INFORMATICS INSTITUTE OF TECHNOLOGY

The Foundation Certificate in Higher Education

Doc 334- Introduction to Programming Coursework Report

Module	DOC 334 MAY Introduction to Programming
Module Leader	Mr. Sudarshana Welihinda
Assessment Type	Individual Coursework
Student ID	20200546
Student Name	Naveen Grero
Date of Submission	15th December 2021

Acknowledgement

While preparing this report, I have received great support and assistance. I would like to extend my special thanks of gratitude to all those who provided me the opportunity to complete this report. I cannot express enough thanks to our lecturers, Mr. Sudarshana Welihinda, Mr. Nishan Saliya, Ms. Keerthiga Rajarathnam and Ms. Tharushi Sandamali for their guidance throughout the report. Finally, I would like to take this as a chance to appreciate my parents, friends and all the others that were involved and helped me with their valuable suggestions

Table of Contents

List of Figures

1. Question

1.1 Problem

You are to create a console Python 3.x program which will allow users to demonstrate the single-player game called “Hangman”.

1.2 How it is played

This is a single player game. A player is presented with some empty spaces to guess a word. Player will have limited number of guesses (known as turns). The initial number of turns are equal to the number of characters in the word. For example, if the word is “Potato”, the number of turns is 6. The player can win the game by guessing the correct word within the given turns. The game will end based on below scenarios.

1.3 Task

Player guesses the correct word within the given number of turns. The result - Player wins. Else player lose.

1.4 Tasks to Complete

- You must use proper Python 3.x program constructs such as packages, modules, functions, variables, data structures, etc. to develop this program.
- The player must be able to do below tasks

• View the past game play history

  - This is stored by using either a file base or a database (a DB will carry more marks)
  - If a DB is used, username must be “root” with no passwords and the server must be “localhost”
  - You should also provide means to import/restore your current database for marking purposes. Failing to do this will result lower marks or no marksfor that component.
  - If a text file is used, the extension must be of .TXT
  - A single player must be able to play this game as many times he/she likes. Players must also have the ability to exit the game as they desire at any time.

- Your program should keep track of wins and losses of each game play. This can be for the whole game play history or per session or both!
- Your program must be able to display stats like below. This can be for the whole game play history or per session.

1. Total number of games played
2. Total wins
3. Total loses

  - ➢ You can use external packages which will help you to develop the game.
- Such package uses must be explained in the report. 1. Purpose of using it and which parts were implemented with it
- Proper instructions must be given to do the installation of such packages.
- Links where you install/download the packages must be provided in the report
- Failing to do above tasks will result lower marks or zero marks for those components
- You CANNOT use ready-made game packages for this game! Such attempts will get zero marks for this whole ICW.

1.5 Problem Understanding

Player runs out of turns before guessing the word. The result – Player loses. If the player guesses a character properly, his/her turn will not exhaust. Only the wrong guesses will exhaust turns. If the word consists of duplicate characters, a single guess will fill all the occurrence of that letter. For example, if the hidden word is "banana" and the player guesses the character "a". This will show the filled guesses as "_ a _ a _ a" revealing all 3 positions of the letter "a". Your program should have 20 stored words which will randomly appear during the game. All 20 words should have different number of turns based on the word size. As an addition you can display a small hint for the player about the word. Your program should also collect the player information and keep records of the games played. the collected information may have,

• Player's name

• Word guessed

- Turns provided
- Turns used
- Win/lost status

1.6 Solution

 First step was created the “Hangman” program. Then get the player name from the player and choose a random word. Next get a letter from the user if user entered a incorrect letter, chance will reduce one by one time and print a wrong. When user entered the correct letter, will show the correct. If user possible to the gives correct all the letters it will show the win. finally create five different program files to get user inputs, create a database, add a table to the database and insert game information to the database.

1.7 Hangman Game Program

```
import random
from datetime import datetime
import create_database
user_name=""
element="_"
#create dictionary
word=["cat","rat","dog","elephant","tiger","cow","lion","rabbit","shark","dolphin","parro
t","crow","frog","cheeta","monkey"]
word_num=0
name=[]
won=0
lost=0
games_played=0
h1="HANGMAN GAME"
print(h1.center(75,'*'))
print()

now=datetime.now()#current date and time
```

```
date_time=now.strftime("%d/%m/%Y , %H:%M:%S")
print("date and time: ",date_time)
print()

def main():
    global user_name,element
    if __name__=='__main__':
      if element in user_name:
        pass
      else:
       user_name=input("Enter Player Name:- ")
       element=user_name
      start()
      choice=input("Do you want to run again(if yes select y if no select n):-  ")
      if choice in 'y':
        main()

      else:
          choice=='n'
          create_database.base(user_name,games_played,lost,won)


def start():
  letter=" "
  global word,word_num,won,games_played,lost
  random_word=random.choice(word)
  word_num=len(random_word)
  games_played+=1
  count = word_num+1
  while count > 0:
```

```
win=word_num

print()

print("\nword:-",end=" ")
for element in random_word:



  if element in letter:
     win=win-1
     print(element,end="")

  else:
     print("_",end=" ")



if win == 0:

   print("\nYou Won!")
   print("Word is ",random_word)
   won+=1
   break


elif count == 1:
       print("\nYou Lost!")
       print("Word is ",random_word)
       lost+=1
       break
```

```
        print()
        print(count-1, "turns remaining")
        user  = input("\nletter:")
        letter += user

        if user  not in random_word:
              count -= 1
              print("Wrong")


main()
```

