JUMBLED WORD GAME

A
Mini Project Report

Submitted in partial fulfilment of the
Requirements for the award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

By
AJAY KUMAR

1602-20-737-002

Raj Charan Reddy

1602-20-737-029

Irfan

1602-20-737-052

Department of Information Technology

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Ibrahimbagh, Hyderabad-31

2022

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

DECLARATION BY THE CANDIDATE

We**, Ajay Kumar, Raj Charan Reddy, Irfan** bearing hall ticket number, **1602-20-737-002**, **1602-20-737-029** and **1602-20-737-052**, hereby declare that the project report entitled **"JUMBLED WORD GAME" ,** is submitted in partial fulfilment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology.**

This is a record of bonafide work carried out by me and the results embedded in this project report have not been submitted to any other university or institute for the award of any other degree or diploma.

Vasavi College of Engineering (Autonomous)

(Affiliated to Osmania University)

Hyderabad-500 031

Department of Information Technology

TABLE OF CONTENTS

ACKNOWLEDGEMENT

We extend our sincere thanks to Dr. S. V. Ramana, Principal, Vasavi College of Engineering for his encouragement.
We express our sincere gratitude to Dr. K. Ram Mohan Rao, Professor & Head, Department of Information Technology, Vasavi College of Engineering, for introducing the Mini-Project module in our curriculum, and also for his suggestions, motivation, and co-operation for the successful completion of our Mini Project. We also want to thank and convey our gratitude towards our mini project coordinators Divya mam and Rajya Laxmi mam, for guiding us in understanding the process of project development & giving us timely suggestions at every phase. We would also like to sincerely thank the project reviewers for their valuable inputs and suggestions.

INTRODUCTION

The Jumbled Words Quiz in Python is a simple project for helping kids grow in IQ. The project contains only the user side. The user can start the quiz by clicking on the start button. Also, you can choose the type of words, you want to solve in quiz. The user can change jumbled words if they do not know the correct word for it.

ABSTRACT

This is a GUI application. The user will have multiple categories of words to choose. Based on the category they select the jumbled words will be displayed for which they will have to type their answers in the entry box. On submitting the answer, they will get either 5 points or 0 points added to their current score. In addition, the pop- up shows if the answer is correct or not. They can change their word if they want to. And if they opt viewing the answer, their score will be reduced by 5. It displays "not enough points" if the score is not greater than 5. Furthermore, the user can go back to choose another category of words at any point of time.

Tkinter Module

Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit. Import the Tkinter module.

Figure 1

Tkinter tutorial provides basic and advanced concepts of Python Tkinter. Our Tkinter tutorial is designed for beginners and professionals.

Python provides the standard library Tkinter for creating the graphical user interface for desktop-based applications.

Developing desktop-based applications with python Tkinter is not a complex task. An empty Tkinter top-level window can be created by using the following steps.

1. import the Tkinter module.
2. Create the main application window.
3. Add the widgets like labels, buttons, frames, etc. to the window.
4. Call the main event loop so that the actions can take place on the user's computer screen.

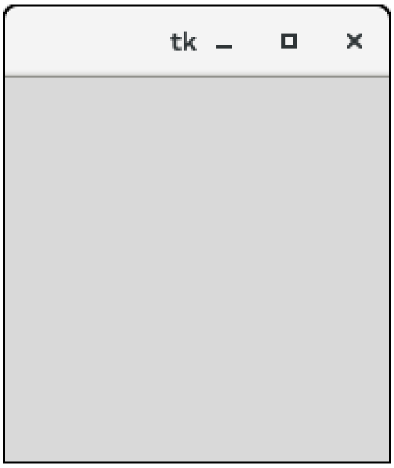


Figure 2

Python Tkinter grid() method

The grid() geometry manager organizes the widgets in the tabular form. We can specify the rows and columns as the options in the method call. We can also specify the column span (width) or rowspan(height) of a widget.

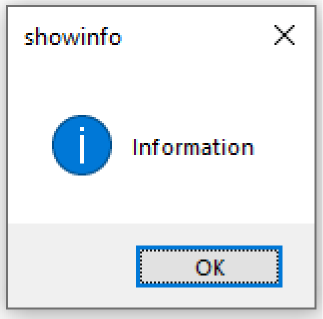
Tkinter widgets

In general, **Widget** is an element of Graphical User Interface (GUI) that displays/illustrates information or gives a way for the user to interact with the OS. In **Tkinter** , **Widgets** are objects ; instances of classes that represent buttons, frames, and so on. Each separate widget is a Python object. When creating a widget, you must pass its parent as a parameter to the widget creation function.

The only exception is the "root" window, which is the top-level window that will contain everything else and it does not have a parent.

MessageBox Widget

Python Tkinter – MessageBox Widget is used to display the message boxes in the python applications. This module is used to display a message using provides a number of functions.



showinfo

Information

Figure 3

Python Tkinter Button

The button widget is used to add various types of buttons to the python application. Python allows us to configure the look of the button according to our requirements. Various options can be set or reset depending upon the requirements. We can also associate a method or function with a button which is called when the button is pressed.

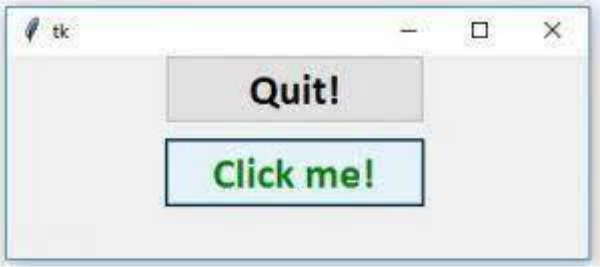


Figure 4

Resizable() method in Tkinter

Resizable() method is used to allow Tkinter root window to change it's size according to the users need as well we can prohibit resizing of the Tkinter window.
So, basically, if user wants to create a fixed size window, this method can be used.

Geometry method in Tkinter

Tkinter provides many methods; one of them is the **geometry()** method. This method is used to set the dimensions of the Tkinter window and is used to set the position of the main window on the user's desktop.

Tkinter title

- Python Tkinter '**title**' refers to the name provided to the window. It appears on the top of the window & mostly found on the top left or center of the screen.
- In the below picture you can notice that 'PythonGuides' is a title for the application.
- It set the title of this widget

FILES AND FOLDERS USED

In options folder, we have the following python files:

