

QUIZ GAME

Grp-10 Presentation

Group - 10

Prasanna kumar Dheekonda

AP21110010174

Sahithya Chilukuri

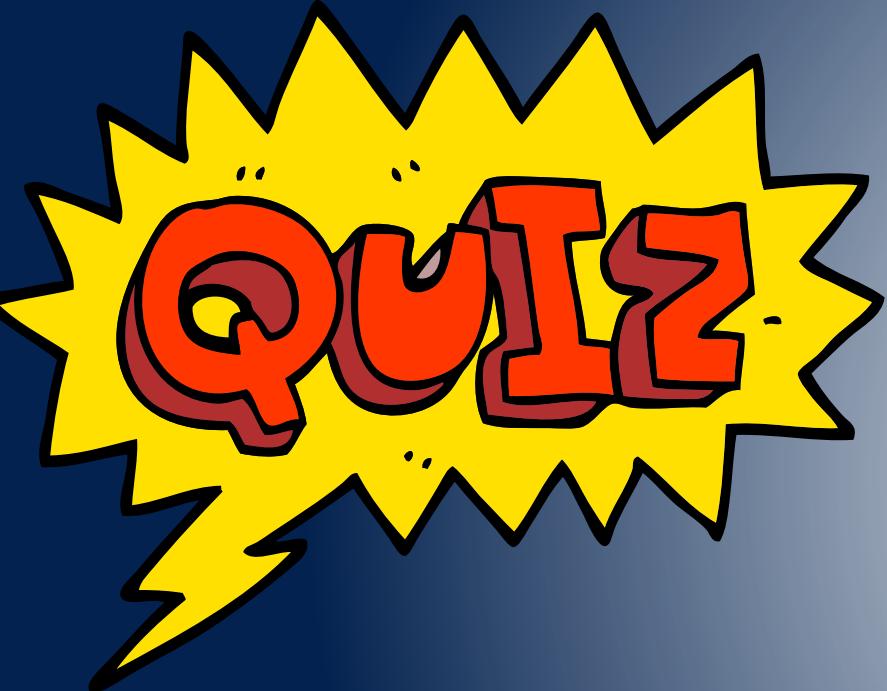
AP21110010181

Narasimha Rao Yandrapragada

AP21110010190

Sivamani Gopavarapu

AP21110010195



Abstract



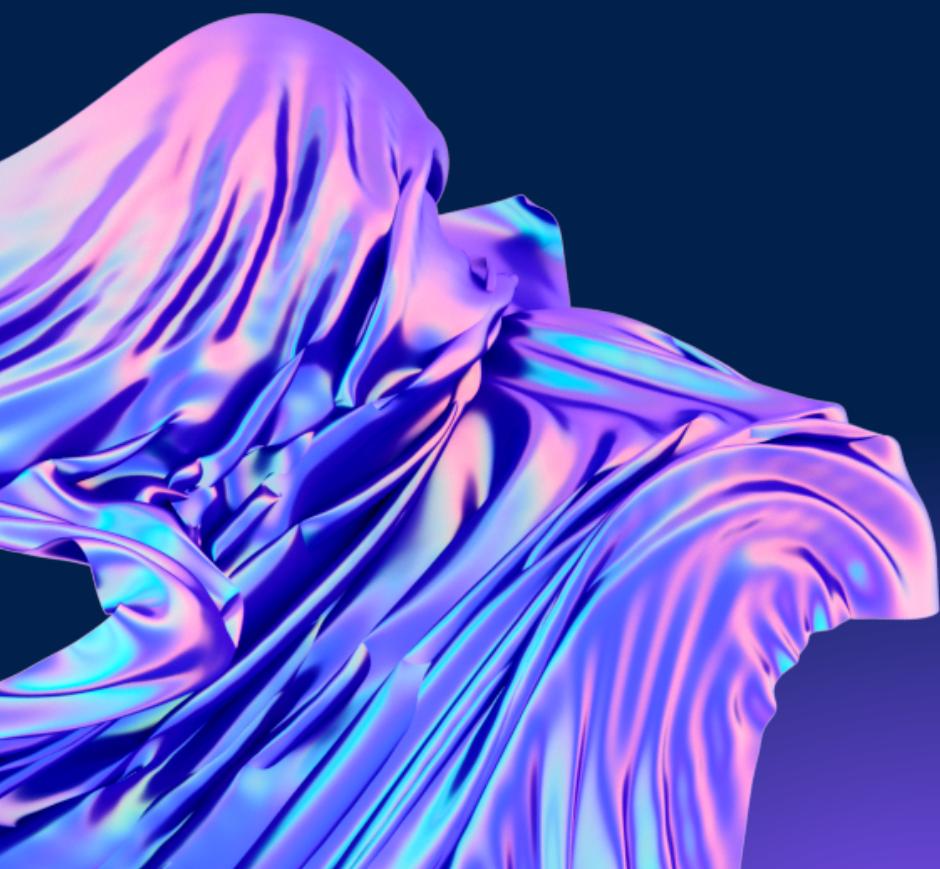
Quiz Game is simple to use and requires less manual labour. In the past, quizzes were administered manually, but today we can obtain automatic results thanks to technology. A

