



Lesson 6

Python Fundamental

Revisit

Text File Handling: Read, Create

Variable Scope: with block

Define a Function

Project: Hangman Game



Technique We've Learned

Random Library

Decision Making: if-elif-else

Logical Operators (==, !=, <, <=, >, >=)

Boolean Operators (and, or, not)

Boolean Values (True / False)

range(start, stop, step) function

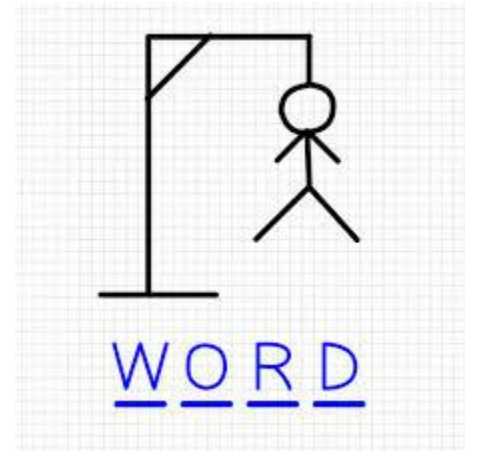
Control Structure: for loop, while loop

Text File Handling: create, write, append, read

Define a Function

Hangman ... continue

- Create a Hangman game:
 - Computer to pick a word from LIST
 - Give the player a hint
 - Player to guess the word one letter at a time
 - Allowed a limited number of incorrect guesses



<https://poki.com/en/g/hangman>

What We Need?



Define a list of word



Set number of attempts



Start a while loop for user to input a character until no attempts allowed and not bingo



If the letter in the word, print it out. If all letters match the word, print a “Congratulations” message.



Otherwise, deduct attempts. If attempts reaching zero, game over

1. Hangman – Pick a Word



Use data type **list** to store list of word



Use random library **random.choice(listOfWord)** to pick one word from the list

2. Hangman – Setup Variables

Variables we need:

Usage	Variable Name	Data Type	Initial Value
The word pick randomly	answer	String	Return a random word
User's guess	guesses	List	[]
Number of attempts	attempts	Integer	5 (mx 5 attempts)
Boolean of guess match status	guess_matches	Boolean	False

3. Hangman – Main Loop

Create	Create a while loop to execute for 2 conditions (using and): attempts > 0 and not guess_matches
Ask	Ask user to guess one character – make sure user only input 1 character
Append	Append the character to the list variable guesses
Subtract	Subtract a chance from attempts

3.1 Hangman – Main Loop Continue

- Ask user input ONE character
- Append the letter to user guess variable
- Deduct attempts
- Loop through each character of the answer and compare with input letter
- Generate the prompt. Example, '_pp__' (for the prompt of apple)



3.2 Hangman – Main Loop Continue

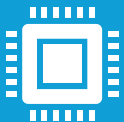


Check if the prompt exactly match variable answer.

Assign True to variable **guess_matches** and print a congratulations message then exit the game.



Otherwise, show the prompt



At the end, if guess_matches is False after attempts reach the maximum number of attempts, print failure and game over message.

Exercise

No exercise for this week

If you didn't finish hangman, finish it at home