Office/Excel Usage Tips
2nd Edition

These details are derived from extensive work with Microsoft Office, especially Office 2004 (Macintosh). Note that *document* means any type of Office file.

Contents

Office Usage: OS Sizing, Buttons, Version/Hardware

Excel Usage: Vocabulary, Unavailable Items, Borders, Hidden Cells, Used Cells, Cell Merges, Cell Formats, Data Formats, References, Calculations, Automatic Navigation, Copies and Moves, Searches, Protections, Multiple Sheets, Maximums, Macro Options, Hidden Books, File Sizes

Customization: Useful Steps, Toolbar Buttons, Key Shortcuts, Backup and Restore, User-defined Operations

Excel Sample Code

Office Usage

OS Sizing

- While all Office formats are compatible between Windows and Macintosh, there is a spacing issue, at the system level: Each Windows font is generally one point larger than the equivalent Macintosh font.

Buttons

- While custom control buttons can be set to any height, they have a minimum actual height.
  - Reducing the document zoom setting does not reduce this minimum height.

Version/Hardware

- Office 2004 on an Intel-based Macintosh requires conversion from PowerPC operations. In Word 2004, this often causes a multi-second delay for some actions (for examples, Next Page and Find). But, for some reason, moving the mouse anywhere on the screen causes an immediate resumption.

Excel Usage

Vocabulary

- Excel unit vocabulary used by Magnafy Software includes:
  - A *book* is called a Workbook by the environment.
  - Unless otherwise indicated, any *selection* is a single cell or a collection of cells.
    - To contrast, other common selections include a collection of sheets.

Unavailable Items

- Some menu items that pertain to the host are unavailable unless a book is active: Excel > Preferences... and Tools > Add-ins....

Mouse Clicks

- Option-clicking a range performs an insert.
  - This can be strange behavior when intending to expand a selection, with <main key>-clicking. (The main key is Command in Macintosh and Control in Windows.)
- Opposite-clicking ("right-clicking") the sheet tab scrolling buttons presents a list of sheets.

Borders

- Even though cell borders are between cells, each can possibly belong to only one of the cells. So, this effects relevant format copies, which is especially significant for cells that are used as a template. Which cell the border belongs to is evident when a format copy (or insert) is performed; and then it can be adjusted, for future copying.
- While row heights are calculated in *points*, which is a loose publishing term that Excel defines as 1/72 of an inch; column widths are calculated in "characters", which Excel defines as the width of a zero — of an unidentified font.

Hidden Cells

- When cells that are drag-selected (or shift-selected) have hidden rows or columns between them, formatting effects the hidden cells. This does not occur when the visible cells are selected separately.
  - Because the indicators for hidden cells are fairly subtle, only in the row or column headings, it can be easy to forget that the hidden cells are there.
- The contents of non-hidden cells can be made invisible by matching the text color to the

background color.

Used Cells

- Specifying any non-default characteristic to empty cells can mark those cells as used.
  - When any empty used cells are in the last row or column, it's not possible to insert a cell in that direction, because Excel will not delete the used cells (logically allowing them to roll off an end of the table).

Cell Merges

- Cell merging and unmerging operates on only a single contiguous (rectangular) selection.
- Merge Across on a whole-columns selection operates to only the last row that Excel has marked as used.

Cell Formats

- When the contents of a cell are too long for its size, they visually overflow into the next cell(s), unless the next cell has any contents.
  - This does not occur if the overflowing cell is formatted to wrap text.
  - Wrapping text expands the height of the cell to accommodate its contents, unless the height is set explicitly.
- Formatting *within* a cell (cell-part formatting) is possible, but only up to the first 1023 characters.

Data Formats

- Custom number formats, which have several combinations of options, are explained in the Help item "About custom number formats".
- Cells whose actual value begins with - (or + or =) must be preceded by an apostrophe or formatted as Text; otherwise, they are interpreted as a formula.
- Cells whose actual value begins with - (or + or =) do not allow cell-part formatting.
  - The leading character can be preceded with a space to enable the formatting.
- A formula can have spacing, including new line characters, but only after separator characters (most commonly, a comma or either parenthesis), not before.

