I eat macros for lunch!

3/17/2020

msd
Safe, clean waterways

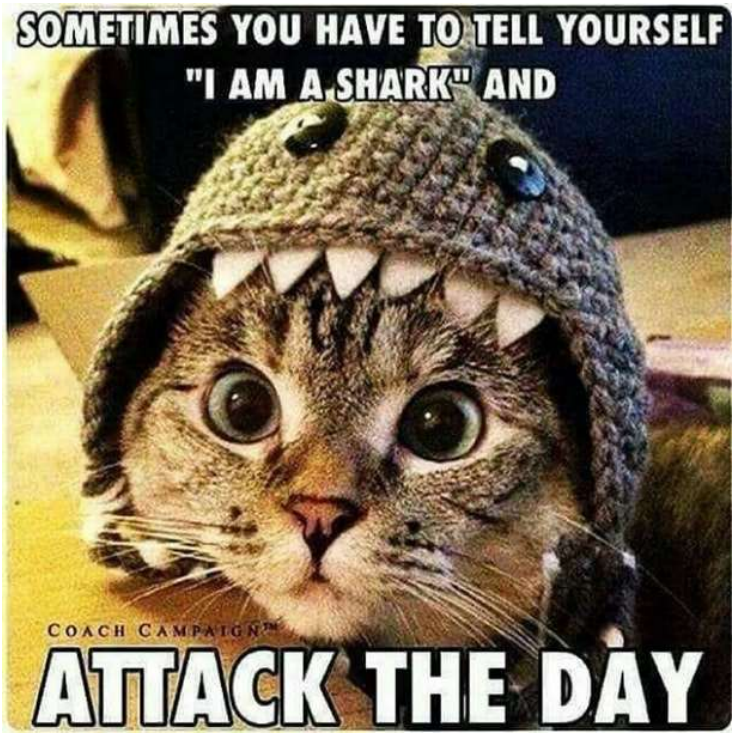
Disclaimer!!!

- I am not an expert.
- I am not I.T.
- I may (or may not) show you the best or efficient way to develop or use macros
- The main skill you need is indeed the newest buzzword: “GRIT”

msd
Safe, clean waterways

Who is this discussion for?

- "I am at a small WQTC/Lab and have no Laboratory Information System (LIMS)"
- "I know a little bit about excel… how can I automate a simple step / process?"



msd
Safe, clean waterways

Goal(s): (… Cause we all need'em…)

- Define: What is a Macro?
- Demonstrate a Macro (in "layman's terms")
- Modifying Tabs / "XML Menus" and the excel "Ribbon"
- Code Basics: Loops, Input Boxes and IF/THEN statements
- Show some real world examples!

  (If possible, open excel and follow along during the demonstration!!!)

msd
Safe, clean waterways

2nd Disclaimer!!! (before we start)

We have a short period of time with a LOT of information, and it will go fast.

This Power Point demonstration details the step by step process…. Feel free download and follow along.

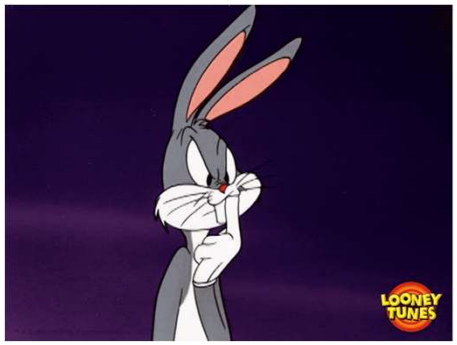
msd
Safe, clean waterways

What is a macro?

- Techopedia: **macro** is an automated input sequence that imitates keystrokes or mouse actions. A **macro** is typically used to replace a repetitive series of keyboard and mouse actions and are common in spreadsheet and word processing applications like MS Excel and MS Word.

  - OR –

  Put simply: "A macro is a bit of code that automates your keystrokes/process"



