Your Financial Dashboard

Ryker Willie

12/9/2010

Executive Summary:

Mint.com does a terrific job with categorizing one's financial transactions. However, when presenting data spanning more than two months, the user is unable to quickly determine how much of each category was accumulated in each particular month. Instead, the user finds that Mint aggregates the amounts from each month and presents a *totaled* number for the period. This is a problem if the user wishes to see a quick categorized summary for multiple months at one time.

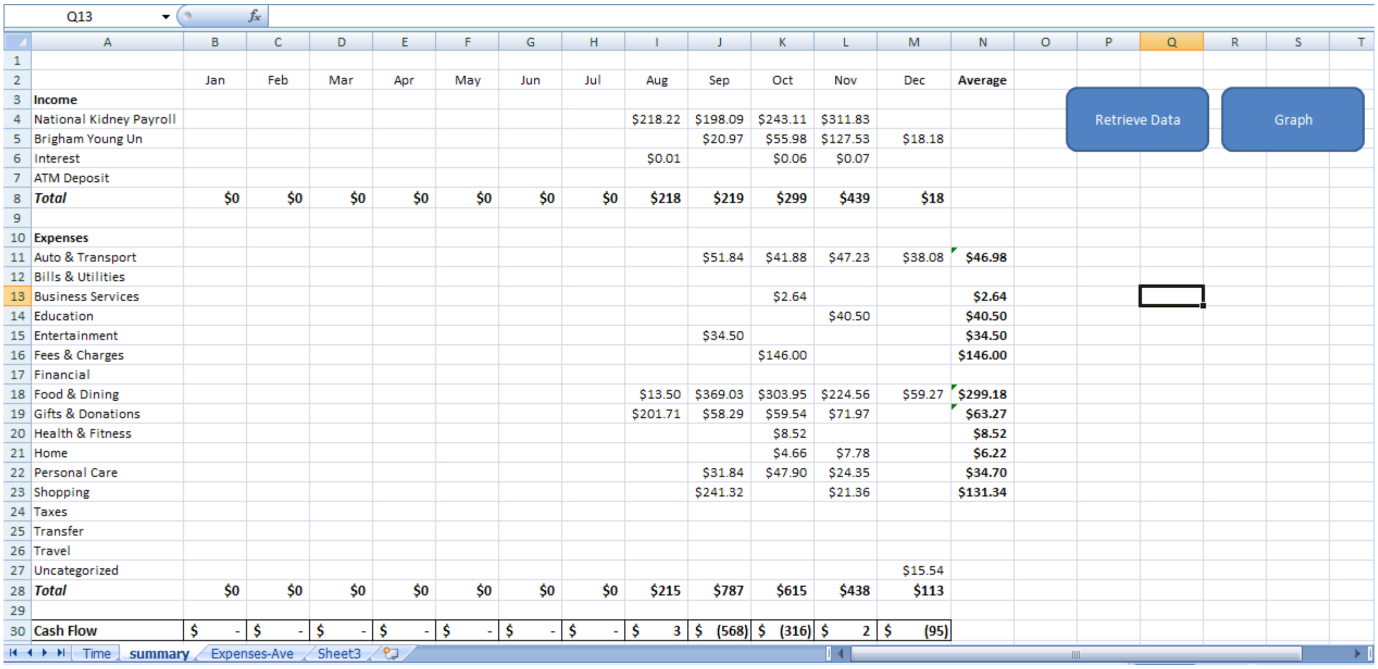
My project involved producing a 12-month dashboard summary that will assist the user in quickly capturing and maintaining the categorized numbers for 12 months of financial data. This required accessing one's Mint.com account, navigating through the Mint.com webpage, importing the data, and presenting the results in a customized dashboard summary sheet. The dashboard also automatically produces graphs of importance for the user.

Introduction

As I walk you through my project implementation, throughout the process I will make note of the difficulties that I encountered and the lessons that I learned.