References

- References to cells that get deleted actually change to just a reference error, so the formula must be written again — and it's just a guess as to what the references should be. Therefore, new cells must be referenced before the old ones are deleted.

    - A replace-all that results in reference errors causes an Open dialog for each error. This Excel-level loop becomes annoying very quickly, and there's no way to exit it, except to force quit.
        - Therefore, a safeguard is to save just before any replace-all operation.
- Usage of relative or absolute references effects only copying the formula. In normal usage (outside of copying the formula), each type has behavior that might be expected of the other:
    - Relative references that get moved change to continue to refer to the same cells.
    - Absolute references to cells that get moved change to continue to refer to the moved cells.
- For relative references, R1C1 strings (R[±r]C[±c]) are in reference to the cell that they occur in; A1 strings are always in reference to the sheet (logically, R0C0).
    - In other words, relative R1C1 references list the calculations, and relative A1 references list the results.
- The preference setting for reference format (style) is at the host level, but it's also stored in each book.
    - If no book is open, then the host format is changed when a book that has the other format is opened.
- Excel has inconsistent handling of file names that have brackets (which also surround a reference to a file); this causes reference problems. So, brackets must not be in file names.
- While specifying a formula, navigation within the field is by mouse click only. Any keys are applied to cell navigation.

Calculations

- Formulas don't allow functions on the results of calculations, so intermediate calculations must be stored.
    - While an "array formula" accomplishes this, it's behavior is difficult to interpret.
- Charts don't have calculation mechanisms, so calculations for charts must be stored in cells.
- Averaging the days per month, including all leap years, provides a reliable constant: 30.436875. This can be especially useful for chart lines.
    - Alternately, the average number of days per month for most sets of four years (99%) is: 30.4375. The next time that this won't be accurate is 2400.

Automatic Navigation

- The preference setting to move the selection after the Return key is pressed applies to only a single-cell selection. For a multi-cell selection, pressing the Return key always keeps the selection the same and moves the active cell to the next cell of the selection vertically.
    - Pressing the Tab key moves the active cell to the next cell of the selection

horizontally.
    - Shift-Return and Shift-Tab move in the reverse directions.

Copies and Moves

- Most desired "cut-and-paste" operations regard the content of the cells. Unfortunately, Cut also removes formatting from the origin cells. So, the best steps for a move are usually Copy, Paste, and Clear Contents (on the origin cells).
    - Cut and Paste are sufficient for whole rows or columns.
- After inserting copied cells, inserting them again requires Copy to be set on the copied cells again, even though the copy mode indicator (animated border) is still on them.
- Sort moves only the values, not the rows (or columns) as a whole; this corrupts references to the sorted cells.
- The option for a control to move with cells is convenient for formatting, but cell copies also copy the control, adding it to the destination. This can create several copies of the control.
    - While control positioning is nearly always with particular cells, control sizing is an appearance issue, nearly always independent of cells. So, the control option to move but not size with cells is more-often appropriate.
- A sheet-level copy is simple, but it has some often-unwanted effects:
    - A sheet-level copy doesn't copy cell contents beyond 255 characters. A data-level copy does.
    - A sheet-level copy includes column widths, hidden cells, view type, zoom amount, page setup items, and cell protection.
        - A sheet-level copy does not include row heights.
    - A sheet-level copy also includes the sheet-anchored code. This causes the Macros-enable dialog on open, even when the code is subsequently deleted.
        - A sheet-level copy also includes the sheet module name.

Searches

- The search (Find, Replace) is always in wildcard mode. * searches for any number of non-empty characters, and ? searches for a single non-empty character.
    - ~ negates the wildcard character; it indicates that the * or ? that follows it is text.

Protections

- Object protection is one level lower than the applicable name (Tools > Protection > Protect object...): cell-level protection — locking — is called sheet protection, and sheet-level protection — naming, order — is called workbook protection.
    - Sheet protection operations are available when only a single sheet is selected.