msd
Safe, clean waterways

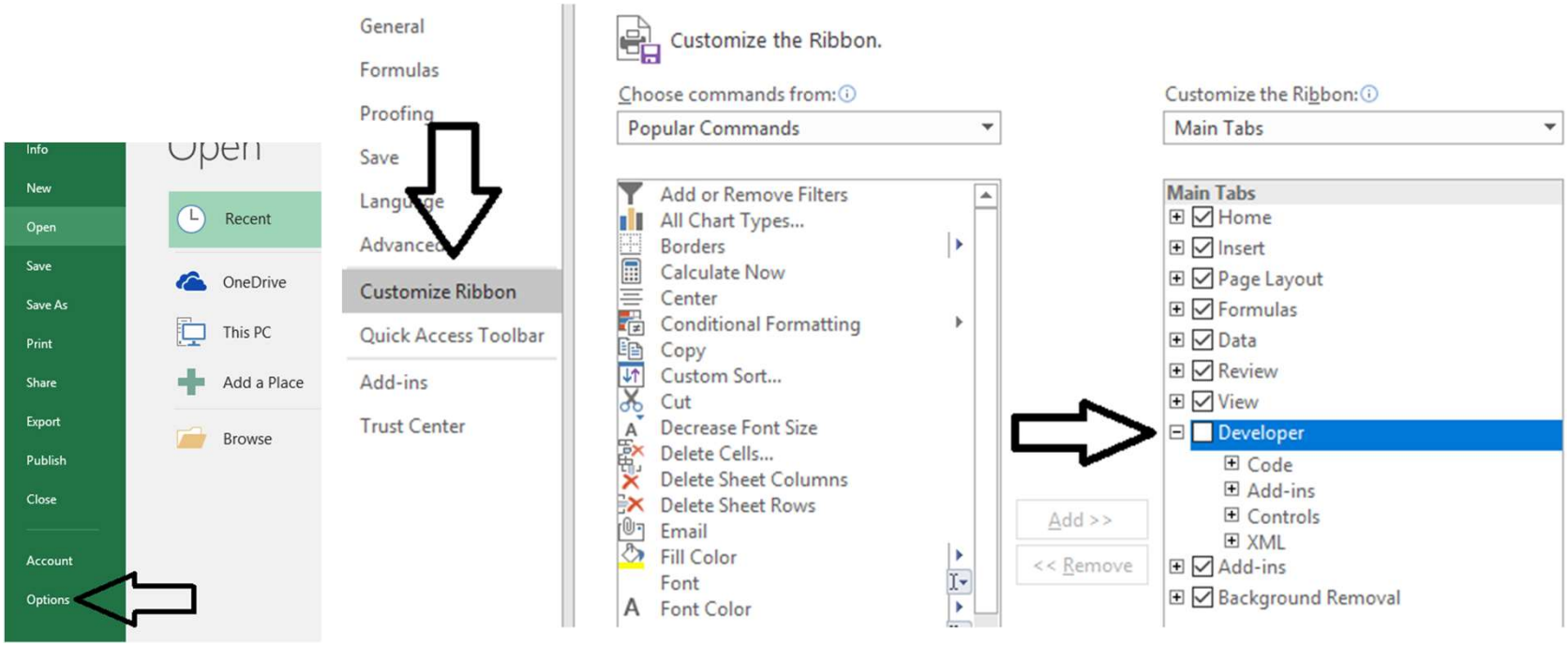
What is a macro?.... More stuff

- So what does that mean?
  - These “bits” of code are stored within excel inside the VBA environment
  - VBA: Visual Basic for Applications, a programming language within office (excel, word, powerpoint) used to automate “stuff” (ie: “Events” within the application)

msd
Safe, clean waterways

Step #1: Get to know excel a little better…

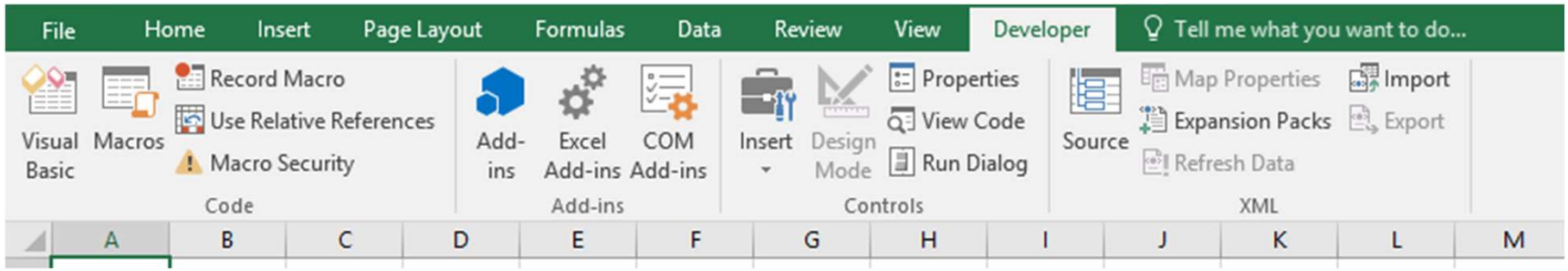
- Getting Excel Ready: Expose the “Developer Tab”:



msd
Safe, clean waterways

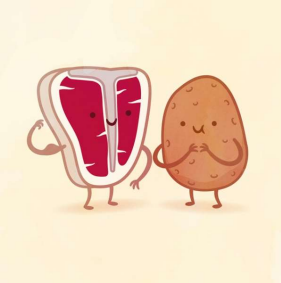
Developer Tab Overview:

- Visual Basic for Applications: “VBA” programming language to control office.
- “Visual Basic”, “Macros”, “Record Macro” and “Insert” Buttons:

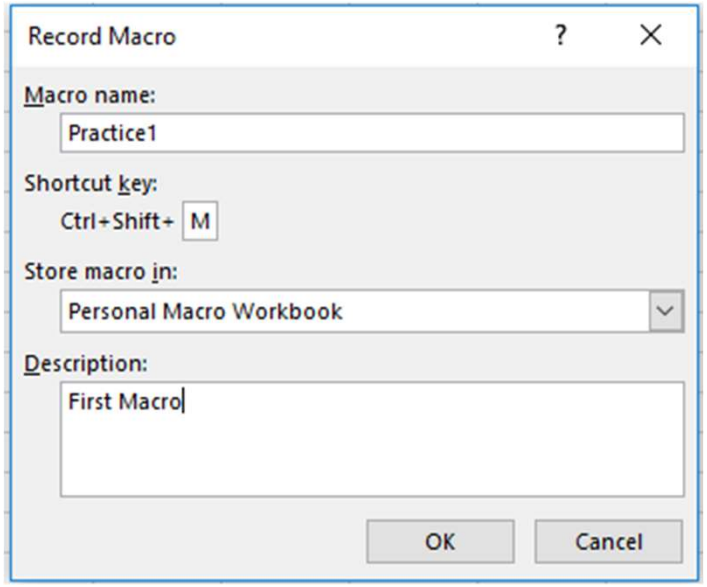


msd
Safe, clean waterways

Lets Make a macro: Step 1, Meat and Potatoes

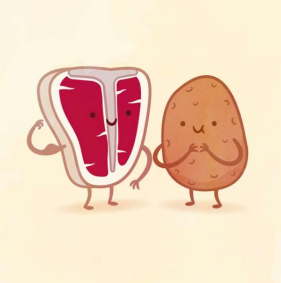


- From “Developer Tab”
  - Select “Record” button.
  - The “Record Macro” window will appear.
  - Select “Personal Macro”



msd
Safe, clean waterways

Lets Make a macro: Step 1, (Continued)