quiz game was created for this project. The idea of quizzes is quite well-liked in both academic circles and some entertainment programmes. People can learn more by taking quizzes, which are a popular kind of entertainment.

This user-friendly quiz game will allow users to learn something new while still having fun.

This game features numerous questions and computes the scoring for each accurate response.

It benefits students of all age groups since it helps them learn more about the world, sports, computers, and other topics. You can login with the same user name without having to register; just provide it. It aids the user in expanding his or her knowledge. A lot of universities have been using quizzes as part of internal assessments.



Introduction

By inputting your name, which is also entered to the records, you can access the quiz game before the test's start. The player will receive questions in our quiz game and must provide a response. Each question will have four possible answers; the player must select the right one.

The correct response is printed if the player doesn't provide the proper response, in which case they lose points. However, if the player responds to the question accurately, he will receive credit for his effort. The overall score is shown, and we also show the player's selected options as well as the correct answers.

The player should enter "skip" to skip the current question and "submit" to finish the test. Lastly, in this project, we keep the earlier records so that we can view the earlier information.

Functions



Design

This project has been designed majorly with the help of function concept and the OS module which is available in python.

The OS module is used in this project. The system() method of the OS module with the string cls or clear as a parameter is used to clear the screen in windows and macOS/Linux, respectively.

Many functions are used to do specific tasks to run the program.

Code with Explanation

```
import os
```

In this project we import the OS module into the code. From this module we will use the system() method to clear the screen. By adding the line "os.system('cls')".

Code with Explanation

```
questions = {  
    "Who created Python?: ": "A",  
    "What year was Python created?: ": "B",  
    "Which of the following is not a Python Data Type?: ": "B",  
    "Which of the following is not a bitwise operator?: ": "C"  
}  
  
options = [[{"A. Guido van Rossum", "B. Elon Musk", "C. Bill Gates", "D. Mark Zuckerberg"},  
           [{"A. 1989", "B. 1991", "C. 2000", "D. 2016"}],  
           [{"A. int", "B. char", "C. float ", "D. string"}],  
           [{"A. &", "B. |", "C. /", "D. ^"}]]  
quizes = []
```

In this part we will add questions in dictionary and options in a list and we create quizes empty list

Code with Explanation

```
while(1):
    os.system('cls')
    print("\n\n\t\t=====")
    print("\t\t===== WELCOME =====")
    print("\t\t===== QUIZ ON PYTHON =====")
    print("\t\t=====")
    print("1.Play Game")
    print("2.Show Previous Records")
    print("3.Exit")
    print("=====")
    choice = int(input("Enter your choice\n> "))
    if choice == 1:
        new_game()
    elif choice == 2:
        show_previousQuizes()
    elif choice == 3:
        print("\nThank you!!!")
        break
```

Any option from the display choices should be entered.
If choice is one, new game() will be called and the statement
will be executed. If the answer is two, show previousQuizes()
will be called and the statement will be executed. If there are
three options, it prints "Thank you!!!" and leaves.

