



VBA Final Project

Kyle Musgrove

Section 2

Executive Summary

Mint.com is a free personal finance service that is provided by Intuit. It easily syncs all your various financial accounts in one place. You can also setup budgets and Mint.com will automatically categorize your purchases in the various budgeted categories. Mint.com has been on Time Magazines 50 top websites for the last three years in a row.

The issue that I identified with the website was the lack of detailed spending information and easy ways to calculate personal financial ratios. With my project I set out to create a MS Excel spreadsheet to automatically retrieve my information from Mint.com. With this information I wanted to better know how much I had spent and how much left I had to spend in the month. Also, I wanted to get up to date key personal financial ratios. Previously I had to copy and paste the information I needed into MS Excel to be able to calculate the ratios.

To be able to more easily see the how much I had spent and how much I had left was to convert the numbers into percentages. The first percentage that I calculated was my percentage spent of the monthly allowance. This indicates my monthly progress and how well I am performing. The second percentage is how much I have spent on a weekly basis. This will help me see my weekly performance and improve my spending habits.

The key personal financial ratios that I want to track are: Housing Payment Ratio, Liquidity Ratio, Solvency Ratio, and Debt to Income Ratio. As a college student that is close to graduation I am becoming more concerned with these financial ratios because I am going to start saving for a home. By tracking these ratios I can easily see my true financial health, and strive to improve them.

The program that I built will automatically go out to Mint.com and import all of my budget categories and the various purchases that I have made in the corresponding categories. The program will then calculate the percentage spent in each category on a monthly and weekly basis. The program will then calculate the key personal financial ratios, mentioned above. The program also gives the user the ability to send an e-mail or text to a significant other to invite them to a personal financial meeting to review the current progress.

Implementation Documentation

- The first task that I performed was to create a template sheet where the data could be pasted and presented to the user.

Monthly Spending						Total Money			Key Ratios	
Category:	Spent	Budgeted	% of Monthly Total	% of Weekly Total	Total Cash	Total Debt	Total Assets	Housing Payment Ratio		
Income: Bonus	\$0 of	\$50	0%	0%	\$21,848.09	\$785.66	\$51,665.06	Liquidity Ratio	21%	
Income: Interest Income	\$0 of	\$16	0%	0%				Solvency Ratio	0%	
Income: Paycheck	\$664 of	\$2,650	25%	100%				Debt to Income	6576%	
Auto & Transport: Auto Insurance	\$0 of	\$92	0%	0%					30%	
Auto & Transport: Gas & Fuel	\$86 of	\$330	26%	104%						
Auto & Transport: Service & Parts	\$0 of	\$55	0%	0%						
Bills & Utilities: Utilities	\$65 of	\$310	21%	84%						
Entertainment: Movies & DVDs	\$1 of	\$40	3%	10%						
Entertainment: Music	\$0 of	\$1	0%	0%						
Food & Dining: Fast Food	\$33 of	\$65	51%	203%						
Food & Dining: Groceries	\$34 of	\$150	23%	91%						
Food & Dining: Restaurants	\$0 of	\$50	0%	0%						
Gifts & Donations: Charity	\$0 of	\$850	0%	0%						
Health & Fitness: Doctor	\$0 of	\$0	0%	0%						
Home: Mortgage & Rent	\$0 of	\$550	0%	0%						
Shopping: Clothing	\$0 of	\$100	0%	0%						

-The next task was to go out to Mint.com's website to and get the information needed. First I had to gather the Mint.com username and password in order to gain access to their website. I used a userform to gather this information from the user.

The image shows a screenshot of a web-based login form. The form has a title bar that says "login Credentials" with a close button (X) in the top right corner. Below the title bar, there are two input fields: "Mint ID:" and "Password:". The "Mint ID:" field is a single-line text box, and the "Password:" field is a single-line text box. Below these fields is a "Login" button. A mouse cursor is pointing at the "login Credentials" title bar.

-After gaining access to the website I needed to navigate to two parts of the website in order to obtain the information that I needed. After logging in, I needed to import the information on the first landing page and information under the budgets tab. I had my program create two new sheets in order to copy the information into MS Excel. In order to obtain the exact information I needed I had to make the program loop through the raw data and find the budget titles and then find the financial information that related to the

respective budget. To accomplish this task I used do loops and copied and pasted the information to the template sheet. Below is a shot of the raw data that is transferred to the template sheet.

Monthly Spending						Total Money		
Get Mint Info	Category:	Spent	Budgeted	% of Monthly Total	% of Weekly Total	Total Cash	Total Debt	Total Assets
	Income: Bonus	-\$0 of -\$50				\$21,848.09	\$785.66	\$51,665.06
E-mail	Income: Interest Income	-\$0 of -\$16						
	Income: Paycheck	-\$664 of -\$2,650						
Text	Auto & Transport: Auto Insurance	-\$0 of -\$92						
	Auto & Transport: Gas & Fuel	-\$86 of -\$330						
	Auto & Transport: Service & Parts	-\$0 of -\$55						
	Bills & Utilities: Utilities	-\$65 of -\$310						
	Entertainment: Movies & DVDs	-\$1 of -\$40						
	Entertainment: Music	-\$0 of -\$1						
	Food & Dining: Fast Food	-\$33 of -\$65						
	Food & Dining: Groceries	-\$34 of -\$150						
	Food & Dining: Restaurants	-\$0 of -\$50						
	Gifts & Donations: Charity	-\$0 of -\$850						
	Health & Fitness: Doctor	-\$0 of -\$0						
	Home: Mortgage & Rent	-\$0 of -\$550						
	Shopping: Clothing	-\$0 of -\$100						

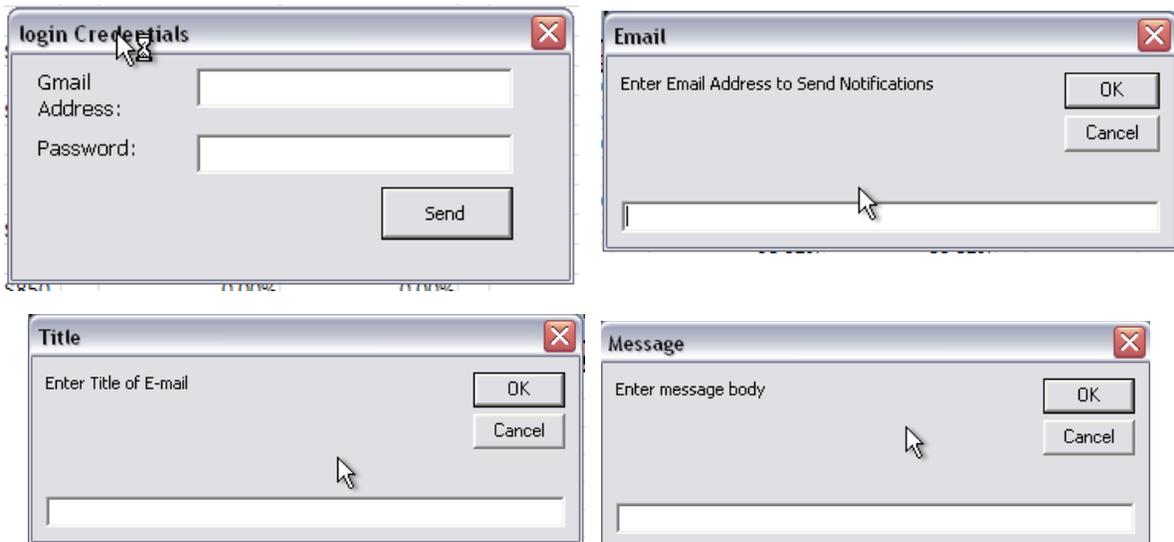
-After getting the data to the template sheet it needed to be formatted in order to do calculations with the amounts. The amounts are copied in as one large string. I first had to split up the data into columns in order to break out what had been spent and what had been budgeted. The total money columns also had extra text added to them from the website that needed to be removed in order to make the numbers useful for calculations. The raw data also had negative signs or dashes, I had to replace these with blank text strings in order for MS excel to see them as positive numbers. Below is the shot of the data formatted after running a macro to clean up the data for calculations.

Monthly Spending						Total Money		
Get Mint Info	Category:	Spent	Budgeted	% of Monthly Total	% of Weekly Total	Total Cash	Total Debt	Total Assets
	Income: Bonus	\$0 of \$50		0.00%	0.00%	\$21,848.09	\$785.66	\$51,665.06
E-mail	Income: Interest Income	\$0 of \$16		0.00%	0.00%			
	Income: Paycheck	\$664 of \$2,650		25.06%	100.23%			
Text	Auto & Transport: Auto Insurance	\$0 of \$92		0.00%	0.00%			
	Auto & Transport: Gas & Fuel	\$86 of \$330		26.06%	104.24%			
	Auto & Transport: Service & Parts	\$0 of \$55		0.00%	0.00%			
	Bills & Utilities: Utilities	\$65 of \$310		20.97%	83.87%			
	Entertainment: Movies & DVDs	\$1 of \$40		2.50%	10.00%			
	Entertainment: Music	\$0 of \$1		0.00%	0.00%			
	Food & Dining: Fast Food	\$33 of \$65		50.77%	203.08%			
	Food & Dining: Groceries	\$34 of \$150		22.67%	90.67%			
	Food & Dining: Restaurants	\$0 of \$50		0.00%	0.00%			
	Gifts & Donations: Charity	\$0 of \$850		0.00%	0.00%			
	Health & Fitness: Doctor	\$0 of \$0		0.00%	0.00%			
	Home: Mortgage & Rent	\$0 of \$550		0.00%	0.00%			
	Shopping: Clothing	\$0 of \$100		0.00%	0.00%			

-After cleaning the data you can use it to make the percentage calculations and the ratio calculations. I also imported total debt, cash, and total assets from the website in order to calculate the personal financial ratios. Below is the completed form that includes all the calculations and ratios.

