PROJECT REPORT

ON

HANGMAN GAME

(Python Application Programming)

TABLE OF CONTENTS

TOPIC

INTRODUCTION

1.1 Python Programming language Python is one of the many open source object oriented programming application software available in the market. Python is developed by Guido van Rossum. Guido van Rossum started implementing Python in 1989. Python is a very simple programming language so even if you are new to programming, you can learn python without facing any issues. Some of the many uses of Python are application development, implementation of automation testing process, allows multiple programming build, fully constructed programming library, can be used in all the major operating systems and platforms, database system accessibility, simple and readable code, easy to apply on complex software development processes, aids in test driven software application development approach, machine learning/ data analytics, helps pattern recognitions, supported in multiple tools, permitted by many of the provisioned frameworks, etc.

Features:

1. **Readable:** Python is a very readable language.

2. **Easy to Learn:** Learning python is easy as this is an expressive and high level programming language, which means it is easy to understand the language and thus easy to learn.

3. **Cross platform:** Python is available and can run on various operating systems such as Mac, Windows, Linux, UNIX etc. This makes it a cross platform and portable language.

4. **Open Source:** Python is an open source programming language.

5. Large standard library: Python comes with a large standard library that has some handy codes and functions which we can use while writing code in Python.

1.2 Applications of Python programming language:

Python can be used to develop different applications like web applications, graphic user interface based applications, software development application, scientific and numeric applications, network programming, Games and 3D applications and other business applications. It makes an interactive interface and easy development of applications. You may be wondering what all are the applications of Python. There are so many applications of Python, here are some of them.

1. **Web development** – Web framework like Django and Flask are based on Python. They help you write server side code which helps you manage database, write backend programming logic, mapping etc.

2. **Machine learning** – There are many machine learning applications written in Python. Machine learning is a way to write a logic so that a machine can learn and solve a particular problem on its own. For example, products recommendation in websites like Amazon, Flipkart, eBay etc. is a machine learning algorithm that recognizes user's interest. Face recognition and Voice recognition in your phone is another example of machine learning.

3. **Data Analysis** – Data analysis and data visualization in form of charts can also be developed using Python.

4. **Scripting** – Scripting is writing small programs to automate simple tasks such as sending automated response emails etc. Such type of applications can also be written in Python programming language.

5. **Game development** – You can develop games using Python.
6. You can develop **embedded applications** in Python.

7. Desktop applications – You can develop desktop application in Python using library like Tkinter.

SYSTEM REQUIREMENTS

2.1 Software Requirements

- Jupyter Notebook version 6.4.5

2.2 Hardware Requirements Operating system:

- Windows 10 Processor:
- Intel core i5
- Disk space: 1 GB

IMPLEMENTATION AND RESULTS

3.1 About Project

This is a simple Hangman game using Python programming language. Beginners can use this as a small project to boost their programming skills and understanding logic.

1. The Hangman program randomly selects a secret word from a list of secret words. The random module will provide this ability, so line 1 in program imports it.
2. The Game: Here, a random word (a fruit name) is picked up from our collection and the player gets limited chances to win the game.
3. When a letter in that word is guessed correctly, that letter position in the word is made visible. In this way, all letters of the word are to be guessed before all the chances are over.

3.2 Project code:

```
#  importing required library. i.e. random

import random

hang = ["""
H A N G M A N - Word Puzzle

  +---+
  |   |
      |
      |
      |
      |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
      |
      |
      |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
  |   |
      |
      |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
 /|   |
```

