

Workbook exercise #3: First VBA exercise

Due dates for book exercises can be found on the main grid of the course webpage:

- https://pages.cpsc.ucalgary.ca/~tamj/2020/203F/index.html#Main_grid

Location of starting document with VBA program and the Word document that will be opened by your VBA program:

https://pages.cpsc.ucalgary.ca/~tamj/books/chop_suey_chop_chop/example_s/VBA/step_by_step/starting/

Starting document: “Typing a program, opening a specified document - starting”. It’s not mandatory that you use this starting document. However it is provided to ensure that you enter your program into the type of Word document that can contain a macro. (By default Word documents cannot have macros saved within them).

Sample data document (used by the program to be created): Example document1 (make sure that this document is located in the same folder that contains your VBA program). It’s not mandatory that you use this file for credit but it’s provided so you have something available.

Writing a VBA macro program for Word:

You will be writing a program using the VBA programming language that will prompt the user for a path (location) and a name of a Word document. The program will then try to open document at the specified location.

- **1.0 GPA** for having the program automatically find the path to the currently active Word document.
- **1.0 GPA** for getting the name of the document from the user.
- **2.0 GPA** for opening the document using Word.

Step 1: Open Word and create a new blank document or open the starting document. Close any other Word documents that you may have opened.

Step 2: Save the document under a different name. Under the **File** tab select the **Save As** menu option.

Step 3: Select a location that you can remember.

Step 4: In the file name field rename the document as “Typing a program, opening a specified document - solution.docm”

Step 5: To create a new macro in Word look at the **View** tab (**Macros** group) and click on the ‘down’ arrow triangle when you can then select the “**View Macros**” option in the menu that appears.

Step 6: Look for the input field under the label “**Macro Name**” and enter the following ‘openingSpecifiedDocument’ but without the quotes.

Step 7: Look for the label “**Macros in**” and select under the pulldown menu the name of the starting document **Typing a program, opening a specified document - solution.**

Step 8: Under the text Macro name type in ‘openingSpecifiedDocument’

Step 9: Click on the ‘**Create**’ button.

Step 10: In the VBA code editor, after line of red colored documentation and before the line containing “**End Sub**” enter the following instructions for your VBA program. As a reminder: before typing in each line containing an instruction you should first hit tab so the instruction is indented.

Step 10A: `Dim documentName As String`

Step 10B: `Dim path As String`

Step 10C (long instructions such as the following should be entered all on a single line in the VBA editor): `documentName = InputBox("Type in the name of the document to open (e.g. Example document1): ")`

Step 10D: `Documents.Open (documentName)`

Step 11: Run the program, click on the ‘**Play**’ icon. An error message appears. The specific content of the message will vary. The gist of the message indicates that the file with the name you just typed in could not be found. The error message should also specify the path or location where it tried to open this file. You should be able to verify that this path (or at least the part that you can see in the popup if the path is a long one) is the location of the Word document containing this program.

Step 12: Go back to the code editor and type in the following after what you entered in Step 10C `documentName = InputBox(...)` and before what you entered in Step 10D `Documents.Open(...): path = ActiveDocument.path`

Step 13: Delete the instruction that you previously entered in Step 10D `Documents.Open (documentName)`. Retype on this line the following new instruction: `Documents.Open (path & documentName)`

Step 14: Run the program, click on the ‘**Play**’ icon. An error message appears. Look carefully at the information specifying the path and file name. You should see that the name of the last folder and the file name you entered are shown as one word. For instance if program tried to open a file at the location: `C:\203\Ex1` and you typed in `Example document1` for the name of the file then the path and the file name would appear as `C:\203\Ex1Example document1`. In this example it’s trying to open a file called `Ex1Example document1` in the location `C:\203`. A slash is needed to separate the name of the last folder `Ex1` from the name of the file that the program is trying to open `Example document1`.

