

## Bank Account Tracking

### Executive Summary

My wife and I are very careful with our money, but our credit union's website does not have the best tools to help us see what is happening with our money in a graphical way. The goal of this project was to create a tool that I could use that would log in to my bank account, extract a set of data, write it to a worksheet and then create a chart that represents that data. This was the bulk of the project and I found ways to make my code do all of those things.

In addition, I wanted a way to keep track of receipts I get by category, so I made a user forms to enter in receipts as well as edit them if I've made a mistake. The form gathers the date, store, amount, and category of each receipt, making it easy to keep track of where our money is going. Rather than manually enter the categories, they are validated by checkboxes so the user cannot enter in a category I'm not tracking.

Finally I created a new ribbon titled 'Financials' that runs all of this code. The result is a powerful tool that's both easy to use and looks clean, and I'm very excited about it.

### Implementation

#### Bank Account

The first step I had to take with my project was just opening up the website. This was simple enough, but after opening up the website, being my bank account, it of course was password protected. This turned out to be harder than we had gone over in class. On my credit union's website, the username and password are entered on separate pages, unlike the Learning Suite example we covered in class. I was able to successfully submit the form with my username, but when I attempted the same process for the password, it simply refreshed the page without moving forward. This was a roadblock, but thanks to some Google searches, I found a new way of clicking through a website by using the document("SubmitNext").click syntax. By clicking the button titled "Submit Next" I successfully navigated through the login process. Naturally, however, I was very wary of hard coding my login credentials into the procedure, so I created a userform that captured that information, and simply used the text entered by the user each time the procedure is run to log in to the account. I did not realize at first how easy it was to reference data from the userform, and originally created a hidden sheet, writing the entered values on the userform to that sheet, then using that sheet's data to log in to my account. I didn't like that approach, so after some looking around I found how easy it was to access the text directly from the userform. One issue I don't know how to solve or that I need to solve is that this procedure will only work on this device, or at least a device that has been set to 'trusted' by either my wife or me. If the device is not trusted, then after submitting the password it will ask an additional security question that my code is not prepared to handle.

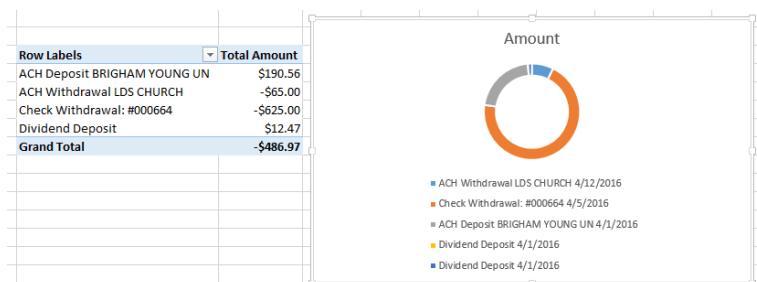
The next step I had to take was to enter in the search criteria for the time period I wanted. This is probably the most inefficient part of my code, and I would take a different approach in the future. I realized that any search I did with a space in it, such as "this month", put a "%20" in place of the space between words. I realize now that a simple replace function would have worked, but I took a different approach. I created a frame on the userform with option buttons for each month, as well as a few other options, like "this week" and "last quarter." Using a select case statement, I generated the correct

syntax for the url based on the time period the user chose, and subsequently opened the new url with the time period inserted. The following is a screenshot of the userform I created.

After finding the page with the time period I wanted, the next step was to get the actual data and assign that data to variables I could use to write to the worksheet. The data I ended up pulling was the title of the transaction, its date, and the total amount. I found in the html source of the page that before each of these were the titles “Search on this” and either “Description,” “Date,” or “Amount.” Once I found these titles it was relatively easy to write a loop using the a.moveto and a.gettext functions of the agent. I recorded the data to variables, created a new worksheet with the name of the search criteria used, and wrote the data to that worksheet, creating an easy-to-read table of the data I wanted. The following picture is an example of data pulled for the month of April.

Description	Date	Amount
ACH Withdrawal LDS CHURCH	4/12/2016	(\$65.00)
Check Withdrawal: #000664	4/5/2016	(\$625.00)
ACH Deposit BRIGHAM YOUNG UN	4/1/2016	\$190.56
Dividend Deposit	4/1/2016	\$0.93
Dividend Deposit	4/1/2016	\$11.54

Following the creating of a new sheet with this table, I wanted to create a pivot chart that allows me to look at the data in different ways. This was straightforward and followed well with what we had done in class. I created the pivot table and put the description as the row and the amount as the value. In the future if I decide I want more data, it will be easy to add that into the pivot table. With the pivot table I created a pivot chart as well. For this process I recorded myself creating the pivot chart, then edited the code to make it more robust and accurately create the chart for each time this procedure is run. The following screenshot is the pivot table and chart I created. I don’t believe this chart is not the best one, but I’m still trying to find one that I like better.

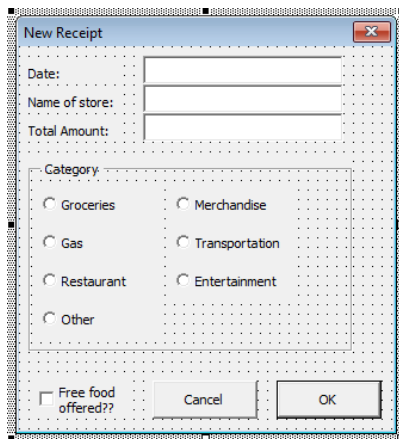


## Receipts

In addition to this tool that I really like, I wanted a way to capture individual transactions that I make. The majority of purchases that I make are on my American Express credit card, and on their website I can get details on each transaction I've made with that card. There are still a significant number of transactions that I make on either visa cards, checks, or with cash, that I'd like to be able to record as well. In order to accomplish this goal, I created a userform that I can open and enter in the details of a receipt.

This userform currently captures the date of the transaction, the total amount spent, the store where the transaction was made, the category, and whether or not free food was offered for filling out a survey. There are seven possible categories the user can pick from, with each category being an option button the user can select. The categories are groceries, gas, restaurant, merchandise, transportation, entertainment, or other.

I also created a userform that the user can use to edit the information on a receipt. Since each receipt recorded by the previous form is recorded onto a single row on the sheet "Receipts," this form uses the row of the active cell to fill in the data on the form when it is initialized. The user can then edit any information on the form, and upon hitting the "Edit" button that information will write over what was there previously. I will provide a picture of the form here. There are only slight difference between the forms. One difference is the "OK" button. When creating a new form, it reads "OK," and when editing an existing receipt, it reads "Edit." The other difference is that when editing a receipt, the default button is cancel instead of "OK," helping the user to avoid accidentally writing over information.



The last step I took was to create a new tab on the workbook that will run all of these procedures. The tab is titled "Financials" and has two groups: "Bank Account" and "Receipts." Clicking any of the buttons will immediately pull up the corresponding userform and subsequently run the procedure called by that form.

## Learning

I learned a great deal by doing this project and I'm very excited to have this tool working. I learned a lot about how to navigate websites with VBA. In class we used the Learning Suite example, but this site was considerably more complex than Learning Suite and required more work to get to the data I wanted. I first got stuck because when I tried submitting the form that contained the password to get to the next

page, all it did was refresh the page rather than move on. After some time I finally got past that by actually clicking the submit next button.

As I mentioned above, the search criteria is a field I'll probably write another way as I use this tool. Originally I wanted it to be a text box that I could enter in and use in the url. The problem I found was that the url with the search criteria contained text in the middle of the it that I had not entered. Rather than face this problem directly, I found a workaround in option buttons. This really restricts the number of searches I can do, but it was sufficient to meet my basic needs, and as I couldn't think of another solution, that's what I ended up using. The next approach I'll try is to use a replace function, replacing spaces with "%20" as I think that's the major change I need to deal with. I'll give that a try and hopefully it makes my tool much more robust.

For the receipts, I decided to keep it simple and not track each individual line item on receipts, but instead just record the total amount of the receipt. I don't anticipate that I would actually analyze each item. I currently keep most of the receipts I get from grocery stores with the thought that I will see how prices are changing, but I very rarely if ever look at those receipts again. Because of this, I decided that was functionality that I did not need.

I did not have any help in creating this tool besides my trusty friend Google. I really am very excited to start using this to be able to watch my money in a very easy way. I also love the clean look of adding a new tab. I've created buttons for all my macros in the past, but adding a new tab was surprisingly simple to accomplish and looks much better.