Code with Explanation

```
def new_game():
    intro_message()
    name = input("\nEnter your name: ")
    guesses = []
    correct_guesses = 0
    question_num = 1

    for key in questions:
        print("\nQuestion no-", question_num)
        print("-----")
        print(key)
        for i in options[question_num-1]:
            print(i)
        guess = input("Enter option(A or B or C or D)\n> ")
        guess = guess.upper()
        if guess == "SKIP":
            guesses.append('-')
            question_num += 1
            continue
        elif guess == "SUBMIT":
            print("Submitting Test...")
            for i in range(len(questions) - len(guesses)):
                guesses.append('-')
            display_score(correct_guesses, guesses)
            break
        guesses.append(guess)
        correct_guesses += check_answer(questions.get(key), guess)
        question_num += 1
    else:
        print("Submitting Test...")
        display_score(correct_guesses, guesses)

    quizzes.append([name, list(questions.values()), guesses, ])
```

In this Function the player should enter the name to start the quiz the questions and options are displayed on the screen.

Player should enter the option if entered option is SKIP then next question will displayed.

If entered option is SUBMIT then display_score() is called.

Code with Explanation

```
def new_game():
    intro_message()
    name = input("\nEnter your name: ")
    guesses = []
    correct_guesses = 0
    question_num = 1

    for key in questions:
        print("\nQuestion no- ",question_num)
        print("-----")
        print(key)
        for i in options[question_num-1]:
            print(i)
        guess = input("Enter option(A or B or C or D)\n> ")
        guess = guess.upper()
        if guess == "SKIP":
            guesses.append('-')
            question_num += 1
            continue
        elif guess == "SUBMIT":
            print("Submitting Test...")
            for i in range(len(questions) - len(guesses)):
                guesses.append('-')
            display_score(correct_guesses, guesses)
            break
        guesses.append(guess)
        correct_guesses += check_answer(questions.get(key), guess)
        question_num += 1
    else:
        print("Submitting Test...")
        display_score(correct_guesses, guesses)

    quizzes.append([name,list(questions.values()),guesses,])
```

If the **input option isn't one of the above cases, the check_answer() Fuction is used to determine whether the entered response is accurate or not.**

Finally, **name, a list of submitted responses, and the correct responses are put to the quizzes list.**

Code with Explanation

```
def intro_message():
    os.system('cls')
    print("\n\n\t\t===== [ INSTRUCTIONS ] =====")
    print("\t\t1) FOR EACH QUESTION THERE ARE 4 OPTIONS SELECT ANY ONE OPTION FROM IT")
    print("\t\t2) FOR EVERY CORRECT ANSWER YOU WILL BE AWARDED 4 MARKS")
    print("\t\t3) NO NEGATIVE MARK")
    print("\t\t4) ENTER \"SKIP\" TO SKIP THE CURRENT QUESTION")
    print("\t\t5) ENTER \"SUBMIT\" TO SUBMIT THE TEST")
    print("\t\t=====")
```

The screen is clean and the INSTRUCTIONS are printed there so that the player can understand the guidelines and perform well on the quiz.

Code with Explanation

```
#-----  
def check_answer(answer, guess):#used to check answer  
    if answer == guess:  
        print("CORRECT!")  
        return 1  
    elif guess.upper() not in ['A', 'B', 'C', 'D']:  
        print("Invalid option")  
        print("\nCorrect answer is", answer)  
        return 0  
    else:  
        print("WRONG!")  
        print("\nCorrect answer is", answer)  
        return 0
```

The **check_answer()** method is used to determine whether or not the entered answer is accurate. If the answer matches the guess, "CORRECT!" is printed and 1 is returned. If the entered estimate does not fall under "A, B, C, or D," it outputs "Invalid option" and the right response is printed before returning 0. If the first two tests are unsuccessful, "WRONG!" is printed, followed by the right response and a return value of 0.

Code with Explanation

```
def display_score(correct_guesses, guesses):
    print("\n-----")
    print("RESULTS")
    print("-----")
    print("Answers: ", end="")
    for i in questions:
        print(questions.get(i), end=" ")
    print()
    print("Guesses: ", end="")
    for i in guesses:
        print(i, end=" ")
    print()
    score = int((correct_guesses/len(questions))*100)
    print("Your score is:", (correct_guesses*4))
    print("Your percentage is: " + str(score) + "%")
    input("Press Enter...")
```

This function is used to display scores, entered guesses and the correct answers. It calculates the percentage and displays . At the end, we should press enter then it will enter into the while loop.

Code with Explanation

```
def show_previousQuizes():
    os.system('cls')
    print(f"===== [Previous Quiz Details] =====")
    count = 1
    for quiz in quizzes:
        score = 0
        print(f"\n----- [Quiz {count}] -----")
        print(f"Name:",quiz[0])
        print("Selected Guesses:",end = " ")
        for guess in quiz[2]:print(guess,end = " ")
        print()
        print(f"Correct Guesses :",end = " ")
        for guess in quiz[1]:print(guess,end = " ")
        print()
        for i in range(len(quiz[1])):
            if quiz[1][i] == quiz[2][i]:
                score = score + 1
        print(f"Score      :",score * 4)
        print("-----")
        count += 1
    print("=====")
    input("Press Enter...")
```

This function is used to print the previous data. In the new_game() function at last we created a list named as quizzes. In that list we add elements for each element we again created a list having names, entered guesses , and correct answers. After comparing the guesses entered and the correct responses, the score is increased to 1 (quiz[1][i] == quiz[2][i]). To print the score we have to multiply the score with 4.

Results

```
===== WELCOME =====
===== QUIZ ON PYTHON =====
=====
1.Play Game
2.Show Previous Records
3.Exit
=====
Enter your choice
> 1
```

Results

```
===== [ INSTRUCTIONS ] =====
1) FOR EACH QUESTION THERE ARE 4 OPTIONS SELECT ANY ONE OPTION FROM IT
2) FOR EVERY CORRECT ANSWER YOU WILL BE AWARDED 4 MARKS
3) NO NEGATIVE MARK
4) ENTER "SKIP" TO SKIP THE CURRENT QUESTION
5) ENTER "SUBMIT" TO SUBMIT THE TEST
-----
Enter your name: Narasimha rao
Question no- 1
-----
Who created Python?:
A. Guido van Rossum
B. Elon Musk
C. Bill Gates
D. Mark Zuckerburg
Enter option(A or B or C or D)
> a
CORRECT!
Question no- 2
-----
What year was Python created?:
A. 1989
B. 1991
C. 2000
D. 2016
Enter option(A or B or C or D)
> c
WRONG!
Correct answer is B
Question no- 3
-----
Which of the following is not a Python Data Type?:
A. int
B. char
C. float
D. string
Enter option(A or B or C or D)
> skip
```

Question no- 4

Which of the following is not a bitwise operator?:

- A. &
- B. |
- C. /
- D. ^

Enter option(A or B or C or D)

> submit

Submitting Test...

RESULTS

Answers: A B B C

Guesses: A C - -

Your score is: 4

Your percentage is: 25%

Press Enter...|

Results

```
===== WELCOME =====
===== QUIZ ON PYTHON =====
=====
1.Play Game
2.Show Previous Records
3.Exit
=====
Enter your choice
> 2
```

```
===== WELCOME =====
===== QUIZ ON PYTHON =====
=====
1.Play Game
2.Show Previous Records
3.Exit
=====
Enter your choice
> 3

Thank you!!!
```

```
===== [Previous Quiz Details] =====
----- [Quiz 1] -----
Name: Narasimha rao
Selected Guesses: A C - -
Correct Guesses : A B B C
Score           : 4
-----
===== Press Enter... =====
```

Thank You

Group 10