JLUFE Spring 2021 (Feb-July)

Final Assignment Report

JILIN UNIVERSITY OF FINANCE AND ECONOMICS

Department of College of Managment Science and Information Engineering

BSc in Information management and information system

(2021)

Final Assignment: Part 02

21/06/2021

MODULE: Data Mining

Submitted by: Jony(郑永健) 0314021805410 (1854)
QQ: 2464304928
Github ID: zhengyongjian123

Instructions:

1. I have added tips and required learning resources for each question, which helps you to solve the exercise.
2. Finish the assignment in **group of two students (Any group find copying/sharing from another group or internet will get '0' points!!!**)
3. **Accept this assignment** from the **Github Clasroom link (https://classroom.github.com/a/yFXO50A4)** This will create private repository of the assignment in your Github account.
4. In your repository `Clone -> Download ZIP` in your computer.
5. Change your: **Major**, **Name**, **Student number**, **Class number**, **QQ number** and **GitHub ID**
6. Once you finish the Assignment **convert your .ipynb file into PDF (https://github.com/milaan9/91_Python_Tips/blob/main/000_Convert_Jupyter_Notebook_to_PDF.ipyn** (both .pynb and .pdf file will be required!)
7. Create Folder name "**Solution**" and copy your 3 files:
   A. Your Jupyter Notebook file (**.ipynb**).
   B. Your PDF converted file (**.pdf**).
   C. **.zip** file containing both .ipynb and .pdf files and name your .zip file as your student number and name. For example: **0318021907632-3086215267 Milan(米兰)-Nina(妮娜).zip**
8. Finally, in your repository `Add files -> upload files` upload the "**Solution**" folder and `Commit changes` .

Python Assignment 02

Question: Hangman Game

Write a python program to create a Hangman game.

About Game: Going back to our old school days, some of the pen-paper games were always a top for our leisure time. In Hangman user has to guess words according to the guesses determined and as soon as they lost all their wrong guesses, they were hanged (not really, but on paper 😉). In the game of Hangman, the player only has 7 incorrect guesses (head, body, 2 legs, and 2 arms, hang) before they lose the game.

Structure:

1. In Part 1, you will require to load a random word from a **dictionary (https://github.com/milaan9/92_Python_Assignments/blob/main/sowpods.txt)**.
2. In Part 2, you will require the logic for guessing the letter and displaying that information to the user.

After completing part 1 and part 2 you will need to add the following features:

Features:

- Only let the user guess 7 times, and tell the user how many guesses they have left. Example: "You have 6 guesses left!"
- No restriction in uppercase and lowercase letters.
  - Example: user can guess "a" and it will be equal to "A" or vice-versa.
- If user guesses a numbers or a special characters, don't penalize them - ask them again to choose only letter.
  - Example: user guess "9" or "?" then ask user again to choose a letter.
- If the guess letter appear more than one time in the word display it.
  - Example: Word is "Apple" and user guess the word 'p' so --> __ P P __ __
- Keep track of the letters the user guessed incorrectly. If the user guesses a letter they already guessed, don't penalize them - let them guess again.
- Display some picture art for the Hangman. This is challenging - do the other parts of the exercise first!
- When the player wins or loses, let them start a new game.

Expected/Similar Output:

```
**********
Welcome to Hangman!
**********
Guess one letter at a time
Game is not case sensitive


_ _ _ _ _ _ _ _

What is your guess?: a

_  A  _ _  A  _ _ _

What is your guess?: 9
Please chose just a letter: e
e is not in this word!
   ____
  |    |

You have 6 guesses left!
your previous wrong guesses: ['E']

_  A  _ _  A  _ _ _

What is your guess?: e
You have already guessed e!

_  A  _ _  A  _ _ _

What is your guess?: h
h is not in this word!
   ____
  |    |
  |    O
  |    |

You have 5 guesses left!
your previous wrong guesses: ['E', 'H']

_  A  _ _  A  _ _ _

What is your guess?: d
d is not in this word!
   ____
  |    |
  |    O
  |   /|

You have 4 guesses left!
your previous wrong guesses: ['E', 'H', 'D']

_  A  _ _  A  _ _ _
```

```
What is your guess?: b
b is not in this word!

  ____
 |    |
 |    0
 |   /|\

You have 3 guesses left!
your previous wrong guesses: ['E', 'H', 'D', 'B']

__  A  __ __  A  __ __ __

What is your guess?: k

K  A  __ __  A  __ __ __

What is your guess?: r

K  A  __ __  A  R __ __

What is your guess?: t
t is not in this word!


  ____
 |    |
 |    0
 |   /|\
 |   /

You have 2 guesses left!
your previous wrong guesses: ['E', 'H', 'D', 'B', 'T']

K  A  __ __  A  R __ __

What is your guess?: l
l is not in this word!


  ____
 |    |
 |    0
 |   /|\
 |   / \

You have 1 guesses left!
your previous wrong guesses: ['E', 'H', 'D', 'B', 'T', 'L']

K  A  __ __  A  R __ __

What is your guess?: p
p is not in this word!


  ____
```

```
   |    |
   |    |
   |    O
   |   /|\
   |   / \
__|__
```

```
You have 0 guesses left!
You lose!
your previous wrong guesses: ['E', 'H', 'D', 'B', 'T', 'L', 'P']

The word was  ['K', 'A', 'N', 'G', 'A', 'R', 'O', 'O']

Would you like to play again? [y|n]: n
```