Step 15: Delete the instruction from Step 13 `Documents.Open (path & documentName)`. Retype on this line the following new instruction: `Documents.Open (path & "\" & documentName)`

Step 16: Run the program, click on the 'Play' icon. Another error message appears. This one will likely be more challenging for most to decipher because it requires knowledge of file name suffixes used in Windows. (File name suffix = last few characters at the end of a file name after the period e.g. for image file `me.jpg` the file name suffix is `jpg`). By default the file name suffix is hidden from the user in Windows. The file name suffix used for Word 2007+ documents is 'docx'. The complete filename and the suffix is needed to open a Word document via `Documents.Open`

Step 17: Delete the instruction from Step 15: `Documents.Open (path & "\" & documentName)`. Retype on this line the following new instruction (that includes the filename suffix): `Documents.Open (path & "\" & documentName & ".docx")`

Step 18: Run the program, click on the 'Play' icon. If you completed the steps correctly and what you entered for the document name exactly matches a document residing in the location where your VBA program is located then you should not see any error message. Instead you should note that Word has opened the specified document.

Submitting your work:

- The file must be electronically submitted using [D2L](#).
- You don't need any special naming conventions for this exercise. Keep in mind only the latest file is the one that will be marked, everything else will be ignored (because it is not fair to your marker to sort through multiple versions of your files). Whatever name you have for the latest make sure that it's what you truly want marked!
- D2L configuration for this course
 - Multiple submissions are allowed for this assignment: You can (and really should) submit work as many times as you wish before the due date. Due dates are strict, only what is in D2L by the deadline is what will be marked. **Other methods of verifying that your work was completed on time (e.g. checking timestamps, emailed files etc.) will NOT be accepted.**
- Do not use compression utilities (such as zip) or archiving utilities (such as tar) otherwise your submission may not be marked. The space savings in D2L is not worth the extra time required by the marker to process each submission.
- Make sure that you [[check the contents of your submitted files](#)] (e.g., is the file okay or was it corrupted, is it the correct version etc.). It's your responsibility to do this! (Make sure that you submit your assignment with enough time before it comes due for you to do a check).

Important points to keep in mind:

1. **Extensions** may be granted for reasonable cases by the course instructor with the receipt of the appropriate documentation (e.g., a completed [Statutory declaration form](#) that has been signed by appropriate Deponent, you can get an appointment via the [Office of the Registrar](#)). Typical examples of reasonable cases for an extension include: illness or a death

in the family. Cases where extensions will NOT be granted include situations that are typical of student life: having multiple due dates, work commitments etc. Teaching Assistants (the people working in the 203 lab room) will not be able to provide extension on their own and must receive written permission from the course instructor first. (Note: Forgetting to submit/not properly submitting your assignment or a component of your assignment in does not constitute a sufficient reason for special grading considerations).

2. **Collaboration:** [Assignments must reflect individual work](#), group work is not allowed in this class nor can you copy the work of others. To avoid problems students should not see each other's assignment solution.
3. **Execution:** the submission must work on the machines on campus Windows computers. (For the remote learning semester since access to the lab computers is more challenging: the requirement is that your document works on any Windows computer). It's up to you to test and check this is the case. Non-functional submissions will receive only partial credit (if any at all).
4. **Late submissions:**

Submission received:	On time	Hours late : >0 and <=24	Hours late: >24 and <=48	Hours late: >48 hours
Penalty:	None	-1.5 GPA	-3.0 GPA	No credit

Marking:

- If you have questions about your marking then the first person to talk to is your marker and that will be the person who teaches the tutorial in which you are officially registered. [\[Tutorial information\]](#) If you still have questions after this first step then feel free to contact your course instructor, just let me know that you talked to your TA first.

Marking feedback. This workbook exercise has far fewer features than an assignment so there is a need for a marking spreadsheet. If any feedback is needed then you can find it directly in the D2L Dropbox folder for this exercise along with the