I'm still having a hard time to code program to be as I wished.

What I can do

- Closing/ened the application when user enters '5' or lose the game / enters 'no'/'n' to end the application.
- The application can check and display match value.
- The application displays stage (index) of the hangman list correctly.
- The application will end the game loop and prompt to replay correctly.

What I am still having issues:

- The application doesn't end the game loop when user enters all correct values & wins.
- I think, I'm still not quite confident to code validation input function (integer / character )
- I cannot figure out how to apply OOP concept into this program
- I cannot make program rejecting duplicate enter letter

Here are the details of my files

1. Words.py for 3-4 collections of words in 'String' data type, inside lists
    - 1.1 A list of words in '4-5 letters' Possible at least 20 words
    - 1.2 A list of words in '6 letters' Possible at least 20 words
    - 1.3 A list of words in '7 letters' Possible at least 20 words
    - 1.4 A list of words in '4-7 or more letters' Possible at least 20 words.

I ended up more than 100 words in each list.

2. show_images.py for image of hangman in each state; 6 {7-1} indexes, inside a list
    - Showing with a vertical pole with rope
    - Showing a hanging pole with rope, & a head.
    - Showing a hanging pole with rope, head, and 1 arm
    - Showing a hanging pole with rope, head, 2 arms
    - Showing a hanging pole with rope, head, 2 arms, body part
        - Showing a hanging pole with rope, head, 2 arms, body part, 1 leg.
        - Showing a hanging pole with rope, head, 2 arms, body part, 2 legs
    - I finally adding logo for starting of the game
3. all_methods.py for
    - method to randomly pick words after got number input from user.
    - showing '_' (underscore) to match with guessing word that got assigned from random in user specific number of letters list.
    - checking input value/ verify if enter value is a letter, not an integer
    - make sure input got in and set to 'lowercase' after verify it's a letter, not an integer.
    - if it's an integer, display an error msg. But won't count as a guessing letter.
    - comparing input value with guessing word.
    - if input value matches within guessing word, then replace a letter on '_' in that specific index of the word.
    - if input value doesn't match within guessing word, show hangman imager from index 0, 1,2,3, 4,5, 6

        - This will be within while loop.
        - minus user_life (-1) each time user guessed the wrong letter.
    - checking if user win/loss
    - if user_life is equal 0, guess word still word still have '_', user loss.
    - if user_life is more than 0, guess guess word no longer have '_' showing, then, user win.
4. The app.py will have starting set values, and will import those files from package.
    - in this file will also have input asking for how many letters in a word user would like to play
    - With in while loop, the game will continue until user win/loss. Or if user enter 'quit' word.
    - After showing user win/loss, should ask user to replay the game again, or quit.