- What is the “Personal Macro Workbook”
  - This is a “behind the scenes” workbook that stores code in a “Module” specific to you and your user login.
  - Alternatively, you can select “This Workbook”. This is a “Module” block of code that is only available (and is stored within) the specific workbook.

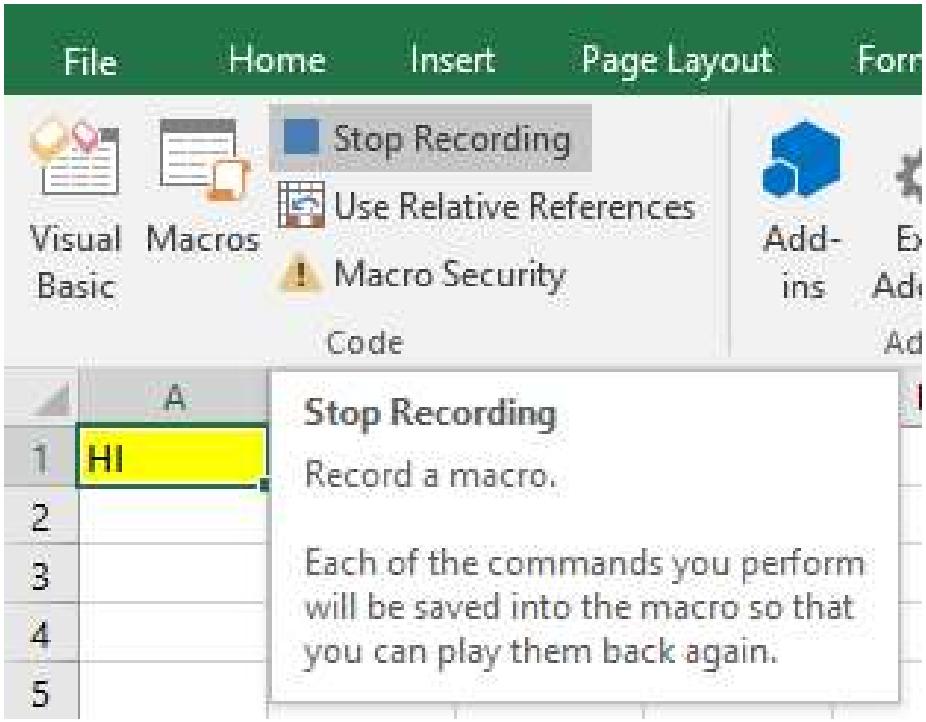
DEMONSTRATION START

(Resume at Slide 23)

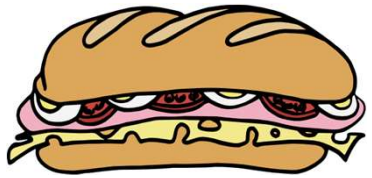
msd
Safe, clean waterways

…. Next, perform the following….

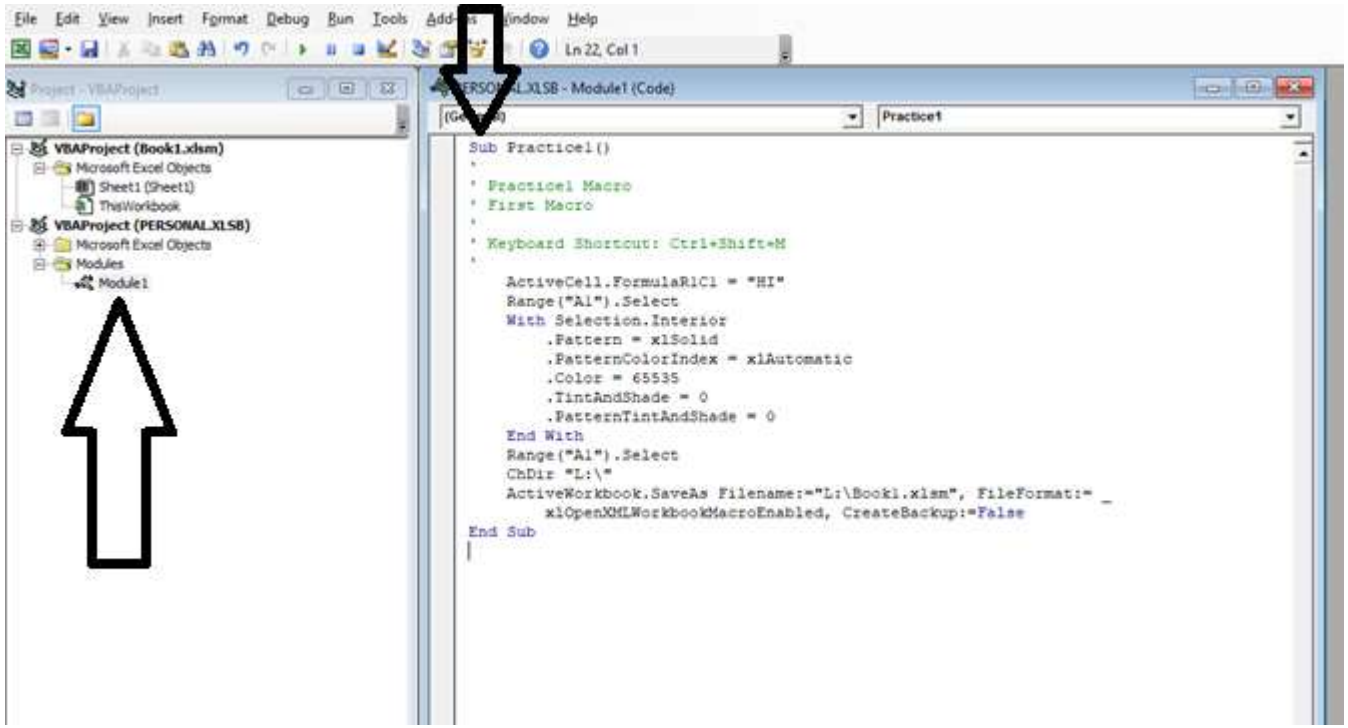
- Type “Hi” into cell “A1”
- Change the color of cell “A1” to yellow.
- Save the document
- Hit “Stop Recording” to finish macro!



