can help to make it easier to maintain and update the code over time by providing a record of changes and updates that have been made.

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

Ans. A call statement is not a commonly used term. Excel does have a function called cell, which can be used in more advanced scenarios to call other Excel functions or macros from within a formula.

The CALL function is used when you want to execute a function or macro that is not part of the standard Excel functions or macros.

This function allows you to call custom functions or macros that have been written in Visual Basic for Applications (VBA) code, and execute them from within a formula.

For example:

We have a custom function called "MyFunction" that has been written in VBA code. This function takes two parameters and returns the sum of those parameters multiplied by 2. To call this function from within a formula in Excel, we use the following syntax: =CALL("MyFunction", 2, 3)

In this example, the first parameter "MyFunction" is the name of the custom function that we want to call. The second and third parameters are the input values that we want to pass to the function.

Please note that the call function is an advanced function and should be used with caution. It is generally recommended to use the standard Excel functions whenever possible, and to only use the call function when we have a specific need to call a custom function or macro.

1. 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?

Ans. To compile a code in VBA, follow these steps:

* Open the Visual Basic Editor (VBE) by pressing Alt + F11.
* Select the module or project that we want to compile.
* From the menu bar, select Debug -> Compile <module/project name>.
* If there are any syntax errors or other issues with the code, the VBE will display an error message. We can then correct the errors and recompile the code.

Compiling code in VBA is important because it helps to detect errors in the code before it is executed. When we don't compile a code, we may face the following problems.

* Syntax errors
* Run-time errors
* Slow performance
* Security issues

Compiling our code is an important step in the development process that can help we catch errors early and improve the performance and security of your program.

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

Ans. Hotkeys in VBA are keyboard shortcuts that execute a particular action or command within the VBA editor or Excel application. Hotkeys can save time and make coding more efficient by allowing us to perform common tasks quickly.

Here are some commonly used hotkeys in VBA:

* F5: Run the code
* F8: Step through the code line by line
* Ctrl + G: Display the Immediate window
* Ctrl + R: Display the Project Explorer window
* Ctrl + Shift + A: Insert a new procedure

We can also create our own custom hotkeys in VBA by using the Application. On Key method. This method allows us to assign a keyboard shortcut to a specific VBA macro or function.

Step for how to create a custom hotkey in VBA:

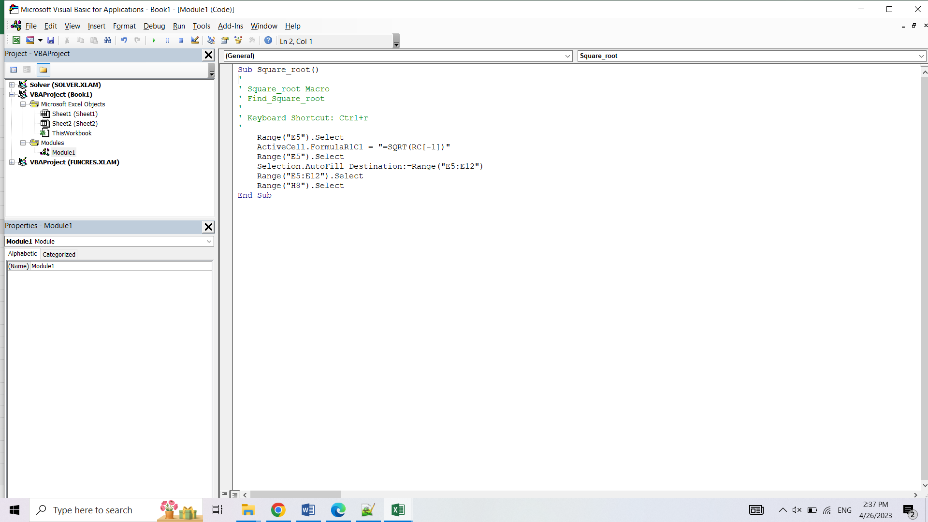
* Open the VBA editor in Excel by pressing Alt + F11.
* Create a new module or open an existing one.
* Write a VBA macro or function that we want to assign a hotkey to.
* Go to the Tools menu and select Customize Keyboard.
* In the Categories box, select Macros.
* In the Commands box, select the macro or function that we want to assign a hotkey to.
* Click in the Press new shortcut key box and type the key combination that we want to assign as a hotkey.
* Click Assign and then Close.