Animals

Body_parts

Colour

Fruit

Shapes

Vegetable

Vehicles

OTHER FILES USED

1. Filename: back , Filetype : PNG

2. Filename : quizee_logo, Filetype: Icon

Quizee
Grow Your Kids with Quizee

3. Filename : quizee_logo, Filetype: PNG

Quizee
Grow Your Kids with Quizee

IMPLEMENTATION

Animals.py file:

```
from tkinter import *

from random import *

from tkinter import messagebox

import time

ANIMALS_WORD = ['DRBI', 'DGO', 'OENDYK', 'GFRIEFA', 'GLOILARTA', 'TAC',
'EHSOR', 'OLIN', 'MYOEKN', 'EEB', 'KDUC',

                'RGFO', 'TPNLEHEA', 'ORCDCIELO', 'POLNIHD', 'LARLIGO', 'EMSUO',
'EGTRI', 'ABRITB', 'ATR', ]


ANIMALS_ANSWER = ['BIRD', 'DOG', 'DONKEY', 'GIRAFFE', 'ALLIGATOR', 'CAT',
'HORSE', 'LION', 'MONKEY', 'BEE', 'DUCK',

                 'FROG', 'ELEPHANT', 'CROCODILE', 'DOLPHIN', 'GORILLA', 'MOUSE',
'TIGER', 'RABBIT', 'RAT', ]

ran_num = randrange(0, (len(ANIMALS_WORD)))

jumbled_rand_word = ANIMALS_WORD[ran_num]

points = 0

def main():

    def back():

        my_window.destroy()

        import main_start

        main_start.start_main_page()

    def change():

        global ran_num

        ran_num = randrange(0, (len(ANIMALS_WORD)))

        word.configure(text=ANIMALS_WORD[ran_num])

        get_input.delete(0, END)

        ans_lab.configure(text="")

    def cheak():

        global points, ran_num
        user_word = get_input.get().upper()
        if user_word == ANIMALS_ANSWER[ran_num]:
            points += 5
            score.configure(text="Score:- " + str(points))
            messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
            ran_num = randrange(0, (len(ANIMALS_WORD)))
            word.configure(text=ANIMALS_WORD[ran_num])
            get_input.delete(0, END)
            ans_lab.configure(text="")
        else:
            messagebox.showerror("Error", "Inorrect Answer..Try your best!")
            get_input.delete(0, END)
    def show_answer():
        global points
        if points > 4:
            points -= 5
            score.configure(text="Score:- " + str(points))
            time.sleep(0.5)
            ans_lab.configure(text=ANIMALS_ANSWER[ran_num])
        else:
            ans_lab.configure(text='Not enough points')
    my_window = Tk()
    my_window.geometry("500x500+500+150")
    my_window.resizable(0, 0)
    my_window.title("Quizee-Animals_jumbled_words ")
    my_window.configure(background="#e6fff5")
    my_window.iconbitmap(r'quizee_logo_.ico')
    img1 = PhotoImage(file="back.png")
    lab_img1 = Button(
        my_window,
        image=img1,
        bg='#e6fff5',
        border=0,
        justify='center',
        command=back,
    )
    lab_img1.pack(anchor='nw', pady=10, padx=10)
    score = Label(
        text="Score:- 0",
        pady=10,
        bg="#e6fff5",
        fg="#000000",
        font="Titillium 14 bold"
    )
    score.pack(anchor="n")
    word = Label(
        text=jumbled_rand_word,
        pady=10,
        bg="#e6fff5",
        fg="#000000",
        font="Titillium 50 bold"
    )
    word.pack()
    get_input = Entry(
        font="none 26 bold",
        borderwidth=10,
        justify='center',
    )
    get_input.pack()
    submit = Button(
        text="Submit",
        width=18,
        borderwidth=8,
        font=("", 13),
        fg="#000000",
        bg="#99ffd6",
        command=cheak,
    )
    submit.pack(pady=(10, 20))
    change = Button(
        text="Change Word",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=change,
    )
    change.pack()
    ans = Button(
        text="Answer",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=show_answer,
    )
    ans.pack(pady=(20, 10))
    ans_lab = Label(
        text="",
        bg="#e6fff5",
        fg="#000000",
        font="Courier 15 bold",
    )
    ans_lab.pack()
    my_window.mainloop()
```

body_parts.py:

```
from tkinter import *

from random import *

from tkinter import messagebox

import time

BODY_PARTS_WORD = ['HEDA', 'IHAR', 'EYE', 'SEAR', 'SEON', 'UHMTO', 'NICH',
'OEEAFHDR', 'AJW', 'EKEHC', 'BOEWEYR',

                  'DHRSULOE', 'RSAM', 'ANDH', 'OBWEL', 'NGEIFR', 'GLE', 'FOTO', 'IHGTH',
'NEKE', ]


BODY_PARTS_ANSWER = ['HEAD', 'HAIR', 'EYE', 'EARS', 'NOSE', 'MOUTH', 'CHIN',
'FOREHEAD', 'JAW', 'CHEEK', 'EYEBROW',

                    'SHOULDER', 'ARMS', 'HAND', 'ELBOW', 'FINGER', 'LEG', 'FOOT',
'THIGH', 'KNEE', ]

ran_num = randrange(0, (len(BODY_PARTS_WORD)))

jumbled_rand_word = BODY_PARTS_WORD[ran_num]

points = 0

def main():

    def back():

        my_window.destroy()

        import main_start

        main_start.start_main_page()

    def change():

        global ran_num

        ran_num = randrange(0, (len(BODY_PARTS_WORD)))

        word.configure(text=BODY_PARTS_WORD[ran_num])

        get_input.delete(0, END)

        ans_lab.configure(text="")
```

```
def cheak():
    global points, ran_num
    user_word = get_input.get().upper()
    if user_word == BODY_PARTS_ANSWER[ran_num]:
        points += 5
        score.configure(text="Score:- " + str(points))
        messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
        ran_num = randrange(0, (len(BODY_PARTS_WORD)))
        word.configure(text=BODY_PARTS_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    else:
        messagebox.showerror("Error", "Inorrect Answer..Try your best!")
        get_input.delete(0, END)
def show_answer():
    global points
    if points > 4:
        points -= 5
        score.configure(text="Score:- " + str(points))
        time.sleep(0.5)
        ans_lab.configure(text=BODY_PARTS_ANSWER[ran_num])
    else:
        ans_lab.configure(text='Not enough points')
my_window = Tk()
my_window.geometry("500x500+500+150")
my_window.resizable(0, 0)
my_window.title("Quizee-Body_parts_jumbled_words")
my_window.configure(background="#e6fff5")
my_window.iconbitmap(r'quizee_logo_.ico')
img1 = PhotoImage(file="back.png")
lab_img1 = Button(
    my_window,
    image=img1,
    bg='#e6fff5',
    border=0,
    justify='center',
    command=back,
)
lab_img1.pack(anchor='nw', pady=10,padx=10)
score = Label(
    text="Score:- 0",
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 14 bold"
)
score.pack(anchor="n")
word = Label(
    text=jumbled_rand_word,
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 50 bold"
)
```

```
word.pack()
get_input = Entry(
    font="none 26 bold",
    borderwidth=10,
    justify='center',
)
  get_input.pack()
 submit = Button(
     text="Submit",
    width=18,
    borderwidth=8,
    font=("", 13),
    fg="#000000",
    bg="#99ffd6",
    command=cheak,
)
submit.pack(pady=(10,20))
change = Button(
    text="Change Word",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=change,
)
change.pack()
```

```
  ans = Button(
      text="Answer",
      width=18,
      borderwidth=8,
      fg="#000000",
      bg="#99ffd6",
      font=("", 13),
      command=show_answer,
  )
  ans.pack(pady=(20, 10))
  ans_lab = Label(
      text="",
      bg="#e6fff5",
      fg="#000000",
      font="Courier 15 bold",
  )
  ans_lab.pack()
  my_window.mainloop()
```