Multiple Sheets

- Nearly no cell-level actions apply to multiple sheets.
  - The only exception is Fill > Across Worksheets, which is a copy and paste action.
  - Also, the range selection of the active sheet is applied to any other sheet as it's added to the sheet selection.

Maximums

- Many maximum settings can be found in the Help item "Microsoft Excel ___ specifications and limitations".
  - A maximum of very common interest is row x column. This varies by file format, not Excel version. For xls series (non-XML-based) files , the maximum is 65,536 x 256; for xlsx (XML-based) files, the maximum is 1,048,576 x 16,384.
  - One maximum that isn't listed in the Help item is sheet name length, which is 31 characters.

Macro Options

- The key combination for a Macro can not be set in the Customize Keyboard dialog; it must be set in the Macro dialog Options... sub-dialog. Only a letter — either lower or higher case — can be specified; the modifier key (or combination) is constant.
  - On Windows, the modifier key is Control; On Macintosh, the modifier key combination is Command-Option.

Hidden Books

- A hidden book can not be closed by the user. It must be unhidden first, so that it can be active.

File Sizes

- Excel versions before 2007 for Windows and 2008 for Macintosh do not effectively release unused file space, creating file sizes that grow to many times larger than the books need.

Customization

These are just a few, significant, customizations. (For Windows, the Toolbar and Menu customizations are not valid for Office 2007 and later.)

Useful Steps

- In the Customize Toolbars/Menus window, Commands tab, the All Commands list is usually best to use — first, because it's alphabetical, and clicking in the list and then pressing a letter jumps in the list; and second, because it has some commands that aren't in any other list.
- Buttons are generally easier to display as just icons. This is specified as contextual-click Properties — View: Default Style.
- Button icons can be copied from one button to another or from a small list to a button. This is through the contextual-click Properties — icon drop-down.

Toolbar Buttons

- In the Standard toolbar:
  - For VB code development, the Visual Basic Editor button, and the ViewVBCode button in Word, in the first position simplify going back and forth between documents and code, because the buttons are then in the same place for both.
  - The Print... button, in place of the Print button, helps to ensure that the document is printed as desired, saving paper and time.
  - In Excel, the Paste Values button, in place of the Paste button, typically serves workflow better. Placing the Paste Formatting button next to it, and using both buttons, serves most of the remaining workflow. However, their icons are not very clear in this context; the Paste and Scrapbook icons, respectively, are more expected.
- In the Format toolbar:
  - The Cells... button in Excel, and the FormatBordersAndShading button in Word (in place of the Borders drop-down), are logical next to the Font Color and then Fill/Highlight buttons.
  - Some buttons are helpful partly because they show the state of the object that they pertain to:
    - The Justify button completes the alignment buttons, between the Align Left and Center buttons.
    - In Excel, for sheets that are to be protected, the Lock Cell button saves having to go to the Format Cells window, Protection tab, for each cell.
    - In Word, the ParaWidowOrphanControl, ParaKeepLinesTogether, and ParaKeepWithNext buttons are frequent formats, well-placed at the end of the toolbar.
      - These buttons need to be changed to just icons, but they don't have icons. Reasonable icons to copy are from OutlineMoveDown, OutlineMoveUp, and DecreaseParagraphSpacing, respectively.

Key Shortcuts

- For Excel, in combination with the environment's main key — Command in Macintosh and Control in Windows — the following keys parallel the various paste buttons:
  - -V, for Paste Values

- -Shift-V, for Paste Formatting
- -Option-V, for Paste Special...

- Also, for consistent workflow, including between Word and Excel, <main key>-Shift-C can be set to Copy.

Backup and Restore

- Just as all of the toolbar, menu, and key settings are stored in the Normal template for Word, they're stored in the Excel Toolbars (<version number>) settings file for Excel. While Word preferences are also stored in the Normal template, workspace settings are in the com.microsoft.Word.prefs.plist file, and both of these things for Excel are stored in the com.microsoft.Excel.prefs.plist file. Also, both hosts store the Help workspace in the com.microsoft.Office.prefs.plist file. Backing up all of these files can eliminate a lot of tedious resetting if there's ever a corruption.
  - These files are in differing folders, from each other and from version to version. Some of these folders are excluded from searches, so finding the file locations for a specific version requires some investigation.

