



JULY 9, 2024

PYTHON QUIZ GAME

SUBMITTED BY-PRISHA D



CODE:

```
questions = {  
    "What is the capital of France?": {  
        "options": ["Paris", "Berlin", "London", "Madrid"],  
        "answer": "Paris"  
    },  
    "Who is the president of the USA?": {  
        "options": ["Joe Biden", "Donald Trump", "Barack Obama", "George Washington"],  
        "answer": "Joe Biden"  
    },  
    "What is the capital of England?": {  
        "options": ["London", "Paris", "Berlin", "Rome"],  
        "answer": "London"  
    },  
    "What is the largest planet in our solar system?": {  
        "options": ["Earth", "Saturn", "Jupiter", "Uranus"],  
        "answer": "Jupiter"  
    },  
    "Which of the following is NOT one of the Great Lakes?": {  
        "options": ["Lake Michigan", "Lake Huron", "Lake Ontario", "Lake Tahoe"],  
        "answer": "Lake Tahoe"  
    },  
    "What is the chemical symbol for gold?": {  
        "options": ["Ag", "Au", "Hg", "Pb"],  
        "answer": "Au"  
    },  
    "What is the smallest country in the world?": {  
        "options": ["Vatican City", "Monaco", "Nauru", "Tuvalu"],  
        "answer": "Vatican City"  
    },  
}
```

```
"What is the largest living species of lizard?": {  
    "options": ["Komodo dragon", "Saltwater crocodile", "Black mamba", "Green  
anaconda"],  
    "answer": "Komodo dragon"  
}  
}
```

Function to display the rules

```
def display_rules():  
    print("Welcome to the Quiz Game!")  
    print("Here are the rules:")  
    print("1. Answer the questions with the corresponding number.")  
    print("2. You will earn 10 points for each correct answer.")  
    print("3. If you answer 3 questions correctly in a row, you will earn a bonus of 50 points!")  
    print("4. The game will end after all questions have been answered.")  
    print("5. Your final score will be displayed at the end of the game.")  
    input("Press Enter to start the game...")
```

Function to display the questions and options

```
def display_question(question_text, options):  
    print(question_text)  
    for i, option in enumerate(options, 1):  
        print(f"{i}. {option}")
```

Function to handle user input and validate the answer

```
def get_user_answer(question):  
    display_question(question["text"], question["options"])  
    while True:  
        user_input = input("Enter your answer (1-{}): ".format(len(question["options"])))  
        if user_input.isdigit() and 1 <= int(user_input) <= len(question["options"]):  
            return question["options"][int(user_input) - 1]
```

```

        else:

            print("Invalid input. Please enter a number between 1 and
            {}".format(len(question["options"])))

# Function to check the user's answer
def check_answer(user_answer, correct_answer):

    if user_answer.strip().lower() == correct_answer.strip().lower():

        print("Correct!")

        return True

    else:

        print("Incorrect. The correct answer is '{}'.format(correct_answer))

        return False

# Function to calculate the user's final score
def calculate_score(correct_count, bonus):

    score = correct_count * 10

    if bonus:

        score += 50

    print("\nYour final score is: {}".format(score))

# Main function to run the quiz
def main():

    display_rules()

    correct_count = 0

    correct_in_a_row = 0

    bonus = False

    for question_text, question_data in questions.items():

        question = {"text": question_text, "options": question_data["options"]}

        user_answer = get_user_answer(question)

        if check_answer(user_answer, question_data["answer"]):