Colour.py:

```
from tkinter import *
from random import *
from tkinter import messagebox
import time
COLOUR_WORD = ['DRE', 'OELLYW', 'LBEU', 'ERGNE', 'OREAGN', 'RPEULP', 'EMIL',
'ORWBN', 'NYAV', 'PNIK', 'LDGO', 'VRLSEI',
            'BLKAC', 'HTWIE', ]
```

```
COLOUR_ANSWER = ['RED', 'YELLOW', 'BLUE', 'GREEN', 'ORANGE', 'PURPLE',
'LIME', 'BROWN', 'NAVY', 'PINK', 'GOLD',
                'SILVER', 'BLACK', 'WHITE', ]
ran_num = randrange(0, (len(COLOUR_WORD)))
jumbled_rand_word = COLOUR_WORD[ran_num]
points = 0
def main():
    def back():
        my_window.destroy()
        import main_start
        main_start.start_main_page()
    def change():
        global ran_num
        ran_num = randrange(0, (len(COLOUR_WORD)))
        word.configure(text=COLOUR_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    def cheak():
        global points, ran_num
        user_word = get_input.get().upper()
        if user_word == COLOUR_ANSWER[ran_num]:
            points += 5
            score.configure(text="Score:- " + str(points))
            messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
            ran_num = randrange(0, (len(COLOUR_WORD)))
            word.configure(text=COLOUR_WORD[ran_num])
            get_input.delete(0, END)
            ans_lab.configure(text="")
        else:
            messagebox.showerror("Error", "Inorrect Answer..Try your best!")
            get_input.delete(0, END)
```

```
def show_answer():
    global points
    if points > 4:
        points -= 5
        score.configure(text="Score:- " + str(points))
        time.sleep(0.5)
        ans_lab.configure(text=COLOUR_ANSWER[ran_num])
    else:
        ans_lab.configure(text='Not enough points')
my_window = Tk()
my_window.geometry("500x500+500+150")
my_window.resizable(0, 0)
my_window.title("Quizee-Colours_jumbled_words")
my_window.configure(background="#e6fff5")
my_window.iconbitmap(r'quizee_logo_.ico')
img1 = PhotoImage(file="back.png")
lab_img1 = Button(
    my_window,
    image=img1,
    bg='#e6fff5',
    border=0,
    justify='center',
    command=back,
)
lab_img1.pack(anchor='nw', pady=10, padx=10)
score = Label(
    text="Score:- 0",
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 14 bold"
)
score.pack(anchor="n")
word = Label(
    text=jumbled_rand_word,
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 30 bold"
)
word.pack()
get_input = Entry(
    font="none 26 bold",
    borderwidth=10,
    justify='center',
)
  get_input.pack()
 submit = Button(
     text="Submit",
    width=18,
    borderwidth=8,
    font=("", 13),
    fg="#000000",
    bg="#99ffd6",
    command=cheak,
)
submit.pack(pady=(10, 20))

change = Button(
    text="Change Word",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=change,
)
change.pack()
ans = Button(
    text="Answer",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=show_answer,
)
ans.pack(pady=(20, 10))
ans_lab = Label(
    text="",
    bg="#e6fff5",
    fg="#000000",
    font="Courier 15 bold",
)
ans_lab.pack()
my_window.mainloop()
```

Fruit.py:

```
from tkinter import *
from random import *
from tkinter import messagebox
```

```
import time
FRUITS_WORD = ['EALPP', 'VAAGU', 'PEHCA', 'PRAE', 'NGMAO', 'PAAPYA',
'EGNRAO', 'AGESPR', 'WIKI', 'RRYCHE', 'LRTOANWEME',
               'PEELIPNPA', 'BELUEBRRY', 'NABAAN', 'NTUOCCO', 'SCURATD EALPP',
'MONLE', 'EREYBMUL', 'MARTAIND']

FRUITS_ANSWER = ['APPLE', 'GUAVA', 'PEACH', 'PEAR', 'MANGO', 'PAPAYA',
'ORANGE', 'GRAPES', 'KIWI', 'CHERRY',
                 'WATERMELON', 'PINEAPPLE', 'BLUEBERRY', 'BANANA', 'COCONUT',
'CUSTARD APPLE', 'LEMON', 'MULBERRY',
                 'TAMARIND']
ran_num = randrange(0, (len(FRUITS_WORD)))
jumbled_rand_word = FRUITS_WORD[ran_num]
points = 0
def main():
    def back():
        my_window.destroy()
        import main_start
        main_start.start_main_page()
    def change():
        global ran_num
        ran_num = randrange(0, (len(FRUITS_WORD)))
        word.configure(text=FRUITS_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    def cheak():
        global points, ran_num
        user_word = get_input.get().upper()
        if user_word == FRUITS_ANSWER[ran_num]:
            points += 5
            score.configure(text="Score:- " + str(points))
            messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
            ran_num = randrange(0, (len(FRUITS_WORD)))
            word.configure(text=FRUITS_WORD[ran_num])
            get_input.delete(0, END)
            ans_lab.configure(text="")
        else:
            messagebox.showerror("Error", "Inorrect Answer..Try your best!")
            get_input.delete(0, END)
    def show_answer():
        global points
        if points > 4:
            points -= 5
            score.configure(text="Score:- " + str(points))
            time.sleep(0.5)
            ans_lab.configure(text=FRUITS_ANSWER[ran_num])
        else:
            ans_lab.configure(text='Not enough points')
    my_window = Tk()
    my_window.geometry("500x500+500+150")
    my_window.resizable(0, 0)
    my_window.title("Quizee-Friuts_jumbled_words")
    my_window.configure(background="#e6fff5")
    my_window.iconbitmap(r'quizee_logo_.ico')
    img1 = PhotoImage(file="back.png")
    lab_img1 = Button(
        my_window,
        image=img1,
        bg='#e6fff5',
        border=0,
        justify='center',
        command=back,
    )
    lab_img1.pack(anchor='nw', pady=10, padx=10)
    score = Label(
        text="Score:- 0",
        pady=10,
        bg="#e6fff5",
        fg="#000000",
        font="Titillium 14 bold"
    )
    score.pack(anchor="n")
    word = Label(
        text=jumbled_rand_word,
        pady=10,
        bg="#e6fff5",
        fg="#000000",
        font="Titillium 30 bold"
    )
    word.pack()
    get_input = Entry(
        font="none 26 bold",
        borderwidth=10,
        justify='center',
    )
      get_input.pack()
     submit = Button(
         text="Submit",
        width=18,
        borderwidth=8,
        font=("", 13),
        fg="#000000",
        bg="#99ffd6",
        command=cheak,
    )
    submit.pack(pady=(10,20))
    change = Button(
        text="Change Word",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=change,
    )
    change.pack()
    ans = Button(
        text="Answer",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=show_answer,
    )
    ans.pack(pady=(20, 10))
    ans_lab = Label(
        text="",
        bg="#e6fff5",
        fg="#000000",
        font="Courier 15 bold",
    )
```

```
    ans_lab.pack()
    my_window.mainloop()
```

