Dead Letter

Purpose

To test the ability to perform common string operations

Directions

Write a function called deadLetter with the signature `deadLetter(list, dead_letter)`, where "list" is a list of strings. The function should return a COPY of `list` that has any string containing `dead_letter` removed.

In the main part of your program, prompt the user for a comma-separated list of strings and a letter. Convert the comma-separated strings into a python list and call your deadLetter function. Print out the list that your function returns.
You can assume that the user will follow all directions precisely.

Notes

- If the dead letter is, e.g. n, you only need to remove strings that have n, not N.
- The strings that wind up in the list returned by deadLetter should not have any leading or trailing whitespace

Examples

```
Enter a comma-separated list of strings: Hello,hi,hiya,Hola,goodbye
Enter the dead letter: h
The new list is: ['Hello', 'Hola', 'goodbye']

Enter a comma-separated list of strings: hello world, Michelle, test,
coffee, snorkel
Enter the dead letter: l
The new list is: ['test', 'coffee']
```

Function test

`print(deadLetter(['one', 'two', 'three', 'four'], 't'])`
should display `['one', 'four']`