Week9: First Tutorial

- Describing what is deemed as academic misconduct for this class
- Typing in macros manually
- Writing program documentation
- Declaring and using variables
- Getting user input
- VBA operators
- Branching: IF, IF-THEN
- Logic and branching

Academic Misconduct (Cheating)

- From the online document: https://pages.cpsc.ucalgary.ca/~tamj/resources/misconduct_203_VBA_information.html
- Key points
  - No group work is allowed
  - Do not include the work of other students in your submissions (to be safe students should not view other student assignment solutions)
  - You can include examples from: lecture, tutorials, textbooks or online
    - **Solutions from all of these external sources must clearly specify** which part of the assignment is external and which part was written by yourself

Author: James Tam

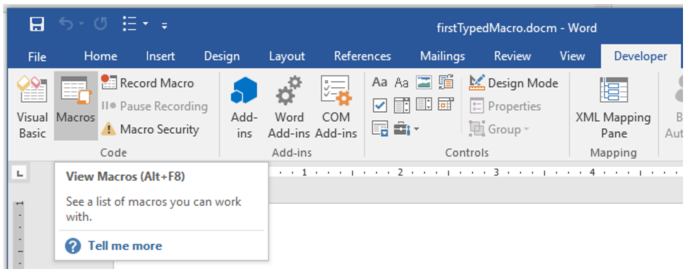
Creating Macros

1. Record the macro automatically: keystrokes and mouse selections will be stored as part of the macro (you will be briefly shown how to do this in tutorial)
2. **Manually enter the Macro** (type it in yourself into the VBA editor)
   - This is how you are to complete your assignment and is how many VBA programs are created.

Author: James Tam

Step 1: Finding The Macro Feature

- To enter a new macro/view previously created macros it's the same as viewing a recorded macro: "Developer->Macro"



View Macros (Alt+F8)

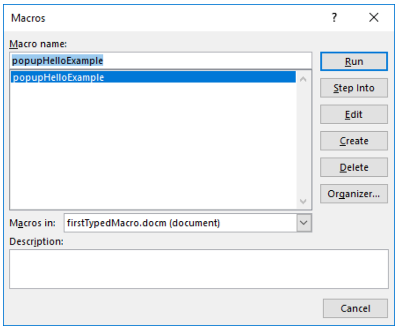
See a list of macros you can work with.

Tell me more

Author: James Tam

Step 2: Naming And Saving The Macro

- Enter the name “Macro name” using a good self-descriptive name (small examples that don’t carry out a useful function such as this example are more challenging to name).



Macro name: popupHelloExample

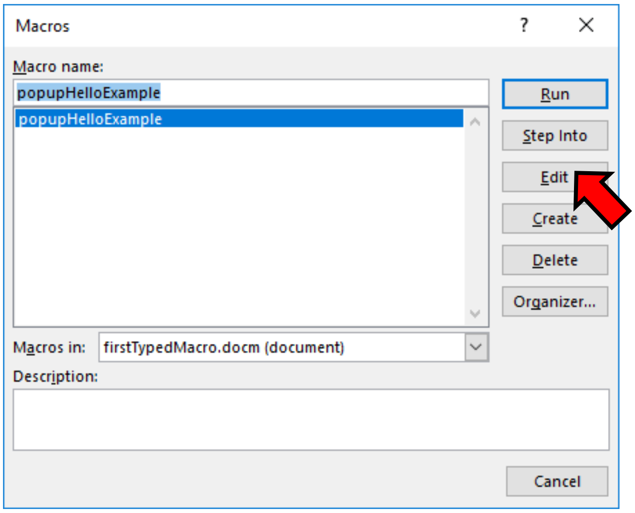
Macros in: firstTypedMacro.docm (document)

Description:

- Important! Make sure you **save the macro in the current document** and not all documents (“`Normal.dotm`”)

Author: James Tam

Step 3: Select The Edit Option



Macro name: popupHelloExample

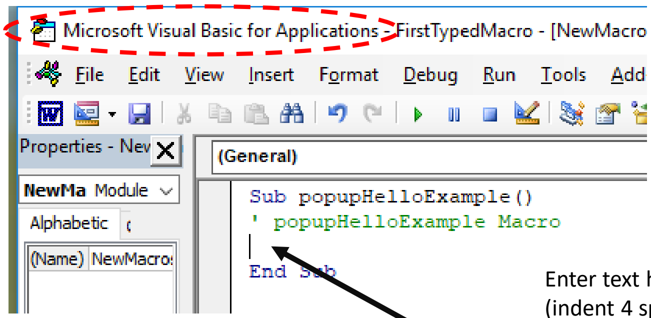
Macros in: firstTypedMacro.docm (document)

Description:

Author: James Tam

Step 4: Enter the Macro In The VBA Editor

- The **VBA editor** is not the same as the Word editor



```
Sub popupHelloExample()
' popupHelloExample Macro

End Sub
```

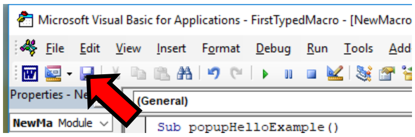
Enter text here
(indent 4 spaces)

```
MsgBox("Hello")
```

Author: James Tam

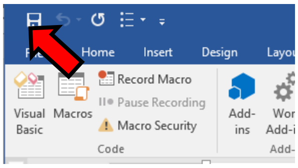
Step 5: Saving The Macro

- Typically if you either save via the VBA editor…



```
Sub popupHelloExample()
```

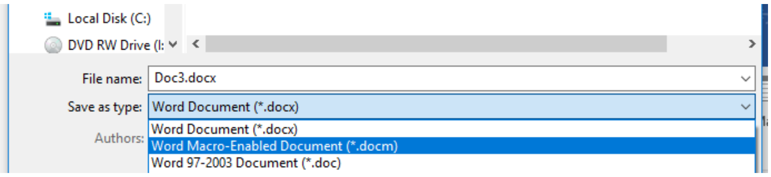
- …or the Word editor then the macro will be saved.