Shapes.py:

```
from tkinter import *
from random import *
from tkinter import messagebox
import time
SHAPES_WORD = ['CCLIER', 'IDANDOM', 'RHATE', 'AOTNOCG', 'USQRAE', 'TARS',
'RITLANEG', ]


SHAPES_ANSWER = ['CIRCLE', 'DIAMOND', 'HEART', 'OCTAGON', 'SQUARE',
'STAR', 'TRIANGLE', ]
ran_num = randrange(0, (len(SHAPES_WORD)))
jumbled_rand_word = SHAPES_WORD[ran_num]
points = 0
def main():
    def back():
        my_window.destroy()
        import main_start
        main_start.start_main_page()
    def change():
        global ran_num
        ran_num = randrange(0, (len(SHAPES_WORD)))
        word.configure(text=SHAPES_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    def cheak():
        global points, ran_num
        user_word = get_input.get().upper()
        if user_word == SHAPES_ANSWER[ran_num]:
```

```
        points += 5
        score.configure(text="Score:- " + str(points))
        messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
        ran_num = randrange(0, (len(SHAPES_WORD)))
        word.configure(text=SHAPES_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    else:
        messagebox.showerror("Error", "Inorrect Answer..Try your best!")
        get_input.delete(0, END)
def show_answer():
    global points
    if points > 4:
        points -= 5
        score.configure(text="Score:- " + str(points))
        time.sleep(0.5)
        ans_lab.configure(text=SHAPES_ANSWER[ran_num])
    else:
        ans_lab.configure(text='Not enough points')
my_window = Tk()
my_window.geometry("500x500+500+150")
my_window.resizable(0, 0)
my_window.title("Quizee-Shapes_jumbled_words")
my_window.configure(background="#e6fff5")
my_window.iconbitmap(r'quizee_logo_.ico')
img1 = PhotoImage(file="back.png")
lab_img1 = Button(
    my_window,
    image=img1,
    bg='#e6fff5',
    border=0,
    justify='center',
    command=back,
)
lab_img1.pack(anchor='nw', pady=10, padx=10)
score = Label(
    text="Score:- 0",
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 14 bold"
)
score.pack(anchor="n")
word = Label(
    text=jumbled_rand_word,
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 50 bold"
)
word.pack()
get_input = Entry(
    font="none 26 bold",
    borderwidth=10,
    justify='center',
)
  get_input.pack()
 submit = Button(
     text="Submit",
    width=18,
    borderwidth=8,
    font=("", 13),
    fg="#000000",
    bg="#99ffd6",
    command=cheak,
)
submit.pack(pady=(10, 20))
change = Button(
    text="Change Word",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=change,
)
change.pack()
ans = Button(
    text="Answer",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=show_answer,
)
ans.pack(pady=(20, 10))
ans_lab = Label(
    text="",
    bg="#e6fff5",
        fg="#000000",
        font="Courier 15 bold",
    )
    ans_lab.pack()
    my_window.mainloop()
```

Vegetable.py:

```
from tkinter import *
from random import *
from tkinter import messagebox
import time
VEGETABLE_WORD = ['OTCRRA', 'OIRCCOBL', 'ROUFCWIALEL', 'RNOC', 'UCMUECRB', 'GGNALETP', 'EEGRN EREPPP', 'ECETTUL',
                  'OSMROSMUH', 'INNOO', 'OATPTO', 'UNPMIPK', 'RED EPEPPR', 'MOTTOA', 'ETEBTROO', 'EPAS', 'HIRSAD',
                  'CEBABAG', 'CLIHI', 'ICRGAL', 'WETSE OTPAOT', 'RERAOCDIN', ]

VEGETABLE_ANSWER = ['CARROT', 'BROCCOLI ', 'CAULIFLOWER ', 'CORN ', 'CUCUMBER ', 'EGGPLANT', 'GREEN PEPPER ',
                    'LETTUCE ', 'MUSHROOMS', 'ONION', 'POTATO', 'PUMPKIN ', 'RED PEPPER', 'TOMATO ', 'BEETROOT', 'PEAS',
                    'RADISH', 'CABBAGE', 'CHILI', 'GARLIC', 'SWEET POTATO', 'CORIANDER', ]
ran_num = randrange(0, (len(VEGETABLE_WORD)))
jumbled_rand_word = VEGETABLE_WORD[ran_num]
points = 0
def main():
    def back():
        my_window.destroy()
        import main_start
        main_start.start_main_page()
```

```
def change():
    global ran_num
    ran_num = randrange(0,  (len(VEGETABLE_WORD)))
    word.configure(text=VEGETABLE_WORD[ran_num])
    get_input.delete(0, END)
    ans_lab.configure(text="")
 def cheak():
    global points, ran_num
    user_word = get_input.get().upper()
    if user_word == VEGETABLE_ANSWER[ran_num]:
        points += 5
        score.configure(text="Score:- " + str(points))
        messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
        ran_num = randrange(0, (len(VEGETABLE_WORD)))
        word.configure(text=VEGETABLE_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    else:
        messagebox.showerror("Error", "Inorrect Answer..Try your best!")
        get_input.delete(0, END)
 def show_answer():
    global points
    if points > 4:
        points -= 5
        score.configure(text="Score:- " + str(points))
        time.sleep(0.5)
        ans_lab.configure(text=VEGETABLE_ANSWER[ran_num])
    else:
        ans_lab.configure(text='Not enough points'
 my_window = Tk()
```

```
my_window.geometry("500x500+500+150")
my_window.resizable(0, 0)
my_window.title("Quizee-Vegetables_jumbled_words")
my_window.configure(background="#e6fff5")
my_window.iconbitmap(r'quizee_logo_.ico')
img1 = PhotoImage(file="back.png")
lab_img1 = Button(
    my_window,
    image=img1,
    bg='#e6fff5',
    border=0,
    justify='center',
    command=back,
)
lab_img1.pack(anchor='nw', pady=10, padx=10)
score = Label(
    text="Score:- 0",
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 14 bold"
)
score.pack(anchor="n")
word = Label(
    text=jumbled_rand_word,
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 40 bold"
)
```

```
word.pack()
get_input = Entry(
    font="none 26 bold",
    borderwidth=10,
    justify='center',
)
  get_input.pack()
 submit = Button(
     text="Submit",
    width=18,
    borderwidth=8,
    font=("", 13),
    fg="#000000",
    bg="#99ffd6",
    command=cheak,
)
submit.pack(pady=(10, 20))
change = Button(
    text="Change Word",
   width=18,
   borderwidth=8,
   fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=change,
)
change.pack()
ans = Button(
    text="Answer",
    width=18,
    borderwidth=8,
    fg="#000000",
    bg="#99ffd6",
    font=("", 13),
    command=show_answer,
)
ans.pack(pady=(20, 10))
ans_lab = Label(
    text="",
    bg="#e6fff5",
    fg="#000000",
    font="Courier 15 bold",
)
ans_lab.pack()
my_window.mainloop()
```