```

```

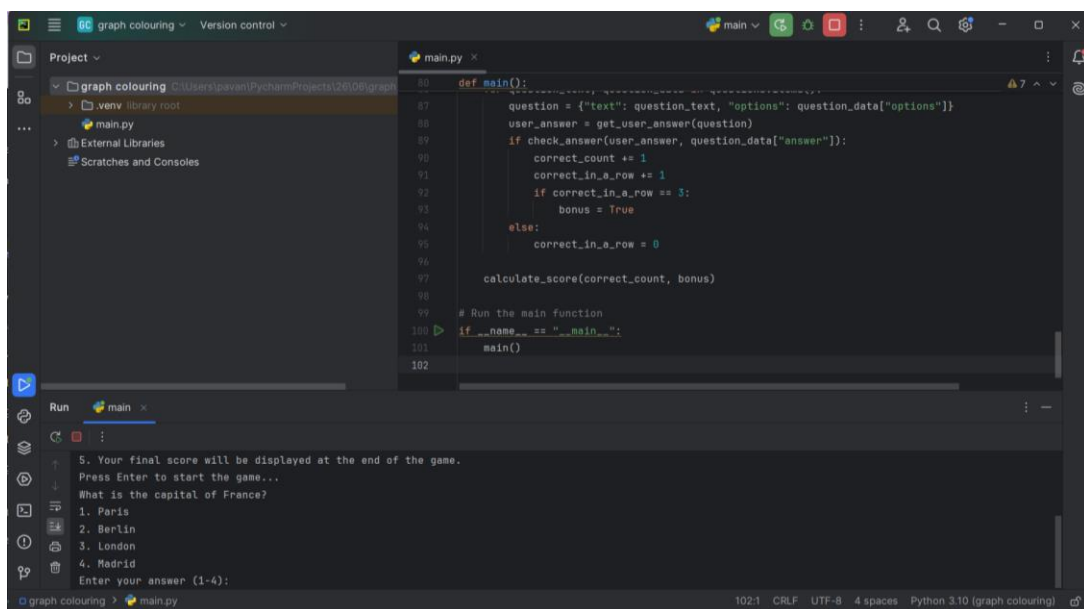
        correct_count += 1
        correct_in_a_row += 1
        if correct_in_a_row == 3:
            bonus = True
    else:
        correct_in_a_row = 0

    calculate_score(correct_count, bonus)

# Run the main function
if __name__ == "__main__":
    main()

```

SCREENSHOTS OF OUTPUT:



graph colouring Version control

Project

- graph colouring C:\Users\pavan\PycharmProjects\20\06\graph
- .venv library root
- main.py
- External Libraries
- Scratches and Consoles

main.py

```
86 def main():
87     question = {"text": question_text, "options": question_data["options"]}
88     user_answer = get_user_answer(question)
89     if check_answer(user_answer, question_data["answer"]):
90         correct_count += 1
91         correct_in_a_row += 1
92         if correct_in_a_row == 3:
93             bonus = True
94     else:
95         correct_in_a_row = 0
96
97     calculate_score(correct_count, bonus)
98
99     # Run the main function
100 if __name__ == "__main__":
101     main()
102
```

Run main

Enter your answer (1-4): 1
Correct!
Who is the president of the USA?
1. Joe Biden
2. Donald Trump
3. Barack Obama
4. George Washington
Enter your answer (1-4): 1

graph colouring > main.py

Page 4 of 4 427 words English (India)

graph colouring Version control

Project

- graph colouring C:\Users\pavan\PycharmProjects\20\06\graph
- .venv library root
- main.py
- External Libraries
- Scratches and Consoles

main.py

```
86 def main():
87     question = {"text": question_text, "options": question_data["options"]}
88     user_answer = get_user_answer(question)
89     if check_answer(user_answer, question_data["answer"]):
90         correct_count += 1
91         correct_in_a_row += 1
92         if correct_in_a_row == 3:
93             bonus = True
94     else:
95         correct_in_a_row = 0
96
97     calculate_score(correct_count, bonus)
98
99     # Run the main function
100 if __name__ == "__main__":
101     main()
102
```

Run main

Enter your answer (1-4): 1
Correct!
What is the capital of England?
1. London
2. Paris
3. Berlin
4. Rome
Enter your answer (1-4): 1

graph colouring > main.py

Page 5 of 5 427 words English (India)

```
def main():
    question = {"text": question_text, "options": question_data["options"]}
    user_answer = get_user_answer(question)
    if check_answer(user_answer, question_data["answer"]):
        correct_count += 1
        correct_in_a_row += 1
        if correct_in_a_row == 3:
            bonus = True
    else:
        correct_in_a_row = 0

    calculate_score(correct_count, bonus)

# Run the main function
if __name__ == "__main__":
    main()
```