Author: James Tam

Step 5: Saving The Macro

- To be safe you can save using both editors but in any case make sure you save the Word document as a “Word Macro-Enabled Document” (*.docm)



File name: Doc3.docx

Save as type: Word Document (*.docx)

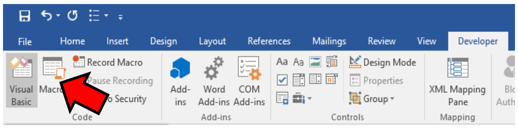
Authors:

- Word Document (*.docx)
- Word Macro-Enabled Document (*.docm)
- Word 97-2003 Document (*.doc)

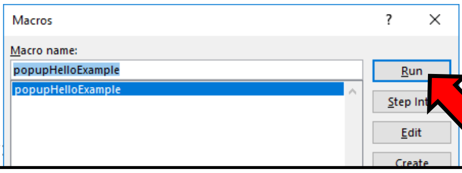
Author: James Tam

Running The Macro

- Regardless of how a macro has been created:
  - Automatically recorded
  - Manually entered
- A macro will be run using the same series of steps.
- Developer->Macros



- The single macro should be highlighted, then click ‘run’



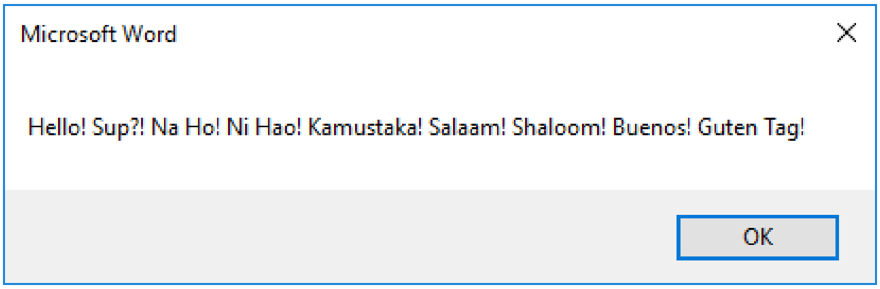
Macros

Macro name: popupHelloExample

popupHelloExample

Author

Result Of Running The Example Macro



Hello! Sup?! Na Ho! Ni Hao! Kamustaka! Salaam! Shaloom! Buenos! Guten Tag!

JT: if you need to see the solution to the exercise look at the document: `1firstTypedMacro.docm`

Author: James Tam

Student Exercise #1

- Writing and running a macro that displays your name
- Name to give your macro: ‘`firstMyName`’
- Name of macro-enabled Word document ‘`first.docm`’
- Write a program that will display your name using a `macro`

Author: James Tam

Program Documentation

- Doesn’t contain executable instructions.
- Created for the reader of the program (other programmers, in this class it’s for your assignment marker)
- **Format**:

```
' <Documentation>
```

- **Example**:

```
' Author: James Tam
```

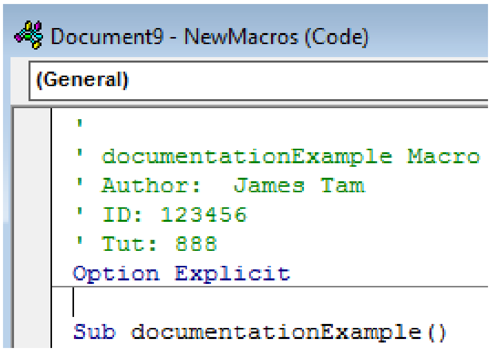
- No error: Everything after the quote until the end of the line will not be translated into machine language/binary
- That means documentation doesn’t have to be a valid and executable instruction

Program Documentation: Assignments (A3)

- Your VBA assignment submission must include identification about you and information the features of your program
  - Full name
  - Student identification number
  - Tutorial number
  - List the program features (from the assignment description) and clearly indicate if the feature was completed or not completed.
  - Program version

Location Of Program Documentation

- Contact information should be located before your program
- Before the ‘sub’ keyword
  - Before ‘Option explicit’ if included



```
'
' documentationExample Macro
' Author:   James Tam
' ID: 123456
' Tut: 888
Option Explicit

Sub documentationExample()
```

Documentation: marked in red

Author: James Tam

Variables

- A location in memory used to store information temporarily.
- At most a variable can store a single piece of information.

num = 1

num = 17

num

Author: James Tam

Creating/Declaring Variables

- Creating and naming memory locations
  - `Dim anInt As Long`
  - `Dim aReal As Double`
  - `Dim aString As String`

- Assigning values to those locations
  - `anInt = 2000`
  - `aReal = 3.14`
  - `aString = "Peter Griffin“`

