

Dharmsinh Desai University

Faculty of Management & Information Science



Course :- Master in Computer Application

Term Work Report

Subject :- Python Programming

Submitted by :-

Mandaliya Dishant Amit

Roll.no :- MA029

Identity No :- 22MAPOT033

Guided By :-

Prof. Narayan Joshi

signature

Index

Sr no.	Topics covered	Page no.
1	Introduction	2
2	How Game Works	2
3	Code	3
4	ScreenShots	7
5	FAQ's	8

Quizzie – The Quiz Taker

Introduction:

Welcome to the Quizzie - The Quiz Game! This interactive application was designed to test your knowledge and challenge your wits with a series of fun and diverse trivia questions. The game fetches questions from the Open Trivia Database API, ensuring a variety of topics and difficulties. Before you dive into the quiz, you need to log in with your username and password. Once you've successfully logged in, the quiz will begin.

Here's how the game works:

Login: You start by providing your username and password to access the quiz. If you enter the correct credentials, you'll be welcomed and granted access to the quiz.

Quiz Time: Once you've logged in, you'll be presented with a series of trivia questions. Each question is followed by multiple-choice answer options. Read the question carefully and select the correct answer by entering the corresponding number.

Scoring: For each correct answer, you earn a point. The game keeps track of your score, and your final score will be displayed at the end of the quiz.

Result: After you've answered all the questions, the game will display your total score out of the number of questions attempted. You can see how well you did and challenge yourself to improve your score next time.

Quizzie – The Quiz Taker

Source Code :-

```
import requests #imported request for fetching the questions from api
import tkinter as tk #imported tkinter for login and result
from tkinter import messagebox

class Quiz:
    def __init__(self):
        self.score = 0 #initializing score variable
    ##
    ## it fetched questions from API
    # API is generated by opendb trivia api
    #also it return json format data
    def fetch_questions_from_api(self):
        api_url = "https://opentdb.com/api.php?amount=10&difficulty=easy"

        response = requests.get(api_url) #get request sent to the url

        #A status code of 200 indicates a successful request. HTTP status code 200 typically
        means "OK."
        if response.status_code == 200:
            data = response.json() #this method converts json data into dictionary
            questions = data["results"]
            return questions
        else:
            print("Failed to fetch questions from the API.")
            return
    ##
    def start_quiz(self):
        #stores value of questions into question variable
        self.questions = self.fetch_questions_from_api()
        if not self.questions:
```

Quizzie – The Quiz Taker

```
return #returns empty if questions not found

##

for i, q in enumerate(self.questions, 1):

    print(f"Question {i}: {q['question']}")#prints the questions to the user.

    #choices" will have all the possible answer options for the current question.

    choices = q["incorrect_answers"] + [q["correct_answer"]]

    #his line processes each answer choice, encoding it as UTF-8 and then decoding it
    #using "unicode_escape."

    #This is a common operation to handle character encoding and escape sequences
    #special character in text data.

    choices = [choice.encode("utf-8").decode("unicode_escape") for choice in choices]

    choices.sort() # Shuffle the answer choices

    # #enumerate this iterate thai sake choices ma and proper print thai2

    for i, choice in enumerate(choices, 1):

        print(f"{i}. {choice}")

        #display the answer choices

        #allows you to create a formatted string with dynamic content without

        #the need for explicit concatenation or string formatting functions.

    userans = input("Enter the number of your answer: ")

    if userans.isdigit():

        userans = int(userans)

        if 1 <= userans <= len(choices): #there should be only one anser

            if choices[userans - 1] == q["correct_answer"]:

                print("Correct!\n")

                self.score += 1

            else:

                print("Wrong answer!\n")

        else:
```

Quizzie – The Quiz Taker

```
        print("Invalid input. Please enter a valid choice.\n")
    else:
        print("Invalid input. Please enter a number.\n")

    self.show_result()

##
def show_result(self):
    #displays the result
    tot_que = len(self.questions)
    print(f"Quiz completed! Your score is: {self.score}/{tot_que}")
    messagebox.showinfo("End",f"Your Score, {self.score}/{tot_que}")

##
def login(self):
    self.login_window = tk.Tk()
    self.login_window.title("Login")

    #using simple label and textbox for username
    self.ulabel = tk.Label(self.login_window, text="Username:")
    self.ulabel.pack()

    self.u_txtbox = tk.Entry(self.login_window)
    self.u_txtbox.pack()

    self.plabel = tk.Label(self.login_window, text="Password:")
    self.plabel.pack()

    #password hidden char is * so value of show is *
    self.p_txtbox = tk.Entry(self.login_window, show="*") # Show asterisks for password
    self.p_txtbox.pack()
```

