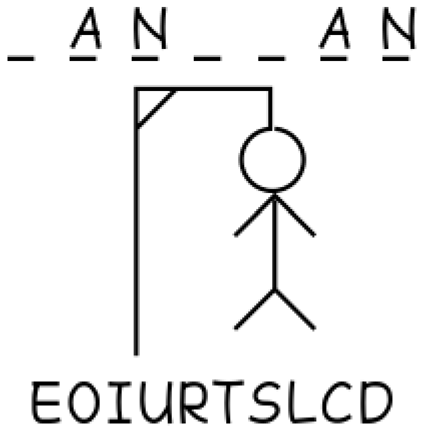
Hangman (game)



#Step 1

word_list = ["aardvark", "baboon", "camel"]

- TODO-1 - Randomly choose a word from the word_list and assign it to a variable called chosen_word.
- TODO-2 - Ask the user to guess a letter and assign their answer to a variable called guess. Make guess lowercase.
- TODO-3 - Check if the letter the user guessed (guess) is one of the letters in the chosen_word.

#Step 2

- TODO-1: - Create an empty List called display. For each letter in the chosen_word, add a "_" to 'display'. So, if the chosen_word was "apple", display should be ["_", "_", "_", "_", "_"] with 5 "_" representing each letter to guess.
- TODO-2: - Loop through each position in the chosen_word; If the letter at that position matches 'guess' then reveal that letter in the display at that position.
  e.g. If the user guessed "p" and the chosen word was "apple", then display should be ["_", "p", "p", "_", "_"].

- TODO-3: - Print 'display' and you should see the guessed letter in the correct position and every other letter replace with "_".

#Step 3

TODO-1: - Use a while loop to let the user guess again. The loop should only stop once the user has guessed all the letters in the chosen_word and 'display' has no more blanks ("_"). Then you can tell the user they've won.

#Step 4

- TODO-1: Import random module and new variable stages. The value of stages variable is given below

```
stages = ['''
  +---+
  |   |
  O   |
 /|\  |
 / \  |
      |
=========
''', '''
  +---+
  |   |
  O   |
 /|\  |
 /    |
      |
=========
''', '''
  +---+
  |   |
  O   |
 /|\  |
      |
      |
=========
''', '''
```

```
  +---+
  |   |
  O   |
 /|   |
      |
      |
=========''', '''
  +---+
  |   |
  O   |
  |   |
      |
      |
=========
''', '''
  +---+
  |   |
  O   |
      |
      |
      |
=========
''', '''
  +---+
  |   |
      |
      |
      |
      |
=========
''']
```

- TODO-2: - Create a variable called 'lives' to keep track of the number of lives left.
  Set 'lives' to equal 6.
- TODO-3: - If guess is not a letter in the chosen_word, then reduce 'lives' by 1.
  If lives goes down to 0 then the game should stop and it should print "You lose."
  Join all the elements in the list and turn it into a String.
  Check if user has got all letters.

- TODO-4: print the ASCII art from 'stages' that corresponds to the current number of 'lives' the user has remaining.

#Step 5

- TODO-1: - If the user has entered a letter they've already guessed, print the letter and let them know.
  Check guessed letter
  Check if user is wrong.
- TODO-2: - If the letter is not in the chosen_word, print out the letter and let them know it's not in the word.