

Fall
2015

Reconciliation Preparation

IS 520 SECTION 001
JOSH TAKABORI

Executive Summary

This program was created to assist Brigham Young University's dining services accounting department in their monthly inventory reconciliations. There are two main software programs that dining services use to account for their inventory: Eatec and Peoplesoft. When food is delivered to the various locations on BYU campus, the locations receive invoices on the deliveries. All of the invoices get sent to the accounting office where they are put in the Eatec system. With the information given in the invoices, Eatec keeps track of all the inventory in each department. For example, when sales are made and physical counts of food are done in the cafeterias, these amounts get put into Eatec, where Eatec will then deplete the inventory in the cafeteria. Peoplesoft, on the other hand, is the software that handles all the payments and money balances for dining services. When invoices get put into Eatec, Eatec initiates payment requests that are finalized on Peoplesoft. Peoplesoft should essentially mirror the transactions Eatec makes so the inventory balance on Eatec and the financial balance on Peoplesoft should ideally always be the same.

The reason for reconciling the Eatec balance to the Peoplesoft balance each month is because the two balances are never the same. Eatec transactions occur that do not get followed through on the Peoplesoft side. For example, sometimes Eatec may put an invoice in for the Cannon Center department but Peoplesoft processes a payment coming from the Creamery on Ninth. Sometimes, an invoice will be put into Eatec and due to the supplier, will not automatically go into Peoplesoft. In this case employees from other departments will have to manually put the transaction into Peoplesoft for payment. Oftentimes these transactions are put in incorrectly or not at all, causing imbalances.

Every month a long workbook of data is prepared that has every single transaction that is not matched up on both the Eatec and Peoplesoft side. The employees are asked to look at all the unreconciled transactions by department and identify the cause of the problem. When I first started working here, preparing the data by separating the transactions by department was nearly as time consuming as the reconciliation process. This macro will take the monthly reports and create a workbook for a specific employee and their designated departments.

Implementation

Before I get into the steps I will explain important elements in the procedure:

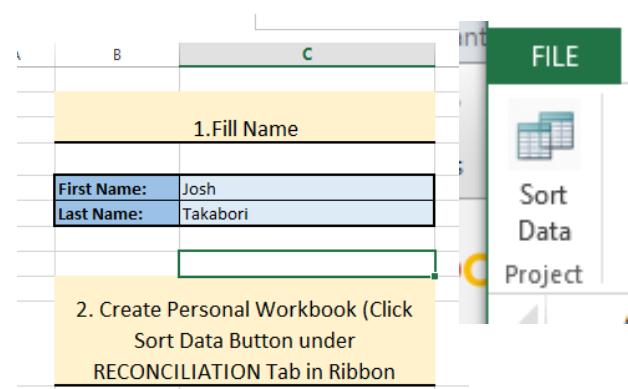
- The master file that runs the macro has a worksheet titled “Assignments” that includes the department, operating unit number, and employee assigned to that department. This is the information that the user form will draw its information from. All new departments and changes to the employees should be edited on this worksheet.

A	B	C	D	E
Column1	Operating Unit #	Department	First Name	Last Name
1	15512132	CONE	Josh	Takabori
2	15512125	Wyview	Warren	Merritt
3	15512105	Cannon	Josh	Takabori
4	15512111	Central Food Stores	Greg	Smith
7				
8				

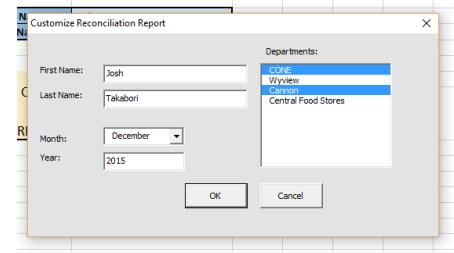
- The monthly reports will all be labeled by the month and year (e.g., January2015, March2011, etc.)
- All files will be opened from and saved onto the same location that the master file is located in.
- Userforms, arrays, arrays within arrays, referencing workbooks, and do until isempty were some of the more significant VBA components to be implemented.
- The code creates a “delete” sheet in the master file and the monthly report that gets pulled up later in the macro. The “delete” sheets will always be deleted by the end of the macro. These sheets contain the data and variables from the userform and other workbook. I found this to be the easiest way to reference variables from anywhere else besides the workbook currently being modified.

Here are the steps that were taken to make preparing the workbooks in a timely manner:

- On the main page, the first and last name of the employee will be put in the appropriate cells. To begin the macro, the user will click on the button on the Reconciliation tab on the ribbon.



- A user form will open up that includes the name of the employee, a list of all the departments, and a textbox and dropdown list that lets the user select the month and year of the reconciliation. The list box of the departments will be initialized by highlighting the departments that the employee are assigned to according to the reference tab. The highlighted departments determine how many worksheets will be created through the macro.



- When the user clicks okay, the macro runs another sub that will open the appropriate workbook and create new worksheets in that workbook. Each worksheet will represent a department and the macro will put the appropriate transactions from the monthly report to the corresponding worksheet. It is important to note that the monthly report carries a lot of unnecessary data. The macro will only pull the data that is relevant for reconciling.

- Each worksheet will be formatted to have a professional look. Most importantly, a formula will be located in two columns, one for Eatec and another for Peoplesoft. The cells in each respective column will only show the dollar amount of that transaction if another cell in the same transaction row has been marked (signifying a transaction has been reconciled). Another formula is also included at the bottom of certain columns to get the sum of all the reconciled transactions.

- Because the monthly report will be used by multiple employees, the final step in the program is to save the file as the name of the employee and the month and year being reconciled (e.g.

“JoshTakaboriDecember2015”).

- The workbook is now customized for the employee and the data is ready to be reconciled.

Lessons Learned/Difficulties Encountered

There were many new nuances of VBA that I learned or appreciated learning about in Professor Allen’s class over the course of this project.

- Validation/Error Handling – what ended up being more of the time-consuming aspects of the project was making sure the program would run into as few errors as possible. Some errors that would come up were if the monthly report the user tried to pull up did not exist and if the macro tried to format a table of data with zero data. I had to go through several steps where errors could occur and just exit certain subs before the error could occur.
- Deciding where to define variables – Because I was working with different workbooks and subs in a userform, I had to determine where to define variables. Variables defined in a userform sub were not easy to bring over to subs outside of the userform so I had to think of different methods to bring variables back and forth between userform subs, regular subs, and different workbooks. What I ended up doing was having the userform clickOK sub create a new worksheet in the main workbook that would contain the values of the variables defined in the userform. Then I would modify the monthly report workbook based off the newly created worksheet from the main workbook. I made sure that the reference sheet would always be deleted or not appear at any point during the execution of the code. This is important because I would always have the reference sheets be titled “delete,” and if a “delete” sheet still existed from a previous execution

of the code the next execution would not work because excel cannot create two sheets of the same name.

- Putting in formulas – I spent a good amount of time figuring out how to put in a formula that was based on relative references of other cells. I stubbornly attempted to treat these formulas like more basic formulas that can just be typed into the code the same way it is written on the formula bar. After a while I gave in and searched the internet to find that I needed to use the R1C1 property.

Assistance

I received help from the lessons learned in Gove Allen's class, his ribbon wizard, and the limitless resource known as the internet that help me overcome several roadblocks on this project. All concepts and procedures were originally conceived by me.

Conclusion

There is a special feeling of satisfaction that comes when completing a VBA project. When I finally completed this assignment I would run the program again and again because it was so fun to watch my code run smoothly. Assignments like this make me appreciate how much/little I know about VBA.