**Hangman Part 2 - The Game / Complex Hangman**

15.0/15.0 points (graded)

Now you will implement the function hangman, which takes one parameter - the secretWord the user is to guess. This starts up an interactive game of Hangman between the user and the computer. Be sure you take advantage of the three helper functions, isWordGuessed, getGuessedWord, and getAvailableLetters, that you've defined in the previous part.

**Hints:**

* You should start by noticing where we're using the provided functions (at the top of ps3\_hangman.py) to load the words and pick a random one. Note that the functions loadWords and chooseWord should only be used on your local machine, not in the tutor. When you enter in your solution in the tutor, you only need to give your hangman function.
* Consider using lower() to convert user input to lower case. For example:

guess = 'A'  
guessInLowerCase = guess.lower()

* Consider writing additional helper functions if you need them!
* There are four important pieces of information you may wish to store:
  1. secretWord: The word to guess.
  2. lettersGuessed: The letters that have been guessed so far.
  3. mistakesMade: The number of incorrect guesses made so far.
  4. availableLetters: The letters that may still be guessed. Every time a player guesses a letter, the guessed letter must be removed from availableLetters (and if they guess a letter that is not in availableLetters, you should print a message telling them they've already guessed that - so try again!).

**Sample Output**

The output of a winning game should look like this...

Loading word list from file...

55900 words loaded.

Welcome to the game, Hangman!

I am thinking of a word that is 4 letters long.

-------------

You have 8 guesses left.

Available letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: a

Good guess: \_ a\_ \_

------------

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: a

Oops! You've already guessed that letter: \_ a\_ \_

------------

You have 8 guesses left.

Available letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: s

Oops! That letter is not in my word: \_ a\_ \_

------------

You have 7 guesses left.

Available letters: bcdefghijklmnopqrtuvwxyz

Please guess a letter: t

Good guess: ta\_ t

------------

You have 7 guesses left.

Available letters: bcdefghijklmnopqruvwxyz

Please guess a letter: r

Oops! That letter is not in my word: ta\_ t

------------

You have 6 guesses left.

Available letters: bcdefghijklmnopquvwxyz

Please guess a letter: m

Oops! That letter is not in my word: ta\_ t

------------

You have 5 guesses left.

Available letters: bcdefghijklnopquvwxyz

Please guess a letter: c

Good guess: tact

------------

Congratulations, you won!

And the output of a losing game should look like this...

Loading word list from file...

55900 words loaded.

Welcome to the game Hangman!

I am thinking of a word that is 4 letters long.

-----------

You have 8 guesses left.

Available Letters: abcdefghijklmnopqrstuvwxyz

Please guess a letter: a

Oops! That letter is not in my word: \_ \_ \_ \_

-----------

You have 7 guesses left.

Available Letters: bcdefghijklmnopqrstuvwxyz

Please guess a letter: b

Oops! That letter is not in my word: \_ \_ \_ \_

-----------

You have 6 guesses left.

Available Letters: cdefghijklmnopqrstuvwxyz

Please guess a letter: c

Oops! That letter is not in my word: \_ \_ \_ \_

-----------

You have 5 guesses left.

Available Letters: defghijklmnopqrstuvwxyz

Please guess a letter: d

Oops! That letter is not in my word: \_ \_ \_ \_

-----------

You have 4 guesses left.

Available Letters: efghijklmnopqrstuvwxyz

Please guess a letter: e

Good guess: e\_ \_ e

-----------

You have 4 guesses left.

Available Letters: fghijklmnopqrstuvwxyz

Please guess a letter: f

Oops! That letter is not in my word: e\_ \_ e

-----------

You have 3 guesses left.

Available Letters: ghijklmnopqrstuvwxyz

Please guess a letter: g

Oops! That letter is not in my word: e\_ \_ e

-----------

You have 2 guesses left.

Available Letters: hijklmnopqrstuvwxyz

Please guess a letter: h

Oops! That letter is not in my word: e\_ \_ e

-----------

You have 1 guesses left.

Available Letters: ijklmnopqrstuvwxyz

Please guess a letter: i

Oops! That letter is not in my word: e\_ \_ e

-----------

Sorry, you ran out of guesses. The word was else.

Note that if you choose to use the helper functions isWordGuessed, getGuessedWord, or getAvailableLetters, you do not need to paste your definitions in the box. We have supplied our implementations of these functions for your use in this part of the problem. If you use additional helper functions, you will need to paste those definitions here.

Your function should include calls to input() to get the user's guess.

def hangman(secretWord):  
 '''  
 secretWord: string, the secret word to guess.  
 Starts up an interactive game of Hangman.  
 \* At the start of the game, let the user know how many   
 letters the secretWord contains.  
 \* Ask the user to supply one guess (i.e. letter) per round.  
 \* The user should receive feedback immediately after each guess   
 about whether their guess appears in the computers word.  
 \* After each round, you should also display to the user the   
 partially guessed word so far, as well as letters that the   
 user has not yet guessed.  
 Follows the other limitations detailed in the problem write-up.  
 '''  
 # add your code here