Vehicles.py:

```
from tkinter import *
from random import *
from tkinter import messagebox
import time
VEHICLES_WORD = ['RIHLPOEETC', 'NELAARIP', 'CKTREO', 'LITSAOBA', 'UIECRS
PIHS', 'ROAGC SPHI', 'TJE SKI', 'PIREAT IHSP',
                'TBOA', 'SHIP', 'RUISAEMNB', 'IYLCECB', 'CAR', 'BUS', 'TIANR', 'UTKCR',
'NVA', 'LRTOMCCYEO', ]


VEHICLES_ANSWER = ['HELICOPTER', 'AIRPLANE', 'ROCKET', 'SAILBOAT',
'CRUISE SHIP', 'CARGO SHIP', 'JET SKI',
                 'PIRATE SHIP', 'BOAT', 'SHIP', 'SUBMARINE', 'BICYCLE', 'CAR', 'BUS',
'TRAIN', 'TRUCK', 'VAN',
                 'MOTORCYCLE', ]
```

```
ran_num = randrange(0, (len(VEHICLES_WORD)))
jumbled_rand_word = VEHICLES_WORD[ran_num]
points = 0
def main():
    def back():
        my_window.destroy()
        import main_start
        main_start.start_main_page()
    def change():
        global ran_num
        ran_num = randrange(0, (len(VEHICLES_WORD)))
        word.configure(text=VEHICLES_WORD[ran_num])
        get_input.delete(0, END)
        ans_lab.configure(text="")
    def cheak():
        global points, ran_num
        user_word = get_input.get().upper()
        if user_word == VEHICLES_ANSWER[ran_num]:
            points += 5
            score.configure(text="Score:- " + str(points))
            messagebox.showinfo('correct', "Correct Answer.. Keep it Up!")
            ran_num = randrange(0, (len(VEHICLES_WORD)))
            word.configure(text=VEHICLES_WORD[ran_num])
            get_input.delete(0, END)
            ans_lab.configure(text="")
        else:
            messagebox.showerror("Error", "Inorrect Answer..Try your best!")
            get_input.delete(0, END)
    def show_answer():
        global points
        if points > 4:
            points -= 5
            score.configure(text="Score:- " + str(points))
            time.sleep(0.5)
            ans_lab.configure(text=VEHICLES_ANSWER[ran_num])
            change()
        else:
            ans_lab.configure(text='Not enough points')
    my_window = Tk()
    my_window.geometry("500x500+500+150")
    my_window.resizable(0, 0)
    my_window.title("Quizee-Vehicles_jumbled_words")
    my_window.configure(background="#e6fff5")
    my_window.iconbitmap(r'quizee_logo_.ico')
    img1 = PhotoImage(file="back.png")
    lab_img1 = Button(
        my_window,
        image=img1,
        bg='#e6fff5',
        border=0,
        justify='center',
        command=back,
    )
    lab_img1.pack(anchor='nw', pady=10, padx=10)
    score = Label(
        text="Score:- 0",
        pady=10,
        bg="#e6fff5",
        fg="#000000",
        font="Titillium 14 bold"
```

```
)
score.pack(anchor="n")
word = Label(
    text=jumbled_rand_word,
    pady=10,
    bg="#e6fff5",
    fg="#000000",
    font="Titillium 45 bold"
)
word.pack()
get_input = Entry(
    font="none 26 bold",
    borderwidth=10,
    justify='center',
)
  get_input.pack()
 submit = Button(
     text="Submit",
    width=18,
    borderwidth=8,
    font=("", 13),
    fg="#000000",
    bg="#99ffd6",
    command=cheak,
)
submit.pack(pady=(10, 20))
change = Button(
    text="Change Word",
    width=18,
    borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=change,
    )
    change.pack()
    ans = Button(
        text="Answer",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        command=show_answer,
    )
    ans.pack(pady=(20, 10))
    ans_lab = Label(
        text="",
        bg="#e6fff5",
        fg="#000000",
        font="Courier 15 bold",
    )
    ans_lab.pack()
    my_window.mainloop()
```

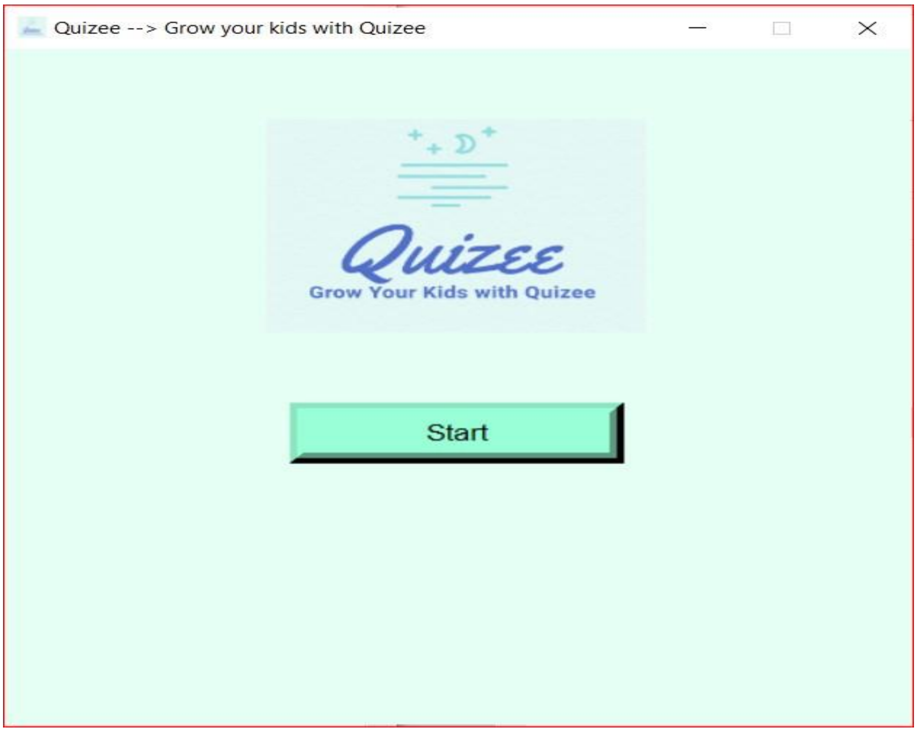
main_start.py:

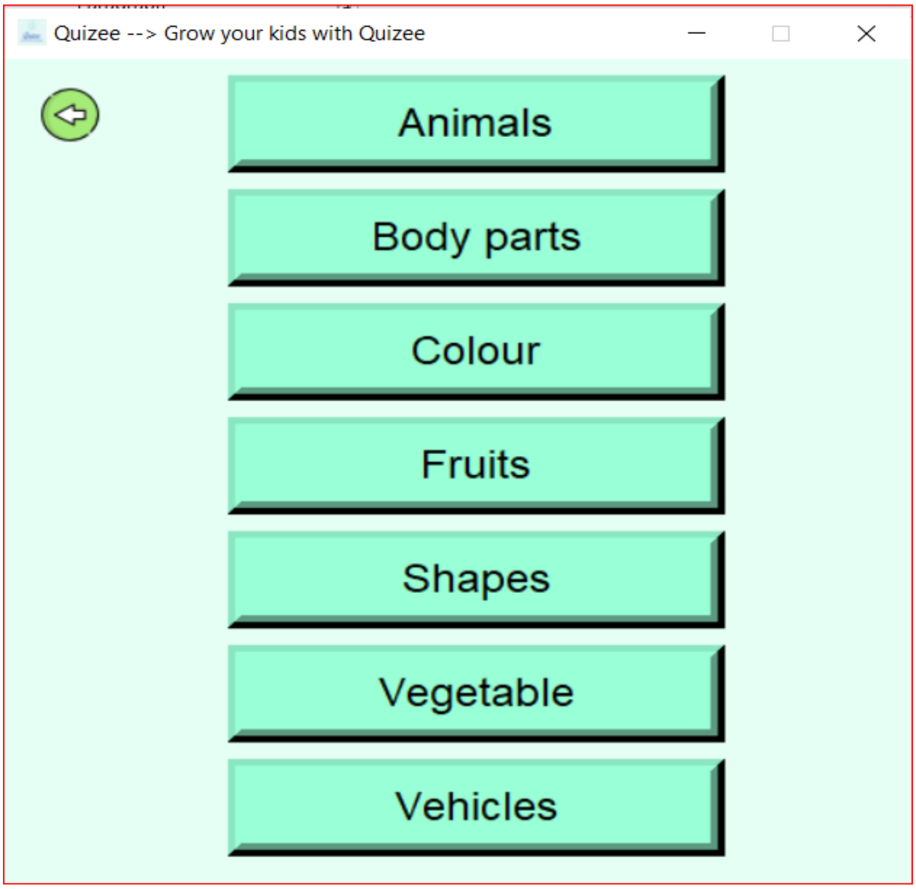
```
from tkinter import *
def start_main_page():
    def start_game(args):
        main_window.destroy()
        if args == 1:
```

