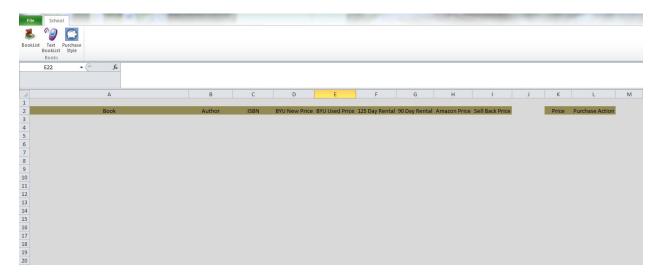
This project consists of creating an interface for students at BYU to have a way to assist students in textbook purchases. It is designed to give all the information needed to allow you to compare the costs of books and make an accurate estimate of how much you will ultimately spend on books after sellback is included. This workbook goes to BYU obtains your book list and then goes to sites such as the bookstore rental store, Amazon, and Bookbyte (an online textbook buyer) to obtain estimates of how much everything will cost. It then gives the option for the student to decide the best course of action for each individual textbook.

What I did to accomplish my project was to first create a worksheet to hold the results of my online searches for the books and their prices. The opening page is somewhat bare in order to just allow for the imported information to populate the page. The below figure illustrates the opening scene of the workbook.



I have also inserted three buttons in the ribbon to facilitate the project. The first button is what will run the sub procedure that will go out and begin to obtain the information for my project. The other button is for after the information is gather the user may decide to have the book information (Book title and author) texted to their phone in case they wish to purchase their books at the bookstore.

When the Booklist button is pressed the user is presented a user form that prompts the user to input their BYU netID and their password. The password's characters are changed so other looking will not know the password used. After this the user has the option of either pressing enter or clicking on the command button "Get Book Info". If the User fails to input one of the two fields or clicks the x in the corner the sub procedure will exit out. The following picture below is an example of the user form that is presented.



After the information is sent, the sub procedure will go to the BYU Bookstore website to get your booklist. If someone is previously signed-in the macro will check to see if it needs to sign-out anyone who has previously been there. It will then authentic using the user's netID and password. An Internet Explorer window will pop-up at sign-in due to some problems that will be discussed later, but this IE window will come to the front and sign the user in and then disappear for the rest of the macro. Once the user is signed in, the macro will begin to scrap the BYU Bookstore website for the user's booklist taking out their book title, author, and ISBN number; and record them to the workbook. Each new entry will alternate between a light and dark blue color in order for the user to analysis the costs of books easier. Also the Total Price and a dark grey box appear in the corner exactly 2 cells below the last book entry underneath the price column.

The macro then uses the ISBN obtained from the BYU Bookstore website and goes to the BYU rental website and obtains prices for both the 125 Day and the 90 Day rental periods. It afterwards uses that same ISBN to go to the Amazon and Bookbyte websites to obtain the prices for a new book and prices for sellback respectively. In the event that you are unable to sell back a book or not able to rent it, the macro will simply return a text that says that "No rental" or "No Sellback" depending on the circumstances. Below is a snapshot of what it looks like after all the website scraping for prices has taken place.



After I accomplished this, I moved on to allowing the user to be able to compare and decide in what way they would like to purchase their books. For this I ended up using a user form to allow the user to control purchasing choices and also if they would like the books info sent to them. The user form contains the information of the new price, used price, rental prices, and total price after sellback along with book title and the author's name. To do the sell back price I took the smallest price of new and used books and subtracted the sellback price. For the new price I took the cheapest of the BYU and Amazon new prices. With all this information I have also included a combobox. From this combobox it only contains the possible option you are allowed to do with the book. For example, if there is no rental price there will be no rental option in the combo box, the same goes for the sellback option. Along with this there are next and previous buttons that will allow the user to scroll through all the book titles

generated by the online searches. There is also a check box that will allow the user to toggle between having the book info texted or not. Though to actually text the info we will use another button. Below is a screenshot of the User form used.

urchasing Choices			X
Book Title MACROECONOMICS 7E		Author MANKIW, N	
New Price	Used Price	125 Day Rental	90 Day Rental
\$166.70	\$125.05	\$62.13	\$54.32
Price After Sellback \$88.05	☐ Text Book Info Select Purchase		Next
Done	125 Rent	•	Previous

As the options are done are changed in the combobox and next or previous are clicked the values of the Purchase Action column are changed accordingly. When this happens the price is also updated in the Price column. As these prices are input the total price also in turn correct to give you the total amount that you are expected to play.

The final part of the project was I made it so that the user could text the books that had been selected from the book user form. What happens here is I bring up a user form much like what was given to us in class. The difference is I now included an option to put a phone number for a Verizon phone. This allows me to be able to text it to different people if needed. I have included the changed user form below.

Google Credentials		
GMail account: @gmail.com		
GMail password:		
Phone Number:		
Cancel OK		

After the information is input you will get the texts that include the book's title and the author's name. I assumed that these would be the things you would need because that is what you normally use when looking for the textbook in the bookstore.

For this project I didn't receive much if any help from other people. I did rely heavily on googling some potential ways to do certain tasks, but not many of those ended up being implemented in the project.

There were a few difficulties I encountered while working on this project. The first problem I found dealt with the BYU authentication. When I first tried to approach this, I found that it did not have a tag on the button for submitting and in a way was not a button I could refer to at all. To work around this I had the Internet Explorer window become visible for the task of submitting using a send key of [Enter]. The next thing I realized was that unless I typed something in also it would not allow me to authenticate. To work around this I simply had a sendkey [K] used before I filled in the Username and password. After doing this I was able to be able to sign-in to BYU. Upon talking to someone after finishing I found out that you had to submit a form in order to get it to work without it popping up.

The next difficulty I encountered was I originally intended on creating a combobox for every book in the column to the right of the price column. I found a few problems with this. First I realized I had the problem of trying to refer to each combobox dynamically to use its value for determining the price in the pricing column, because these are only created if a book exists on that row. I was then unable to dynamically enter in values depending on if rentals and sellback was available. If I created the combobox and then tried to fill it in the same macro it wouldn't work, but when I did it all through the immediate window it was fine. Lastly, I decided to give them all the same inner values but was unable to find a way to refer to these values I had given them. I ran out of time so I instead opted to just use the user form in order to just assign a word value to the Purchase action column to avoid these head aches and used my combobox easily in the userform.