NITTE EDUCATION TRUST | N.M.A.M. INSTITUTE OF TECHNOLOGY
(An Autonomous Institution affiliated to Visvesvaraya Technological University, Belagavi)
Nitte – 574 110, Karnataka, India
(ISO 9001:2015 Certified)
Accredited with 'A' Grade by NAAC

Python Project On

"COLOR GAME"

Submitted

In partial fulfillment requirements for the award of the Degree

Of

BACHELOR OF ENGINEERING

IN

INFORMATION SCIENCE AND ENGINEERING

By

DEEPIKA (4NM18IS036)

ANUSHA B (4NM18IS021)

Staff Incharge

Ms. Anusha N

Assistant Professor Gd.I

Department of Information Science and Engineering

Output of "COLOR GAME"

Output 1:Game starts.

Fig 1: Beginning of the game.

Output 2: Press Enter key,the timer starts.

Fig 2:Enter key is pressed and timer starts.

Output 3: Enter the color of the word given.

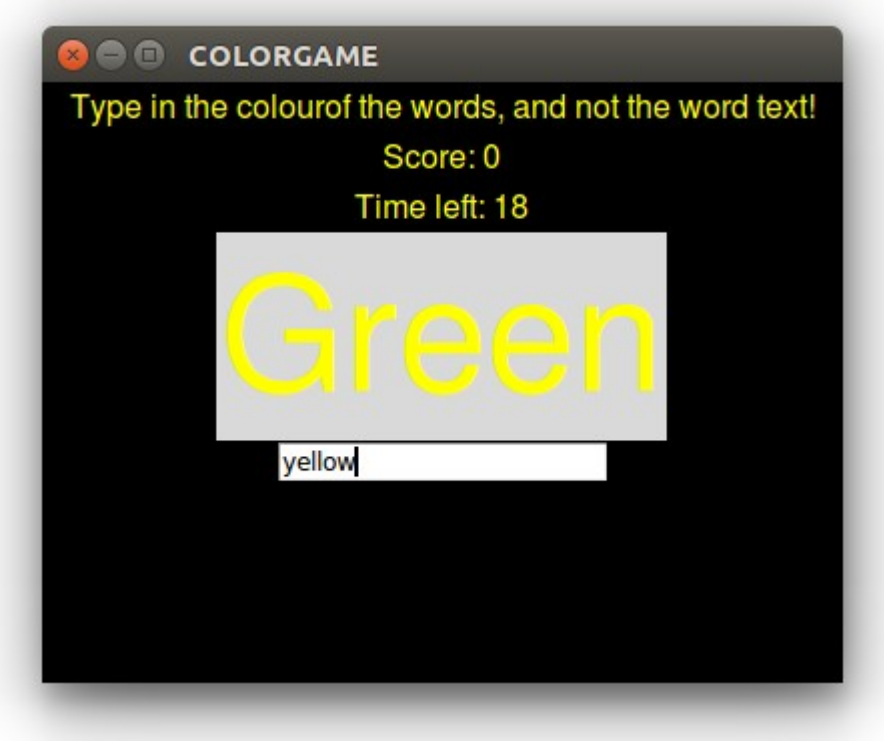

Fig 3:Type the color of the text in the entry box.

Output 4: If the entered color is correct, score increased by 1.

Fig 4:Increment of score, only if color of the text entered is correct.

Output 5: Score increased by 2(correct input).

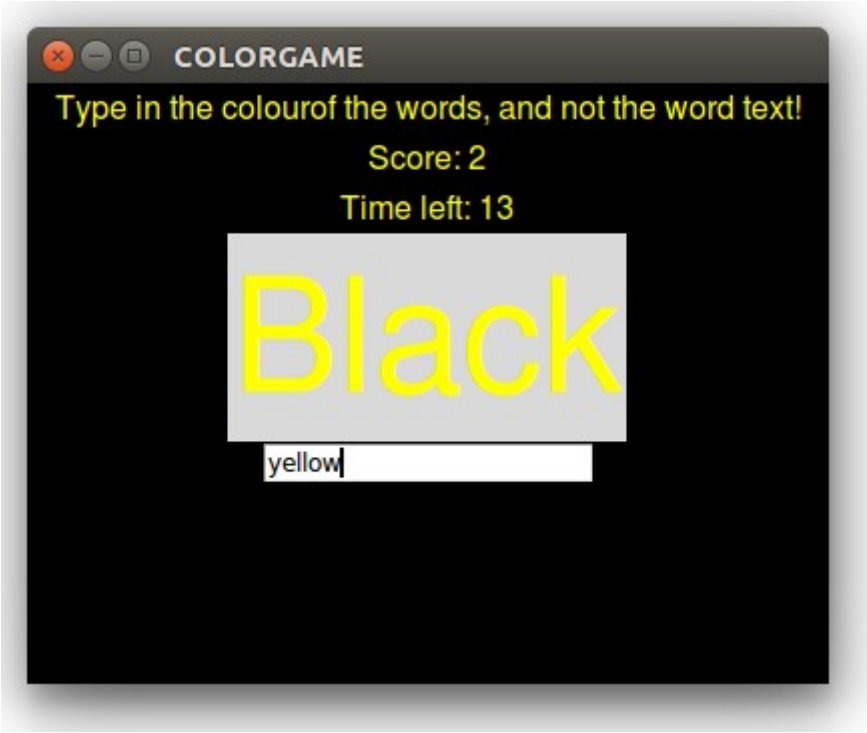

Fig 5:Increment of score(correct input).

Output 6: Wrong input is entered.

Fig 6:Wrong color of text is entered.

Output 7: If the input is wrong, then score remains unchanged.

Fig 7:Score remains same(wrong color is entered).

Output 8: Time over

Fig 8:End of the game(No time left).

Implementation of "COLOR GAME"

```
import tkinter
import random


# list of possible colour.
colours = ['Red','Blue','Green','Pink','Black',
           'Yellow','Orange','White','Purple','Brown']
score = 0


# the game time left, initially 30 seconds.
timeleft = 30


# function that will start the game.
def startGame(event):


    if timeleft == 30:


        # start the countdown timer.
        countdown()


    # run the function to
    # choose the next colour.
    nextColour()
```

```
# Function to choose and
# display the next colour.
def nextColour():

    # use the globally declared 'score'
    # and 'play' variables above.
    global score
    global timeleft

    # if a game is currently in play
    if timeleft > 0:

        # make the text entry box active.
        e.focus_set()

        # if the colour typed is equal
        # to the colour of the text
        if e.get().lower() == colours[1].lower():

            score += 1

        # clear the text entry box.
        e.delete(0, tkinter.END)
```

```
        random.shuffle(colours)

        # change the colour to type, by changing the
        # text _and_ the colour to a random colour value
        label.config(fg = str(colours[1]), text = str(colours[0]))

        # update the score.
        scoreLabel.config(text = "Score: " + str(score))


# Countdown timer function
def countdown():

    global timeleft

    # if a game is in play
    if timeleft > 0:

        # decrement the timer.
        timeleft -= 1

        # update the time left label
        timeLabel.config(text = "Time left: "
```

```
                        + str(timeleft))

        # run the function again after 1 second.
        timeLabel.after(1000, countdown)



# Driver Code

# create a GUI window
root = tkinter.Tk()

# set the title
root.title("COLORGAME")

# set the size
root.geometry("400x300")
root.config(bg="black")

# add an instructions label
instructions = tkinter.Label(root, text = "Type in the colour""of the words, and not the word text!", font = ('Helvetica', 12),bg="black",fg="yellow")
instructions.pack()

# add a score label
```

```
scoreLabel = tkinter.Label(root, text = "Press enter to start", font = ('Helvetica', 12),bg="black",fg="yellow")
scoreLabel.pack()

# add a time left label
timeLabel = tkinter.Label(root, text = "Time left: " +str(timeleft), font = ('Helvetica', 12),bg="black",fg="yellow")

timeLabel.pack()

# add a label for displaying the colours
label = tkinter.Label(root, font = ('Helvetica', 60))
label.pack()

# add a text entry box for
# typing in colours
e = tkinter.Entry(root)

# run the 'startGame' function
# when the enter key is pressed
root.bind('<Return>', startGame)
e.pack()

# set focus on the entry box
```

```
e.focus_set()



# start the GUI

root.mainloop()
```

*** THANK YOU ***