Quizzie – The Quiz Taker

```
#button for login and calling the function check_login on click tp verify the details

self.login_button = tk.Button(self.login_window, text="Login",
command=self.check_login)

self.login_button.pack()


self.login_window.mainloop()

##

def check_login(self):
    username = self.u_txtbox.get()
    password = self.p_txtbox.get()

    user_data = {
        "dishant": "123",
        "ram": "jaishriram",
        "krishna": "jaishrikrishna",
    }

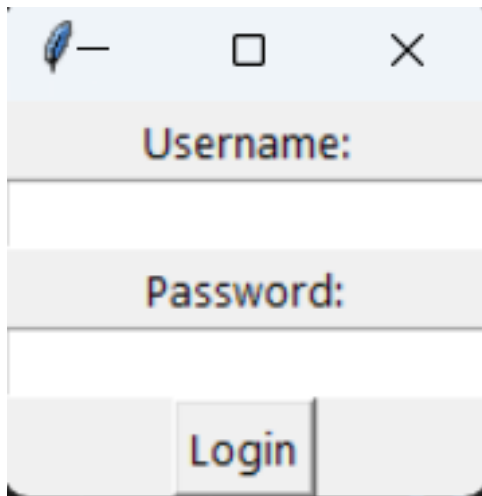
    if username in user_data and user_data[username] == password:
        messagebox.showinfo("Login Successful", f"Welcome, {username}!")
        self.login_window.destroy() # Close the login window and enter the quiz method
        self.start_quiz()
    else:
        messagebox.showerror("Login Failed", "Invalid username or password")

##

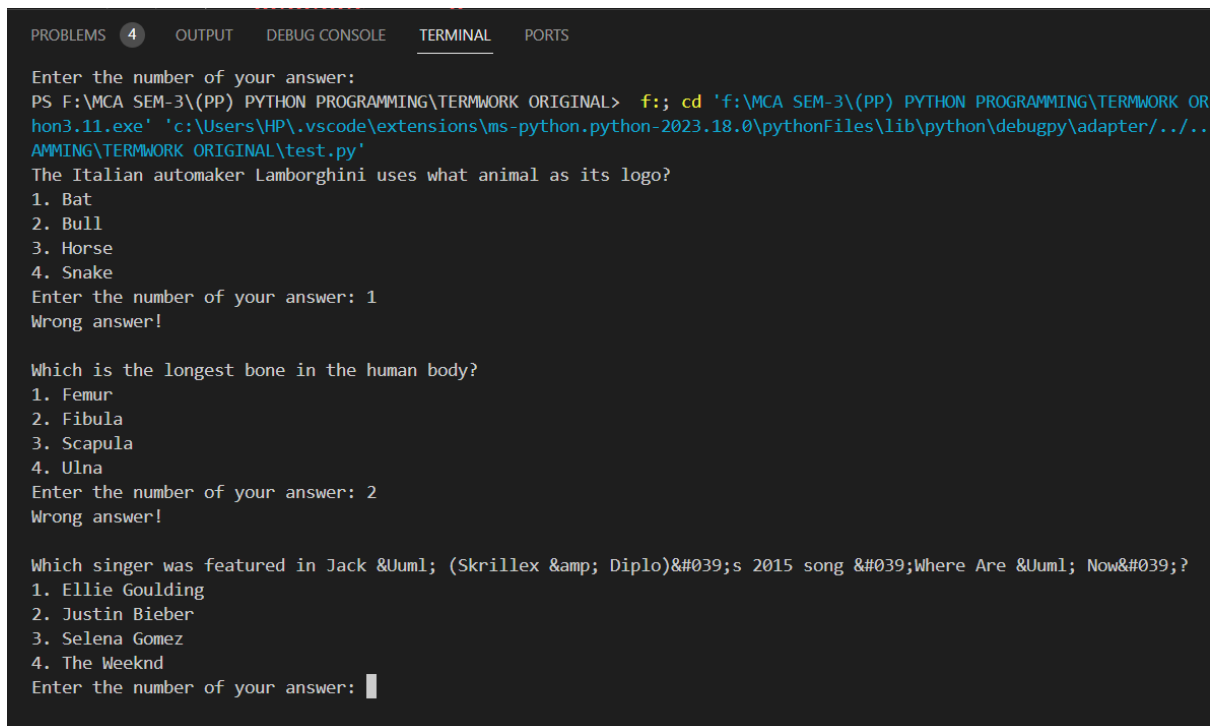
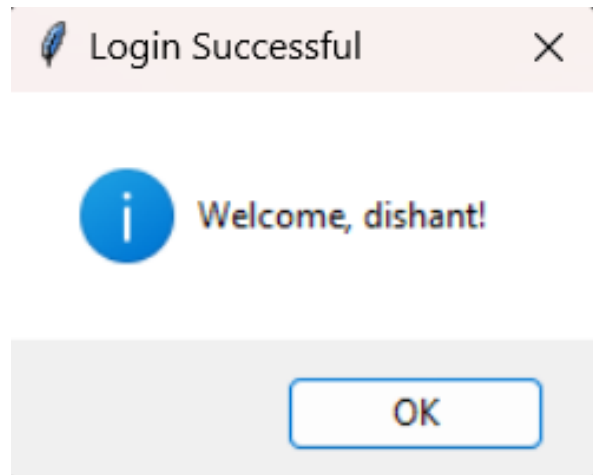
quiz = Quiz() #creatign objecvt of quiz class
q = quiz.login()
if q == True:
    quiz.start_quiz()
```