Author: James Tam

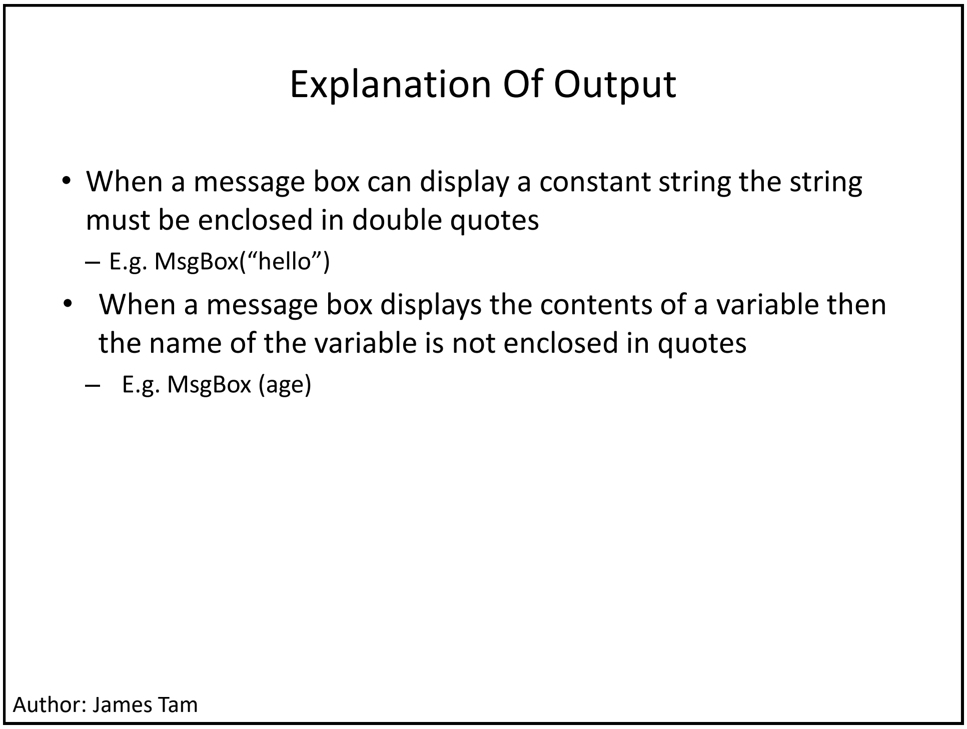
Displaying The Contents Of Variables

- Format:
  - `MsgBox (<Name of the variable>)`

- Example (assumes a variable called ‘age’ has already been created and a value has been assigned to it)
  - MsgBox (age)

- Question for students: What’s the output of the MsgBox
  - Dim name As String
  - Name = "Smith“
  - MsgBox (name)

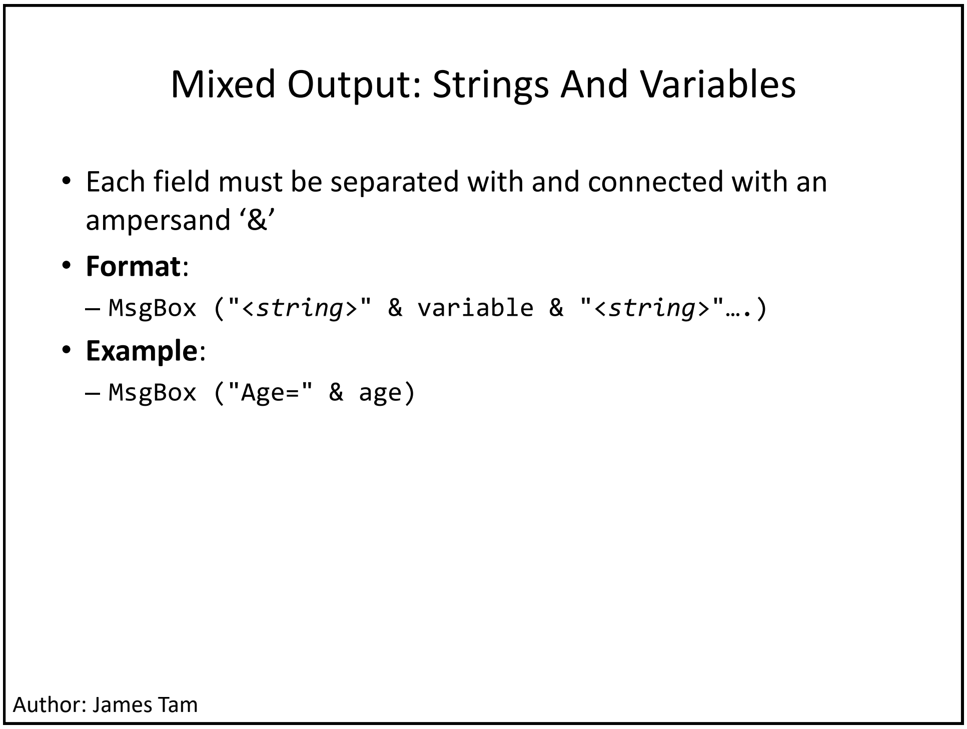
Author: James Tam



Explanation Of Output

- When a message box can display a constant string the string must be enclosed in double quotes
  - E.g. MsgBox(“hello”)
- When a message box displays the contents of a variable then the name of the variable is not enclosed in quotes
  - E.g. MsgBox (age)

Author: James Tam



Mixed Output: Strings And Variables

- Each field must be separated with and connected with an ampersand ‘&’
- **Format**:
  - `MsgBox ("<string>" & variable & "<string>"….)`
- **Example**:
  - `MsgBox ("Age=" & age)`

Author: James Tam

Mixed Output Example

- **`2variablesMixedOutput.docm`**

```
Sub variablesMixedOutput()
    Dim age As Long
    Dim name As String
    age = 37
    name = "Homer J. Simpson"
    MsgBox ("Name: " & name & ", " & "Age: " & age)
End Sub
```

Getting Input

- Getting information from the user as the program runs via an `InputBox`
- **Format**:
  - `InputBox ("<`*`Prompting message`*`>")`
- **Example**:
  - `InputBox ("Tell me your name")`

Input Example

- **3inputExample.docm**

```
Sub inputExample()
    Dim age As Long
    Dim name As String
    age = InputBox("Tell me your age: ")
    name = InputBox("What is your name: ")
    MsgBox ("Name: " & name & ", " & "Age: " & age)
End Sub
```