Step 3: Open Visual Basic Editor to View Code.

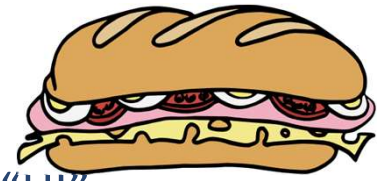


1. Select “Visual Basic” on the Developer Tab
2. Select “Module1” within the VBA Project
3. Notice the “Sub” Routine: “A package of code”



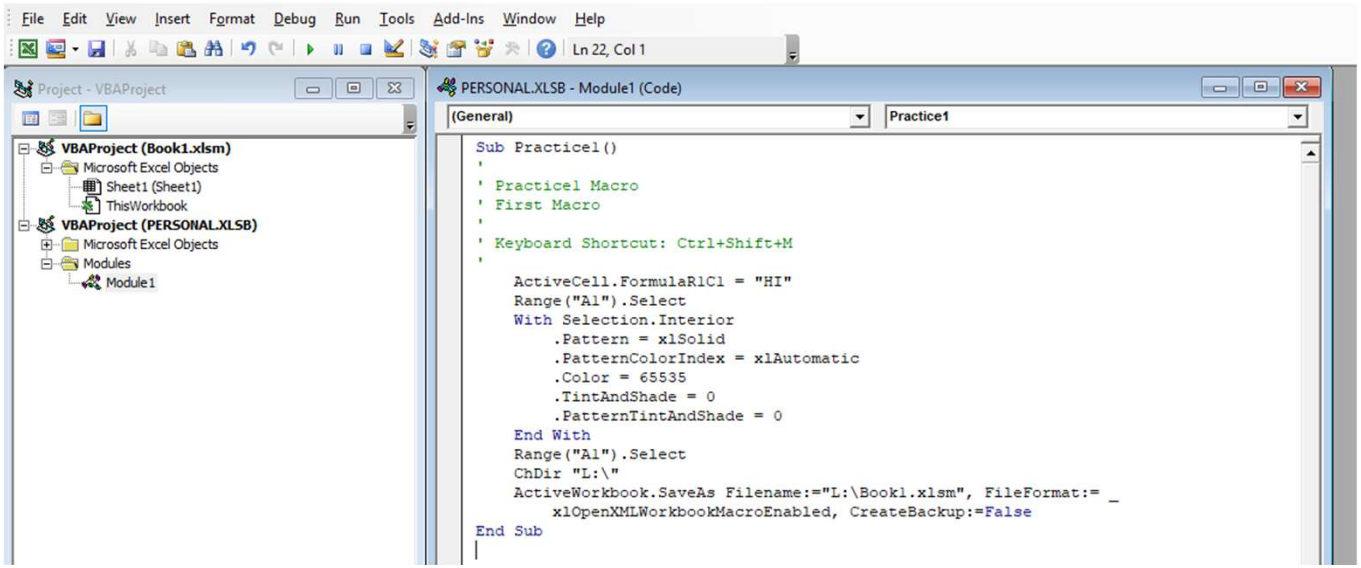
msd
Safe, clean waterways

Lets Look at the code…Notice the 3 steps



The Code Tells a story:

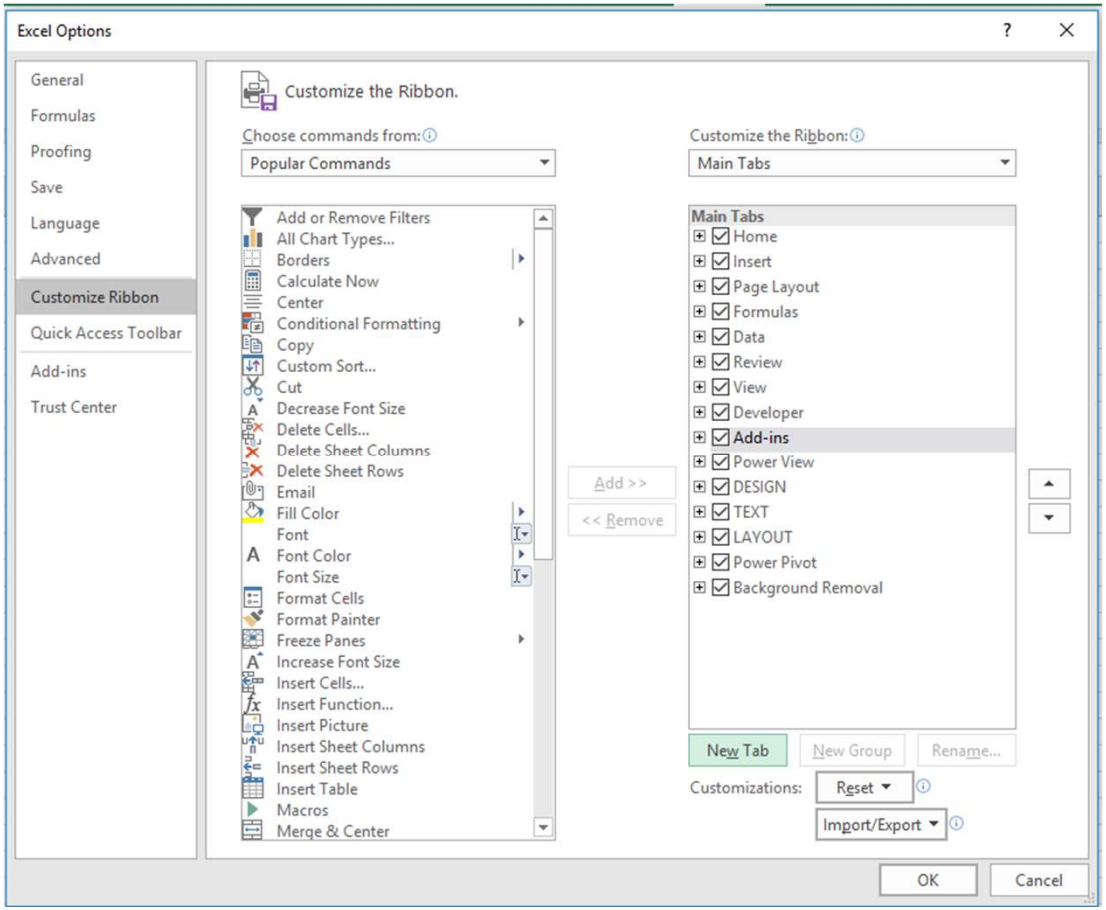
- "ActiveCell.FormulaR1C1": "cell on row one column 1 = "HI"
- "Range("A1").select: "select cell A1"
- "With Selection…. End with": "with the selected cell, change these properties (color, pattern, etc)
- "ChDir / Activeworkbook.SaveAs": "find this directory and save the file as a macro enabled workbook with name "Practice1.xlsm"



msd
Safe, clean waterways

But there's MORE….. Add a Tab to the Ribbon

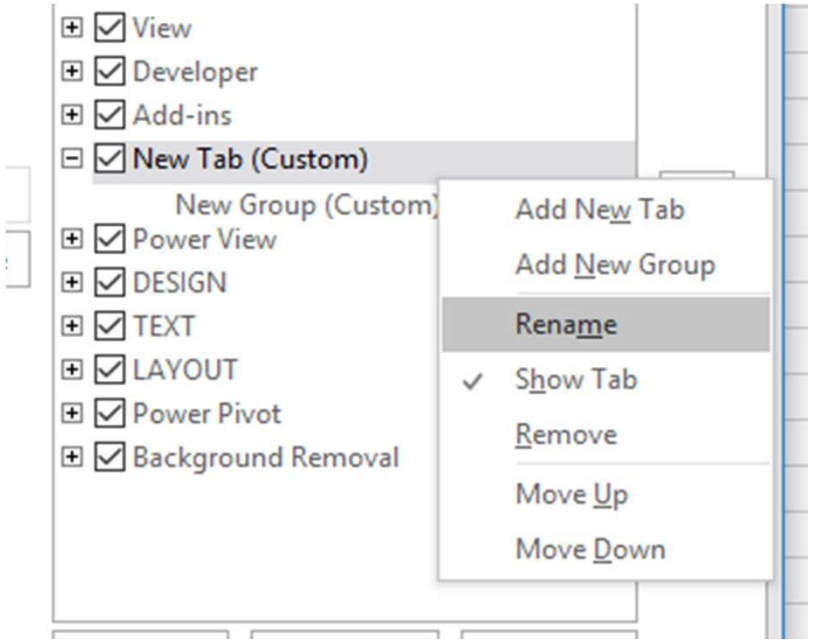
- On the worksheet right click on the ribbon and select: "CUSTOMIZE THE RIBBON"



msd
Safe, clean waterways

…. Adding a Tab to the Ribbon ….