1.8 Create Database Program

```
    #import internal and external packages
    import create_table
    import mysql.connector
    from mysql.connector import Error

    #cretae a database for the game
    def base(user_name,games_played,lost,won):
       try:
          mydb=mysql.connector.connect(host='localhost',user='root',password='')
          mycoursor=mydb.cursor()

          #check connected or not
          if mydb:
             print("Connected")
          else:
             print("Not Connected")

          mycoursor .execute("CREATE DATABASE hangman")
```

```
        create_table.table(user_name,games_played,lost,won)


    except Error as e:
        create_table.table(user_name,games_played,lost,won)
```

1.9 Create Table Program

```
#import internal and external packages
import insert_data
import mysql.connector
from mysql.connector import Error


#create a batabase table for the game
def table(user_name,games_played,lost,won):
   try:

tb=mysql.connector.connect(host='localhost',user='root',password='',database='hangma
n')
       mycoursor=tb.cursor()

       mycoursor.execute("CREATE TABLE player (player_name
VARCHAR(40),played_games int(100),won int(100),lost int(100))")
       insert_data.data(user_name,games_played,lost,won)
   except Error as e:

       insert_data.data(user_name,games_played,lost,won)
```

1.10 Insert Data Program

```
#import mysql connector
import mysql.connector
import html_file
```

```
#insert data into table coloumns
def data(user_name,games_played,lost,won):



da=mysql.connector.connect(host='localhost',user='root',password='',database='hangma
n')
    mycoursor=da.cursor()
    coloumns=("INSERT INTO hangman.player"
        "(player_name,played_games,lost,won)"
        "VALUES(%s,%s,%s,%s)")
    values=(user_name,games_played,lost,won)
    mycoursor.execute(coloumns,values)

    da.commit()
    mycoursor.close()



    da.close()
    html_file.html()#call html function
```

1.11 HTML File Program

```
import mysql.connector
import webbrowser

filename = "play_history.html"
```

```
def main(body, filename):
    #create a body of html page
    output = open(filename,"w")
    output.write(body)
    output.close()


def html():
    global filename
    # connecting to the MYsql and create a data base
    htm = mysql.connector.connect(user='root',
password='',host='localhost',database='hangman')

    #select table in the data base
    select_table = """SELECT * FROM player"""
    cursor = htm.cursor()
    cursor.execute(select_table)
    result = cursor.fetchall()

    lis = []

    tbl =
"<tr><td>player_name</td><td><td>played_games</td><td><td>won</td><td><td>l
ost</td><td></tr>"
    lis.append(tbl)

    for rows in result:
        a = "<tr><td>%s</td>"%rows[0]
        lis.append(a)
        b = "<td><td>%s</td>"%rows[1]
        lis.append(b)
```

```
c = "<td><td>%s</td>"%rows[2]
lis.append(c)
d = "<td><td>%s</td></tr>"%rows[3]
lis.append(d)
```

2. External Packages

Import mysql.connector/ Error: MySQL connector is imported to arrange and create a connectivity between MySQL and the created Python program because the data in this program are being saved inside to MySQL databases which is a free database server of Oracle and its affiliates. And Error is a class inside this module which is used to identify the error while hosting and connecting to databases.

Import random: Using 'import random' imports the package, which you can then use the function from this package: You can use any other function from the 'random' package as well. You can also tell python to specifically import only the random function from the package random: 'from random import random'.

Import datetime: This module identifies the current date and time of the operating system and is displayed in string format using the strftime() method.

Import webbrowser: Creates a link between the program and a given URL of a website or any variety of a webpage and interfaces for launching and controlling the web browsers which are given inside the program.