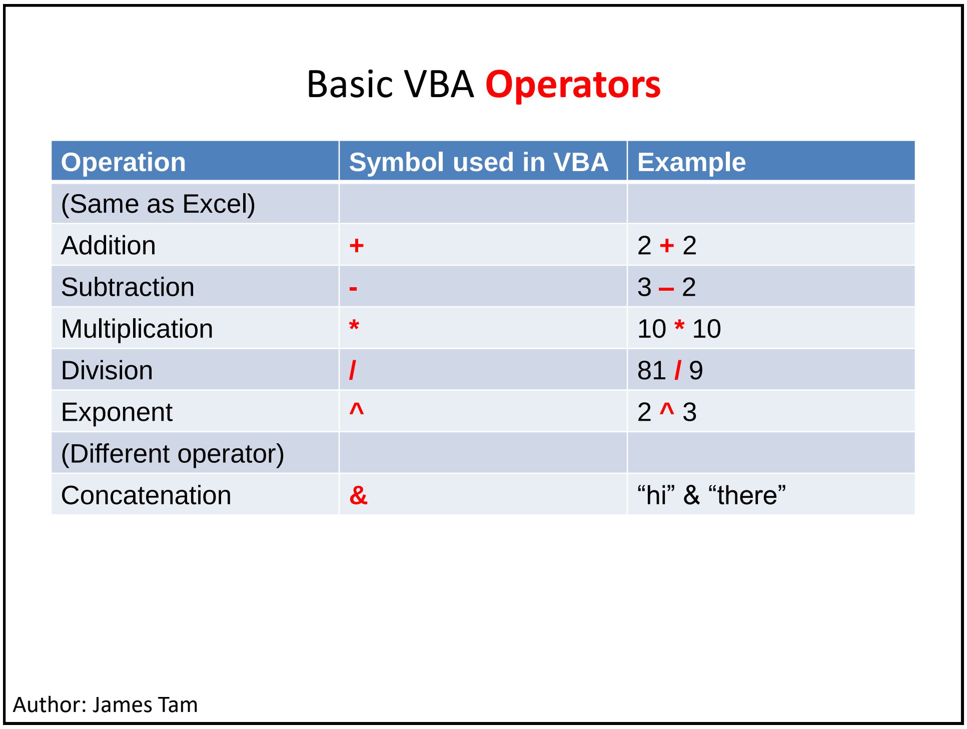
Notice how the values displayed in the MsgBox will vary in this example according to user input

Author: James Tam

Student Exercise #2

- Write a new VBA program in a new Word document
- Declare a String variable called ‘name’
- Write a program that will ask the user for their name using an InputBox and store the input in the variable called ‘name’
- The program will then display the value entered by the user using a MsgBox

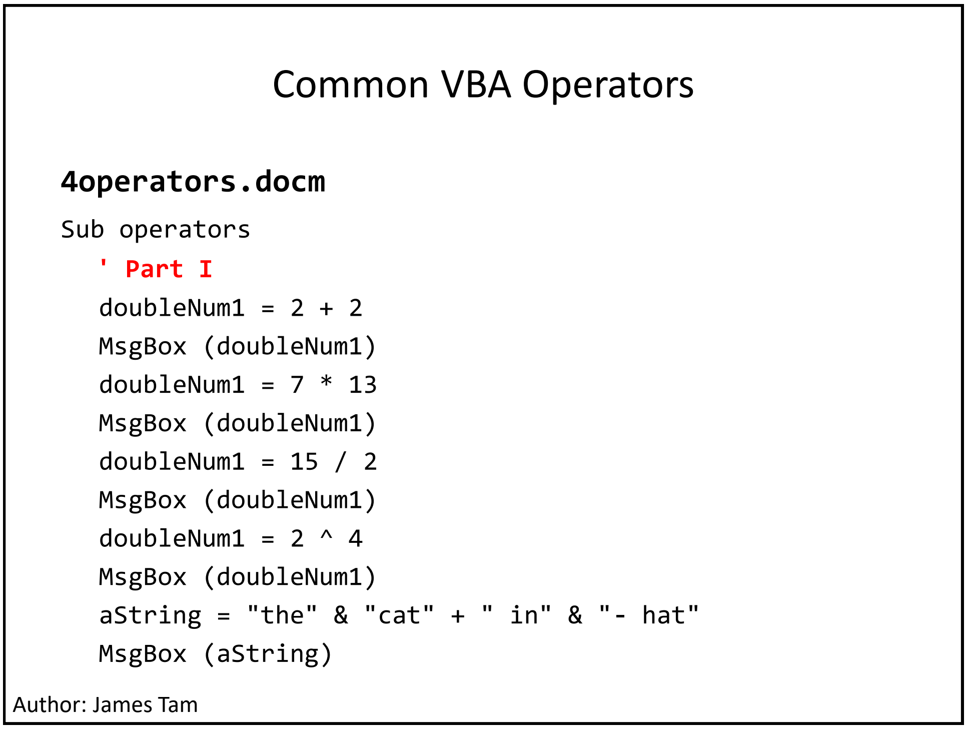
Author: James Tam



Basic VBA **Operators**

Operation	Symbol used in VBA	Example
(Same as Excel)		
Addition	**+**	2 **+** 2
Subtraction	**-**	3 **–** 2
Multiplication	*****	10 ***** 10
Division	**/**	81 **/** 9
Exponent	**^**	2 **^** 3
(Different operator)		
Concatenation	**&**	“hi” & “there”

Author: James Tam



Common VBA Operators

4operators.docm

```
Sub operators
   ' Part I
   doubleNum1 = 2 + 2
   MsgBox (doubleNum1)
   doubleNum1 = 7 * 13
   MsgBox (doubleNum1)
   doubleNum1 = 15 / 2
   MsgBox (doubleNum1)
   doubleNum1 = 2 ^ 4
   MsgBox (doubleNum1)
   aString = "the" & "cat" + " in" & "- hat"
   MsgBox (aString)
```