User-defined Operations

- Sometimes, a combination of operations is needed on a regular basis. This can be very specific to each user. As opposed to having to rely on, and wait for, a package to include every possible command (causing the package to be very bloated), flexible commands for Excel can be fairly simply created in three general steps with ExcelExtension:
  - Select a single cell, row, or column.
  - Start Macro recording (Tools > Macro > Record New Macro — Store macro in: Personal Macro Workbook). Perform the combination. Press the Stop button.
  - If individual cells or multiple pages need to be operated on, then go to the recorded Macro (Tools > Macro > Macros... — select — Edit). Place the recorded code in one, or a combination, of the structures in the sample code section.
    - In the Visual Basic Environment, the sample code just needs a Reference to the ExcelExtension add-in (Tools > References...), to use its data items and functions. It's best to save the Personal Macro Workbook after this.
- Any Macros can be made into a toolbar button using the steps in the Useful Steps section.

Excel Sample Code

- setting all arrays to start at 1, unless otherwise defined:
  - This needs to be at the top of the code module (before any other lines of code).

```
Option Base 1
```

- range processing when every used cell in the range needs to be processed individually:
  - Note that the cell attributes can be used at the range level, in the previous sample, if all of the cells in the range are to be the same.
  - Also note that Cell.Text is the formatted content, while Cell.Value is the actual content. This is most significant for a date, which is stored as a single number.

```
Dim sValue, sFormula As String
Dim iColorIndex As Integer
For Each Cell In CellRange(Selection)  ‘ CellRange: eliminates unused cells
    ' . . .
    ' cell attribute examples
     With Cell
    .Value = Trim(.Text)  ‘ Trim: eliminates leading and trailing spaces
    .Value = "prefix" & .Text & "suffix"
    sFormula = .Formula  ' A1
    sFormula = .FormulaR1C1
    .Font.ColorIndex = xlColorIndexAutomatic  ' user default, usually black
    .Interior.ColorIndex = xlColorIndexNone  ' background: default, white
     End With
Next
```

- strong overall range processing:
  - The Select structure is the simplest version of entity-type handling. The specific situation might require more-intricate structuring.
  - The previous sample can be put in any Case Is section of the Select structure. `CellRange(Selection)` just needs to be changed to `CellRange(oRange_Segment)`.

```
Dim iLastSegment, iSegment As Integer
Dim oRange_Compound, oRange_Segment As Range
Set oRange_Compound = SimpleRange(Selection)  ‘ eliminates repeated cells
iLastSegment = oRange_Compound.Areas.Count
For iSegment = 1 To iLastSegment
    Set oRange_Segment = oRange_Compound.Areas(iSegment)
    Select Case ihSegmentType_(oRange_Segment)
     Case Is = ichSegmentCells
        ' . . .
     Case Is = ichSegmentRows
        ' . . .
     Case Is = ichSegmentCols
        ' . . .
    End Select
Next
```

- processing of multiple sheets:
  - Either, or a combination, of the other two samples can be put in the Worksheet section of the Select structure.
  - Chart sheets do not have all of the characteristics of standard Worksheet sheets with an embedded chart, so they require frequent separate code.
    - It’s simplest for all sheets to be Worksheet sheets (even when the only thing on the sheet is a chart).

```
Dim oSheets_Selected As Sheets
```

```
Dim oSheet_Current As Object
Dim iSheet As Integer
Set oSheets_Selected = ActiveSheet.SelectedSheets
For Each oSheet_Current In oSheets_Selected
    iSheet = ihSheet_(oSheet_Current)
    oSheet_Current.Activate
    Select Case TypeOf_(oSheet_Current)
     Case Is = "Worksheet"
        ' . . .
     Case Is = "Chart"  ' Chart sheet
        ' . . .
    End Select
Next
```