

## Problem 3 - Valid Words

[Bookmark this page](#)

### Problem 3 - Valid Words

10.0 points possible (graded)

At this point, we have written code to generate a random hand and display that hand to the user. We can also ask the user for a word (Python's `input`) and score the word (using your `getWordScore`). However, at this point we have not written any code to verify that a word given by a player obeys the rules of the game. A *valid* word is in the word list; **and** it is composed entirely of letters from the current hand. Implement the `isValidWord` function.

**Testing:** Make sure the `test_isValidWord` tests pass. In addition, you will want to test your implementation by calling it multiple times on the same hand - what should the correct behavior be? Additionally, the empty string ( `''` ) is not a valid word - if you code this function correctly, you shouldn't need an additional check for this condition.

Fill in the code for `isValidWord` in `ps4a.py` and be sure you've passed the appropriate tests in `test_ps4a.py` before pasting your function definition here.

```
1 def isValidWord(word, hand, wordList):
2     """
3     Returns True if word is in the wordList and is entirely
4     composed of letters in the hand. Otherwise, returns False.
5
6     Does not mutate hand or wordList.
7
8     word: string
9     hand: dictionary (string -> int)
10    wordList: list of lowercase strings
11    """
12    # TO DO ... <-- Remove this comment when you code this function
13
```

Press ESC then TAB or click outside of the code editor to exit

---