Implementation

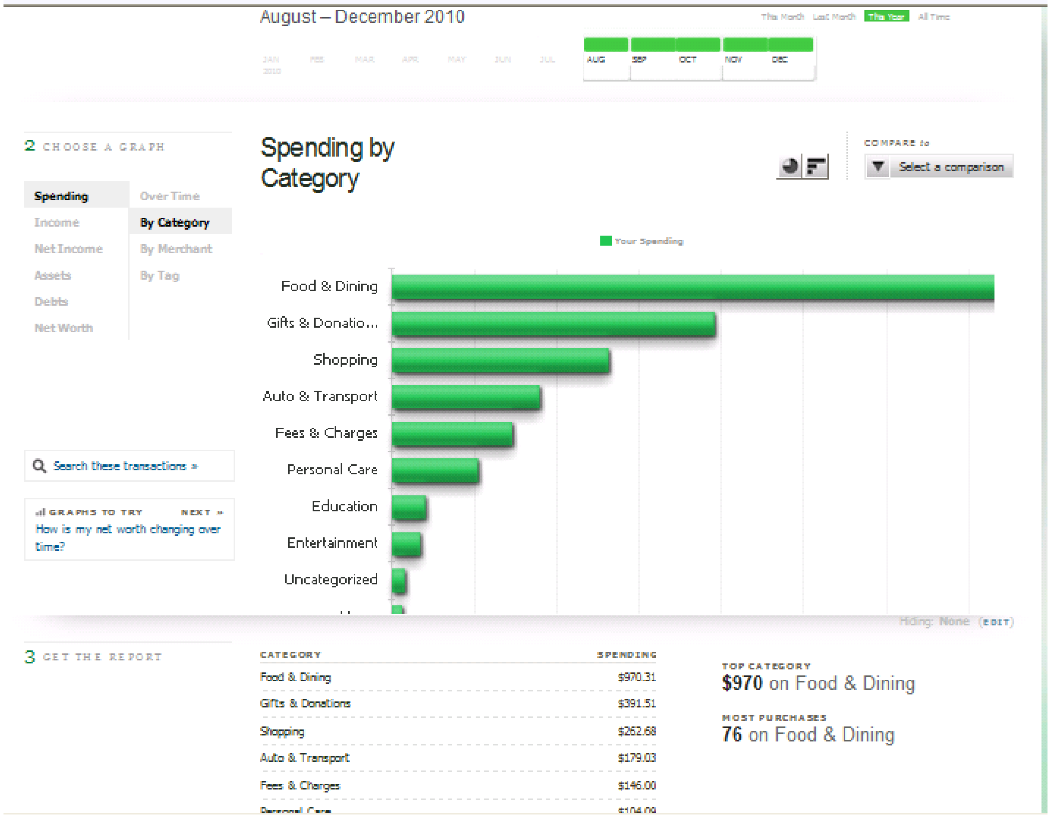
Below is a screen shot of the 12-month financial dashboard



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1														
2		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	**Average**
3	**Income**													
4	National Kidney Payroll								$218.22	$198.09	$243.11	$311.83		
5	Brigham Young Un									$20.97	$55.98	$127.53	$18.18	
6	Interest								$0.01		$0.06	$0.07		
7	ATM Deposit													
8	***Total***	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$218**	**$219**	**$299**	**$439**	**$18**	
9														
10	**Expenses**													
11	Auto & Transport									$51.84	$41.88	$47.23	$38.08	**$46.98**
12	Bills & Utilities													
13	Business Services										$2.64			**$2.64**
14	Education											$40.50		**$40.50**
15	Entertainment									$34.50				**$34.50**
16	Fees & Charges										$146.00			**$146.00**
17	Financial													
18	Food & Dining								$13.50	$369.03	$303.95	$224.56	$59.27	**$299.18**
19	Gifts & Donations								$201.71	$58.29	$59.54	$71.97		**$63.27**
20	Health & Fitness										$8.52			**$8.52**
21	Home										$4.66	$7.78		**$6.22**
22	Personal Care									$31.84	$47.90	$24.35		**$34.70**
23	Shopping									$241.32		$21.36		**$131.34**
24	Taxes													
25	Transfer													
26	Travel													
27	Uncategorized												$15.54	
28	***Total***	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$215**	**$787**	**$615**	**$438**	**$113**	
29														
30	**Cash Flow**	$ -	$ -	$ -	$ -	$ -	$ -	$ -	$ 3	$ (568)	$ (316)	$ 2	$ (95)	

The financial dashboard is the summary page that displays categorized income and expense data for 12 months. To better understand the need of such a dashboard, consider the fact that Mint gives only the following choices: month vs prior month comparison, month vs same month last year comparison, current year totals. The presentation of the financial data is severly limited. The 12 month dashboard summary enables the user to see everything on one page.