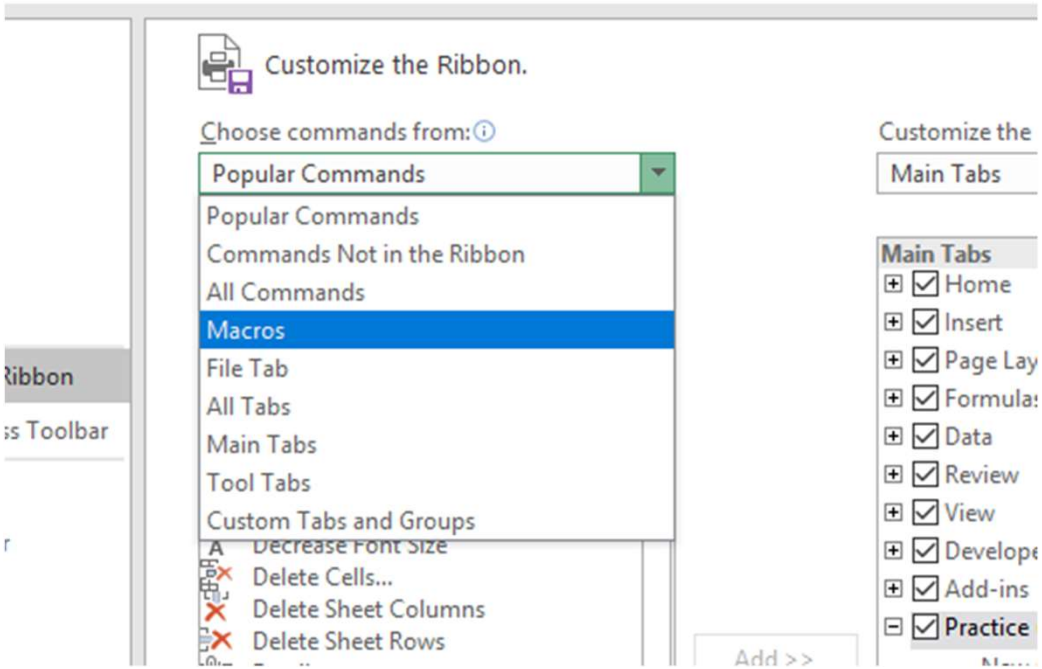
- Select “New Tab” button
- Select “New Tab (Custom)”
- Right-Click and Rename the Tab.
- Name the new tab “Practice”
- Select “New Group (Custom)”
- Right-Click and Rename
- Name the Group “Practice”



