

printout

Keystone MacCentral Macintosh Users Group ❖ www.keystonemac.com

Keystone MacCentral October Meeting

Please see your membership email for the links to this month's Zoom meeting or email us at KeystoneMacCentral@mac.com.

During our program this month we plan to discuss

- iPhone iOS 17 top features
- Mac macOS 14 Sonoma top features
- Personal voice (How to add your voice to your computer.)



We have virtual meetings via Zoom on the third Tuesday of each month:

Emails will be sent out prior to each meeting. Follow the directions/invitation each month on our email – that is, just click on the link to join our meeting.

Contents

Keystone MacCentral October Meeting	1
Mozilla Says Modern Cars Are Data Collection Nightmares on Wheels	
By Adam Engst	3
Do You Use It? Proxy Icons Quietly Boost Productivity By Adam Engst	4 – 6
Discovering the Hidden Power of Google Sheets By Adam Engst	7 – 13

Keystone MacCentral is a not-for-profit group of Macintosh enthusiasts who generally meet the third Tuesday of every month to exchange information, participate in question-and-answer sessions, view product demonstrations, and obtain resource materials that will help them get the most out of their computer systems. Meetings are free and open to the public. **The Keystone MacCentral printout** is the official newsletter of Keystone MacCentral and an independent publication not affiliated or otherwise associated with or sponsored or sanctioned by any for-profit organization, including Apple Inc. Copyright © 2023, Keystone MacCentral, 310 Somerset Drive, Shiresmanstown, PA 17011.

Nonprofit user groups may reproduce articles from the Printout only if the copyright notice is included, the articles have not been edited, are clearly attributed to the original author and to the Keystone MacCentral Printout, and a copy of the publication is mailed to the editor of this newsletter.

The opinions, statements, positions, and views stated herein are those of the author(s) or publisher and are not intended to be the opinions, statements, positions, or views of Apple, Inc.

Throughout this publication, trademarked names are used. Rather than include a trademark symbol in every occurrence of a trademarked name, we are using the trademarked names only for editorial purposes and to the benefit of the trademark owner with no intent of trademark infringement.

Board of Directors

President

Linda J Cober

Recorder

Wendy Adams

Treasurer

Tim Sullivan

Program Director

Dennis McMahon

Membership Chair

Eric Adams

Correspondence Secretary

Sandra Cober

Newsletter Editor

Tim Sullivan

Industry Liaison

Eric Adams

Web Master

Tom Bank II



By Adam Engst

Mozilla Says Modern Cars Are Data Collection Nightmares on Wheels

Mozilla's [*Privacy Not Included](#) team has evaluated the privacy stances of 25 carmakers, introducing its [extensive report](#) with:

Car makers have been bragging about their cars being “computers on wheels” for years to promote their advanced features. However, the conversation about what driving a computer means for its occupants’ privacy hasn’t really caught up. While we worried that our doorbells and watches that connect to the internet might be spying on us, car brands quietly entered the data business by turning their vehicles into powerful data-gobbling machines. Machines that, because of their all those brag-worthy bells and whistles, have an unmatched power to watch, listen, and collect information about what you do and where you go in your car.

*All 25 car brands we researched earned our *Privacy Not Included warning label — making cars the official worst category of products for privacy that we have ever reviewed.*

These findings fall into the category of “I had no idea, but I guess I’m not surprised.” Mozilla concluded that car companies are terrible about privacy because they collect too much personal data, share or sell collected data, give drivers little to no control over their data, and don’t publish useful security details, such as whether all that data is encrypted at rest. Nor are they good at protecting what they collect—Honda, Mercedes-Benz, Nissan, Toyota, and Volkswagen have suffered breaches affecting millions of drivers.

What might that data include? [The full list](#) is far too long to transcribe here, but it’s notable that, for instance, [Nissan says it may collect](#):

Sensitive personal information, including driver’s license number, national or state identification number,

citizenship status, immigration status, race, national origin, religious or philosophical beliefs, sexual orientation, sexual activity, precise geolocation, health diagnosis data, and genetic information.

At least the source of those particular data types is “Direct contact with users”—your Nissan Leaf isn’t detecting backseat nookie. But between today’s sensor- and camera-laden vehicles and their accompanying apps, carmakers can Hoover up a vast amount of information about how you drive. Along with geolocation data, [Hyundai says it may collect](#):

driving data about the operation of a Vehicle, such as speed, acceleration and braking data; direction of travel; trip data (mileage, date, length, conditions); ignition events; steering events; cruise control data; seatbelt status; information about Vehicle incidents or events; other information about how you drive a Vehicle; as well as associated date/time stamps for such information.