Year-to-Date Spending without Monthly Breakdown

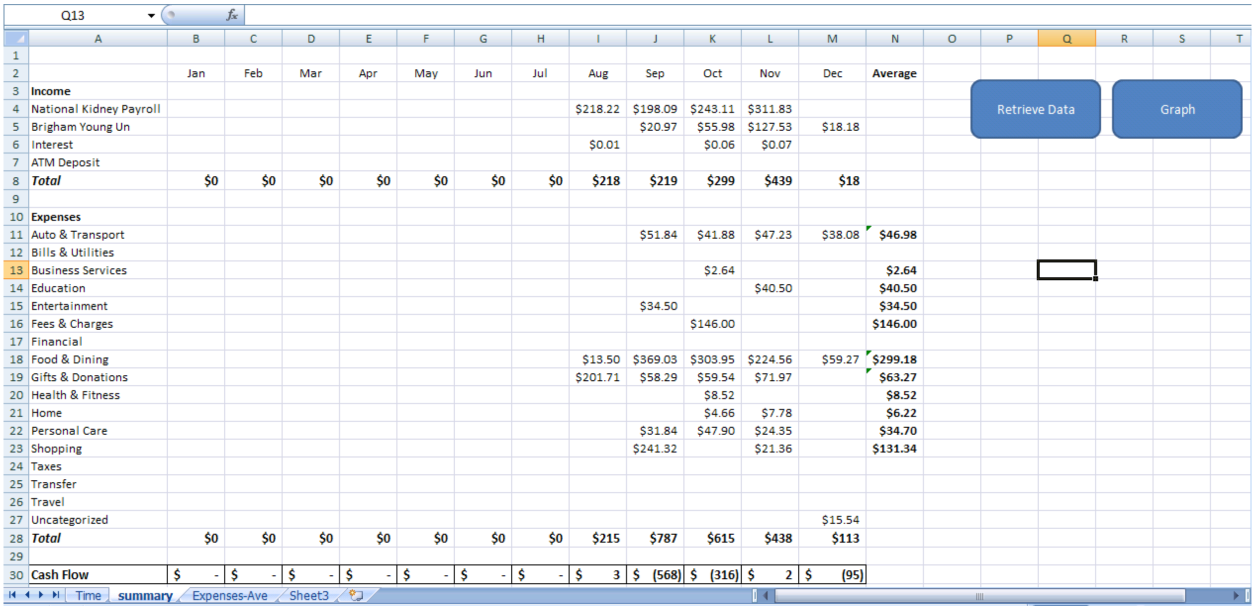


CATEGORY	SPENDING
Food & Dining	$970.31
Gifts & Donations	$391.51
Shopping	$262.68
Auto & Transport	$179.03
Fees & Charges	$146.00
Personal Care	$104.09

TOP CATEGORY
$970 on Food & Dining

MOST PURCHASES
76 on Food & Dining

VS



	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1														
2		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec	**Average**
3	**Income**													
4	National Kidney Payroll								$218.22	$198.09	$243.11	$311.83		
5	Brigham Young Un									$20.97	$55.98	$127.53	$18.18	
6	Interest								$0.01		$0.06	$0.07		
7	ATM Deposit													
8	***Total***	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$218**	**$219**	**$299**	**$439**	**$18**	
9														
10	**Expenses**													
11	Auto & Transport									$51.84	$41.88	$47.23	$38.08	**$46.98**
12	Bills & Utilities													
13	Business Services										$2.64			**$2.64**
14	Education											$40.50		**$40.50**
15	Entertainment									$34.50				**$34.50**
16	Fees & Charges										$146.00			**$146.00**
17	Financial													
18	Food & Dining								$13.50	$369.03	$303.95	$224.56	$59.27	**$299.18**
19	Gifts & Donations								$201.71	$58.29	$59.54	$71.97		**$63.27**
20	Health & Fitness										$8.52			**$8.52**
21	Home										$4.66	$7.78		**$6.22**
22	Personal Care									$31.84	$47.90	$24.35		**$34.70**
23	Shopping									$241.32		$21.36		**$131.34**
24	Taxes													
25	Transfer													
26	Travel													
27	Uncategorized												$15.54	
28	***Total***	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$215**	**$787**	**$615**	**$438**	**$113**	
29														
30	**Cash Flow**	$ -	$ -	$ -	$ -	$ -	$ -	$ -	$ 3	$ (568)	$ (316)	$ 2	$ (95)	

After creating the summary page, the next step was to access Mint.com , import the required data, and place the imported data in the proper column on the financial dashboard.