3.Test Cases

```
C:\Users\POWER\OneDrive\Desktop\IIT LECTURES\FOUNDATION\2nd Sem\Introduction to programming 2 (doc334)\hangman game>game
**********************************HANGMAN GAME**********************************

date and time:  14/12/2021 , 15:04:53

Enter Player Name:- dewmini


word:- _ _ _
3 turns remaining

letter:c


word:- c_ _
3 turns remaining

letter:a
Invaid


word:- c_ _
2 turns remaining

letter:o


word:- co_
2 turns remaining

letter:w


word:- cow
You Won!
Word is  cow
Do you want to run again(if yes select y if no select n):-  y
```

Figure 1-Test Cases

```
word:- cow
You Won!
Word is  cow
Do you want to run again(if yes select y if no select n):-  y


word:- _ _ _ _
4 turns remaining

letter:l
Invaid


word:- _ _ _ _
3 turns remaining

letter:o


word:- _ _ o_
3 turns remaining

letter:r


word:- _ ro_
3 turns remaining

letter:f
Invaid


word:- _ ro_
2 turns remaining

letter:c


word:- cro_
2 turns remaining

letter:w


word:- crow
You Won!
Word is  crow
Do you want to run again(if yes select y if no select n):-  y
```

Figure 2-Test Cases

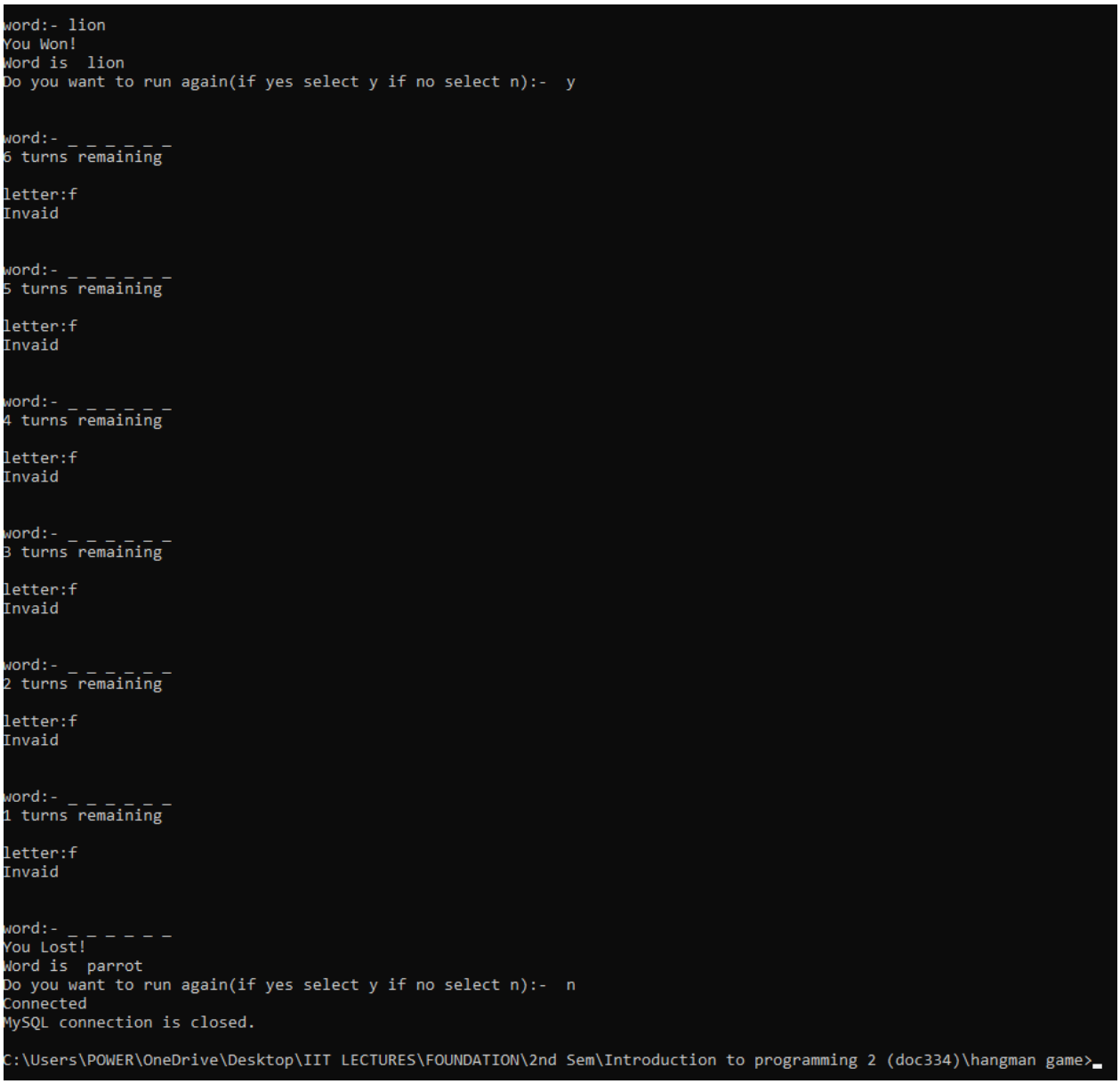

Figure 3-Test Cases

4.MySQL Structure (Game Data Table)