Monthly Spending					Total Money			Key Ratios	
Category:	Spent	Budgeted	% of Monthly Total	% of Weekly Total	Total Cash	Total Debt	Total Assets	Housing Payment Ratio	
Income: Bonus	\$0 of	\$50	0%	0%	\$21,848.09	\$785.66	\$51,665.06	Liquidity Ratio	21%
Income: Interest Income	\$0 of	\$16	0%	0%				Solvency Ratio	0%
Income: Paycheck	\$664 of	\$2,650	25%	100%				Debt to Income	6576%
Auto & Transport: Auto Insurance	\$0 of	\$92	0%	0%					
Auto & Transport: Gas & Fuel	\$86 of	\$330	26%	104%					
Auto & Transport: Service & Parts	\$0 of	\$55	0%	0%					
Bills & Utilities: Utilities	\$65 of	\$310	21%	84%					
Entertainment: Movies & DVDs	\$1 of	\$40	3%	10%					
Entertainment: Music	\$0 of	\$1	0%	0%					
Food & Dining: Fast Food	\$33 of	\$65	51%	203%					
Food & Dining: Groceries	\$34 of	\$150	23%	91%					
Food & Dining: Restaurants	\$0 of	\$50	0%	0%					
Gifts & Donations: Charity	\$0 of	\$850	0%	0%					
Health & Fitness: Doctor	\$0 of	\$0	0%	0%					
Home: Mortgage & Rent	\$0 of	\$550	0%	0%					
Shopping: Clothing	\$0 of	\$100	0%	0%					

-The final part to the project was to give the user the opportunity to invite a significant other to a meeting in order to review the financial progress. In order to complete this task the user must have a Gmail account. The user can only use Sprint to text, as that is my service provider that I am using. I again used a form to gather the username and password in order to log in to the Gmail account of the user to send the e-mail. Then an input box appears asking for the e-mail address of the person the user would like to send the e-mail to. Then two more input boxes appear asking for a title of the e-mail and another for the user to type the body of the e-mail.



-A similar process is followed when the user wishes to send a text message. The same log in box appears asking for the Gmail account information, and then another text box appears asking for the phone number of the recipient. A final text box appears asking for a message to send to the user.

Discussion of Learning and Difficulties

The first and largest difficulty that I encountered was automating Internet Explorer, specifically how to have the code execute a click on the log in button. The code from the website I was interacting with was different than the example given in class. There was no submit button option in the code of the webpage that I could simply tell VBA to click on. I searched on the web for a few hours trying to figure out the solution to my problem. I gave up, figuring that I was going to have to go to the instructor for help.

In the meantime I decided to build a work around so that I could continue coding my project. I had the web page appear and a message box that prompted the user to click the log in button on the browser. This was an extremely hideous way of completing the task but it was sufficient for the time being. After having completed my project I returned to the log in button issue. I consulted with one of my peers and he was able to show me what he had done to overcome this situation. If I had not asked him I do not think I would have ever come up with the solution he had on my own. Now I am fully confident in the future that I can make it work with any website, thanks to his help.

Another major learning point was looping through the raw information to pick out what I needed to display. I decided I would search for colons. All the budget titles contain colons, but the problem was that there was data before the information that I needed that contained colons. So I simply wrote code to erase some of the data that I did not need to make it easier to get at the information.

The other problem with the colon was that I needed a stopping point just after the data so that I could exit the loop. What I did to overcome this problem was to insert a colon onto a text string just after the data ended. I programmed the loop to end when it found that particular string with the colon on the end of it. I had to make sure though that the text string would always exist in order to ensure I would get out of the loop.

A small element that I did not get to work, but that I wish I could change was not using the select method. I used it several times throughout my code in order to copy and paste. I tried several other ways but for some reason I could not get the code to execute if I did not use the select method. This is not detrimental for me while I use this program, but for others it may be an issue.

Another difficulty that I had was executing the code so that it would be flexible for anyone's budget. In Mint.com the user can create different budgets with different titles. It took a lot of thinking and extra code to be able to make it adapt to different situations. I put a maximum on the code however of about 100 rows, which means 100 different budget categories. I feel that this is plenty, as many people usually have between 10-25 different budget categories.

I wanted to be able to attach the spreadsheet to the E-mail as well but I ran into an error that would not let me access the document. I currently run MS XP on my MacBook and I figured this might be the cause of the error. Sometimes the security on my OS X denies access of XP to certain files. I ended up deciding this was not worth my time to investigate further. I assumed that my wife and I can simply look at the file on my computer when we have our personal financial meetings.

Overall I felt that I learned a great deal about interacting with a webpage through VBA. I now have a better understanding for how to automatically log in and import the information I need. I feel that this will be extremely helpful in the future. I also am more comfortable with using do loops and figuring out ways to make the loops adjust for differing circumstances.