Learning and Conceptual Difficulties

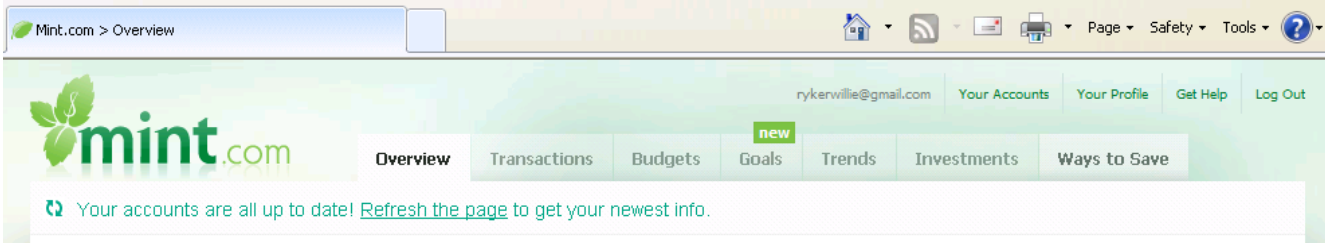
Besides receiving a couple of lectures that introduced me to the basics of accessing password protected sites, this was my first time attempting to interact with a unique webpage through code. In addition, this was my first time working with class modules.

Class Modules

At first, I was unfamiliar with the significance of Professor Allen's web page class module. In fact, I was unaware that I needed to incorporate that same class module into my own project. After a few hours of attempting to create my own class module, I realized that all I needed to do was import Professor Allen's module into my project. Even though I became somewhat side tracked, I learned somewhat concerning class modules. I now know that class modules enable a programmer to design his own object, methods, and properties.

Accessing Password protected sites

After I became familiar with the class module, I was able to successfully log in to Mint. The next step was learning how to navigate around on the web page. This involved learning the code it takes to click on tabs or links. This proved to be a challenge (see section below). Upon entering the web page, I next needed to click on the "Trends" tab below. After clicking on this link, I was able to get to the data that I needed.



Your accounts are all up to date! Refresh the page to get your newest info.