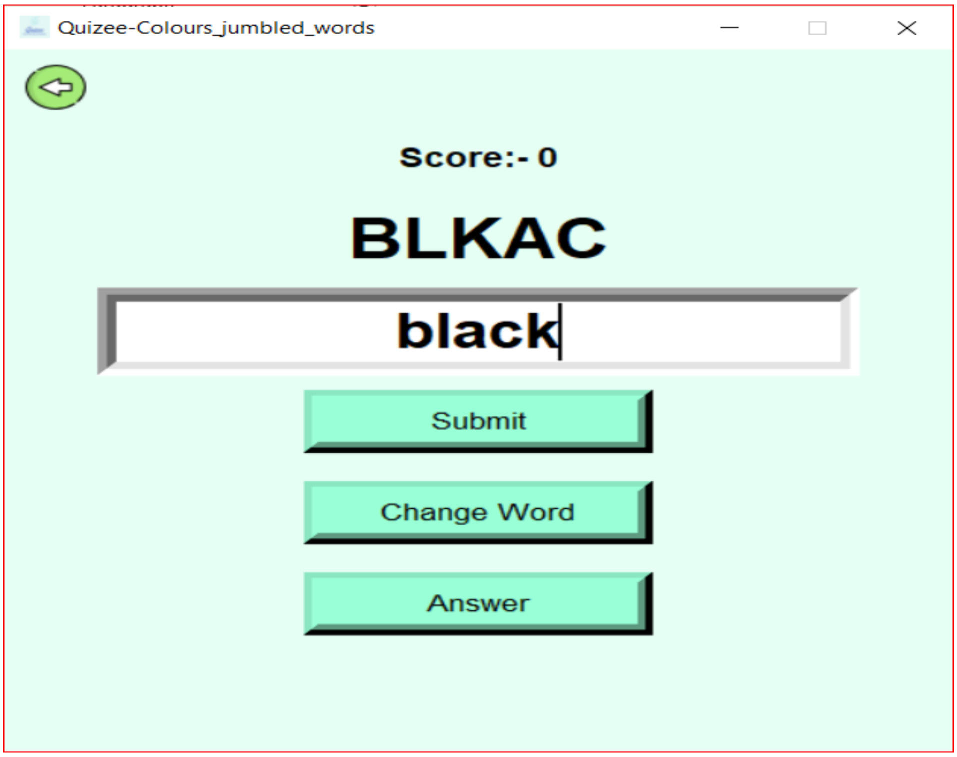
```
        from Options import Animals
        Animals.main()
    elif args == 2:
        from Options import Body_parts
        Body_parts.main()
    elif args == 3:
        from Options import Colour
        Colour.main()
    elif args == 4:
        from Options import Fruit
        Fruit.main()
    elif args == 5:
        from Options import Shapes
        Shapes.main()
    elif args == 6:
        from Options import Vegetable
        Vegetable.main()
    elif args == 7:
        from Options import Vehicles
        Vehicles.main()

def option():

    lab_img1 = Button(
        main_window,
        image=img1,
        bg='#e6fff5',
        border=0,
        justify='center',
)
sel_btn1 = Button(
    text="Animals",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(1),
)
sel_btn2 = Button(
    text="Body parts",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(2),
)
sel_btn3 = Button(
    text="Colour",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(3),
)
sel_btn4 = Button(
    text="Fruits",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(4),
)
sel_btn5 = Button(
    text="Shapes",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(5),
)
sel_btn6 = Button(
    text="Vegetable",
    width=18,
    borderwidth=8,
    font=("", 18),
    fg="#000000",
    bg="#99ffd6",
    cursor="hand2",
    command=lambda: start_game(6),
    )
    sel_btn7 = Button(
        text="Vehicles",
        width=18,
        borderwidth=8,
        font=("", 18),
        fg="#000000",
        bg="#99ffd6",
        cursor="hand2",
        command=lambda: start_game(7),
    )
    lab_img1.grid(row=0, column=0, padx=20)
    sel_btn1.grid(row=0, column=4, pady=(10, 0), padx=50, )
    sel_btn2.grid(row=1, column=4, pady=(10, 0), padx=50, )
    sel_btn3.grid(row=2, column=4, pady=(10, 0), padx=50, )
    sel_btn4.grid(row=3, column=4, pady=(10, 0), padx=50, )
    sel_btn5.grid(row=4, column=4, pady=(10, 0), padx=50, )
    sel_btn6.grid(row=5, column=4, pady=(10, 0), padx=50, )
    sel_btn7.grid(row=6, column=4, pady=(10, 0), padx=50, )
def show_option():
    start_btn.destroy()
    lab_img.destroy()
    option()
main_window = Tk()
main_window.geometry("500x500+500+150")
main_window.resizable(0, 0)
main_window.title("Quizee --> Grow your kids with Quizee")
main_window.configure(background="#e6fff5")
main_window.iconbitmap(r'quizee_logo_.ico')
```

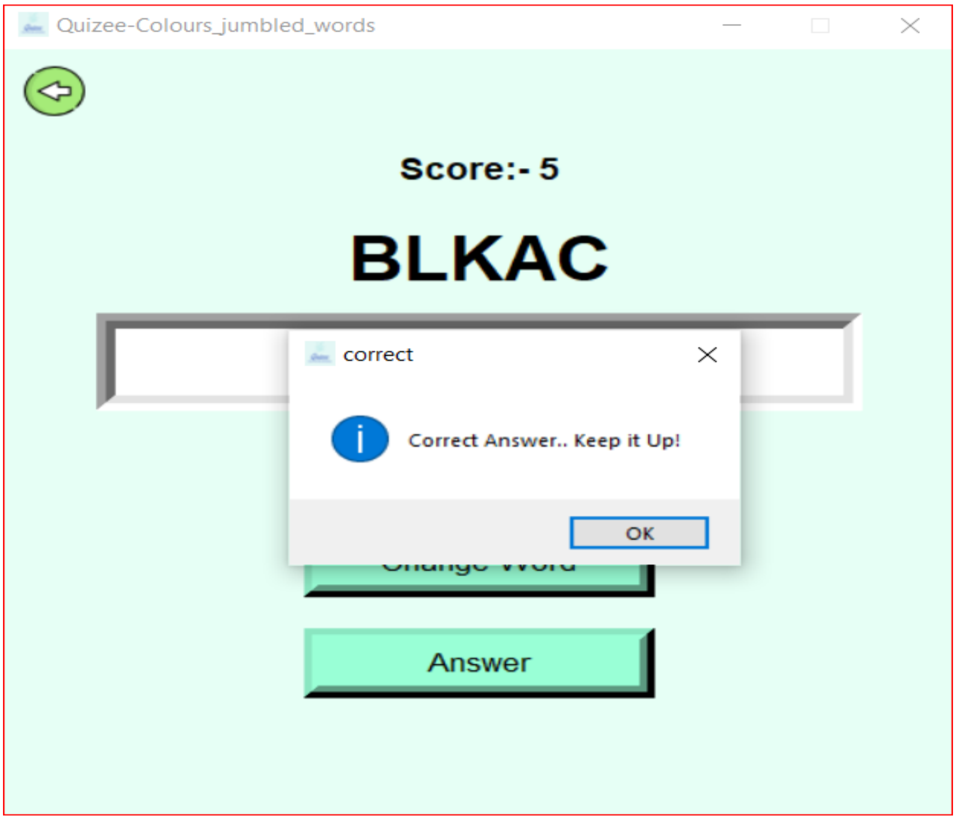
```
    img0 = PhotoImage(file="quizee_logo.png")
    img1 = PhotoImage(file="back.png")
    lab_img = Label(
        main_window,
        image=img0,
        bg='#e6fff5',
    )
    lab_img.pack(pady=(50, 0))
    start_btn = Button(
        main_window,
        text="Start",
        width=18,
        borderwidth=8,
        fg="#000000",
        bg="#99ffd6",
        font=("", 13),
        cursor="hand2",
        command=show_option,
    )
    start_btn.pack(pady=(50, 20))
    main_window.mainloop()
start_main_page()
```

