ISTA 350 Worksheet, 1/16/2020 Name:

Write a function called `word_count` that takes a string and a Boolean with a default value of `True`. If the Boolean is `True`, the function should ignore case when processing the string. The function returns a `Series` that has the words in the string as position labels in alphabetical order and the number of times that they occur as elements. Don't worry about punctuation.

Write a Boolean function called `is_symmetric` that takes a square `DataFrame` and returns `True` if its elements represent a symmetric matrix, `False` if not. Recall that a symmetric matrix is mirrored across the diagonal, i.e. $\mathbf{A} = \mathbf{A}^t$. An example of a symmetric matrix:

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 4 & 5 \\ 3 & 5 & 8 \end{bmatrix}$$

Write a function called `letter_count` that takes a string and a Boolean with a default value of `True`. If the Boolean is `True`, the function will process text in case-insensitive fashion. The function returns a `Series` that maps the letters of the alphabet (position labels) to the number of times that they occur in the string (elements). Ignore non-letter characters. I suggest taking an example string and figuring out by hand what the `Series` should look like as prep.

Write a function called `sum_rows` that takes a frame that has text row labels and numeric elements and a string with a default value of `""`. If the string is the empty string, return the sum of all of the elements in the frame. Otherwise, return the sum of the rows whose labels contain the string as a substring.