

Problem Statement

In this project, you will create from the ground up a user form that will enable one to use real-time exchange rates to convert currency from one unit to another. You will automate a data query in Excel, and link the results to an easy-to-use VBA user form. The default date for the currency exchange is the current date, but you should be able to enter in any historic date (within reason) to obtain the exchange rates/converted values for times in the past. Finally, your user form should generate a plot of exchange rates over the past 30 days.

I have provided a variety of screencasts that will take you step-by-step through many of the main components of this project, but you'll still have to use some things you have learned earlier related to VBA in order to complete everything. You will also likely need to implement your debugging skills to work out all the problems with your code, should they arise.

Prior to project submission, please make sure to read the requirements and grading rubric below. Specifically, there are some major issues with the date format used around the world, so it is important that you take screen shots of your project "in action" (i.e., working properly on your machine) and place these screen shots into your workbook in case the grader's machine has different regional date settings applied.

Requirements/Grading Rubric

- (1 pt) You can start with a list of all the currencies and their descriptions (e.g., "USD" in column A and "US Dollar" in column B, etc.) on a visible worksheet, but the user should never see the imported data. Start with a hidden sheet, which is then unhidden, data imported to it, data used, but then the sheet is hidden. Use `Application.ScreenUpdating = False` such that the user never sees anything that's going on "behind the scenes".
- (0.5 pt) By default, the date shown should be the current date.
- (2.5 pts) Your currency converter user form should successfully make a variety of different exchanges, and display the result in a message box. The grader should verify that several of these conversions work (choose randomly and compare to current data on <http://www.xe.com/currencytables/>).
- (1 pt) Your user form should work on different dates. IMPORTANT: Due to variations in date formatting across the world, this part of your project may not work on the computers of your Peer Reviewers. Therefore, please take a screen shot of your result and place the plot somewhere easily visible in your Excel (.xlsm) file.
- (0.5 pt) The Quit button should work perfectly. When the form is re-opened, you should check to make sure that the combo boxes do not have repeat values.
- (2.5 pts) Your user form should successfully plot the last 30 days of exchange rates. IMPORTANT: Due to variations in date formatting across the world, the 30-day plot may not work on the computers of your Peer Reviewers. Therefore, please take a screen shot of your 30-day plot, and place the plot somewhere easily visible in your Excel (.xlsm) file (for example, on a clearly labeled worksheet entitled something like "30-day plot").
- (2 pts) Input validation (0.5 pts for each of the following)
 - If the user inputs a negative or zero currency, it should provide an error.
 - If the user leaves the input field blank and tries to run the button, it should display an error.
 - If the user puts text in the input field, it will display an error.
 - If the date field is not a date, then an error should be displayed

Hints

- I provide a file that has some excellent code for importing data from tables in web pages. See “Query – STARTER.xlsm” as well as the screencast “How to run a data query”.
- General input validation information for user forms can be found in Week 4 of Part 2 of the course.
- To verify that a string is a date, the IsDate function in VBA can be employed.
- From a Coursera learner (thanks, Leland Remson!): To eliminate the date issues that arise due to different regional settings by the various learners in the course, just use Format(DateBoxValue, "yyyy-mm-dd") as a string and concatenate it into the url string. This should help you a lot!