Import all variables from config

import random

players={0:{"roundtotal":0,"gametotal":0,"name":""},

         1:{"roundtotal":0,"gametotal":0,"name":""},

         2:{"roundtotal":0,"gametotal":0,"name":""},

        }

roundNum = 0

dictionary = []

turntext = ""

wheellist = []

roundWord = ""

blankWord = []

vowels = {"a", "e", "i", "o", "u"}

roundstatus = ""

finalroundtext = ""

def readDictionaryFile():

    global dictionary

    open the dictionary file

read the dictionary file and set to dictionary variable

def readTurnTxtFile():

    global turntext

    open the turn text file

read the file and set to turntext variable

def readFinalRoundTxtFile():

    global finalroundtext

    open the final round text file

read the file and set to a finalroundtext variable

def readRoundStatusTxtFile():

    global roundstatus

    open the roundstatus text file using the config roundstatusloc variable

read the file and set it to a roundstatustext variable

def readWheelTxtFile():

    global wheellist

    open the wheellist text file using the config wheeltextloc variable

read the file lines and set to a wheellist variable

def getPlayerInfo():

    global players

    for each player in their name key in the player dictionary prompt for a name input

def gameSetup():

readDictionaryFile()

    readTurnTxtFile()

    readWheelTxtFile()

    getPlayerInfo()

    readRoundStatusTxtFile()

    readFinalRoundTxtFile()

def getWord():

    global dictionary

    set a roundWord variable to a random choice from the dictionary list

set a roundUnderScoreWord variable to a list putting “\_” for each letter in roundWord

    return roundWord,roundUnderscoreWord

def wofRoundSetup():

    set each player’s roundTotal values to zero

make initPlayer variable equal to a random range of 0 to 3

call the getWord function to get a random word for each round

    return initPlayer

def spinWheel(playerNum):

    create a stillinTurn boolean for making sure the turn is still going

make a spinresult variable and set it equal to a random choice from the wheellist

if the spin result is bankrupt, set the player’s roundTotal to zero and end their turn

if the spin result is loseturn, end their turn

if anything else, prompt to guess a letter and run it through the guessLetter function to check if in roundWord

if it is a valid guess, add the spin value to the roundTotal and prompt again

If it wasn’t a valid guess, end their turn

    return stillinTurn

def guessletter(letter, playerNum):

    keep a Boolean set to true

while loop for if true

if the letter guessed is in the roundWord then replace that “\_” with letter

make it a valid guess and show the roundUnderscoreWord

If its not a valid guess, set it to false

    return goodGuess, count

def buyVowel(playerNum):

    check to see if the players roundTotal is > 250

if it is, prompt to enter a vowel

if it’s a vowel, run through guess letter to validate if in word

if it is a valid guess, subtract 250 from roundTotal

If they don’t have enough money, send back to choice menu

    return goodGuess

def guessWord(playerNum):

    set a guessWord variable to prompt for an input of the word they want to guess

if guessWord == roundWord then replace all “\_” with appropriate letters

if not, tell it wasn’t correct and next player’s turn

    return False

def wofTurn(playerNum):

    print round status

while the player is still in turn, check to see if “\_” in roundUnderscoreWord

set choice variable to asking player if they want to spin, buy vowel, or guess word

go to respective functions based on player input

if(choice.strip().upper() == "S"):

            stillinTurn = spinWheel(playerNum)

        elif(choice.strip().upper() == "B"):

            stillinTurn = buyVowel(playerNum)

        elif(choice.upper() == "G"):

            stillinTurn = guessWord(playerNum)

        else:

            print("Not a correct option")

def wofRound():

    set initPlayer to wofRoundSetup to get a random starting player per round

while round in progress, check to see if word solved yet

if it is solved (no “\_”) transfer roundTotals to gameTotals

then set round in progress to false and break

def wofFinalRound():

    global roundWord

    global blankWord

    global finalroundtext

    winplayer = 0

    amount = 0

    for I in players keys if the players gameTotal is the biggest, that player plays

set amount variable to the playing players gameTotal

print instructions for final round

use getWord function to get a word for the round

set the given RSTLNE to a variable

use guessLetter function to apply those letters where applicable

for i in range 3, prompt for consonants and send through guessLetter

prompt for the 1 vowel and send through guessLetter

print updated roundUnderscoreWord

prompt for final guess and set equal to finalGuess variable

if the finalGuess == roundWord give finalprice and previous earnings (amount)

def main():

    gameSetup()

    for i in range(0,maxrounds):

        if i in [0,1]:

            wofRound()

        else:

            wofFinalRound()

if \_\_name\_\_ == "\_\_main\_\_":

    main()