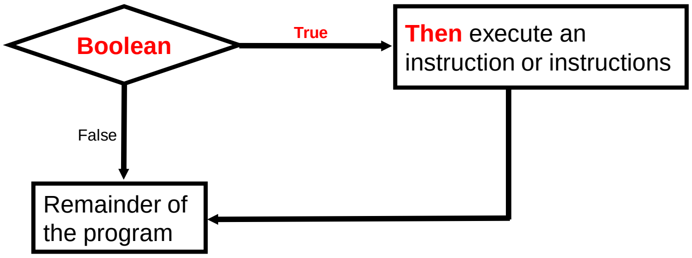
Author: James Tam

Common VBA Operators (2)

```
    ' Part II
    doubleNum1 = 2
    doubleNum2 = 3
    doubleNum3 = doubleNum1 ^ 3
    doubleNum1 = doubleNum1 * doubleNum2
    doubleNum2 = doubleNum1 + doubleNum3
    MsgBox ("doubleNum1=" & doubleNum1 & "-" &
      "doubleNum2=" & doubleNum2 & "-" & "doubleNum3=" &
      doubleNum3)
 End Sub
```

Author: James Tam

Decision Making With ‘**If-Then**’



Author: James Tam

Branching: IF Example

- **`5branchingLogic.docm`**
  - All versions of Part I: program displays an error message if age is not in range from 0 – 114 years.

```
Sub fiveBranchingLogic()
    Dim age As Long
    Dim userInput As String
```

Author: James Tam

Branching: IF Example (2)

```
' Part I:
age = InputBox("Age (0 - 114)?")
If ((age < 0) Or (age > 114)) Then
    MsgBox ("Age is outside allowable range of 0 - 114")
End If
If ((age >= 0) And (age <= 114)) Then
    MsgBox ("Age of " & age & " is OK")
End If
```

Author: James Tam

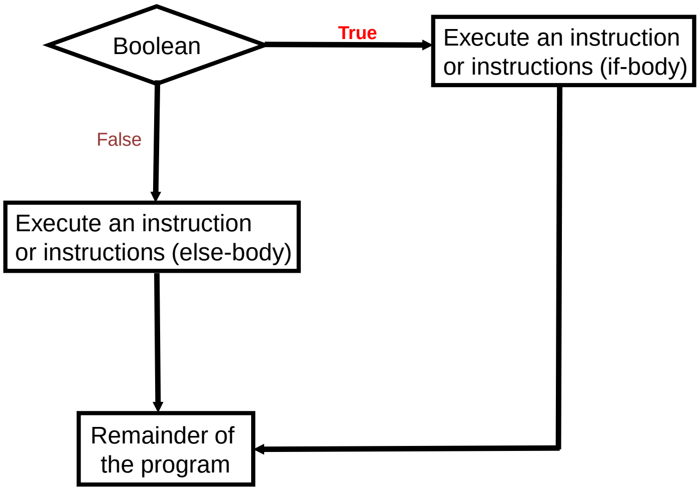
Branching: IF Example (2)

```
    ' Part II: Using negation (NOT)
    userInput = InputBox("Password?")
    If Not (userInput = "password") Then
        MsgBox ("Password incorrect")
    End If

End Sub
```

Author: James Tam

Decision Making With An ‘If, Else’



Boolean

True

Execute an instruction or instructions (if-body)

False

Execute an instruction or instructions (else-body)

Remainder of the program

Author: James Tam

Branching: IF-ELSE Example

- **6ifElse.docm**
  - Part I of last example augmented (using Else) to display an alternate message when the input is valid

```
Sub sixBranchingLogic()
    Dim age As Long
    Dim userInput As String
```

Author: James Tam

Branching: IF-ELSE Example (2)

```
' Part I:
age = InputBox("Age (0 - 114)?")
If ((age < 0) Or (age > 114)) Then
    MsgBox ("Age is outside allowable range of 0 - 114")
Else
    MsgBox ("Age of " & age & " is OK")
End If
If ((age >= 0) And (age <= 114)) Then
    MsgBox ("Age of " & age & " is OK")
Else
    MsgBox ("Age is outside allowable range of 0 - 114")
End If
```

Author: James Tam

Student Exercise #3

- Ask the user for ‘age’ and when the value is negative the program displays an error message via MsgBox “Age cannot be less than zero”

Author: James Tam

Student Exercise #4

- Modify the previous program so **in addition to** displaying the previous error message when age is negative the following message will be displayed **when age is not negative**:
  - “age is…” as well as the age
  - E.g. 1: the user enters -1 for age and the message “`Age cannot be less than zero`” `appears`
  - E.g. 2: the user enters 114 for age and the message “`Age is...114`”

Author: James Tam

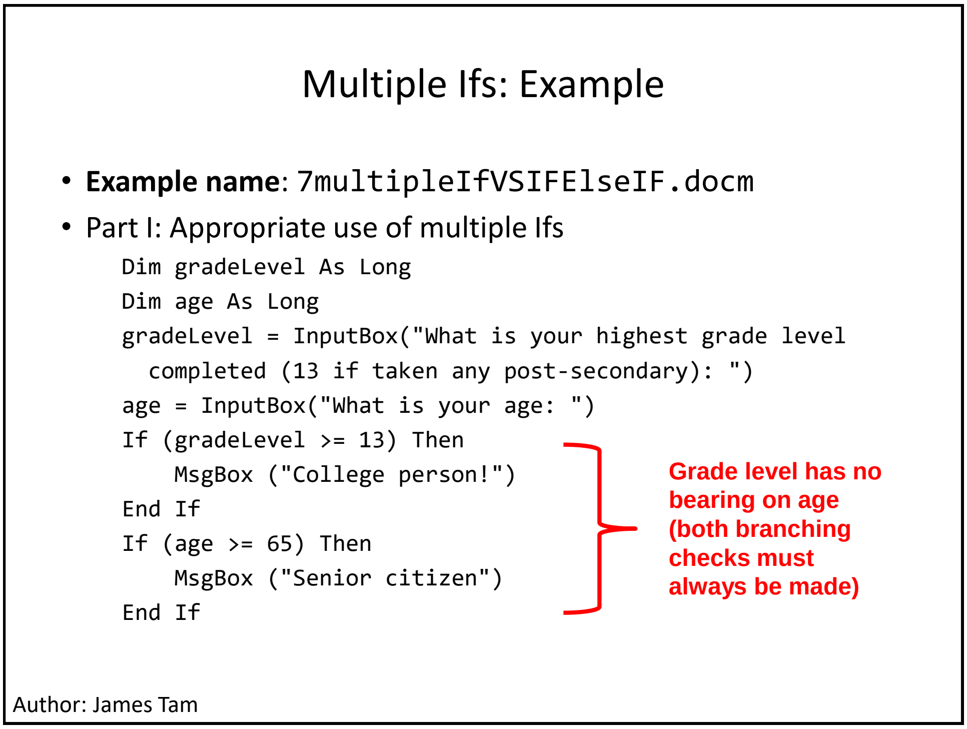
Week9: Second Tutorial

- Branching: Multiple IFs and IF-ELSEIF
- While loops
  - Simple counting loops
  - Loops for error handling
- Loops vs. branches
- Nesting

Multiple Ifs: When To Use

- When each condition must be checked (each ‘question’ must always be ‘asked’)
- Example:
  - Q1: Are you an adult?
  - Q2: Are you a Canadian citizen?
  - Q3: Are you currently employed?

Author: James Tam



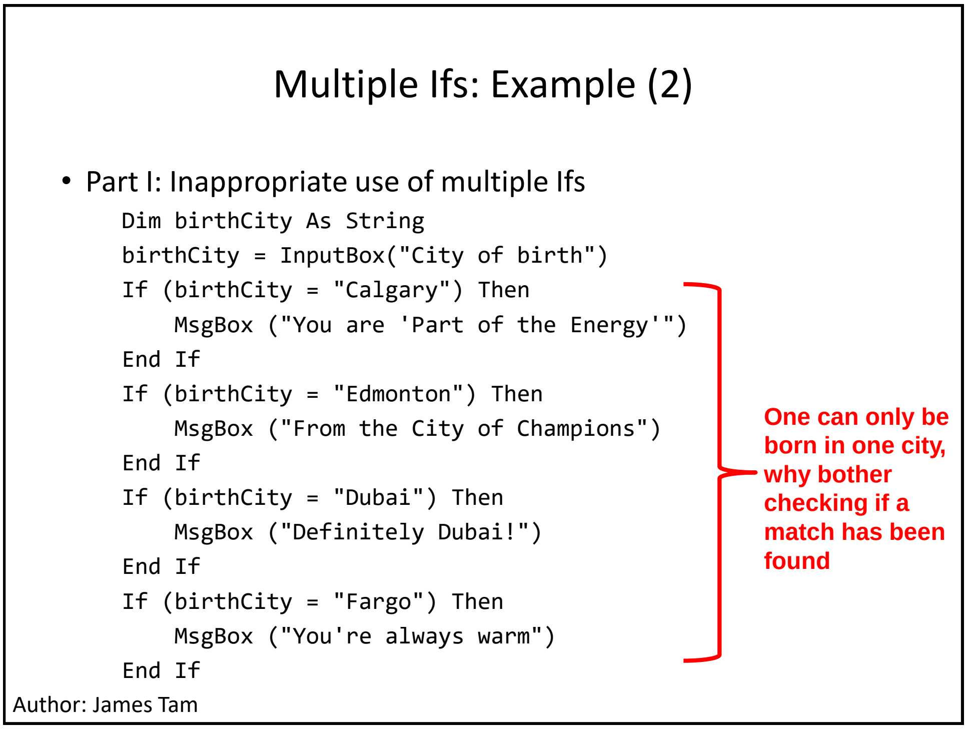
Multiple Ifs: Example

- **Example name**: `7multipleIfVSIFElseIF.docm`
- Part I: Appropriate use of multiple Ifs

```
Dim gradeLevel As Long
Dim age As Long
gradeLevel = InputBox("What is your highest grade level
  completed (13 if taken any post-secondary): ")
age = InputBox("What is your age: ")
If (gradeLevel >= 13) Then
    MsgBox ("College person!")
End If
If (age >= 65) Then
    MsgBox ("Senior citizen")
End If
```

Grade level has no bearing on age (both branching checks must always be made)

Author: James Tam



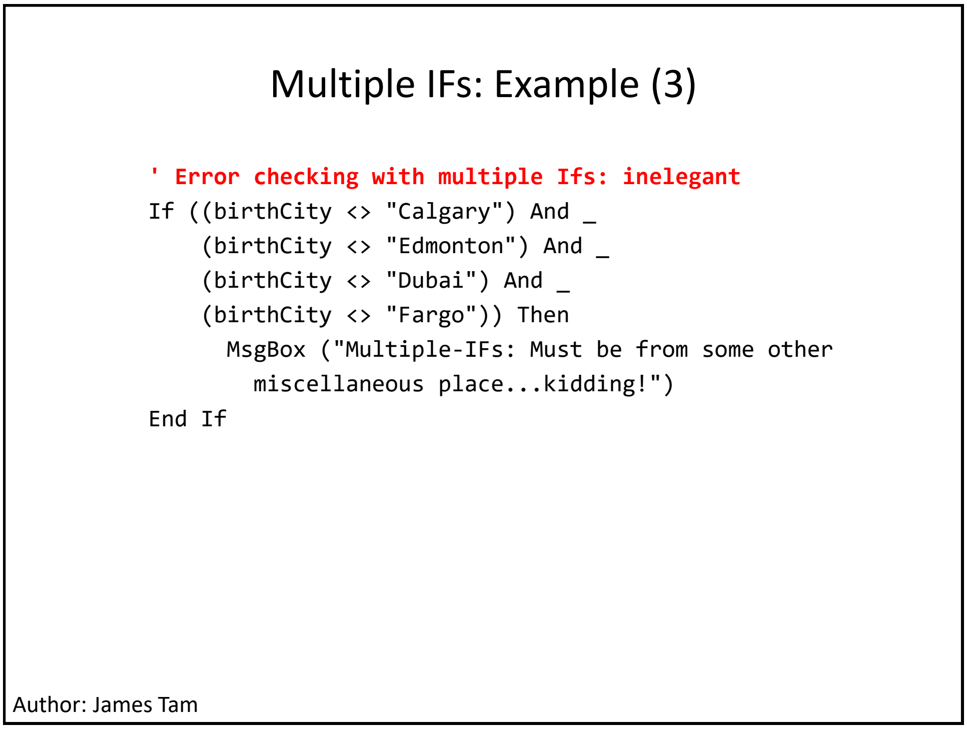
Multiple Ifs: Example (2)