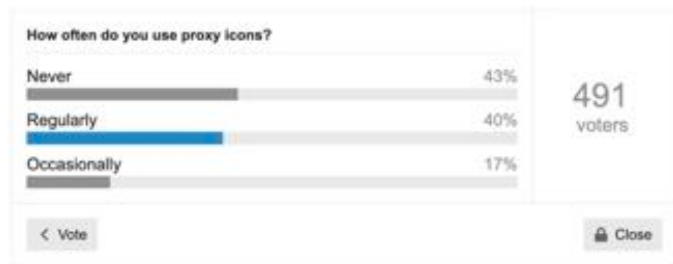
What remains unknown is just how real the privacy risks are. Just because carmakers craft their privacy policies to say they can collect data about your tooth enamel doesn’t mean they’re doing it or sharing the details with fly-by-night dentists. However, even if nothing is actually happening now, it’s still troubling that carmakers are giving themselves legal cover for whatever they decide to do in the future.

Mozilla’s reports on each carmaker offer suggestions for reducing the impact of this data collection, but there’s not much you can do and little difference between manufacturers. Perhaps signing [Mozilla’s petition](#) and helping to spread the word can embarrass some of these companies into doing better. 🗑️

By Adam Engst

Do You Use It? Proxy Icons Quietly Boost Productivity

Our first few Do You Use It? polls focused on prominent features that have starred in keynotes, been mentioned on macOS Web pages, and migrated between platforms. We're switching gears with the results from this next poll, which asked if you use proxy icons. They're one of my favorite macOS features, and I was fully expecting that even many TidBITS readers wouldn't use them, if only because of not knowing what they are or what they're good for. The poll results backed up my expectations, with only [40% of respondents using proxy icons regularly](#) and another 17% using them occasionally. Even the number of respondents was down from previous polls, with just under 500 votes. Allow me to persuade you to give proxy icons a try!



What Are Proxy Icons?

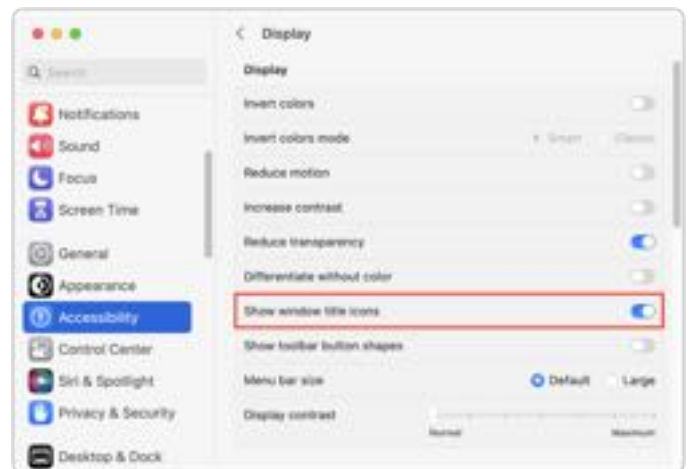
Proxy icons, or as Apple now calls them, “window title icons,” are tiny icons that sit next to the window title in Finder windows and windows from document-focused apps. Dragging the proxy icon to another app or location has the same effect as dragging a standard Finder icon.



Those who weren't already proxy icon users before macOS 11 Big Sur are forgiven for not noticing them. That's because Apple chose to hide the proxy icon in Big Sur by default, revealing it only when you moved your pointer over the window title or clicked in that area. It was undoubtedly an attempt

to make the Mac interface look simpler, but it came at the cost of discoverability, like so many of Apple's interface changes.

A `defaults write` command made it possible to reduce the length of time you had to hover over the window title before the proxy icon appeared, but only in macOS 12 Monterey did Apple acknowledge the overall wrong-headedness of this change and provide an option to keep the proxy icon visible at all times (see [“TipBITS: Always Show Window Proxy Icons,”](#) 26 March 2022). Apple inappropriately relegates this setting to System Settings/Preferences > Accessibility > Display > Show Window Title Icons. While there may be an accessibility argument for the visibility of proxy icons, they're a longstanding functional part of the macOS experience for everyone and shouldn't be hidden.



What Can You Do with Proxy Icons?

The simple answer to that question is anything you can do with standard icons in the Finder. They're called “proxy” icons for just that reason—they stand in for the real thing. What's interesting about proxy icons, however, is that their ready accessibility means you can use them in situations where working with a Finder icon wouldn't be

worth the effort. After all, once you ensure that proxy icons are always visible, they're right in front of you in window title bars, just waiting to be dragged.