OUTPUT SCREENSHOTS:

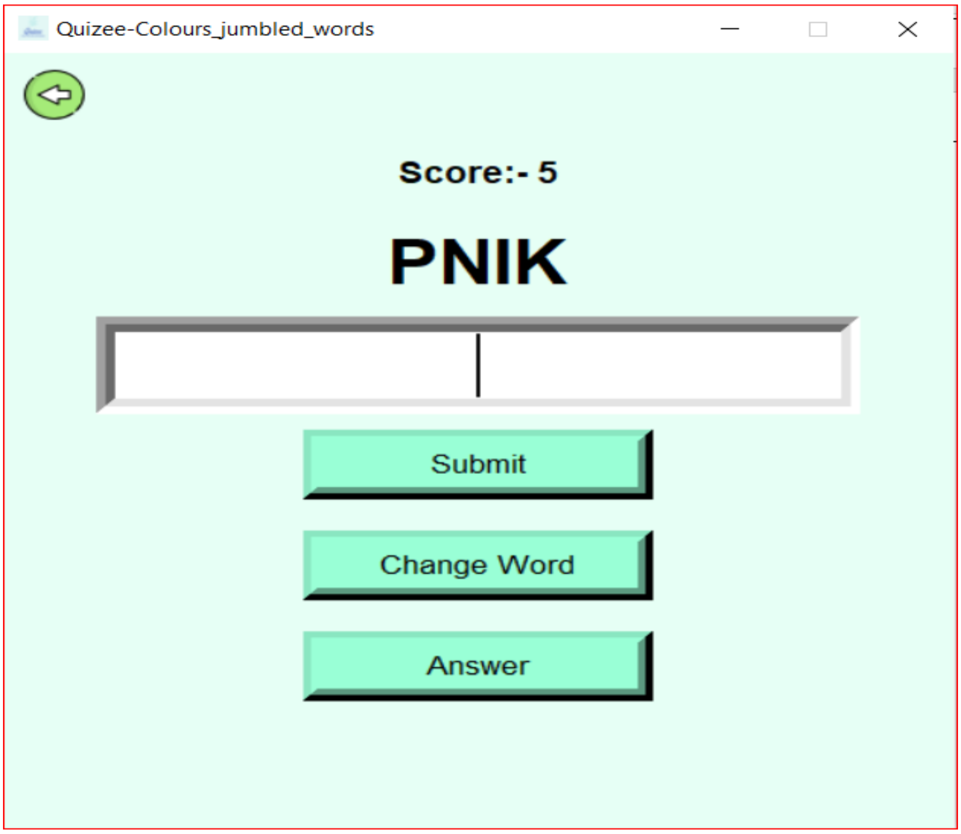

Quizee --> Grow your kids with Quizee
Quizee
Grow Your Kids with Quizee
Start





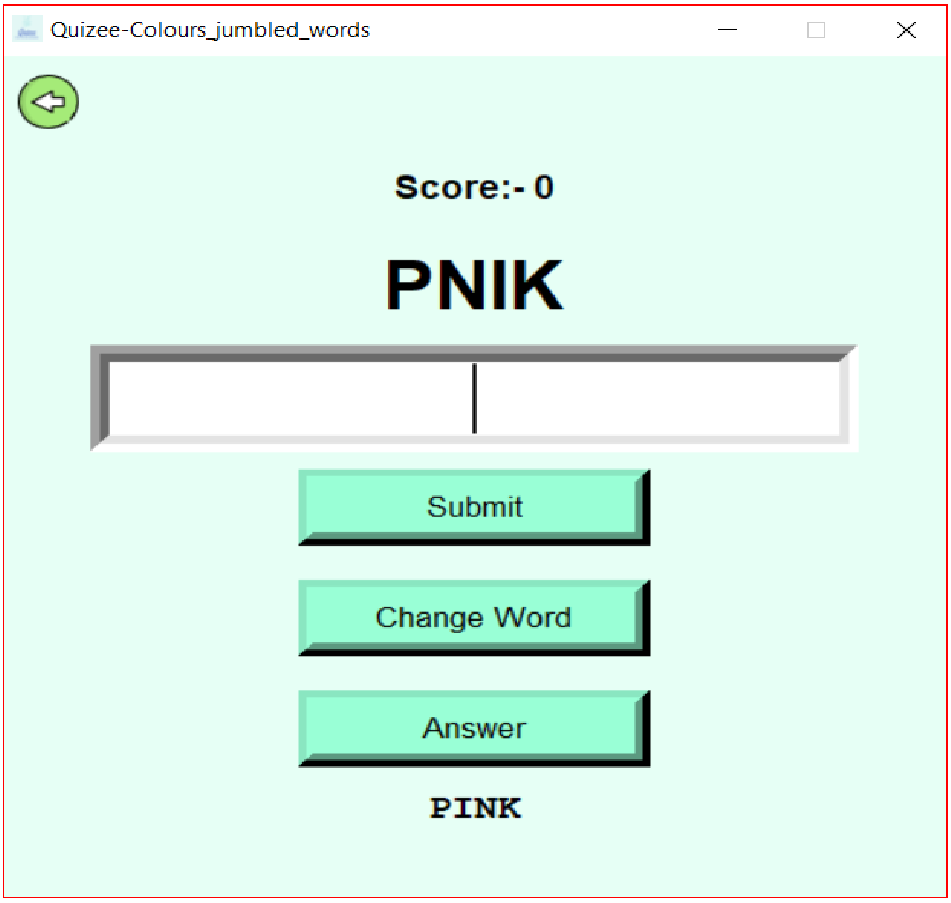


Correct Answer.. Keep it Up!