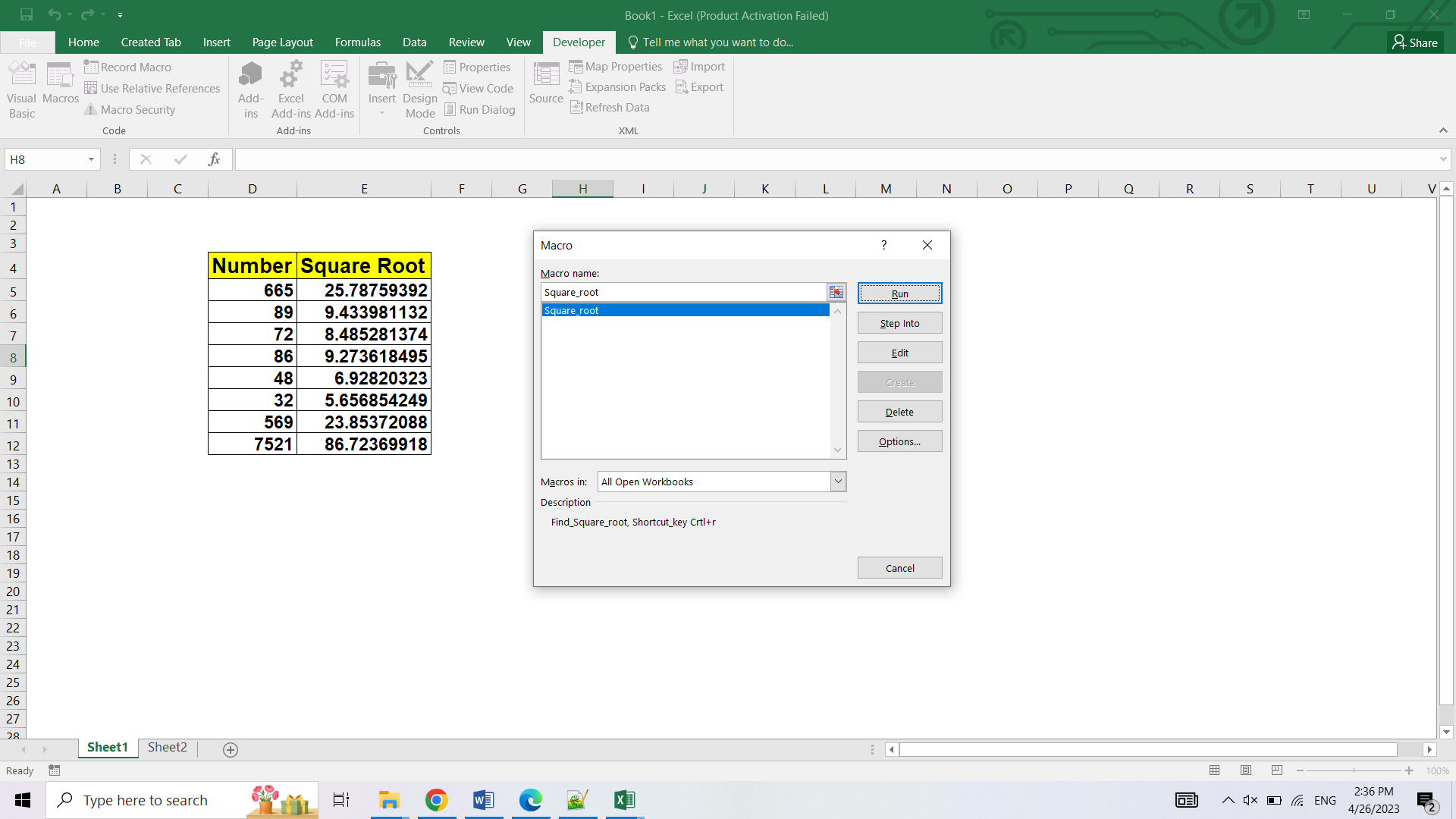
We press the key combination we specified, the VBA macro or function will be executed.

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

|  |  |
| --- | --- |
| **Number** | **Square Root** |
| **665** | **25.78759392** |
| **89** | **9.433981132** |
| **72** | **8.485281374** |
| **86** | **9.273618495** |
| **48** | **6.92820323** |
| **32** | **5.656854249** |
| **569** | **23.85372088** |
| **7521** | **86.72369918** |

Ans.





1. What are the shortcut keys used to

Ans. a. Run the code – F5

b. Step into the code - F8

c. Step out of code – Crtl+Shift+F8

d. Reset the code - Ctrl + Break