Table of Contents

Bleep

tl;dr

Implement a program that censors messages that contain words that appear on a list of supplied "banned words."

```
$ python bleep.py banned.txt
What message would you like to censor?
What the heck
What the ****
$ python bleep.py banned.txt
What message would you like to censor?
gosh darn it
**** **** it
```

Getting Started

Here's how to download this problem's "distribution code" (i.e., starter code) into your own CS50 IDE. Log into CS50 IDE (https://ide.cs50.io/) and then, in a terminal window, execute each of the below.

1. Execute `cd ~/pset6` to ensure that you're in `~/pset6`. If you haven't yet created that directory, create it now (remember how?).
2. Execute `wget https://cdn.cs50.net/2018/fall/psets/6/bleep/bleep.zip (https://cdn.cs50.net/2018/fall/psets/6/bleep/bleep.zip)` to download a (compressed) ZIP file with this problem's distribution.
3. Execute `unzip bleep.zip` to uncompress that file.
4. Execute `rm bleep.zip` followed by `yes` or `y` to delete that ZIP file.
5. Execute `ls`. You should see a directory called `bleep`, which was inside of that ZIP file.
6. Execute `cd bleep` to change into that directory.
7. Execute `ls`. You should see this problem's distribution, including `bleep.py` and `banned.txt`.

Understanding

This program defines only one function, `main`, which gets called per the file's last line. Within `main` … ugh, looks like that's just a big `TODO`!

Specification

Complete the implementation of `bleep.py` in such a way that it:

- Accepts as its sole command-line argument the name (or path) of a dictionary of banned words (i.e., text file).
- Opens and reads from that file the list of words stored therein, one per line, and stores each in a Python data structure for later access. While a Python `list` will work well for this, you may also find a `set` (https://docs.python.org/3/tutorial/datastructures.html#sets) useful here.
- If no command line argument (e.g., `banned.txt`) is provided, be sure to have your program exit (https://docs.python.org/3/library/sys.html#sys.exit) with a status code of `1`.
- You may assume that any text files the staff tests with will have one word per line (each line terminated with a `\n`), and any alphabetic characters in those words will be lowercase.
- Prompts the user to provide a message.
- Tokenizes that message into its individual component words, using the `split` (https://docs.python.org/3/library/stdtypes.html#str.split) method on the provided string, and then iterates over the `list` of

"tokens" (words) that is returned by calling `split`, checking to see whether any of the tokens match, case-insensitively, any of the words in the banned words list.

- Prints back the message that the user provided, except if the message contained any banned words, each of its characters is replaced by a `*`.
- For example, `gosh` should be replaced with four `*` characters, while `fudge` should be replaced with five.
- You should not censor words that merely contain a banned word as a substring. For example, if **bar** is a banned word in the provided list, then none of ***bar**ns* nor *crow**bar*** nor *wheel**bar**row* should be censored.
- You explicitly **do not** need to support input strings that contain punctuation marks. You may assume we will only test your input where each word is only separated by whitespace.

Usage

Your program should behave per the examples below. Assume that the underlined text is what some user has typed.

```
$ python bleep.py
Usage: python bleep.py dictionary
```

```
$ python bleep.py list1.txt list2.txt list3.txt
Usage: python bleep.py dictionary
```

```
$ python bleep.py banned.txt
What message would you like to censor?
hello world
hello world
```

```
$ python bleep.py banned.txt
What message would you like to censor?
what the heck
what the ****
```

```
$ python bleep.py banned.txt
What message would you like to censor?
gosh darn it
**** **** it
```

Testing

Correctness

```
check50 cs50/problems/2019/x/bleep
```

Style

```
style50 bleep.py
```

Staff’s Solution

```
~cs50/2019/x/pset6/bleep
```

How to Submit

Execute the below from within your `~/pset6/bleep` directory, logging in with your GitHub username and password when prompted. For security, you'll see asterisks (`*`) instead of the actual characters in your password.

```
submit50 cs50/problems/2019/x/bleep
```

You can then go to https://cs50.me/cs50x (https://cs50.me/cs50x) to view your current scores!

Hints

- Be sure to test with different banned words lists than the one provided by default—we will!
- When independently researching how to do things on this problem (which is indeed part of the expectation, as you grow in your comfort with programming overall!), be sure your Google searches and the like include "Python 3" in them, and not just "Python", lest you get code examples written in an earlier version of Python!
- Odds are you'll find `str.split` (https://docs.python.org/3/library/stdtypes.html#str.split) of interest.
- Odds are you'll find `str.lower` (https://docs.python.org/3/library/stdtypes.html#str.lower) of interest.

- Odds are you’ll find `str.strip` (https://docs.python.org/3/library/stdtypes.html#str.strip) of interest, to chomp off any trailing newlines that may be attached to words on your "banned words" list.