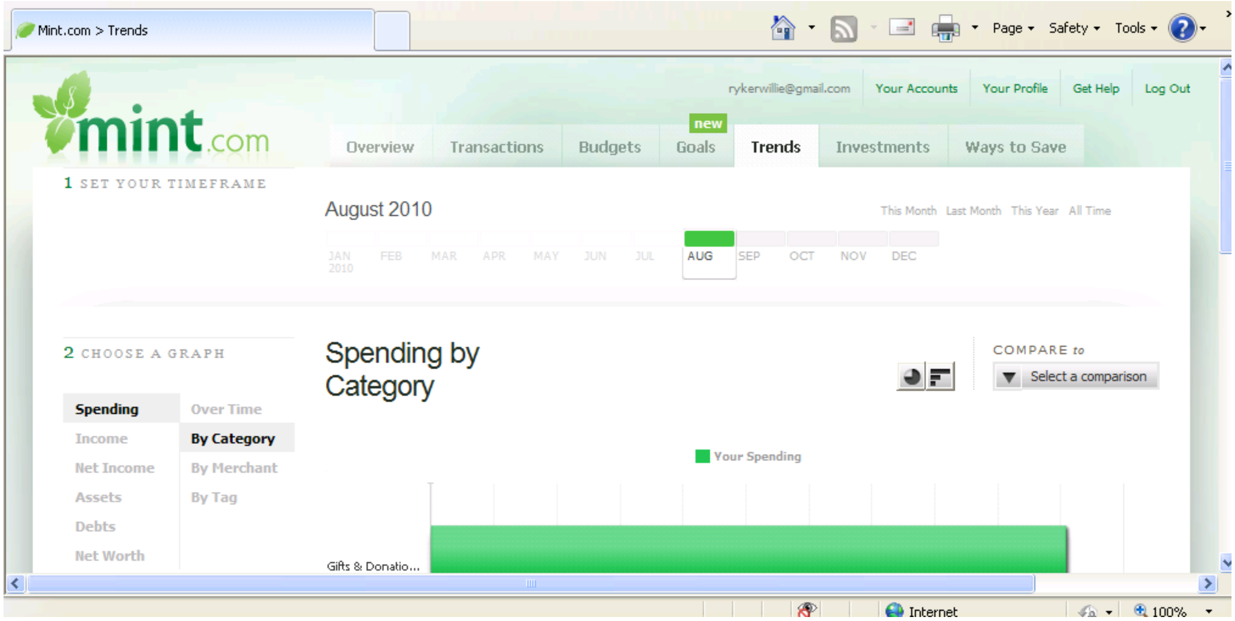
Challenges with navigating on websites

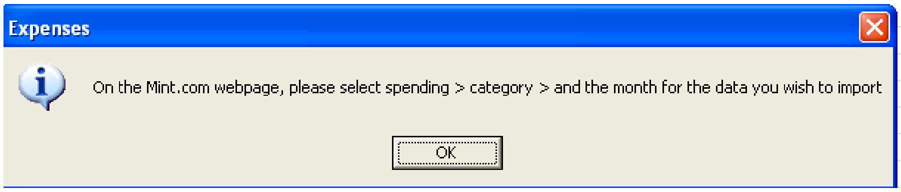
This was probably one of the more difficult parts of my project. After some time I learned that Mint's HTML code employed various means for navigation. These included the following:

- Buttons
- Forms
- HTML links
- Javascript

As one knows, it order to manipulate an HTML object, the user must first discover that object's ID or name. I spent a good amount of time exploring the possibilites. Probably the greastest skill that I developed when working with the HTML code is greater efficiency when locating an object's ID or name in the HTML code. In addition, I know how to manipulate HTML buttons, HTML forms, and HTML input boxes.

The most challenging navigation related to javascript. The screenshot below demonstrates the struggle. Initially, I wanted the project to be completely automated. As such, the user would not have to select the desired month on Mint. However, I was unable to figure out exactly how to command this javascript code. I was somewhat relieved when Professor Allen could not figure it out either (He informed me that he spent about an hour and a half trying to figure it out). At some length, we determined that the best way to deal with the problem would be to introduce a message box that requested the user to manually select the desired month. See the screenshot below:



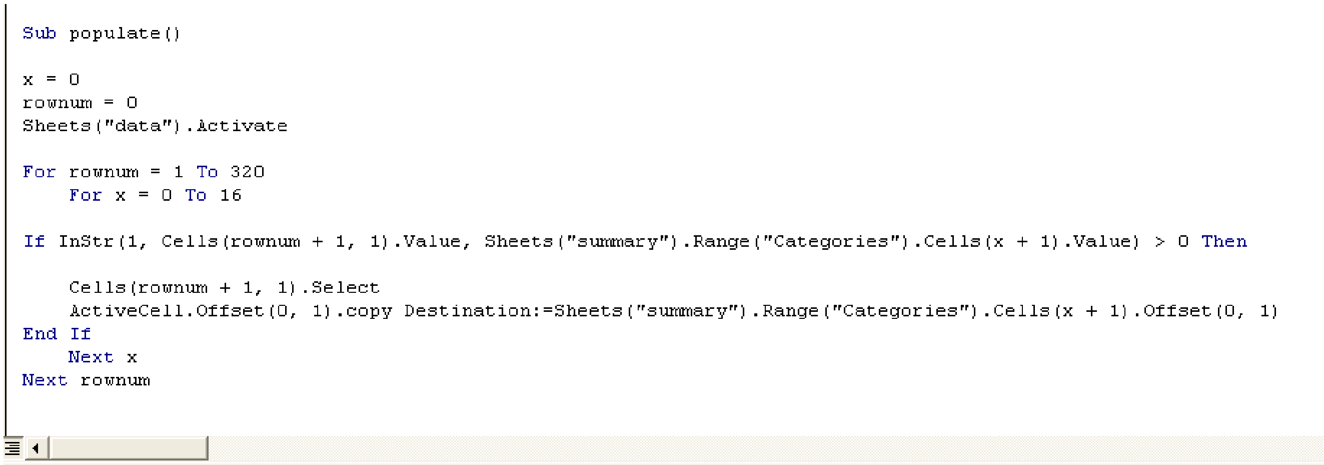


On the Mint.com webpage, please select spending > category > and the month for the data you wish to import

Here I learned an important lesson. If I am unable to figure out how to automate a process, there is no harm in stopping the code so that the user can make necessary adjustments, in this case, the selection of the proper month. At first, it was hard to introduce manual user interference. I resisted because I wanted my project to be completely automated. I have learned, however, that at times it might be most practical to stop code execution and allow the user to make the necessary adjustments. This can be done with message boxes.