Here are some real-world uses that TidBITS readers recommended in our poll:

- **Open files in non-default apps:** Many standard document types can be opened by multiple apps, so if you're looking at a JPEG file in Preview, for instance, but you want to open it in Affinity Photo, you can drag the JPEG's proxy icon to Affinity Photo's Dock icon.
- **Open a folder of files:** If you have a folder of text files you want to open in BBEdit or a collection of images to browse in GraphicConverter, you can drag a folder's proxy icon to the BBEdit or GraphicConverter Dock icon.
- **Move files to different folders:** If a file you're working on isn't located in the right folder in the Finder, you can move it by dragging the proxy icon to the destination folder. Imagine you've downloaded a bunch of stock image photos and need to distribute them to various project folders. You can open them all in Preview, open windows for the Finder destinations, and then drag each photo's proxy icon to the desired folder.
- **Navigate in Open dialogs:** Whenever you're looking at an Open dialog, you can drag a proxy icon into it to make it display the contents of the folder containing the dragged proxy icon. Suppose you are working on a document and want to open another document stored in the same folder. However, when you choose File > Open, the Open dialog shows a different folder. Rather than navigate up and down and all around the folder hierarchy, dragging the open file's proxy icon to the Open dialog takes you directly to the desired folder.
- **Navigate in Save dialogs:** Similarly, when looking at a Save dialog for a new document (or a Save As dialog), you can drag a folder or file proxy icon to the dialog to change the location to its enclosing folder. However, dragging a file proxy icon has the added benefit of pre-filling the name of the

dragged file for the new one. That's handy when the new file's name should be similar to the dragged one, perhaps varying only by a date or version number.

- **Enter paths in Terminal:** Typing long paths at the command line in Terminal is slow and error-prone, whether you're using `cd` to move into a deeply nested directory or specifying a destination for a command that takes a pathname. Drag any proxy icon into a Terminal window to insert a properly formatted path to that folder or file. The tip I gleaned from the discussion is that Command-dragging an icon or proxy icon into a Terminal window has the same effect as typing `cd`, dragging the icon in to get its path, and pressing Return.
- **Attach files to email messages or text chats:** It's common to find yourself working on a file you want to share with someone via email. Drag the file's proxy icon to an email message or your email client's Dock icon to attach it. This trick also works well with Messages and other chat apps. Make sure to save the file before dragging it so your changes are reflected in the shared file.
- **Combine text and word processing files:** Imagine that you want to combine a collection of text files. Rather than a convoluted dance of Select All > Copy > *switch documents* > Paste, you can open the destination document in one window and drag each file's proxy icon into the destination to put its text at the insertion point.
- **Shelve a file in Yoink:** The [Yoink](#) utility provides a "shelf" for storing files and other content you want to work with later. If you're working with a file that you want to access in the future, drag its proxy icon to Yoink to add it.
- **Preview text in Marked:** The [Marked](#) app is a previewer for Markdown and other plain text markup languages. Suppose you're writing in [iA Writer](#), which lets you focus on your text, but you want to see what your document will look like when formatted. Drag the proxy icon from iA Writer to Marked to open a preview.

- **Preview HTML documents:** When writing HTML, you will likely want to preview your work repeatedly. Although some text and HTML editors have their own previewing capabilities, it's important to check that your HTML code renders properly in at least Chrome, Firefox, and Safari. Drag the HTML document's proxy icon to a browser's Dock icon to do that.

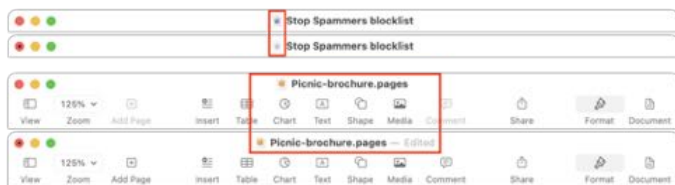
Proxy Icon Quirks and Caveats

The modifier keys you can use while dragging a file or folder in the Finder also work with proxy icons. So, when using a Finder location like the desktop as a destination:

- Command-drag a proxy icon to move the open file there without closing it, even when the destination is on another drive.
- Option-drag a proxy icon to make a new copy of it there.
- Command-Option-drag a proxy icon to create an alias there.

There is one action that always works with a standard Finder icon but may not with a proxy icon: dragging to the Trash. Try that with a proxy icon in Pages or TextEdit, and nothing will happen. However, it's not a universal prohibition—BBEdit allows it, and other apps may as well.

Eagle-eyed users may notice that document proxy icons are sometimes grayed out, as in BBEdition, and such icons may not even be draggable, as in GraphicConverter. That happens when the file has unsaved changes. Not all apps honor this historical interface convention (which is pretty subtle, I'll admit), and apps like Pages that use Apple's newer approach to title bar controls instead append "Edited" to the window title. Either way, save the file before dragging its proxy icon to ensure your changes are incorporated.