- Part I: Inappropriate use of multiple Ifs

```
Dim birthCity As String
birthCity = InputBox("City of birth")
If (birthCity = "Calgary") Then
    MsgBox ("You are 'Part of the Energy'")
End If
If (birthCity = "Edmonton") Then
    MsgBox ("From the City of Champions")
End If
If (birthCity = "Dubai") Then
    MsgBox ("Definitely Dubai!")
End If
If (birthCity = "Fargo") Then
    MsgBox ("You're always warm")
End If
```

One can only be born in one city, why bother checking if a match has been found

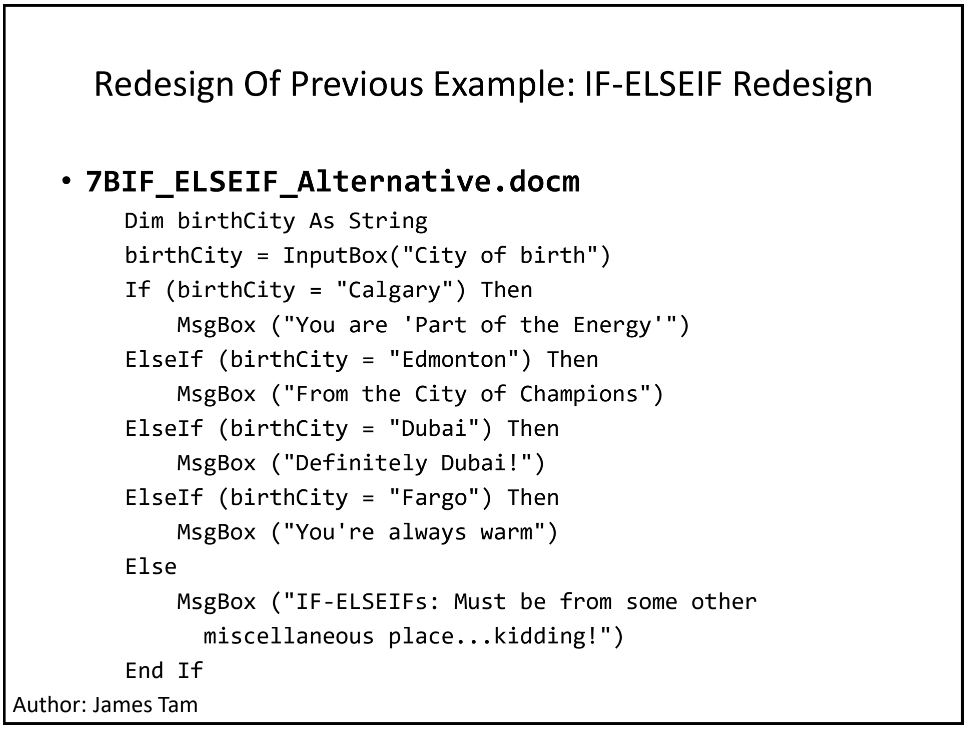
Author: James Tam



Multiple IFs: Example (3)

```
' Error checking with multiple Ifs: inelegant
If ((birthCity <> "Calgary") And _
    (birthCity <> "Edmonton") And _
    (birthCity <> "Dubai") And _
    (birthCity <> "Fargo")) Then
      MsgBox ("Multiple-IFs: Must be from some other
        miscellaneous place...kidding!")
End If
```

Author: James Tam



Redesign Of Previous Example: IF-ELSEIF Redesign

- **7BIF_ELSEIF_Alternative.docm**

```
Dim birthCity As String
birthCity = InputBox("City of birth")
If (birthCity = "Calgary") Then
    MsgBox ("You are 'Part of the Energy'")
ElseIf (birthCity = "Edmonton") Then
    MsgBox ("From the City of Champions")
ElseIf (birthCity = "Dubai") Then
    MsgBox ("Definitely Dubai!")
ElseIf (birthCity = "Fargo") Then
    MsgBox ("You're always warm")
Else
    MsgBox ("IF-ELSEIFs: Must be from some other
      miscellaneous place...kidding!")
End If
```

Author: James Tam

Loops/Repetition

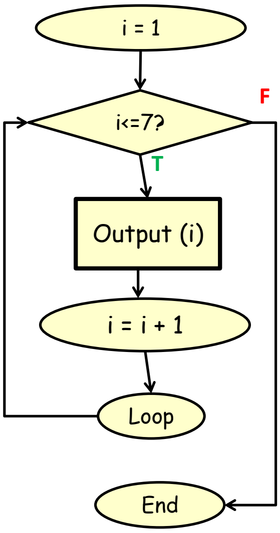
- Getting programs to automatically repeat an instruction or instructions without duplicating parts of the program.
- **New term 'iteration'**: each time that a program repeats using a loop is referred to as an 'iteration'.
- **New term 'loop control'**: a variable that determines if a loop a loop will execute (it 'controls' execution)
  - `while (i < 10):`

Author: James Tam

First Looping Example

- **8loopV1.docm**

```
Sub eightLoopV1()
   Dim i As Long
   i = 1
   Do While (i <= 7)
       MsgBox ("i=" & i)
       i = i + 1
   Loop
End Sub
```



Author: James Tam

Second Looping Example

- **9loopV2.docm**

```
Dim i As Long
i = 0
Do While (i <= 7)
    MsgBox ("i=" & i)
    i = i + 1
Loop
```

