

Programming Project #7

This assignment is worth 50 points and must be **completed and turned in before 11:59 on Tuesday, March 16, 2010**. The non-Monday date is due to Spring Break.

Assignment Overview

The goal of this assignment is for you to become more proficient with file I/O, using functions, and learning to use the dictionary data structure effectively. Dictionaries are very useful data structures when you can map something (e.g. an English word), to something else (e.g. equivalent Spanish words). For this assignment, you will synthesize these concepts together by creating an English-Spanish vocabulary program.

Problem Statement

If you've ever taken a foreign language class, you may have studied for tests by creating flash cards to memorize how a given word in a particular language translates into a different language. For this project, you are to create a vocabulary quiz program that requests the name of a vocabulary file in the current directory, quizzes the user on a subset of the words in the vocabulary file, and then gives the user the option to write out any missed words to a new vocabulary file. While the word lists that we are using are English to Spanish, the program could easily be applied more generally to other languages, giving you a practical replacement to flash cards!

Program Outline

1. Check if there are any vocabulary files in the directory, which by convention are files that end in .txt. Print the list of files found. If there aren't files ending in .txt, display an error message and quit.
2. Have the user select which vocabulary file he or she would like to use. Error check to make sure that they select a valid file (i.e., one of the files listed in step 1).
3. Store the contents of the vocabulary file into a dictionary data structure.
4. Print the number of English words in the dictionary.
5. Prompt the user for the number of words he or she would like to be quizzed on. Error check to make sure the number is valid.
6. Quiz the user by using a randomly generated list of English words from the dictionary.
7. If the user missed any questions, give the option to output the missed words to a new word list file.

Example Output

Note: User input is in **bold**.

```
> python proj07.py
```

```
Welcome to the vocabulary quiz program.  Select one of the following word lists:
    verbs.txt
    places.txt
```

```
Please make your selection: verbs.txt
```

```
10 entries found.  How many words would you like to be quizzed on? 3
```

```

English word: to study
Enter 1 equivalent Spanish word(s).
Word [1]: estudiar
Correct!

---

English word: to search
Enter 2 equivalent Spanish word(s).
Word [1]: buscar
Word [2]: mirar
Correct!

---

English word: to be
Enter 2 equivalent Spanish word(s).
Word [1]: ser
Word [2]: ???
Incorrect.

---

2 out of 3 correct.
Enter an output file (or press enter to quit): wrong.txt

```

Program Requirements

You project will meet the following requirements:

1. Create a function that takes a filename as input and returns a dictionary of vocabulary words. What should be the keys? Since a single English word can be mapped to multiple Spanish words, what data structure can you use for each item that is mapped to a key?
2. Create a function that takes a filename and dictionary as input. It does not have to return anything, but inside the function you should export the dictionary into the filename in the same format as the example word lists.
3. Create another function other than those previously described.
4. You must do some error checking for the project. Specifically, you should check if there are any available word lists in the directory. You should also make sure that the user inputs a valid filename. Finally, when the user is asked how many words to be quizzed on, your program should make sure that it is within the possible range of numbers (i.e. between one and the number of English words in the dictionary).
5. You must use a dictionary for the English-Spanish mapping and use it appropriately.

Notes and Hints:

- Start early on this project so that you know how much work is involved.
- Since the dictionary is so integral to the program, make sure that you can get it working first before you implement any part of the actual user quizzing.
- The `glob.glob(regex)` function is useful for obtaining a list of files that meet a given criteria. You must import the `glob` package to use it.
- To randomly generate a quiz list, check out the Python documentation's [page on random numbers](#), specifically the `random.sample()` function. Don't forget to import the `random` package.

Deliverables

proj07.py -- your source code solution (remember to include your section, the date, project number and comments).

1. Please be sure to use the specified file name, i.e. "proj07.py"
2. Save a copy of your file in your CS account disk space (H drive on CS computers).
3. Electronically submit a copy of the file.