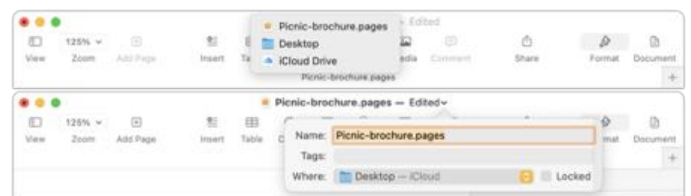


Note the difference in proxy icons and window titles between saved (top in each pair) and unsaved (bottom in each pair) documents

Several people noted that proxy icons don't work in Excel. However, as the discussion progressed, it turned out that the problem was not with Excel but with all Office documents stored in OneDrive. It's unclear why Microsoft disables proxy icons for Office documents in OneDrive, given that they work fine for other types of documents in OneDrive and for Office documents in Dropbox, Google Drive, and iCloud Drive.

Mail used to sport proxy icons that you could drag to the Finder to create an alias to that message. Although Mail messages no longer have proxy icons, you can drag from anywhere in the header of a message to create a mailbox (.eml) file of the message.

Finally, some people spoke up in favor of the directory menu that pops up when you Command-click the window title. This menu is associated with the proxy icon only by proximity, but it's helpful for checking the enclosing folder to see if the file is stored in the proper place or is the correct copy. Similarly, in apps like Pages that adopt Apple's new title bar interface, mousing over the window title reveals a down-pointing arrow that, when clicked, brings up a popover with fields for renaming the file, managing tags, and checking the folder location.



If you haven't previously used proxy icons, I hope this look at one of the Mac's unsung features has given you ideas for speeding up your workflows. Those of us who are accustomed to employing proxy icons in our everyday Mac usage would be distraught to lose such a productivity-enhancing capability. And if you're already a fan of proxy icons, I hope you learned something new, like the trick of Command-dragging an icon into a Terminal window to change to that directory. 🗑️

By Adam Engst

Discovering the Hidden Power of Google Sheets

I've never been a big spreadsheet guy. As a writer, my life has been spent in word processors and text editors. Sure, I know how to use spreadsheets, and I'm a great believer in setting up a simple spreadsheet for calculations that might need to be tweaked or updated. Until recently, I could go months without opening a spreadsheet. Part of my disregard has been due to people in my life using spreadsheets as layout programs because they lacked the tools or skill to produce the same document in a page layout app. That's seductive—you can put text anywhere on the "page" by filling in the right cells—but for anything except one-offs, hammering a nail into the page layout wall with Excel as your screwdriver will eventually result in tears.

In the last year or two, though, I've changed my spots. It turns out that I just needed the right project to unleash my inner spreadsheet maven. I hope the highly specific story of my evolution from a guy who seldom strayed beyond [SUM](#) and [AVERAGE](#) to someone who has discovered the wonders of multi-sheet lookups will give you ideas about how you too can make better use of the astonishing tool that the modern spreadsheet has become.

Speaking of tools, although I suspect that what I've done is possible in Apple's Numbers and Microsoft Excel, I chose to work in [Google Sheets](#). Not because it's necessarily the most powerful of spreadsheets—I have no idea how it stacks up—but because having my spreadsheets always accessible to multiple people with minimal fuss is the killer feature. It's the same reason I do most of my writing in Google Docs these days. It's also helpful to be able to pull up any of my online documents on any of my other devices.

Architecting the Budget Tool

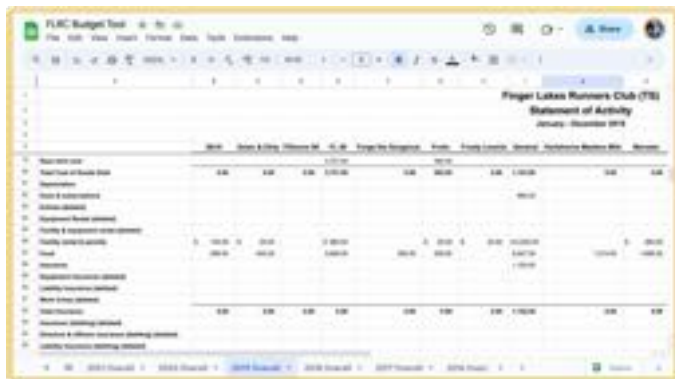
In 2020, I became president of the Finger Lakes Runners Club (FLRC), only to have the pandemic force us to cancel all our races after February, which thoroughly confused the club's financial workings. At the end of 2020, our then-treasurer had to create a 2021 budget based on pure guesswork and update it throughout the year, as some races came back and others didn't. That treasurer stepped down during the year, and I convinced my friend Charlie to take the role. Our first task was to build a budget for 2022. Research revealed that the previous treasurer's approach had been a simple spreadsheet that did little more than add manually entered numbers. I hate retyping numbers—that's what computers are for!—so I started to brainstorm better approaches.

My first question revolved around the purpose of the budget. FLRC puts on about 20 races per year, and we expect most of them to make a small profit or at least break even, with two large races providing the bulk of the club's income for operations, philanthropy, and savings. I realized we didn't care exactly how much race directors spent as long as they stayed within ranges that would lead to a workable outcome for the club as a whole. Problems had occasionally cropped up in the past when a race director came up with an ill-considered plan that resulted in considerable losses. I wanted our budget to provide guardrails that race directors could stay between when contemplating an unusual expense.

My initial thinking was that QuickBooks Online, which is what the club uses for accounting, might provide budgeting capabilities. It does, but it was a non-starter for two reasons. First, its interface doesn't provide significant historical information

we could use to inform our budget estimates for the next year. Second, although we could export the budget in various ways, we would have to do that regularly so race directors could check how their race was doing as we recorded income and expenses. That's when I focused on Google Sheets—I wanted any race director to be able to see the budget for their race and its up-to-date numbers with a click of a link.

The key report to see a race's current numbers is the statement of activity in QuickBooks Online. Each race is a class that turns into a column when exported, and each row is an account (registrations, awards, timing, etc.) from our chart of accounts. It's nice, tabular data, but it's difficult to read in any format because the number of races and accounts force scrolling vertically and horizontally. Also, I wanted each race to appear in isolation—asking race directors to parse their race data from a massive PDF or spreadsheet wouldn't fly.



That's when I had my next epiphany. I may not be a spreadsheet guy or a programmer, but I understand abstraction. When programming, you use variables to avoid hard-coding data—`myVariable` changes based on whatever you put into it. In spreadsheets, you use cell references to operate on whatever the contents of cell C32 or column M happen to be. Even better, a single spreadsheet can have multiple internal sheets, and you can reference cell ranges in one sheet from another.

Suddenly, I could visualize the architecture of the spreadsheet I needed to build. I would have a statement of activity in one sheet—exported directly from QuickBooks Online—and then I'd

have a separate sheet for each race that would extract its data from the statement of activity. Even better, I'd import a statement of activity for every year back to 2016 so each race could display columns of historical data for those years as well. That historical data would inform the actual budgeting process, enabling Charlie and me to look at previous years (and an overall average) as we attempted to predict the next year's revenues and expenses.



Building the Magic Formula

The tricky part was implementing my vision. I knew it was possible to look up data from one sheet and display it in another, but I didn't know how. I quickly came up with a hard-coded proof-of-concept, which was good, because it took me a lot longer to figure out how to abstract it. Eventually, with many Web searches and poring over examples, plus trips to [Help > Function List](#) to see what Google Sheets is capable of, I ended up with this formula.

```
=IFERROR(HLOOKUP($A$2,INDIRECT("'"&E$4&"Overall!$A$5:$X$250"), Utility!$A5, FALSE))
```

Explaining it will take some time, but it's where the magic happens:

- **IFERROR:** Wrapping the entire formula in an **IFERROR** function like this shows a blank instead of an error like `#N/A` when the rest of the formula fails, usually because there's no data in one of the long-ago years.
- **HLOOKUP:** The **HLOOKUP** function does the bulk of the work, searching across a range for a key and returning the value of the cell in the column that

it finds. Configuring it properly took the most time.

- **\$A\$2**: The first task was to specify the search key for **HLOOKUP**. Since my goal was to look up the information for a particular race, I initially hard-coded the race name in this spot so **HLOOKUP** could find it in the annual statements of activity. That worked but would have forced me to customize the formula on each race sheet. Doable, but fussy and brittle. Instead, I created a variable of sorts by putting the race name in cell **A2** and using the absolute reference **\$A\$2** to look it up. (\$) characters ensure that the specified column and row don't change when the formula is copied into other cells.) When I wanted to create a new sheet for a race, all I had to do was put the new race name in that cell to enable the formula to look up the correct data.

- **INDIRECT**: The next information that **HLOOKUP** needed was the range to search. I could hard-code something like **2023 Overall!A5:X250** to reference the necessary data in this year's statement of activity, but abstracting that to any given year was tough. (I had to start at **A5** rather than **A1** because the first four rows come in as merged cells from the QuickBooks Online export, and starting at **A1** fails.) The **INDIRECT** function turned out to be the key because it returns a cell reference specified by a string. Building its string took me a long time due to the exactitude required.

- **"&E\$4&" Overall'!\$A\$5:\$X\$250"**: Each historical column needed to look up data from the associated year's sheet, but I wanted to avoid hard-coding the years. My solution was to use **"&E\$4&" Overall'** to pull the year from cell **E4** and mash it onto the word **Overall** to specify the title of the correct sheet. Because **E** was relative (not prefixed with **\$**), copying the formula to other columns continued to look up the year from row **4** (absolute, because it was prefixed with **\$**). Specifying the absolute range of **!\$A\$5:\$X\$250** was easy. The hardest part was figuring out the combination of double and

single hash marks and putting the **&** (string concatenation operator) in the right place. It probably took me an hour of trial and error; I understand why it works now, but I didn't until it worked. I'm a shameless stealer of examples, and one of them must have convinced me it was the right approach.