Quizee-Colours_jumbled_words

Score:- 5

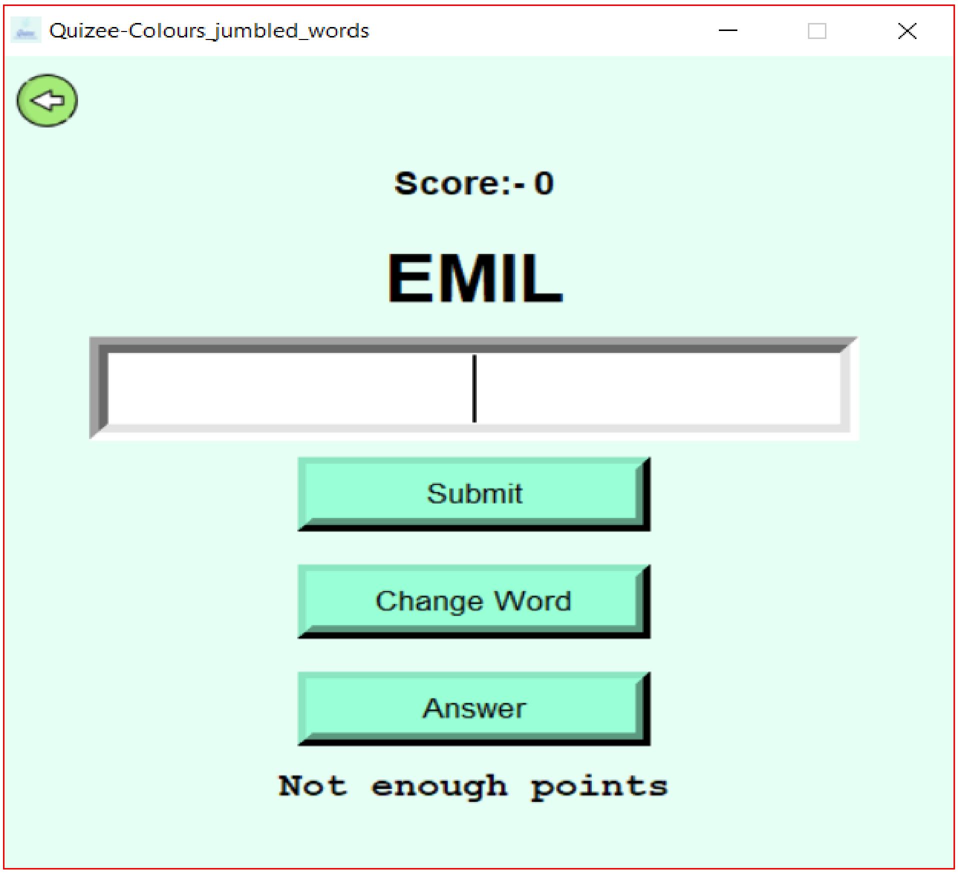
PNIK


Quizee-Colours_jumbled_words

Score:- 0

PNIK

PINK

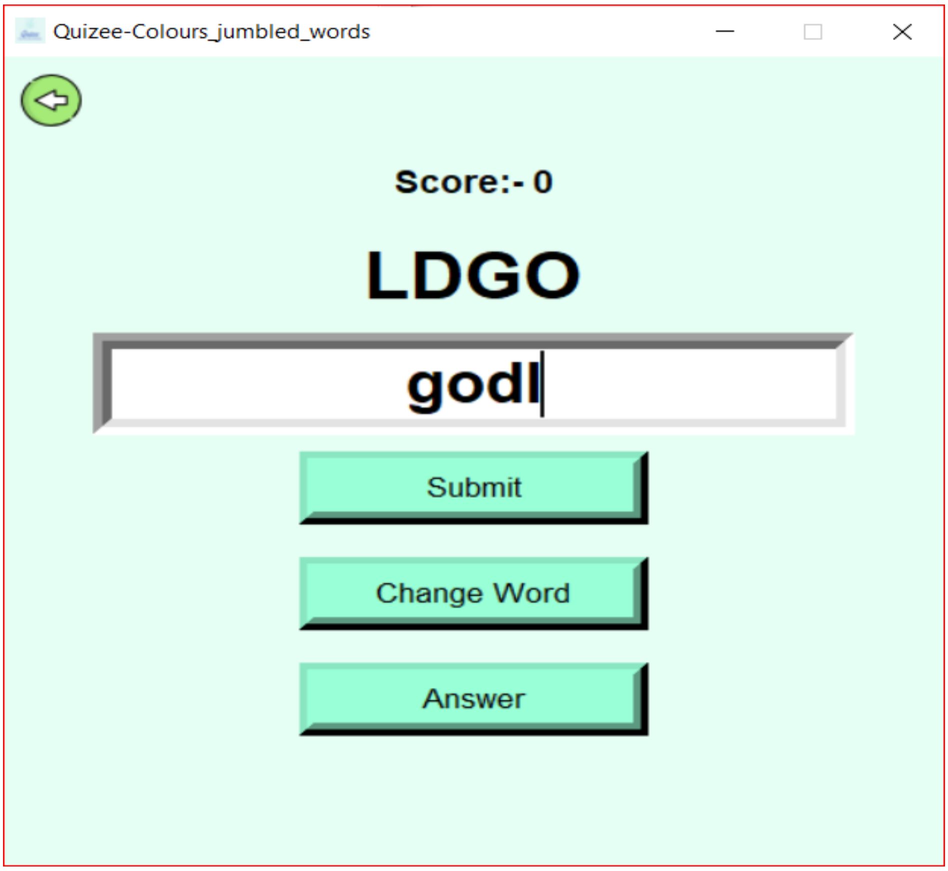


Quizee-Colours_jumbled_words

Score:- 0

EMIL

Not enough points

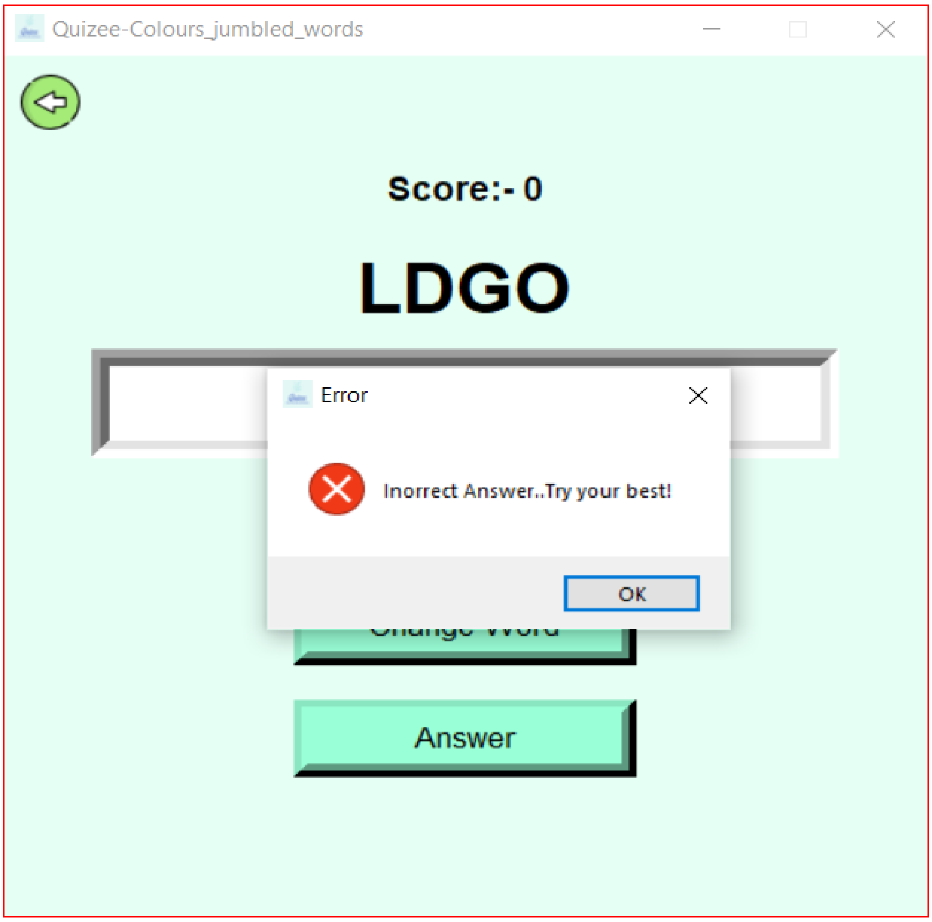


Quizee-Colours_jumbled_words

Score:- 0

LDGO

godl



Inorrect Answer..Try your best!

CONCLUSION AND FUTURE WORK:

To conclude, this game lets children have fun while also increasing their IQ. We wish to carry this spirit and work on many more projects in future.

REFERENCES

TkinterWidgets- https://www.geeksforgeeks.org/python-gui-tkinter/