CP212 Midterm Review

Original on Google Drive

Details 

Date: Wednesday, Oct 18, 2017
Weight: 15%
Location: At home or in class
Format: Short answer, some code writing, given some code find the errors, what does this code do?
Content: Chapters 1 - 8

New for 2017: Online “quiz” format - midterm available from 9am - 9pm.

1 hour to complete the exam once you start. You cannot stop and continue later.

Details 

Short answer:
VBA, Excel, Excel Object Model Hierarchy
Using / navigating Excel and the VBE

Writing Code:
- Collections
- Search for items in a range
- Working with arrays

Finding Errors

Your code won't have to compile correctly, but it should be mostly correct, and in VBA ;)

Practical Study Technique #1- Keywords

- Make a list of VBA keywords we've studied so far
- Understand how they are used - how you use them in code

Practical Study Technique #2 - Concepts

- Make a list of the concepts we've covered
- If they completely make sense, don't spend a lot of time on them - you already know them!
- If there is confusion, look it up, study, ask for help... remove the confusion

Practical Study Technique #3 - Chunking

Chunking

This is a process of reducing your textbook and lecture notes over and over again. You make summaries of your summaries of your summaries of your .... It is best to do this at least three times, preferably five. Each time you make a new summary by reducing the previous one, you are revising the material and learning more of both the detail and main ideas. Each time you summarise you need less writing, headings, outline or detail to remember the same amount of information. Eventually you can lock up whole chunks of material behind a few well-chosen terms or brief lists or skeleton diagrams. (http://www.usq.edu.au/learningcentre/alsonline/effstudy/studytech)

- Break the topic into pieces (ie. chapters)
- Summarize each section
  - then summarize the summary
    - then summarize the summary

Content

- Excel / IDE basics - Chapters 1 - 3
  - opening the VBE, parts of the VBE
- 3 Types of Errors

Content cont'd

Chapter 5

- Subroutines
- Variables (declaring them, basic data types)
- Simple user input and output
- Strings
- Objects, Properties and Methods
- The **With** construct

```
Sub Basic2()
    Range("A1").Select
    Selection.Font.Bold = True
    Selection.Font.Italic = False
    With Selection.Font
        .Color = -16776961
        .TintAndShade = 0
    End With
End Sub
```

```
Dim wks As Worksheet
Set wks = ActiveSheet

With wks.Range("A1").Font
    .Bold = True
    .Italic = False
    .Color = rgb(255,0,0)
End With
```

Chapter 6

- Ranges
  - Properties
  - Methods
- Properties
  - Address
  - Cells
  - Font: Range("A2").Font.Bold = True
    - Range("A3").Font.Italic = True
  - Formula
  - Name: Range("A3:A10").Name = "Sales"
  - Value: Range("B15").Value = 17
- Methods: Clear, Copy, Select, ClearContents
- Specifying a range, Offset
- Using the End property

Chapter 6 Con't

- use top left and bottom right arguments
  - Range(Range("C2"),Range("D5"))
  - combined with the With command
    - With Range("A1")
      Range(.Offset(1,1), .Offset(3,3)).Select
      End With
- use the End property
  - with built-in constants (xlDown, xlUp, xlToRight, xlToLeft) it will select the entire used range, regardless of size
  - if you don't know the size of the range:
    - With Range("A1")
      Range(.Offset(0,0), .End(xlDown).End(xlToRight)).Select
      End With

Chapter 6

Write a line of code to do the following:

- Change cell A1 to Bold.
- Change cell B5 to Bold and Italic. (requires 2 lines of code)
- Set the formula in cell K9 to calculate the sum from K1 to K8.
- Set the value in cell B8 to total of the average from L1 to L7 (use WorksheetFunction)

Chapter 7 - Logic and Loops

- R1C1 notation
- If... Then ... Else and variations
  - ElseIf and End If
- Select Case
- For ... Next loops
- For Each
- Do While and Do Until
- Basic error handling with logic
- Searching example: Given a range, how do you find a value?
- Given a column of values, how do you add a new item to the end?