msd
Safe, clean waterways

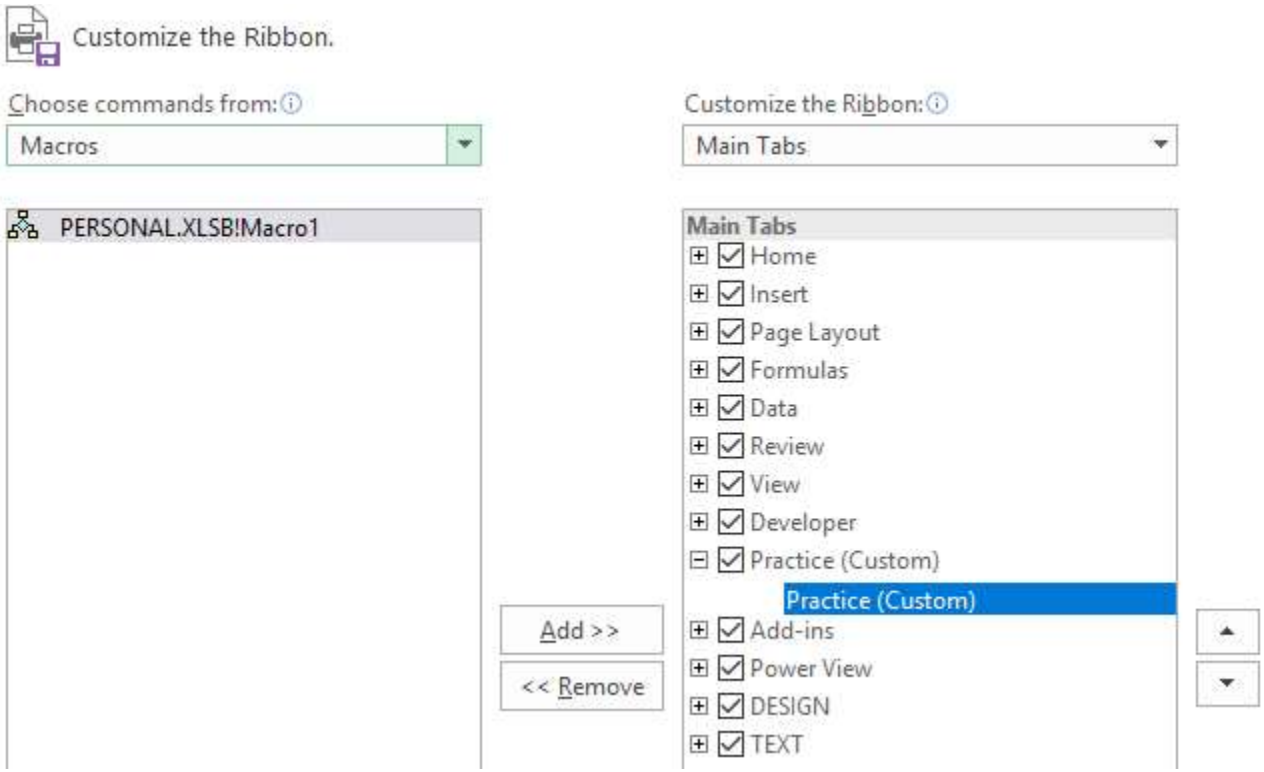
… Add our new macro to a Tab…

- Select “Macros” from the “Choose Commands from:” drop menu



… Add our new macro to a Tab…

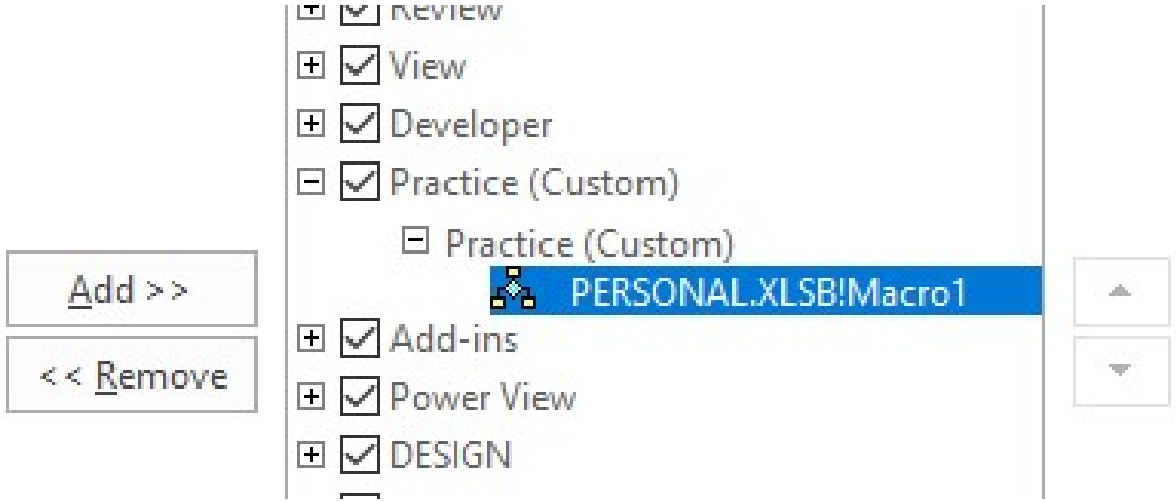
- Select / Highlight the “Practice (Custom)”
- Select “Add>>” to add the practice1 macro to the tab.



msd
Safe, clean waterways

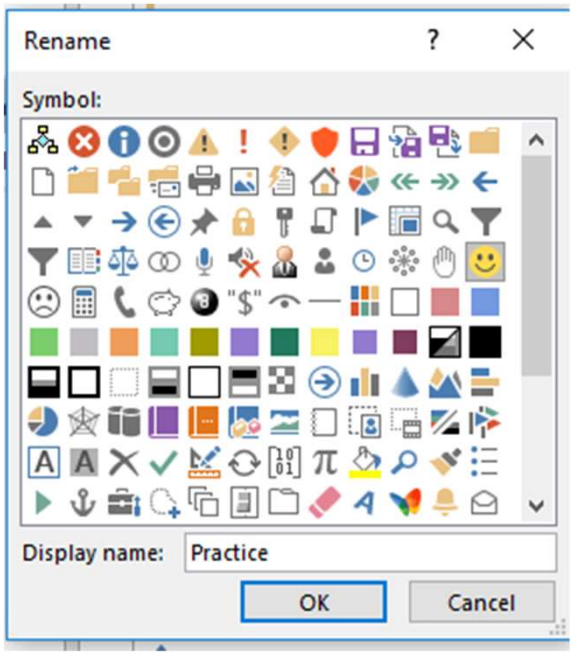
… Add our new macro to a Tab…

- The macro is now added to the new tab called “Practice”



… Add our new macro to a Tab…

- Rename the macro:
  - Right-Click; Select “Rename”
  - Change the “Display name” to “Practice”
  - Select the “Smiley Face” icon
  - Select “OK” to save”

