Course Project

Name : Bhagya Patel

Student Number : 200558135

Description :

Hello, my name is Bhagya Patel, and I've created a game called Zephyro. This game is designed for children in the age group of 7-12 years. To explain, in this game, we initially check if the user is a human player by asking them to retype the number displayed on the screen. If the user fails to enter the correct value, the game will end. To play again, you need to rerun the code. However, if the user enters the correct value, the game name and rules will be displayed. To start the game, the user needs to press enter. Once they do, they will be prompted with a shape, and they must answer with the name of the shape from the list provided in the rules section. It's crucial to be careful because everything is case-sensitive, so the spelling must match exactly as provided in the list. If the user answers correctly, their score will increase by 1 for each correct response. To avoid repetition, the game will remove the shapes from the list once user guess it correct . In the case of an easy shape, the user gets another chance to guess. In total, the user has 8 chances. Once they have guessed more than 50% for easy shapes, the game will ask if they want to increase the difficulty level. If the user agrees, the game will present slightly more challenging shapes. If the user guesses any shape wrong, the game will end, displaying the result. If the user guesses correctly, it will ask if they still want to increase the difficulty. If the user is satisfied with the result, they may end the game by saying no. The end result which is provided is out of number of chances provided to guess which is 8 and percentage display accordingly. Lastly, it's crucial to enter the spelling correctly. The user need to score more than 80% to win .

Code :

```
# Importing the modules to use certain funtions in code

import random

import os

import openpyxl

# User defined funtion usage

#This funtion is used to generate random column to generate random integer from the excel file

def randomColum():

    randomNumber = random.randint(1,15)

    return 'A' + str(randomNumber)
```

```
# Using the XL to check the person who is playing the game is not robot if they retype value correctly the code will execute else it ends
resultDataBook = openpyxl.load_workbook(r"C:\Users\pbhag\OneDrive\Desktop\Python\CourseProject\Randomnumber.xlsx")
activateSheet = resultDataBook.active
randomNumber = activateSheet[randomColum()].value
print("This number is generated to check you are not a robot : "+str(randomNumber))
# Intializing userCheck as none because if error occur it directly jump to
# code of exept and we get anther error at if(randomNumber == userCheck) this
# line of code as for this userCheck is not defined
userCheck = None
# Execption handling using key words try and except
try :
  userCheck = int(input("If you are not robot enter the number dispalyed above to start the game : "))
except ValueError:
  print("Invalid input you should enter Integer value !!\nPlease try again by running the program again !!")
# Declaring the global variable count which get increse by one for every correct answer
count = 0
# 8 user defined funtion to print different shapes
# This could print solid square made of star
def square():
 for i in range(6):
   # for every value of i j will execute three times
  for j in range(6):
   print("*", end=" ")
   # This print funtion enable the user define funtion to print in next line once for every value of i j is been executed
  print()



def rectangle():
 for i in range(5):
   # We are using string replication here as it is context sensetive dataType
  print("*" * 5)

# This funtion will print right angle triangle means a triangle having one angle of 90 degree
def rightAngleTriangle():
 for i in range(5):
  for j in range(i + 1):
   print("*", end="")
  print()


def triangle():
 for i in range(1, 6 + 1):
   # Here to print actual triangle we have print some spaces before and after the star
   # for that reson we are printing n-i paces in each row followed by stars which will increse consecutively with increse in i
  spaces = " " * (6 - i)
  stars = "*" * (2 * i - 1)
  print(spaces + stars)


def invertedTriangle():
  # Here to print the inverted triangle we starting range of 5 because in first row we do not need to
  # print any spaces that why intial spaces will become n-n = 0 hence here spaces gonna increse and * gonna deacrese
 for i in range(5, 0, -1):
  spaces = " " * (5 - i)
  stars = "*" * (2 * i - 1)
  print(spaces + stars)


def hollowRectangle():
  # Here by using the matrix terminology we are printing the hollow rectangle
 for i in range(1, 5 + 1):
  for j in range(1, 5 + 1):
    # AS in matrix when the value of row and colum is min and max print stars
    # else print spaces in this way we can able to achive hollow pattern
   if(i == 1 or i == 5 or j == 1 or j == 5):
    print("*", end="")
   else:
    print(" ", end="")
    # This print funtion used to start the code from next line
  print()
```

```
# This pattern of butterfly is consist of two chuck of code
# Upper most parts and lower most part
def butterfly():
    # Upper part of butterfly
    # This code print firstHalf of the butterfly which is basically gonna be right angle triangle
    # This for loop is used to iterate overs row
  for i in range(1, 7):
      # This for loop take care of printing the stars in each row
      for j in range(1, i + 1):
          print("*", end="")
      space = 2 * (6 - i)
      # This for loop will print spaces for n-i spaces(where n is size of butterfly you want  to print)
      # and spaces is gonna deacrese the number of spaces as value of i increse
      # This for loop is not a part of body of upper one because these three for loop executed for same value of row
      for j in range(1, space + 1):
          print(" ", end="")
      # This for loop taken care of printing the reflected right angle triangle  in this way we can print upper part of butterfly
      for j in range(1, i + 1):
          print("*", end="")
      print()
    # Lower part of butterfly
    # Similarly as above we are iterate on row to print a star on bottom part here value to i goes from n-->0(n is size)
  for i in range(6, 0, -1):
      # Below three for loop executed same time for same row to print star spaces star
      for j in range(1, i + 1):
          print("*", end="")
      space = 2 * (6 - i)
      # This loop will print n-i spaces in each row
      for j in range(1, space + 1):
          print(" ", end="")
      for j in range(1, i + 1):
          print("*", end="")
      print()
```

```
# This chunk of code prints rhombus. This code is combination of the triangle and inverted triangle
def rhombus():
   # Upper half
   # This will print iterate on row follwed by spaces and stars
  for i in range(5):
   for j in range(5 - i - 1):
     print(" ", end="")
   for j in range(2 * i + 1):
     print("*", end="")

   print()
   #Lower half
   # This will print n-i spaces follwed star here in first row  0 spaces and it will increse
  for i in range(5 - 2, -1, -1):
   for j in range(5 - i - 1):
     print(" ", end="")
   for j in range(2 * i + 1):
     print("*", end="")

   print()
# This will check human appearence if user didn't able to type write integer they need to run game again
if(randomNumber == userCheck):
# Starting of game

print("""*************************************************************************************************************************
*****************************************************************************
                              ************      Welcome to Zephyro      ************
                              ************     Guessing the shape game    ************
********************************************************************************************************************************
********************************************************************""")
   # Handling exception as it throw error if file not found
   try:
     rulesFile = open(r"C:\Users\pbhag\OneDrive\Desktop\Python\CourseProject\Rules.txt")
     rules = rulesFile.read()
     print(rules)
   except FileNotFoundError:
     print("Rules file doesn't exsist in you system please kindly add it and then run your code ")
     # This while loop don't allow the user to go forward until they enter anything else than pressing enter
   while (input( "Please press enter to start the game and set out on a magical adventure!") != ''):
     print("Invalid argument ! please enter value correctly when prompted .")
   # This is list is consist of easy shape at 0th index and hard shapes at 1st index
   listOfShapes = [["square", "rectangle", "triangle", "rhombus"],["HollowRectangle", "butterfly", "rightangletriangle","invertedtriangle"]]
   # Comined lenght of lists
   numberOfRound = len(listOfShapes[0]) + len(listOfShapes[1])
   # Number of time upto which want to give user a chance it will be numberOfRound-1 to avoid index error
   for i in range(numberOfRound):
   # This while loop will execute when score is lees than 50%
     while (count < 4):
       # String slicing and shuffling the list
     shuffledList = random.shuffle(listOfShapes[0])
     shapePicked =  listOfShapes[0][0]
     # String Comparision
     if shapePicked == "square":
       square()
     elif shapePicked == "rectangle":
       rectangle()
     elif shapePicked == "triangle":
       triangle()
     else:
       rhombus()
     userGuess = input("Alright little champ!\nCould you tell me what shape is being displayed on the screen by taking a wild guess : ")
     # Comapring answers
     if (userGuess.lower() == shapePicked.lower()):
       print("Yeppy! Your answer is correct, master blaster.")
       # As got correct score increased
       count = count + 1
       print("Excellent! Your current score is : " +  str(count))
       # Removing the correct gussed shape to avoid repetation
       del listOfShapes[0][0]
```

```
     # break to start the loop again
     break
   # break as user got wrong by printing suituable result
   else:
     print("sorry you got it wrong please try to guess again")
     break
 # As loop breaks and count is still smaller than 50% then it will start again else ask to increse the level
 if (count < 4):
   continue
# If score exceeds 50% it will ask to increse the level else if they donot want it will end the game
 else:
     # Taking user permission
   increasingDifficulty = input("Do you want to increse difficulty and still want to continue game (yes/no) : ")
   if (increasingDifficulty.lower() == "YES".lower() and len(listOfShapes[1])!=0):
       # Shuffling and Slicing list of hard shape at 1st index
     shuffuleUpdatedList = random.shuffle(listOfShapes[1])
     difficultShape = listOfShapes[1][0]
     # String comparision
     if difficultShape == "HollowRectangle":
       hollowRectangle()
     elif difficultShape == "rightangletriangle":
       rightAngleTriangle()
     elif difficultShape == "invertedtriangle":
       invertedTriangle()
     else:
       butterfly()
     # User Input to guess the shape
     userInput = input("AHA !! It's time to guess little difficult champ : ")
     if (userInput.lower() == difficultShape.lower()):
         # Incresing score if got write
       count = count + 1
       # Printing current score
       print("Your current score is : " + str(count))
       # Avoiding repetation
       del listOfShapes[1][0]
```

```
      # If got correct continue from above
      continue
    # Got wrong break and over the game
     else:
      print("Nice try !! Sorry to say you got tough one wrong ")
      break
    # Braking the top loop to exit
    else:
     break
  # Ending of game and print the result
  # Requirement of game is to score more than 80% to win else they lose game
  print("Game Over !!")
  percentage = (count/numberOfRound)*100
  if(percentage > 80):
   print("Congratulation !! You win and your score is : "+str(percentage))
  else:
   print("Sorry better luck next time !! You lose and your score is : "+str(percentage))
   print("Required Score for winning the game is more than 80% ")
 # This thing will printed if user unable to print number displayed on screen. if system   think there might be chance of robot playing game
else :
  print("Failed to start game !!Some AI interferernce detected!!\nAs you unable enter the correct value asked for!! ")
```