- **Utility!\$A5**: What's this Utility sheet? It contains a single column of numbers, with **1** in **A1**, **2** in **A2**, and so on, down to **250** in **A250**. It exists because I needed some way to give **HLOOKUP** the index, the row number of the value to be returned, and I couldn't figure out any other way to increment a number without referring to the contents of a cell. Confusingly, the index has to start from row **5** because my range to search is **A5:X250**. So the index for row **5** needs to be **1**, **6** has to be **2**, and so on. This was mind-bending, and, honestly, I only figured it out through trial and error by slotting in specific index numbers until the correct results came back. The **\$** before the **A** ensures that it's absolute to stay focused on that column, whereas the lack of the dollar sign before **5** makes it relative so it can increment as need be.

- **FALSE**: Finally, the **FALSE** option for **HLOOKUP** specifies that the row to be searched isn't sorted. I'm unsure what that would entail, but since my data isn't sorted, I just went with it.

The beauty of this formula is that it pulls in all the data I need in one fell swoop. I first create a sheet with the race name in **A2** and columns for **2016** through **2023**. Then I put the formula in the first cell under **2023**, copy it to the right to fill the first row for each year, and finally copy it down to fill the rest of the rows.

(For those who don't do much with spreadsheets, there's always a trick for copying a formula—in Google Sheets and Excel, you drag the handle in the lower-right corner of the cell; in Numbers, you drag a yellow handle on the desired side of the cell. That copies the formula to the destination cells, appropriately incrementing relative row and column references. If you want to put a formula

from one cell into another without incrementing cell references, you must either prefix both the row and column references with \$ or copy and paste the text of the formula rather than the cell that contains it.)

Filtering and Formatting the Data

If you looked closely at the statement of activity screenshot earlier in the article, you might have noticed that many of the accounts are suffixed with `(deleted)` or even `(deleting)(deleted)`. Those are an artifact of a quirk of QuickBooks Online that required a lot of fuss. Here's why.

Our chart of accounts hasn't remained static over time, with various treasurers and bookkeepers adding and removing accounts, some of which haven't been used since 2016, the first year I pull in for the Budget Tool. But if you think about my magic formula, it works by looking up cell references. I need the Food account to be in row 87 in the Overall sheets for each year. A standard QuickBooks Online statement of activity shows only rows with activity, but if I create such a report, there's no predicting which row Food will end up in for any given year.

Luckily, there is an option in QuickBooks Online to show all accounts in reports, and when I use that, the statement of activity report puts each category in the same row every time. That's possible because QuickBooks Online never *deletes* any account; it only marks it as deleted and hides it from view.

However, having all these unnecessary rows littering up my raw data presents other problems. Most of our races have activity in 10–20 accounts at most, but the statements of activity have a whopping 204 rows. That's a serious presentation problem.

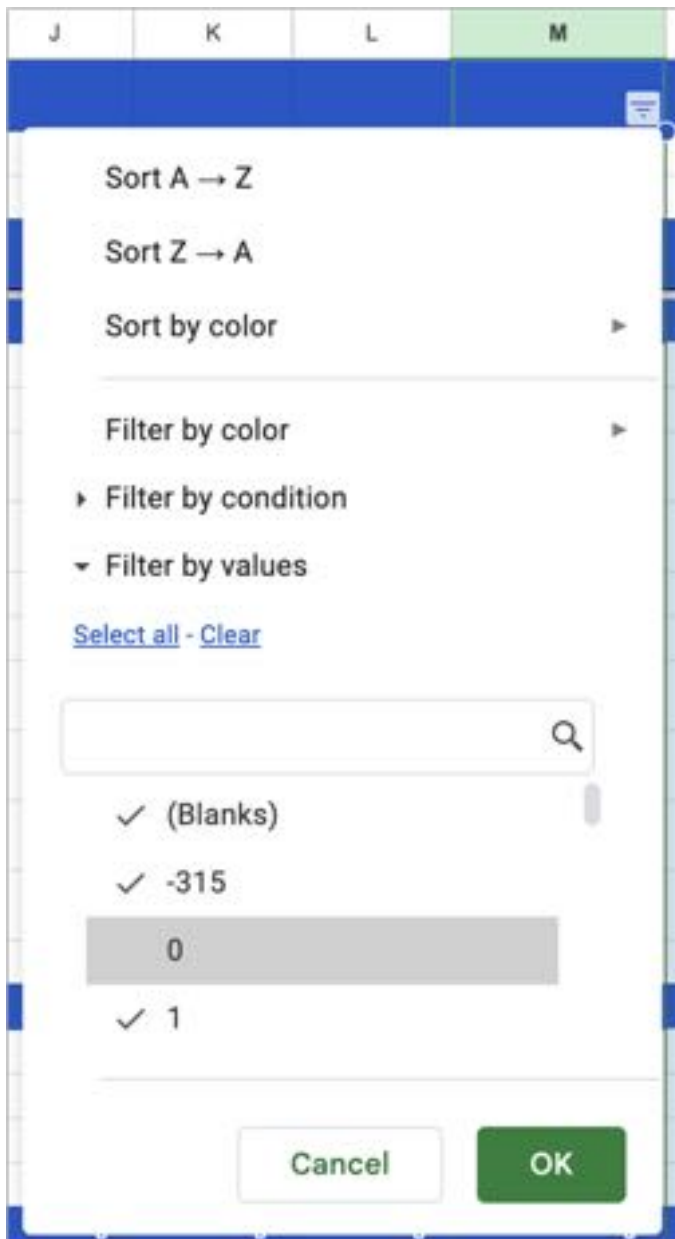
Why so many accounts? Earlier this year, we realized we could save over \$800 per year by switching from a regular QuickBooks Online account to one through [TechSoup](#), which provides discounted technology for nonprofits. I vaguely knew about TechSoup because Jeff Porten wrote about it for TidBITS years ago (see "[TechSoup: Get](#)

