Notes

Part 1
Only program the rules. The exceptions can be ignored to keep things simple. However, feel free to program them if you want the challenge.

Regarding comparisons, if you have the last two characters of a string, say `word[-2:]`, compared to two or more things, you should use the `in` operator instead of the == operator repeatedly. So, for example, you can say `if word[-2:] in ["sh", "ch"]: ...` or the like. Notice that this takes care of two cases in one. (Watch the video for the lecture if you need more help.)

Any interaction with the user including input and output should be done in the main script.

Part 2
I recorded a demo showing what the script should be doing. It is in the Assignments section of Brightspace.

If your second file turns out empty, consider that once the file has been read, the pointer is at the end. You need to reset the pointer to the beginning in order to read the file again. One option is to close the original prices file once you are done with the first run and reopen it before the second run. (The header line will be read again since the file goes back to the beginning when it is reopened. Read that line using readline() and reprocess it or discard it.) Another option is to use the tell() and seek() methods explained in part 3.

Part 3
I recorded a demo showing what the script should be doing. It is in the Assignments section of Brightspace.

As requested in Task 3.1.d, remove any spaces from each label using strip(): some of the column labels have spaces that might interfere with proper recognition.

Note: by mistake I labeled two parts Task 3.2. The one referenced below is the first one.
For identifying the columns in Task 3.2, one strategy is to assume that the column names won't change. In that case the following code is useful to get the correct indices without hard coding the numbers themselves. Verify that the column names correspond to those in the csv file.

```
# Key columns identified.
district_index = col_labels.index("district")
given_name_index = col_labels.index("given name")
family_name_index = col_labels.index("family name")
phone_index = col_labels.index("phone")
```