What is the output?

```
Sub SimpleLogic()

    Dim answer As VbMsgBoxResult

    answer = MsgBox("Do you want to continue?", vbYesNoCancel, "Yes or No")

    If answer = vbYes Then
        MsgBox "Great, thanks!"
    ElseIf answer = vbNo Then
        MsgBox "Fine, be that way."
    Else
        MsgBox "Away you go then."
    End If

End Sub
```

What is the output?

```
Sub SelectCase()
    Dim result As Integer
    Dim output As String

    result = InputBox("Enter a number between 1 and 10:", "Choose a number", "7")

    Select Case result
        Case 1
            output = "You chose a good one."
        Case 2 To 3
            output = "You chose either 2 or 3."
        Case 4, 6
            output = "You chose either 4 or 6."
        Case Else
            output = "There was some other choice."
    End Select

    ' Output the result based on their choice.
    MsgBox output, vbOKOnly, "Your choice"

End Sub
```

What is the output if the user enters 5?
Practice changing the code to change it to a Do.While loop and to check the condition at the end of the loop.

```
Sub DoUntilExample()

    Dim result As Integer
    Dim output As String

    ' You can modify this exampe to a Do While loop, and check the
    ' condition at the end of the loop or the top to see how the
    ' result differs.

    Do Until result = -1
        result = InputBox("Enter a number between 1 and 10. Enter -1 to quit.", "Choose a number", "7")

        Select Case result

            Case Is < 0
                output = "Thank you for quitting."
            Case 1
                output = "You chose a good one."
            Case 2 To 3
                output = "You chose either 2 or 3."
            Case 4, 6
                output = "You chose either 4 or 6."
            Case Else
                output = "There was some other choice."
        End Select

        ' Output the result based on their choice.
        MsgBox output, vbOKOnly, "Your choice"

    Loop

End Sub
```

What is the output?

```
Sub ForExample()

    Dim start As Integer
    Dim theEnd As Integer
    Dim counter As Integer
    Dim output As String

    start = 1
    theEnd = 4

    For counter = start To theEnd
        If counter = theEnd Then
            output = output & counter & "."
        Else
            output = output & counter & ", "
        End If
    Next

    MsgBox output, vbOKOnly
End Sub
```

Chapter 8 - Collections

- Excel Object Hierarchy
- Object describing the file in those applications?
- What object represents an Excel file?
- Workbooks
- Worksheets

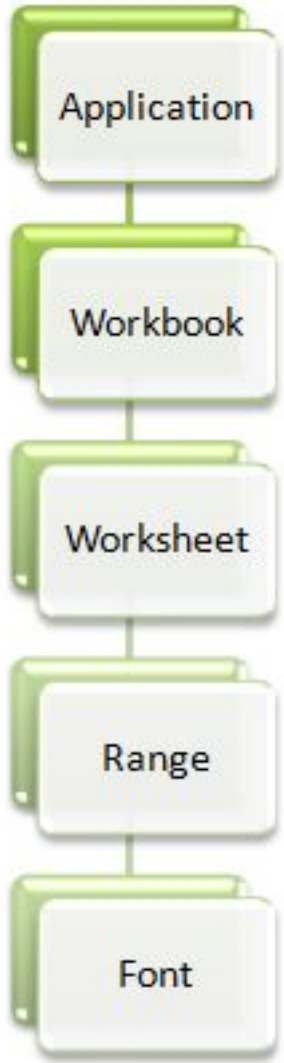
Chapter 8 - Collections

Write 1 or more lines of code to do the following.

- Set the value of the range called "Total" on the worksheet called "Sales" to be the sum from A1:A8
- Write a loop that counts the number of worksheets in the current workbook. Don’t use Worksheets.Count.

