7_Loops_1_solution

September 19, 2019

1 Hangman

The idea is simple, the implementation is a little harder. Make a game of hangman that you and a friend can play!

New function: input

[3]:
```
# example of using input
name = input('Type in your name and hit enter: ')
print('Hi, ' + name + '!')
```

```
Type in your name and hit enter: Zena
Hi, Zena!
```

[2]:
```
# Learn more about input
?input # this works for me in Jupyter Notebook, if it doesn't for you, try:
help(input)
```

```
Help on method raw_input in module ipykernel.kernelbase:

raw_input(prompt='') method of ipykernel.ipkernel.IPythonKernel instance
    Forward raw_input to frontends

    Raises
    ------
    StdinNotImplentedError if active frontend doesn't support stdin.
```

1.0.1 Approach:

1) Write down how a game of hangman works. What is the first step? What is the next step?

- How does the game start?
- How does the game end?
- Are there certain parts where you have to make different decisions based on the input?
- Are there parts that get repeated multiple times?

2) Write pseudocode based on the steps you wrote in (1).

3) Write up your code below.

4) Try it out on a friend!

Hints: - Make sure you have a variable you set that is the word your friend is trying to guess. - Ex: word = "python" – you can manually change this word each time you want to play another game of hangman - You have to tell your friend how long the word is - When your friend guesses a correct letter, you have to add it in the correct place in the word - When your friend guesses an incorrect letter, you have to let them know - The game is over when all of the letters are guessed (or you can set a maximum number of turns your friend can take to guess the word)

[2]:
```
# Set word
word = 'python'

# Initialize list to store guesses
guesses = list()

# Make a list to store letters in the word
empty_word = list('_'*len(word))

# Print out blanks for word
print(empty_word)

# Set number of guesses
n_turns = 10
print('\nYou have ' + str(n_turns) + ' guesses.')

# While there are still guesses left and all letters haven't been guessed
while(n_turns >= 0):
    # If you're out of guesses
    if(n_turns == 0):
        print('Sorry, you lose. The word was ' + word + '.')
        break
    # Have the user guess a letter
    guess = input('Guess a letter: ')
    # Check to see if the user input is a letter
    if not guess.isalpha() or not len(guess) == 1:
        print(guess, 'is not a single letter. Please input a letter from A to Z.
 ↪')
        continue
    # Check to see if the user has already guessed that letter
    if guess in guesses:
        print('You have already guessed '+ guess +
              '!\nYou have guessed: ' + str(guesses) +
              '.\nGuess another letter.')
        continue
    # Add the guess to the previous guesses
    guesses.append(guess)
```

```
    # If the guess is in the word (upper or lower case guess)
    if guess in word or guess.lower() in word:
        # Add letter to empty word and print it out
        for i,letter in enumerate(word):
            if letter == guess or letter == guess.lower():
                empty_word[i] = guess.lower()
        print(empty_word)
        if '_' in empty_word:
            continue
        # If the user has guessed the whole word, end the game
        else:
            print('You guessed the word (' + word + '). Nice job!')
            break
    # If the guess is wrong, subtract a guess and start over
    else:
        n_turns -= 1
        print('Sorry, the letter', guess,
              'is not in the word.\nYou have guessed: ' + str(guesses) +
             '.\nYou have ' + str(n_turns) + ' guess(es) left.')
```

```
['_', '_', '_', '_']

You have 10 guesses.
Guess a letter: e
Sorry, the letter e is not in the word.
You have guessed: ['e'].
You have 9 guess(es) left.
Guess a letter: a
['a', '_', '_', 'a']
Guess a letter: t
Sorry, the letter t is not in the word.
You have guessed: ['e', 'a', 't'].
You have 8 guess(es) left.
Guess a letter: a
You have already guessed a!
You have guessed: ['e', 'a', 't'].
Guess another letter.
Guess a letter: n
['a', 'n', 'n', 'a']
You guessed the word (anna). Nice job!
```

[ ]:

[ ]: