import tkinter as tk

from tkinter import messagebox

import random

# Quiz data (questions, options, and correct answers)

quiz\_data = [

{

"question": "Who develope python?",

"options": ['Zim den', 'Niene stom', 'Guido van rossum', 'Wick van rossum'],

"correct\_answer": "Guido van rossum"

},

{

"question": "Which extension we use in python?",

"options": ['.python', '.py', '.pyl', '.p'],

"correct\_answer": ".py"

},

{

"question": "use of print()?",

"options": ['All', 'to take i/p', 'to read', 'to print'],

"correct\_answer": "to print"

},

{

"question": "python is case sensitive?",

"options": ['yes', 'no', 'machine depedant', 'none of this'],

"correct\_answer": "yes"

},

{

"question": "print(x<<2)?",

"options": ['1', '4', '8', '2'],

"correct\_answer": "4"

}

]

class QuizApp:

def \_\_init\_\_(self, root):

self.root = root

self.root.title("Quiz App")

self.root.geometry("400x400")

# Colors

self.bg\_color = "grey" # Light purple background

self.question\_color = "black" # Deep purple text color

self.option\_color = "black" # Purple text color

self.correct\_color = "#2E7D32" # Green text color for correct answer

self.incorrect\_color = "#C62828" # Red text color for incorrect answer

self.score = 0

self.question\_number = 0

self.seconds\_left = 15 # Change this to set the timer duration

self.timer\_running = False

self.label\_intro = tk.Label(root, text="Welcome to the Quiz App!\nYou have 15 seconds to answer each question.", fg='White', bg='grey')

self.label\_intro.pack(pady=10)

self.label\_question = tk.Label(root, text="", height=5, width=28, bg='black', fg="#fff", font=('Verdana', 15),wraplength=300)

self.label\_question.pack(pady=10)

self.radio\_var = tk.IntVar()

self.radio\_var.set(-1)

self.radio\_buttons = []

for i in range(4):

radio\_btn = tk.Radiobutton(root, text="", variable=self.radio\_var, value=i, fg=self.option\_color, bg=self.bg\_color)

radio\_btn.pack(anchor=tk.W)

self.radio\_buttons.append(radio\_btn)

self.label\_timer = tk.Label(root, text="Time left: {}s".format(self.seconds\_left), fg=self.option\_color, bg=self.bg\_color)

self.label\_timer.pack()

self.btn\_submit = tk.Button(root, text="Submit", command=self.check\_answer, bg=self.option\_color, fg="white")

self.btn\_submit.pack(pady=10)

self.load\_question()

def load\_question(self):

if self.question\_number < len(quiz\_data):

self.label\_question.config(text=quiz\_data[self.question\_number]["question"])

options = quiz\_data[self.question\_number]["options"]

random.shuffle(options)

for i in range(4):

self.radio\_buttons[i].config(text=options[i], fg=self.option\_color)

self.start\_timer()

else:

self.show\_score()

def start\_timer(self):

if not self.timer\_running:

self.timer\_running = True

self.countdown()

def countdown(self):

if self.seconds\_left > 0:

self.label\_timer.config(text="Time left: {}s".format(self.seconds\_left))

self.seconds\_left -= 1

self.root.after(1000, self.countdown)

else:

self.timer\_running = False

self.check\_answer()

def check\_answer(self):

if self.timer\_running:

self.timer\_running = False

selected\_option = self.radio\_var.get()

if selected\_option == -1:

messagebox.showinfo("Time's up!", "You didn't select an option. The correct answer is: {}".format(quiz\_data[self.question\_number]["correct\_answer"]))

else:

selected\_answer = self.radio\_buttons[selected\_option].cget("text")

correct\_answer = quiz\_data[self.question\_number]["correct\_answer"]

if selected\_answer == correct\_answer:

self.score += 1

self.radio\_buttons[selected\_option].config(fg=self.correct\_color)

else:

self.radio\_buttons[selected\_option].config(fg=self.incorrect\_color)

for i in range(4):

if self.radio\_buttons[i]["text"] == correct\_answer:

self.radio\_buttons[i].config(fg=self.correct\_color)

self.question\_number += 1

self.radio\_var.set(-1)

self.seconds\_left = 15 # Reset the timer for the next question

self.label\_timer.config(text="Time left: {}s".format(self.seconds\_left))

self.load\_question()

def show\_score(self):

messagebox.showinfo("Quiz Completed", "Your score: {}/{}".format(self.score, len(quiz\_data)))

self.root.destroy()

if \_\_name\_\_ == "\_\_main\_\_":

root = tk.Tk()

app = QuizApp(root)

root.configure(bg=app.bg\_color)

root.mainloop()