Run main

Enter your answer (1-4): 1
Correct!
What is the largest planet in our solar system?
1. Earth
2. Saturn
3. Jupiter
4. Uranus
Enter your answer (1-4): 2

```
def main():
    question = {"text": question_text, "options": question_data["options"]}
    user_answer = get_user_answer(question)
    if check_answer(user_answer, question_data["answer"]):
        correct_count += 1
        correct_in_a_row += 1
        if correct_in_a_row == 3:
            bonus = True
    else:
        correct_in_a_row = 0

    calculate_score(correct_count, bonus)

# Run the main function
if __name__ == "__main__":
    main()
```

Run main

Enter your answer (1-4): 2
Incorrect. The correct answer is 'Jupiter'.
Which of the following is NOT one of the Great Lakes?
1. Lake Michigan
2. Lake Huron
3. Lake Ontario
4. Lake Tahoe
Enter your answer (1-4): 1

```
def main():
    question = {"text": question_text, "options": question_data["options"]}
    user_answer = get_user_answer(question)
    if check_answer(user_answer, question_data["answer"]):
        correct_count += 1
        correct_in_a_row += 1
        if correct_in_a_row == 3:
            bonus = True
    else:
        correct_in_a_row = 0

    calculate_score(correct_count, bonus)

# Run the main function
if __name__ == "__main__":
    main()
```

Run main

Enter your answer (1-4): 3
Incorrect. The correct answer is 'Lake Tahoe'.
What is the chemical symbol for gold?
1. Ag
2. Au
3. Hg
4. Pb
Enter your answer (1-4): 2

graph colouring > main.py

```
def main():
    question = {"text": question_text, "options": question_data["options"]}
    user_answer = get_user_answer(question)
    if check_answer(user_answer, question_data["answer"]):
        correct_count += 1
        correct_in_a_row += 1
        if correct_in_a_row == 3:
            bonus = True
    else:
        correct_in_a_row = 0

    calculate_score(correct_count, bonus)

# Run the main function
if __name__ == "__main__":
    main()
```

Run main

Enter your answer (1-4): 2
Correct!
What is the smallest country in the world?
1. Vatican City
2. Monaco
3. Nauru
4. Tuvalu
Enter your answer (1-4): 1

graph colouring > main.py

graph colouring Version control

Project ▾

- graph colouring C:\Users\pavan\PycharmProjects\20\00\graph
- venv library root
- main.py
- External Libraries
- Scratches and Consoles

main.py

```
86 def main():
87     question = {"text": question_text, "options": question_data["options"]}
88     user_answer = get_user_answer(question)
89     if check_answer(user_answer, question_data["answer"]):
90         correct_count += 1
91         correct_in_a_row += 1
92         if correct_in_a_row == 3:
93             bonus = True
94     else:
95         correct_in_a_row = 0
96
97     calculate_score(correct_count, bonus)
98
99     # Run the main function
100 if __name__ == "__main__":
101     main()
102
```

Run main

Enter your answer (1-4): 1
Correct!
What is the largest living species of lizard?
1. Komodo dragon
2. Saltwater crocodile
3. Black mamba
4. Green anaconda
Enter your answer (1-4): 3

graph colouring > main.py 1021 CRLF UTF-8 4 spaces Python 3.10 (graph colouring)

graph colouring Version control

Project ▾

- graph colouring C:\Users\pavan\PycharmProjects\20\00\graph
- venv library root
- main.py
- External Libraries
- Scratches and Consoles

main.py

```
86 def main():
87     question = {"text": question_text, "options": question_data["options"]}
88     user_answer = get_user_answer(question)
89     if check_answer(user_answer, question_data["answer"]):
90         correct_count += 1
91         correct_in_a_row += 1
92         if correct_in_a_row == 3:
93             bonus = True
94     else:
95         correct_in_a_row = 0
96
97     calculate_score(correct_count, bonus)
98
99     # Run the main function
100 if __name__ == "__main__":
101     main()
102
```

Run main

4. Green anaconda
Enter your answer (1-4): 3
Incorrect. The correct answer is 'Komodo dragon'.
Your final score is: 100
Process finished with exit code 0

graph colouring > main.py 1021 CRLF UTF-8 4 spaces Python 3.10 (graph colouring)