In [1]:

```
# Solution:

import random
import linecache
import string
# you can use more libraries if you want
def print1(strrr):
    for i in range(len(strrr)):
        print(strrr[i],end=' ')
    print('')
    return
def th(str1,loc,rep):
    string=list(str1)
    string[loc]=rep
    z=''.join(string)
    return z
def picture(x):
    if x==6:
        print('    ____ \n   |    | \n   |    | ')
    elif x==5:
        print('    ____ \n   |    | \n   |    | \n   |    0 ')
    elif x==4:
        print('    ____ \n   |    | \n   |    | \n   |    0 \n   |   /|')
    elif x==3:
        print('    ____ \n   |    | \n   |    | \n   |    0 \n   |   /|\\')
    elif x==2:
        print('    ____ \n   |    | \n   |    | \n   |    0 \n   |   /|\\\n   |   /  ')
    elif x==1:
        print('    ____ \n   |    | \n   |    | \n   |    0 \n   |   /|\\\n   |   / \\  ')
    elif x==0:
        print('    ____ \n   |    | \n   |    | \n   |    0 \n   |   /|\\\n   |   / \\  \n__|__ ')
    return
def getword():
    count1 = len(open('sowpods.txt','r').readlines())
    hs = random.randrange(1,count1,1)
    word=linecache.getline('sowpods.txt',hs)
    word=word.strip('\n')
    return word
def play():
    wd='QWERTYUIOPASDFGHJKLZXCVBNM'
    word=getword()
    ok_word='_'*len(word)
    trueges=[]
    wrongges=[]
    count=7
    print('*************************\nWelcome to Hangman!\n*************************\nGuess one letter a
    while count!=0:
        a=input('What is your guess?:')
        a=a.upper()
        while a not in wd or len(a)>1:
            a=input('Please chose just a letter:')
            a=a.upper()
        if a in trueges or a in wrongges:
            print('You have already guessed {}!'.format(a))
            continue
        if a in word:
            for i in range(len(word)):
```

```
                if word[i]==a:
                    ok_word=th(ok_word,i,a)
                    #ok_word[i]=word[i]
            print1(ok_word)
            trueges.append(a)
            if '_' not in ok_word:
                print('GAME OVER,YOU PASSED!!!!!!!!!!!!!')
                return 1
        if a not in word:
            count-=1
            print('{} is not in this word!'.format(a))
            picture(count)
            print('You have {} guesses left!'.format(count))
            wrongges.append(a)
            if count==0:
                print('You lose!')
            print('your previous wrong guesses: ',wrongges)
            print1(ok_word)
            if count==0:
                print('the word was  ',word)
                return 0
play()
again=input('Would you like to play again? [y|n]:')
while again!='n':
    play()
    again=input('Would you like to play again? [y|n]:')
```

```
**********************
Welcome to Hangman!
**********************
Guess one letter at a time
Game is not case sensitive


__ __ __ __ __ __ __ __
What is your guess?:a
A _ _ _ _ _ _ _ A _
What is your guess?:d
A _ _ _ _ _ D _ A _
What is your guess?:s
S is not in this word!

  ____
  |    |
  |    |
You have 6 guesses left!
your previous wrong guesses:  ['S']
A _ _ _ _ _ D _ A _
What is your guess?:f
F is not in this word!

  ____
  |    |
  |    |
  |    O
You have 5 guesses left!
your previous wrong guesses:  ['S', 'F']
A _ _ _ _ _ D _ A _
What is your guess?:g
G is not in this word!

  ____
  |    |
  |    |
```

```
   |    |
   |    O
   |   /|
You have 4 guesses left!
your previous wrong guesses:  ['S', 'F', 'G']
A _ _ _ _ _ D _ A _
What is your guess?:w
W is not in this word!

    ____
   |    |
   |    |
   |    O
   |   /|\
You have 3 guesses left!
your previous wrong guesses:  ['S', 'F', 'G', 'W']
A _ _ _ _ _ D _ A _
What is your guess?:e
E is not in this word!

    ____
   |    |
   |    |
   |    O
   |   /|\
   |   /
You have 2 guesses left!
your previous wrong guesses:  ['S', 'F', 'G', 'W', 'E']
A _ _ _ _ _ D _ A _
What is your guess?:t
A _ T _ _ _ D _ A _
What is your guess?:e
You have already guessed E!
What is your guess?:e
You have already guessed E!
What is your guess?:r
A R T _ R _ D _ A _
What is your guess?:t
You have already guessed T!
What is your guess?:i
A R T _ R _ D I A _
What is your guess?:m
M is not in this word!

    ____
   |    |
   |    |
   |    O
   |   /|\
   |   / \
You have 1 guesses left!
your previous wrong guesses:  ['S', 'F', 'G', 'W', 'E', 'M']
A R T _ R _ D I A _
What is your guess?:s
You have already guessed S!
What is your guess?:z
Z is not in this word!

    ____
   |    |
   |    |
   |    O
   |   /|\
   |   / \
 __|__
You have 0 guesses left!
```

```
You have 0 guesses left.
You lose!
your previous wrong guesses:  ['S', 'F', 'G', 'W', 'E', 'M', 'Z']
A R T _ R _ D I A _
the word was   ARTHRODIAL
Would you like to play again? [y|n]:n
```

In [ ]: