SKILLS DEVELOPMENT LAB ASSIGNMENT 2 – HANGMAN

BY

VAIBHAV JAISWAL

PRN: 17070122071

FINAL YEAR, 2017-2021

Problem Statement

Build the Hangman Game using Python

Logic

- 1. Starts the game using play_hangman function.
- 2. Selects a word at random from words.py file
- 3. Hides the characters in the word randomly and replaces them with "_"
- 4. Displays the hangman states and asks the user to guess a character
- 5. On a correct guess, reveals the hidden states of those character and updates the list of hidden characters
- 6. On a wrong guess, reduces the number of tries by 1 and based on that changes the hangman states
- 7. If all the hidden words are guessed correctly then sets the guessed_word to True and displays the win message
- 8. If all the tries are gone I.e., tries = 0 and word is not yet guessed then displays the loser message as well as the loser state of hangman
- 9. End the game
- 10. Asks the user if they want to play another game or not.

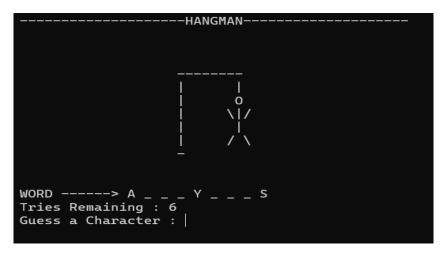
Code

```
\\|/
# head, torso, both arms, and one leg
         \\|/
         \\|
# head and torso
```

```
# initial empty state
                11 11 11
    def get_word(self) -> str:
        returns a random word at random
        Returns:
           str: random word
        return str(choice(word_list).upper())
   def display_state(self,tries : int,word_completion_state : list) -> None:
        This Function printes the current state of hangman with
        some preset messages as well as print the remaining tries
        Args:
            tries ([int]): Number of tries remaining
            word_completion_state ([list]): The current state of the hidden character
                                            in the word
        print(f"{'-' * 20}HANGMAN{ '-' * 20}\n\n")
        print(self.hangman_states[-(tries+1)] + "\n")
        print(f"WORD -----> {' '.join(word_completion_state)}")
        print(f"Tries Remaining : {tries}")
    def play_hangman(self) -> None:
        Initiates a game of hangman
        Use has 6 tries to complete the word selected at random
        after tries run out or word is guessed , display a win
        or loss message and exit the game
        tries=6
        current_word=self.get_word()
        guessed_word = False
        word_hidden_states = [current_word[indx] for indx in sample(range(0, len(current_w
ord)-1), randint(1, len(current_word)-2))]
        word_completion_state = [letter if letter not in word_hidden_states else "_" for l
etter in current_word]
       while tries > 0 and not guessed_word:
```

```
os.system('cls' if os.name == 'nt' else 'clear') ## Clear the terminal for new
 lines to be printed
            self.display state(tries,word completion state)
            guessed char=str(input("Guess a Character : ")).upper()
            if guessed char in word hidden states :
                print("\nCorrect Guess !!!!!! Updating.....")
                for indx,_ in enumerate(word_completion_state) :
                    if guessed_char == current_word[indx]:
                        word completion state[indx]=guessed char
               word_hidden_states = [char for char in word_hidden_states if char != guess
ed_char]
                guessed_word = False if "_" in word_completion_state else True
                sleep(1)
            else :
               print("\nIncorrect Guess!!! Updating!!!!!")
                sleep(1)
                tries=tries-1
       if tries == 0 and not guessed_word:
            os.system('cls' if os.name == 'nt' else 'clear') ## Clear the terminal for new
 lines to be printed
           print(f"{'-' * 20}HANGMAN{ '-' * 20}\n\n")
            print(self.hangman_states[-1] + "\n")
            print(f"No Tries Remaining , YOU LOST !!!!!")
            print(f"CORRECT WORD was ----> {current_word}")
            print(f"GAME OVER")
       if guessed_word:
            os.system('cls' if os.name == 'nt' else 'clear') ## Clear the terminal for new
 lines to be printed
            print(f"{'-' * 20}HANGMAN{ '-' * 20}\n\n")
            print(self.hangman_states[-tries] + "\n")
            print(f"YOU GUESSED THE WORD CORRECTLY !!!")
            print(f"WORD was ----> {current_word}")
            print(f"Congratulations You win")
if __name__ == "__main__":
    hangman=Hangman()
    continue_choice='y'
   while continue_choice == 'y':
       hangman.play_hangman()
       continue_choice=str(input("Do you want to play another game (Y/n) : "))
       os.system('cls' if os.name == 'nt' else 'clear') ## Clear the terminal for new lin
es to be printed
```

Pass Condition







```
YOU GUESSED THE WORD CORRECTLY !!!

WORD was ----> ANONYMOUS
Congratulations You win
Do you want to play another game (Y/n) :
```

FAIL

```
0
WORD -----> _ _ _ Y M _ U S
Tries Remaining : 6
Guess a Character : l
Incorrect Guess!!! Updating!!!!!!
 ----HANGMAN-----
WORD -----> _ _ _ Y M _ U S
Tries Remaining : 5
Guess a Character : p
Incorrect Guess!!! Updating!!!!!!
   -----HANGMAN------
WORD -----> _ _ _ Y M _ U S Tries Remaining : 3
Guess a Character : c
Incorrect Guess!!! Updating!!!!!!
```

-----HANGMAN-----



```
No Tries Remaining , YOU LOST !!!!!

CORRECT WORD was ----> ANONYMOUS

GAME OVER

Do you want to play another game (Y/n) :
```