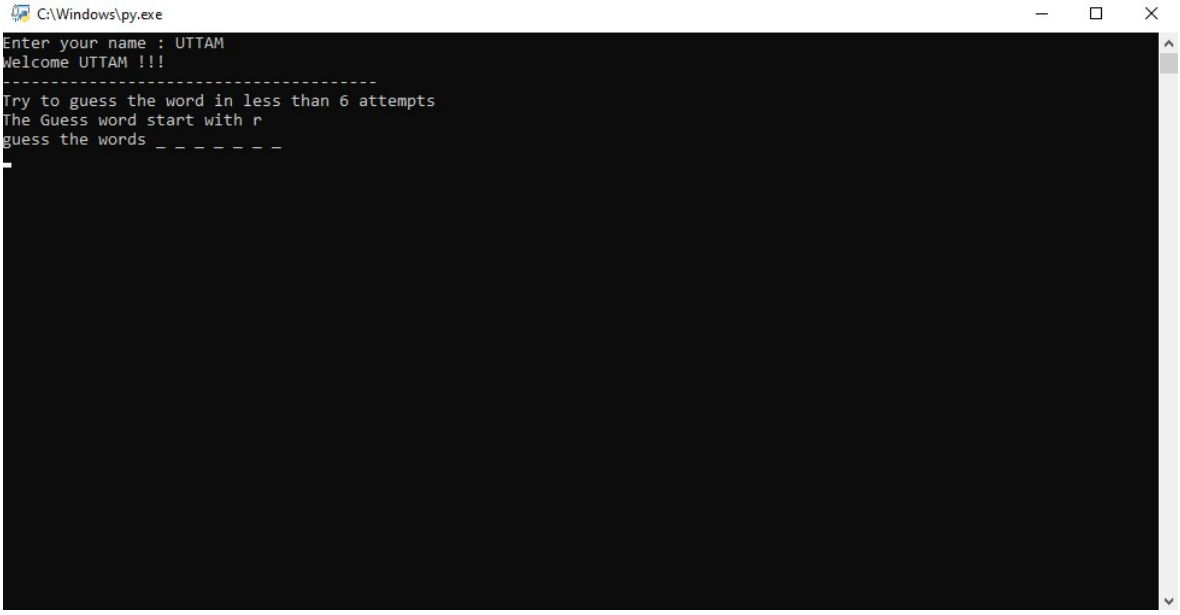
```
     |
     |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
 /|\  |
      |
      |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
 /|\  |
 /    |
      |
=========""", """
H A N G M A N - Word Puzzle

  +---+
  |   |
  O   |
 /|\  |
 / \  |
      |
=========""", """]


# create a function named hangman (to play a hangman)
def hangman():

    list_of_words=['rainbow', 'computer', 'science', 'programming',
        'python', 'mathematics', 'player', 'condition',
        'reverse', 'water', 'board', 'geeks']

    word=random.choice(list_of_words).lower()
```

```
print('The Guess word start with',word[0])

guessmade=''

count=0

valid_entry=set('abcdefghijklmnopqrstuvwxyz')

while len(word)>0:
    main_word =''

    for letter in word:

        if letter in guessmade:
            main_word=main_word+letter
        else:
         main_word=main_word+"_ "



    if main_word==word:
        print(main_word)
        print('You win !!')
        break



    print('guess the words', main_word)
    guess=input()

    if guess in valid_entry:
        guessmade=guessmade+guess
    else:
        print("Enter valid character : ")
        guess=input()


    if guess not in word:
```

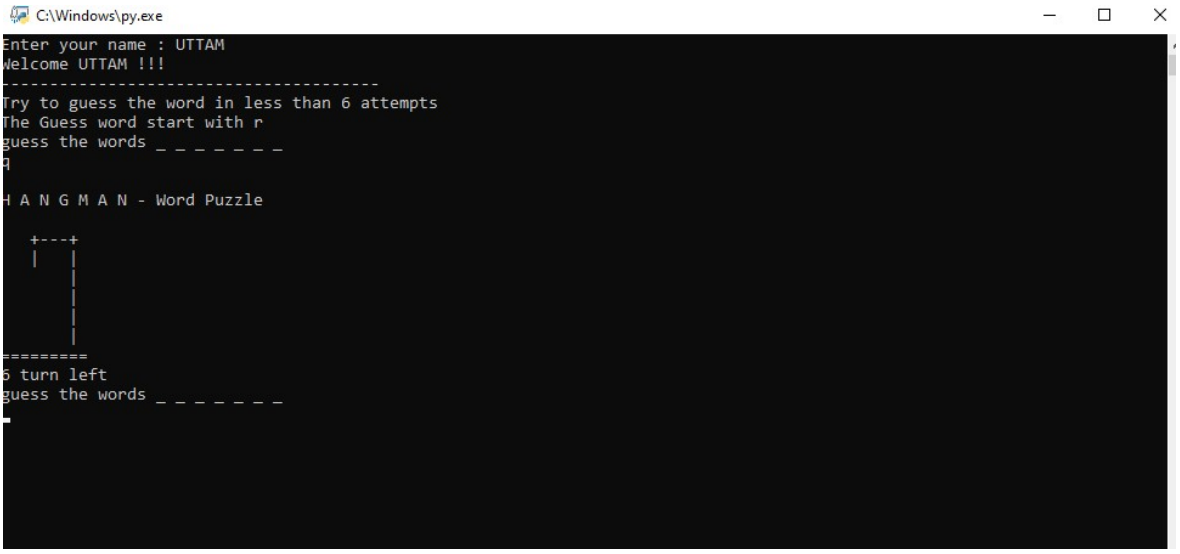
```
        if count==0:
            print(hang[0])
            count=count+1
            print("6 turn left")

        elif count==1:
            print(hang[1])
            count=count+1
            print("5 turn left")

        elif count==2:
            print(hang[2])
            count=count+1
            print("4 turn left")

        elif count==3:
            print(hang[3])
            count=count+1
            print("3 turn left")

        elif count==4:
            print(hang[4])
            count=count+1
            print("2 turn left")

        elif count==5:
            print(hang[5])
            count=count+1
            print("only 1 turn left")

        elif count==6:
            print('You lose!!!')
            print(hang[6])

            break



name=input("Enter your name : ")
```

```
print("Welcome",name,'!!!')
print('-------------------------------------')
print('Try to guess the word in less than 6 attempts')

hangman()
```