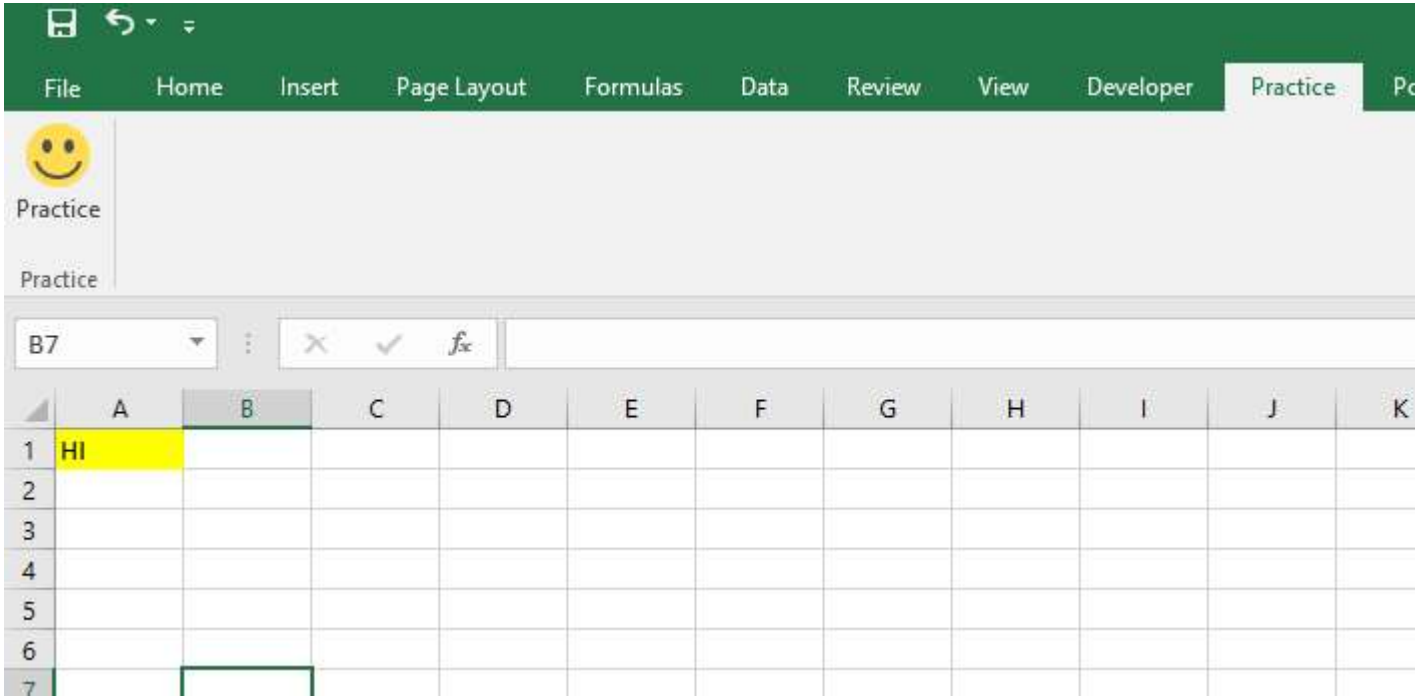


- Select “OK” to exit the customize Ribbon area.

msd
Safe, clean waterways

… Add our new macro to a Tab…

- The macro is now added to the new tab called “Practice”



DEMONSTRATION END

msd
Safe, clean waterways

A look at Module Code Basic Examples:

- Loops
- Input Boxes
- If / Then Statements

msd
Safe, clean waterways

Background Coding Speak:

- “DIM” means to declare or assign
- “Variable” is a block of memory that holds info; its result can vary or change. There are MANY types of variables: Dates, Integers, Strings, Long, etc.
- Our focus for today:
  - “Integer” a variable that holds numbers
  - “String” a variable that holds a combination of number and letters; assigned to “non-numerical values”

msd
Safe, clean waterways

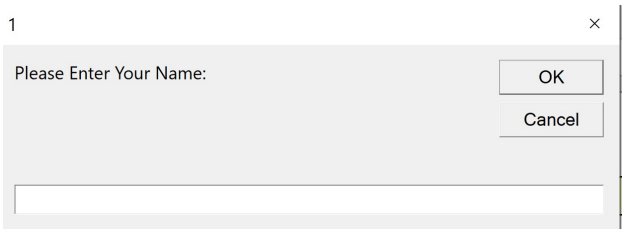
Loops: “Go Round and Round”

```
'Loops  Many types:  For/Next,  Do While, Do Until

Sub ForNextLoop1()

Dim Num1 As Integer

   For Num1 = 1 To 5

      MsgBox ("Num1 value: " & Num1)

   Next Num1


End Sub
```

msd
Safe, clean waterways

Input Boxes: “A way to assign variables”

```
'Input box:  A way to enter information.

Sub InputBox1()

Dim Input1 As String

    Input1 = InputBox("Please Enter Your Name:", vbOK)

    MsgBox ("Hello there " & Input1)

End Sub
```



msd
Safe, clean waterways

If / Then Statements: “If this… then that”

```
'If then statements:  A way to make decsions

Sub IfThenStatements()

Dim Animal As String
Animal = "Dog"


    If Animal = "Dog" Then MsgBox ("Your animal is a dog")
    If Animal <> "Dog" Then MsgBox ("Your animal is not a dog")


End Sub
```

msd
Safe, clean waterways

Let's put it all together:

```
Sub Combined() 'Combines all from above;  Asks the question two times (within a loop).

Dim Animal As String
Dim I As Integer

   For I = 1 To 2 'Question will be asked two times

      Animal = InputBox("Please name an animal: ", vbOKCancel)

      If Animal = "Dog" Then MsgBox ("Your animal is a dog")
      If Animal = "Cat" Then MsgBox ("Your animal is a cat")
      If Animal = "" Then MsgBox ("No animal selected.")

   Next I

End Sub
```

msd
Safe, clean waterways

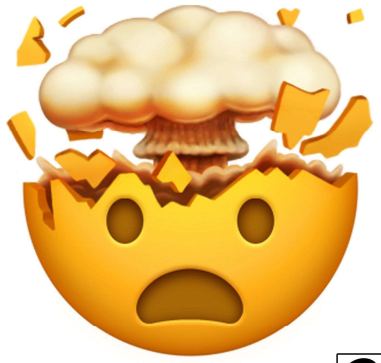
What are the possibilities?

- Balance Logs?
- Control Charts?
- Data Uploads?
- Reagent Logs?
- Receiving Logs?
- Etc. etc. Etc.

msd
Safe, clean waterways

Demonstration:

- Loops, Input Boxes and IF/Then Statements
- TSS Bench Sheet
  - Balance Logbook tied to sheet
  - Reagent/Standards Logbook tied to sheet
  - Control Charts
  - Audit Trail

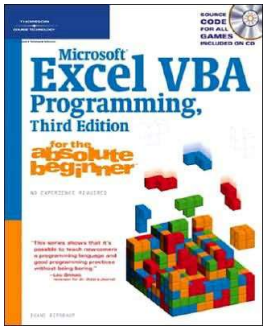


msd
Safe, clean waterways

Advice and What I've Found….

- Start Slow and "Play often"
- Resources and links:
  - Book: "For the absolute beginner" Series
  - Book: "Power Programing With VBA (J.Walkenbach)
  - GOOGLE!
    - Search: "Excel Macro to do "XYZ"; Paste code into Subs and play
  - https://www.youtube.com/watch?v=dmdolFcS-fI

    (Link to video showing how to customize ribbon)



msd
Safe, clean waterways

msd
Safe, clean waterways

THANK YOU!

QUESTIONS?

Contact: David.Radke@LouisvilleMSD.org

msd
Safe, clean waterways