Quizzie – The Quiz Taker

ScreenShots :-



A login form window with a title bar containing a feather icon, a minus button, a maximize button, and a close button. The form has two input fields: 'Username:' and 'Password:'. Below the password field is a 'Login' button.

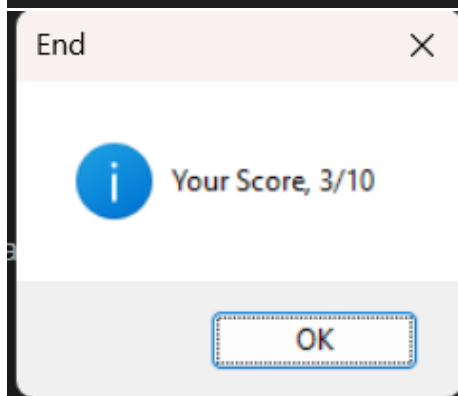


A screenshot of a VS Code terminal window showing a Python script execution. The terminal output includes a prompt to enter a number, a list of animals (Bat, Bull, Horse, Snake), a question about the Lamborghini logo, and a list of bones (Femur, Fibula, Scapula, Ulna). The user has entered '1' and '2', both of which are marked as 'Wrong answer!'. The terminal also shows the file path for the script and the location of the Python interpreter.

```
PROBLEMS 4 OUTPUT DEBUG CONSOLE TERMINAL PORTS
Enter the number of your answer:
PS F:\MCA SEM-3\PP) PYTHON PROGRAMMING\TERMWORK ORIGINAL> f;; cd 'f:\MCA SEM-3\PP) PYTHON PROGRAMMING\TERMWORK ORIGINAL\test.py'
The Italian automaker Lamborghini uses what animal as its logo?
1. Bat
2. Bull
3. Horse
4. Snake
Enter the number of your answer: 1
Wrong answer!

Which is the longest bone in the human body?
1. Femur
2. Fibula
3. Scapula
4. Ulna
Enter the number of your answer: 2
Wrong answer!

Which singer was featured in Jack & (Skrillex & Diplo)'s 2015 song 'Where Are You Now'?
1. Ellie Goulding
2. Justin Bieber
3. Selena Gomez
4. The Weeknd
Enter the number of your answer: 
```



Quizzie – The Quiz Taker

FAQ'S – Frequently Asked Questions

1. What is the purpose of this quiz game?

The purpose of this quiz game is to entertain and challenge users by presenting a series of trivia questions from various categories and difficulties. Users can test their knowledge, earn points for correct answers, and see their final scores.

2. How do I start playing the quiz?

To start playing the quiz, you need to log in with your username and password. Once you've successfully logged in, the quiz will begin.

3. Where do the quiz questions come from?

The quiz questions are fetched from the Open Trivia Database API. This ensures a variety of topics and difficulties in the questions.

4. How do I answer the quiz questions?

After each question is presented, you'll see multiple-choice answer options. Simply enter the corresponding number of your chosen answer to submit your response.

5. How is my score calculated?

For each correct answer, you earn one point. Your total score is calculated based on the number of questions you answered correctly.

6. What happens if I enter the wrong username or password during login?

If you enter incorrect login credentials, you will receive an error message indicating that the login has failed. You can attempt to log in again with the correct username and password.

7. Can I play the quiz more than once?

Yes, you can play the quiz multiple times. After you've completed one round, you can log in and start a new game.

8. Is there a time limit for answering each question?

No, there is no time limit for answering the questions. You can take your time to read the questions and choose your answers.

Quizzie – The Quiz Taker

9. Can I customize the number of questions or difficulty level in the quiz?

Currently, the quiz fetches 10 questions with an "easy" difficulty level from the API. If you want to customize these settings, you would need to modify the code.

10. What happens if I close the login window during the game?

If you close the login window without successfully logging in, the game will not start. You will need to reopen the application and log in again.

11. How do I know my final score?

After you've answered all the questions, the game will display your total score out of the number of questions attempted. You can see how well you did and challenge yourself to improve your score next time.

12. Can I see a list of high scores or compare my score with others?

The code provided does not include a feature to view high scores or compare scores with other players. It's a single-player quiz game.