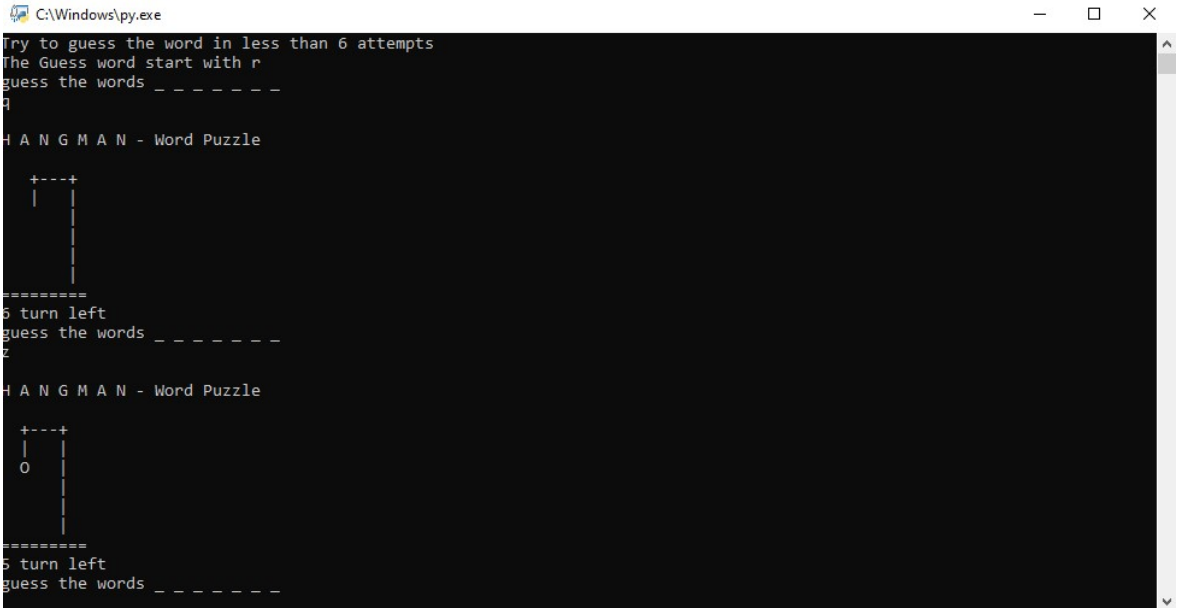
3.2 SCREENSHOT AND RESULT:



First wrong attempt



Second wrong attempt



Third wrong attempt

Fourth wrong attempt

Fifth attempt

Last attempt

Conclusion:

In the conclusion of this project, Hangman is a traditional game, typically played with words. It's possible, however, to play Category Hangman rather than guessing words the player might guess names of fruits. You'll be writing a program to play a "guess a word letter-by-letter" version of hangman as shown above. You'll also be doing some statistical analysis of the words used in the Hangman game.