Author: James Tam

Student Exercise #5

- Write a loop that it iterates (steps through the numbers) 1 – 100 (1, 2, 3…100)
- A message box will only display the value of the loop control **after** the loop has ended (i.e. once)

Author: James Tam

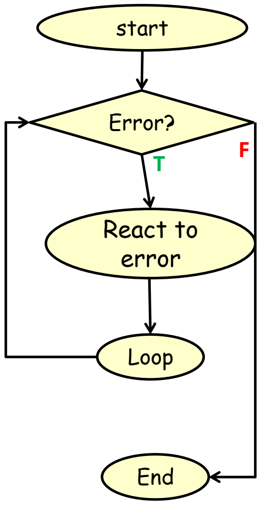
Student Exercise #6

- Modify this program so that it instead iterates through the sequence 5 – 8 (5, 6, 7, 8)
- This time the message box can appear inside the body of the loop and display the value of the loop control as it takes on the four values in the above sequence.

Author: James Tam

Error Handling Loops

- Repeat/iterate so long as an error condition occurs



Author: James Tam

NAFTA Countries Only

- Error checking loop: N_AMERICAN country not one of America, Canada, Mexico
- **`10errorCheckingLoopCountry.docm`**

(Subroutine Version1)

```
Dim country As String
country = InputBox("Enter your North American country")
Do While (country <> "america") And _
   (country <> "canada") And _
   (country <> "mexico")
    country = InputBox("Enter your North American
      country")
Loop
MsgBox ("Your country: " & country)
```

Author: James Tam

NAFTA: A Slightly More Elegant Version

- Doesn't ask user before the loop
- Instead the loop is primed so that it is guaranteed to execute at least once
- (Subroutine Version2)

```
Dim country As String
country = "None of the above"
Do While (country <> "america") And _
   (country <> "canada") And _
   (country <> "mexico")
    country = InputBox("Enter your North American
     country")
Loop
MsgBox ("Your country: " & country)
```

Author: James Tam

Student Exercise #7

- (Subroutine Version3)
  - Changes 'And' to 'Or': what happens? Hand trace it first!
  - **Don't run it if you have other important Word documents open**

```
Dim country As String
country = InputBox("Enter your North American country")
Do While (country <> "america") Or _
   (country <> "canada") Or _
   (country <> "mexico")
    country = InputBox("Enter your North American
      country")
Loop
MsgBox ("Your country: " & country)
```

Author: James Tam

Common Mistake

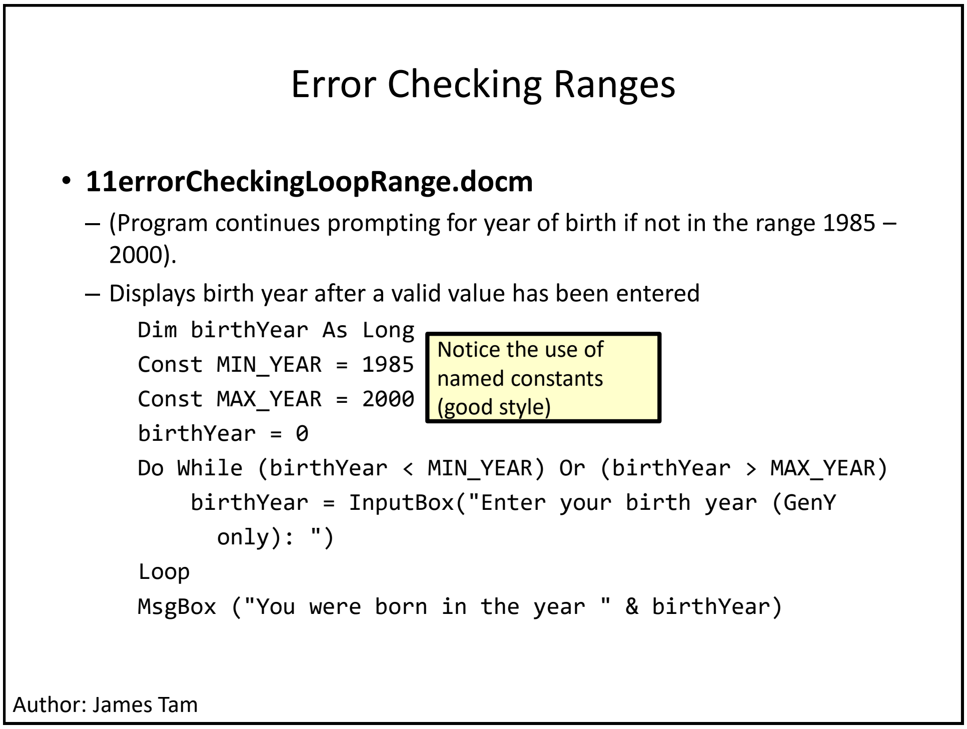
Try the following g values for the country:

- No Error: canada
- Error in data: USA (and then USA again)

- Using a branching IF instead of a loop (while)
- Remember a loop repeats, a branch just splits execution one way or another
  - (Subroutine Version 4)

```
Dim country As String
country = InputBox("Enter your North American country")
If (country <> "america") And _
   (country <> "canada") And _
   (country <> "mexico") Then
     MsgBox ("Your country: " & country)
     country = InputBox("Enter your North American
       country")
End If
```

Author: James Tam



Error Checking Ranges

- **11errorCheckingLoopRange.docm**
  - (Program continues prompting for year of birth if not in the range 1985 – 2000).
  - Displays birth year after a valid value has been entered

```
Dim birthYear As Long
Const MIN_YEAR = 1985
Const MAX_YEAR = 2000
birthYear = 0
Do While (birthYear < MIN_YEAR) Or (birthYear > MAX_YEAR)
    birthYear = InputBox("Enter your birth year (GenY
      only): ")
Loop
MsgBox ("You were born in the year " & birthYear)
```

Notice the use of named constants (good style)

Author: James Tam