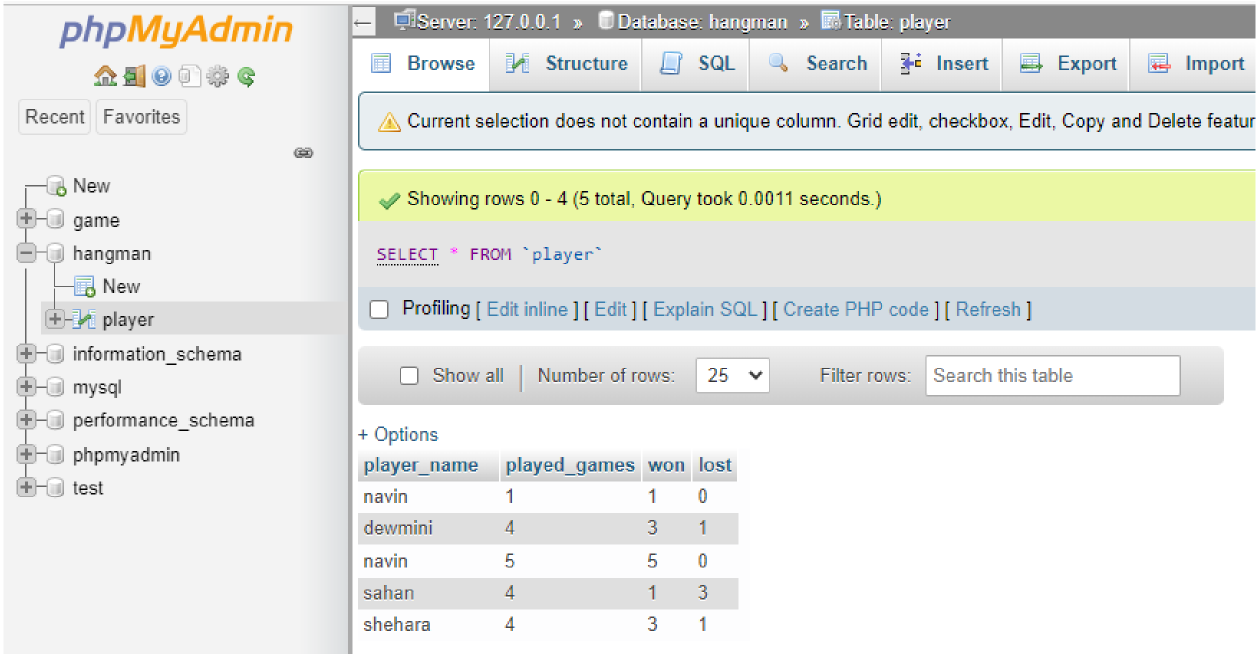

Figure 4-MySQL Structure (Game Data Table)

5. Web browser Structure

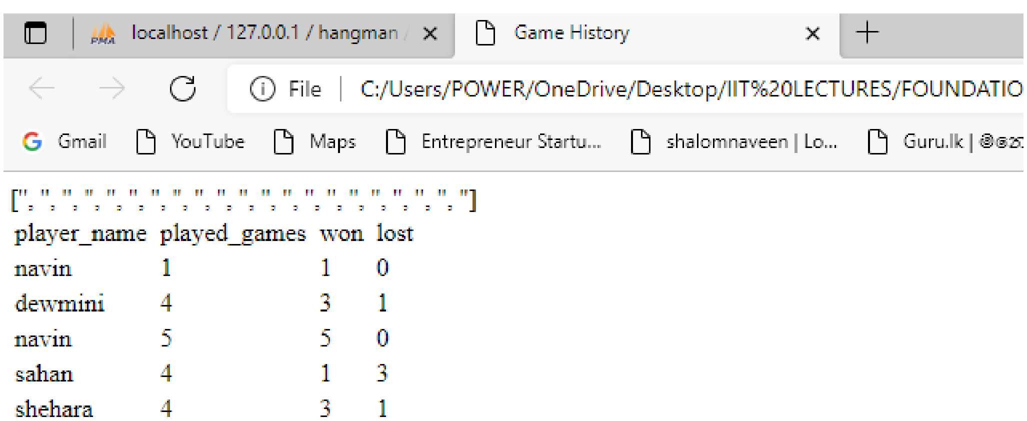

Figure 5-Web browser Structure

References

- programiz.com, n.d. Programiz. [Online] Available at: https://www.programiz.com/python-programming/datetime/strftime [Accessed 16 August 2021].

- https://itsourcecode.com/free-projects/python-projects/hangman-game-in-pythonwith-sourcecode/#:~:text=A%20Hangman%20Game%20On%20Python,then%20the%20gam e%20is%20over.

- https://www.codegrepper.com/code-examples/python/python+hangman

- https://datamakessense.com/mysql-to-html-with-python