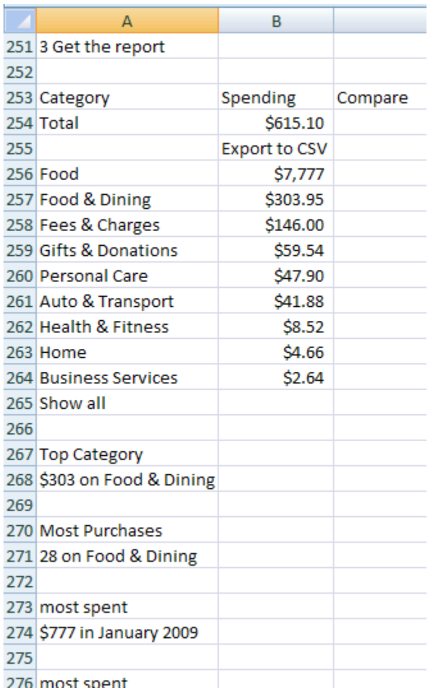
Locating and Placing the Expense Data

Once the data was imported into my workbook, I had to figure out how to locate the data needed. Once I found the data, the next step was writing the code that properly placed the code on the dashboard summary. For me this was my crowning acheivement I was able to both locate and paste the data onto the summary page with just one subprocedure. I employed the use of the INSTR function and a nested for loop. See the code below:



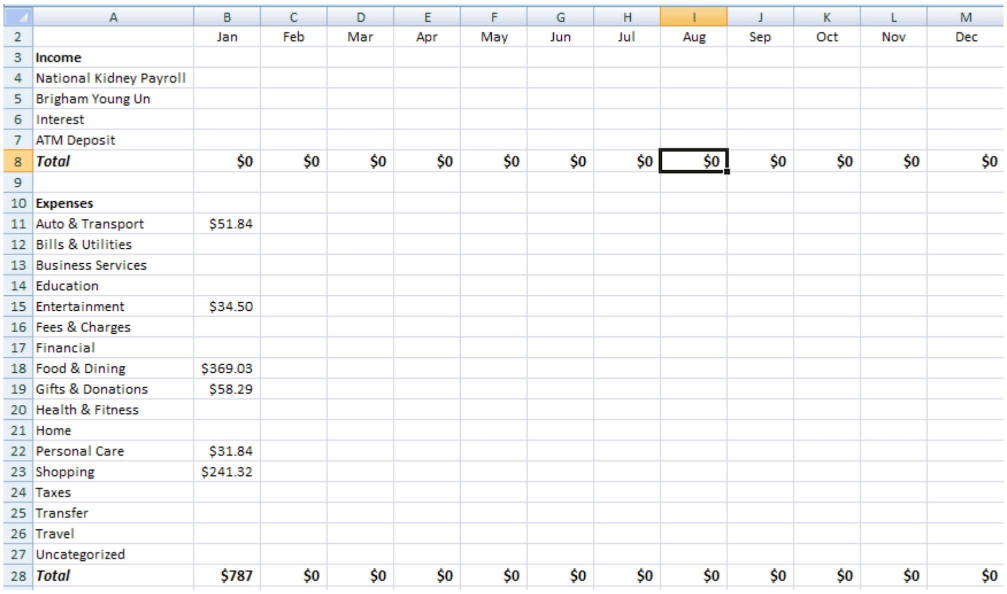
```
Sub populate()

x = 0
rownum = 0
Sheets("data").Activate

For rownum = 1 To 320
    For x = 0 To 16

If InStr(1, Cells(rownum + 1, 1).Value, Sheets("summary").Range("Categories").Cells(x + 1).Value) > 0 Then

    Cells(rownum + 1, 1).Select
    ActiveCell.Offset(0, 1).copy Destination:=Sheets("summary").Range("Categories").Cells(x + 1).Offset(0, 1)
End If
    Next x
Next rownum
```

However, I had to make one adjustment before executing the code above. The screenshot below is a shot of the imported page. Notice cell A271. When I executed the code above, the subprocedure always selected cell A271 when searching for “Food & Dining”. As a result, the correct amount of “Food & Dining” expense never got pasted to the dashboard summary. I still don’t know why the INSTR function skipped cell A257, especially since it is encountered before cell A271 (I faced the same issue when searching my income items i.e. the top income source would always be skipped). What I decided to do was find the cell containing “Top Category” and delete the items below it. This resolved the issue and all items were able to be properly pasted to the dashboard summary.



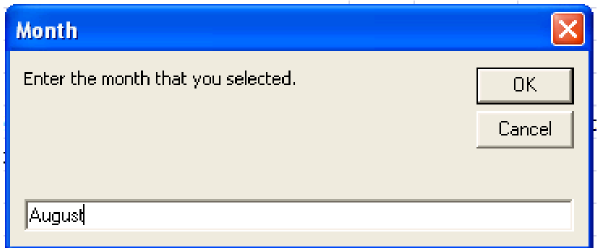
	A	B	
251	3 Get the report		
252			
253	Category	Spending	Compare
254	Total	$615.10	
255		Export to CSV	
256	Food	$7,777	
257	Food & Dining	$303.95	
258	Fees & Charges	$146.00	
259	Gifts & Donations	$59.54	
260	Personal Care	$47.90	
261	Auto & Transport	$41.88	
262	Health & Fitness	$8.52	
263	Home	$4.66	
264	Business Services	$2.64	
265	Show all		
266			
267	Top Category		
268	$303 on Food & Dining		
269			
270	Most Purchases		
271	28 on Food & Dining		
272			
273	most spent		
274	$777 in January 2009		
275			
276	most spent		

The next challenge was figuring out how to get the data in the correct column on the dashboard summary. The nested for loop, by default, placed the data in the January column as shown below. Placing the data in the correct column would have been easy if the imported page contained a unique identifier associated with the selected month. However, suprisingly, the imported web page contained no such identification.



	A	B	C	D	E	F	G	H	I	J	K	L	M
2		Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
3	**Income**												
4	National Kidney Payroll												
5	Brigham Young Un												
6	Interest												
7	ATM Deposit												
8	***Total***	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**
9													
10	**Expenses**												
11	Auto & Transport	$51.84											
12	Bills & Utilities												
13	Business Services												
14	Education												
15	Entertainment	$34.50											
16	Fees & Charges												
17	Financial												
18	Food & Dining	$369.03											
19	Gifts & Donations	$58.29											
20	Health & Fitness												
21	Home												
22	Personal Care	$31.84											
23	Shopping	$241.32											
24	Taxes												
25	Transfer												
26	Travel												
27	Uncategorized												
28	***Total***	**$787**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**	**$0**

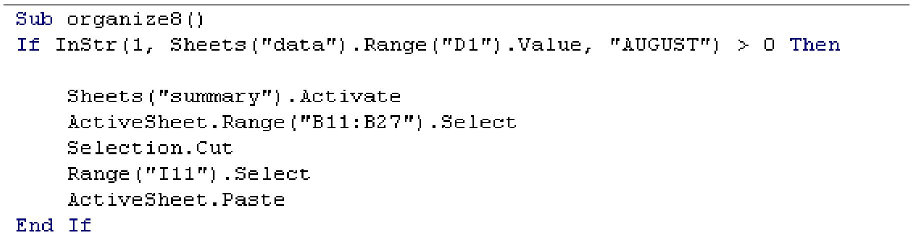
After some time, I realized that I could allow the user to specify which month he had imported. Here, I employed the use of an input box where the user types in the name of the month imported. See the screenshot below:



Enter the month that you selected.

August

The resulting value is copied onto the imported page and used as the page's unique identifier. Now, with a unique identifier, I was able to assign the data to the correct column utilizing the following code:

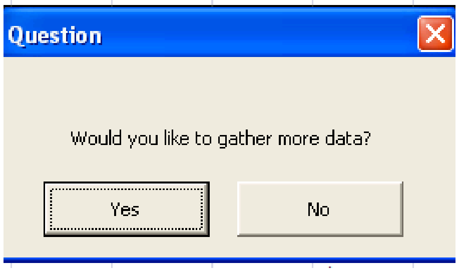


```
Sub organize8()
If InStr(1, Sheets("data").Range("D1").Value, "AUGUST") > 0 Then

    Sheets("summary").Activate
    ActiveSheet.Range("B11:B27").Select
    Selection.Cut
    Range("I11").Select
    ActiveSheet.Paste
End If
```

As an aside, I used the UCASE function on the value typed by the user to make sure that the code above finds the month typed by the user.   In addition, here I would have liked to have developed the code in such a way as to not have a separate subprocedure for every month. I am still unaware of how this might be done.

Locating and Placing the Income Data

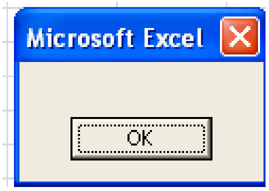
Once the expense data is placed in the proper column on the dashboard summary, the code performs the same procedures but this time imports income data and places it properly on the dashboard summary. After the code completes the income data, the user is prompted with the following user form.



Would you like to gather more data?

If the user clicks “Yes”, he restarts the entire process and is able to import data for a different month. If the user clicks “No”, the code ends and he is able to enjoy the benefits of his newly created financial dashboard.