Excel Object Hierarchy

- a simplified model



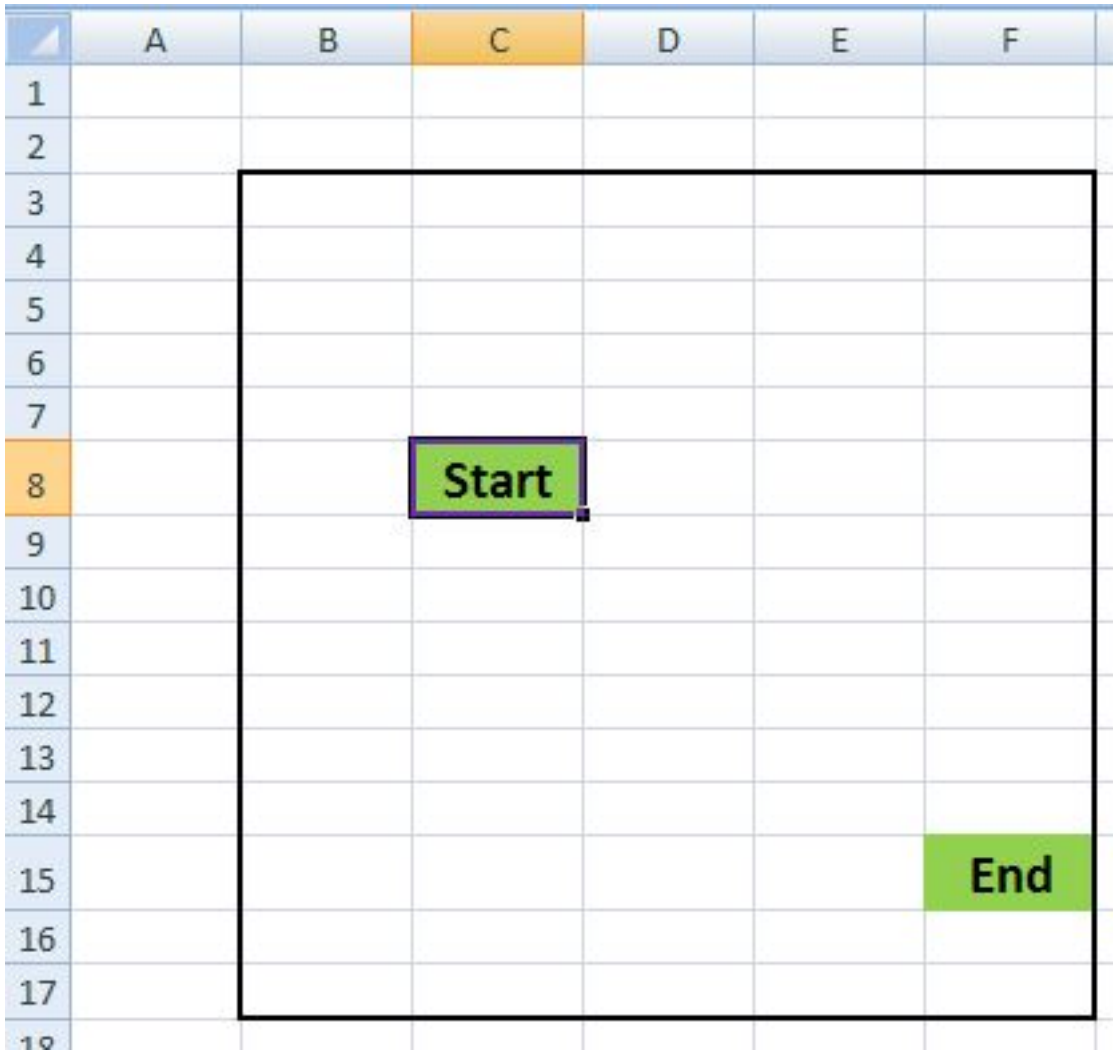
What's wrong with this code?

```
Sub SayHello()
' Declare variables before use when Option Explicit turned on.
    Dim Msg As String
    Dim Ans As VBMsgboxResult

    Msg = "Is your name " & Application.UserName & "?"
    Ans = Msgbox Msg, vbYesNo

    If Ans == vbNo Then
        Msgbox "Oh, never mind."
    Else
        Msgbox "I must be clairvoyant!"
    EndIf
End Sub
```

There are 3 problems.

Use R1C1 Notation

Reminder:
Relative
vs.
Absolute
Notation!



Using the START as a reference, what is the address of the END position in relative R1C1 notation? In absolute?

How about as an Offset? - Range("C8").Offset(?,?)

Recommended Exercises

A list of Recommended Exercises is available on the Lab Home Page.

http://bohr.wlu.ca/rhenderson/cp212/recommended_exercises.html