[Deep Discounts on Technology for Your Nonprofit](#)," 18 October 2018), but I didn't realize it had a \$75-per-year QuickBooks Online subscription on offer. Converting our existing QuickBooks Online subscription to the TechSoup account required setting up a new "company," transferring our data, and then canceling the previous subscription. It was complicated, stressful, and required help from Intuit support to get the promised refunds, but it worked in the end.

Unfortunately, setting up a new QuickBooks Online company gives you a boilerplate chart of accounts even when you're immediately importing data from a backup. All those boilerplate accounts were then marked as deleted, radically increasing the number of deleted accounts cluttering my statement of activity reports. I wanted those accounts to disappear entirely since they never had any activity at all, but the closest I could get to removing them for good was merging them. In theory, I could have merged all of them into a single account that would be marked as deleted. In reality, it was a clumsy, error-prone process that worked only sporadically, so I gave up and decided to suffer with ugly statements of activity.

In an earlier iteration of the Budget Tool, I had manually gone through each race and hidden the rows that didn't contain relevant data, but that was tedious and caused confusion once when a race generated activity in a previously unused account that I'd hidden. When I was refactoring the Budget Tool to handle all those extra deleted rows from the QuickBooks Online boilerplate chart of accounts, I thought of a better approach.

Google Sheets has filtering capabilities that can hide or show rows. So I created a Total column that added up the numbers from each of the years from 2016 through 2023. I don't care about the total itself, but then I created a filter (choose Data > Create Filter) on that column that showed only those rows that weren't zero. Poof! All the unnecessary data immediately disappeared.

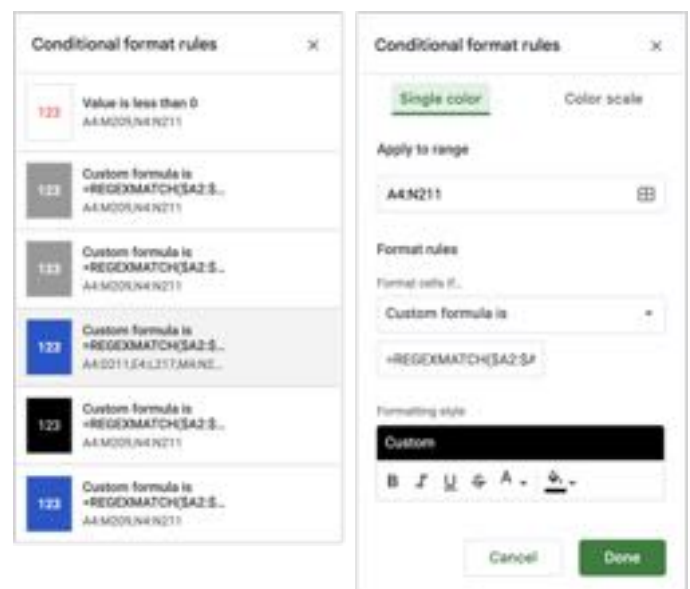


I did end up with one wart. My sheets all have header rows for Revenues and Expenditures. The Revenues header appears above all the data and is thus safe from the filter, but since Expenditures is in the middle of the sheet and has no associated data, the filter hides it. To keep it around, I put a 1 in one of its columns to keep it visible. Perhaps there's a better solution.

You'll also notice that, along with the blue Revenues and Expenditures header rows, I have gray Total Revenues and Total Expenditures rows, and a black Net Revenue row. For ease of reading the sheet, I want those formatted differently.