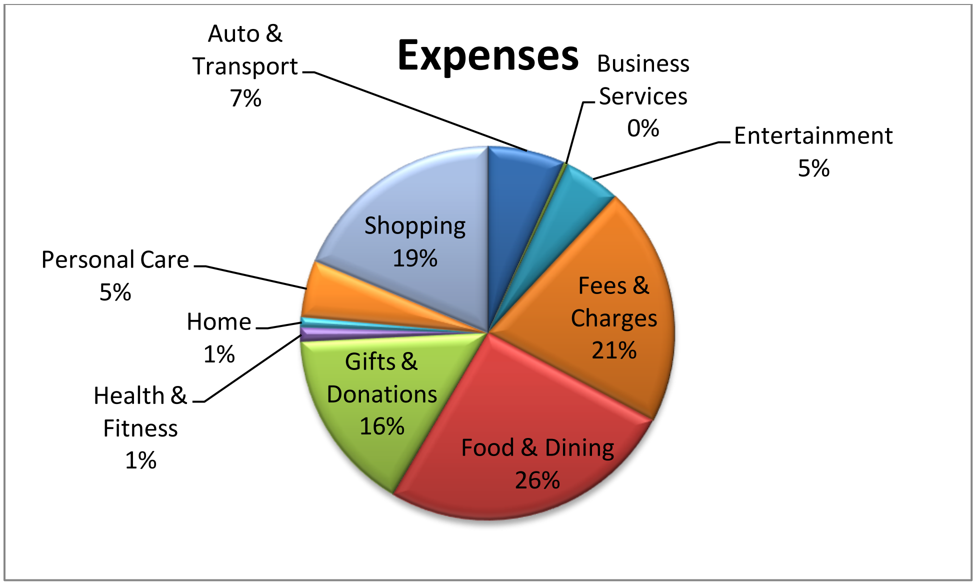
For some reason the code produces this strange message box before the user form.

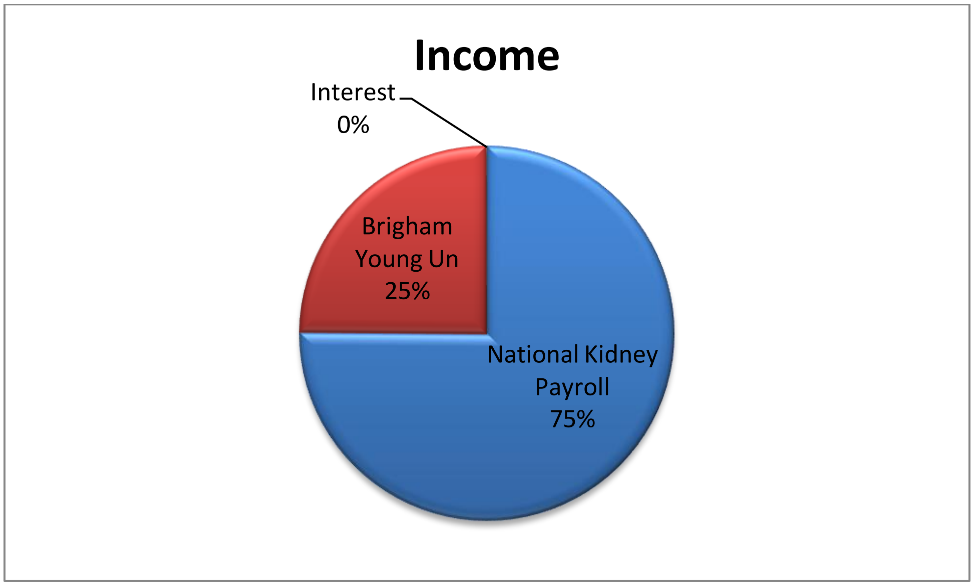


Even though the code continues after clicking “OK”, it would be nice to figure out how to remove this message box.

Graphs

With the click of a button, the user can view a pie chart that helps the user understand where his money is being spent. The graph contains average spending and effectively shows each category’s percentage of total spending. In addition, the user can view a pie chart that helps him better understand his income in much the same way. See an example of the pie charts on the page below.


Expenses
Auto & Transport 7%
Business Services 0%
Entertainment 5%
Fees & Charges 21%
Food & Dining 26%
Gifts & Donations 16%
Health & Fitness 1%
Home 1%
Personal Care 5%
Shopping 19%


Income
Interest 0%
Brigham Young Un 25%
National Kidney Payroll 75%