Initially, I had formatted those manually for each sheet, which is fussy but not as complicated as it might seem, given that Google Sheets has an Edit > Paste Special > Format Only command. I could copy any cell that was formatted the way I wanted and then go through selecting entire rows and pressing Command-Option-V to paste just the formatting of my copied cell.

Nevertheless, in my quest for abstraction, I realized there was a better way. Google Sheets also has a conditional formatting capability, which lets you format a range of cells based on a formula. I've long used it to make negative numbers red, but I realized I could also make rules to format my special rows. One tip: choosing Format > Conditional Formatting displays only the rules that apply to the currently selected cells. So if you choose that command and nothing appears, select a cell that should be in the affected range.



The trick in getting this to work was in the formula: `=REGEXMATCH($A2:$A210, "Net Revenue")`. `REGEXMATCH` is a regular expression (grep) function that looks for text within a range. As with so many other parts of this project, it took some experimentation to get it working. I'm not sure why some of the rules ended up with odd ranges—I changed the one shown to the simpler `A4:N211` with no problem.

Once I had done this, I created several additional columns that Charlie and I could use for budget calculations—the ones in orange in the screenshot. They employ simple calculations to sum the revenues and expenditures and then calculate the net revenue. The notes column helps us remember why we estimated a particular number. Occasionally, I've also made a scratchpad column to preview what different estimates might do to a race's bottom line—it's just another orange column with the same calculations as the budgets.

I've been trying to abstract things like hidden rows and formatting because we have 22 sheets between all the races, **General**, and **Total**. If I decide to make a change, I must make it 22 times, which is tedious. In my last major revision, when I abstracted the hidden rows and formatting, I also had to recreate each sheet. That was easy enough; I duplicated one, changed the display name and race name in **A2**, and refreshed the filter. But having done that, I'm back to needing to make any further changes 22 times. For instance, I'll need to add a **2024** column for each sheet next year. In an ideal world, abstraction capabilities would be easily applied to multiple sheets within a spreadsheet. Perhaps that's possible, and I just haven't discovered it yet!

Regular Usage

I want to leave you with a few notes about how we use our Budget Tool regularly.

First, after our bookkeeper has entered everything for a month into QuickBooks Online, we run a statement of activity report that includes all rows and export it to Excel format. Although it's possible to import that into Google Sheets, overwriting the contents of a particular sheet, I find it easier to open the export in Excel, copy the data, switch to Google Sheets, and paste it over the associated year's sheet.

(Is there an online integration that would move the data between QuickBooks Online and Google Sheets automatically? The closest I've found is [Skyvia](#), but Intuit doesn't provide the necessary API access to reports.)

As soon as Google Sheets has the new data, everything—and I mean everything—instantly updates, so we can spin through the sheets and see how each race is doing. If some numbers differ wildly from our budget estimates, we try to figure out why. (It's usually just a categorization error.)

The only hitch we've encountered is a race getting data in a previously unused account; I have to refresh the filter by selecting all and then deselecting zero again. That's happened only once so far.

When it comes to sharing the data with our race directors, there's one final Google Sheets trick I wanted to share. If you click the blue Share button in the upper-right corner of Google Sheets, the link provided goes to the first sheet, **Total** in our case. That's not bad, but linking directly to a race sheet would be better. To do that, I Control-click cell **A1** for a particular sheet and choose View More Cell Actions > Get Link To This Cell at the bottom. That copies an appropriate custom link to the clipboard for sharing elsewhere.

You might wonder if I'm worried someone will inadvertently or maliciously delete or edit my carefully constructed sheets and formulas. My experience is that most people are sufficiently intimidated by collaborative documents that they won't touch anything, but there are three ways of protecting shared spreadsheets like this:

- **Share access:** I chose to use the blunt force approach and limit the access for everyone but me and Charlie to Commenter. If people want to make changes, they can, but I get to approve or reject them. That hasn't come up yet.
- **Protect sheets and ranges:** For significantly more granular control, the Data > Protect Sheets and Ranges options let you specify exactly who can do what with specific portions of the spreadsheet. I haven't needed such fine-grained control yet.
- **Previous versions:** Google Sheets maintains a complete version history of all changes,

making it easy to revert the entire spreadsheet to a previous version if necessary. Even if someone did make undesirable changes, it's simple to recover. The only problem is if you don't notice immediately and many legitimate changes are made afterward.

I realize everything in this story is highly specific to my project, but I hope you take away the idea that

modern spreadsheets can do much more than just perform calculations—and perhaps some ideas for how to use similar techniques in your own work. Spreadsheets aren't far off from full-fledged programming environments, and in fact, they often support macros or scripts that can go beyond what's possible within the confines of rows and columns. But that's another journey for another day. 🍷

