INTRODUCTION:

My project defines a quiz game that asks the user to solve simple exponentiation problems. The quiz continues until the user chooses to exit.

Each time the quiz is taken, 5 questions are asked and the user's score and the time taken to complete the quiz are displayed at the end. The quiz then asks the user if they want to retake the quiz. If the user chooses not to retake the quiz, the game will exit.

This report presents a simple quiz game that asks the user to solve exponentiation problems. The game continues until the user chooses to exit. The score and time taken to complete each quiz are displayed at the end.

IMPLEMENTATION:

- The game has been implemented using the Python programming language. The **datetime** and **random** modules have been imported to handle date and time operations and to generate random numbers, respectively.
- The game consists of an infinite loop that continues until the user chooses to exit. At the start of each quiz, 5 questions are asked and the user's score and the time taken to complete the quiz are displayed at the end. The user is then asked if they want to retake the quiz. If the user chooses not to retake the quiz, the game exits.

- For each question, two random integers between 0 and 9 are generated and the user is asked to solve the exponentiation problem using these integers. The user's answer is read using the **input()** function and compared with the correct answer. If the answer is correct, a message is displayed and the user's score is incremented. If the answer is incorrect, a message is displayed.
- The code sets up a loop that will run indefinitely until the user chooses to end the game. Inside the loop, several variables are initialized: correct is a counter for the number of correct answers, count is a counter for the total number of answers, and **totalquestions** is the total number of questions for the quiz (which is set to 5). The starting time of the quiz is also recorded using **datetime.datetime.now().**
- Inside the loop, the program generates two random integers a and b using **random.randint()** and prints a math problem using these values. The user is asked to input the answer to the math problem. If the user's answer is correct, the program will increment the correct counter and the count counter, and print **"You are correct!"**. If the user's answer is incorrect, the program will print **"You are wrong! Correct answer is (correct answer)"**, and increment only the count counter. After each question, the **totalquestions** counter is decremented by 1.
- When the **totalquestions** counter reaches 0, the inner loop ends, and the program records the ending time of the quiz using **datetime.datetime.now().** The difference between the starting time and the ending time is calculated and stored as **totalTime.** The program then prints the number of correct answers, the total number of answers, and the time taken to complete the quiz.

HERE IS A BREAKDOWN OF THE CODE:

1. The **datetime** and random modules are imported.
2. An infinite while loop is defined.
3. The variables correct, count, and **totalquestions** are initialized to 0, 0, and 5, respectively. correct will store the number of correct answers, count will store the total number of questions asked, and **totalquestions** will store the total number of questions that should be asked in each quiz.
4. The current date and time is stored in the **startTime** variable using **datetime.datetime.now().**
5. The quiz start message and start time are printed.
6. A while loop is defined that will continue to execute until **totalquestions** is 0.
7. Two random integers between 0 and 9 are generated using **random.randint().**
8. The user is asked to solve the exponentiation problem using the two random integers. The user's answer is read using **input()** and stored in the answer variable as an integer.
9. If the user's answer is correct, a message is displayed and the correct and count variables are incremented. If the answer is incorrect, a message is displayed and only the count variable is incremented.
10. The **totalquestions** variable is decremented by 1.
11. When the inner while loop exits, the current date and time is stored in the **endTime** variable. The difference between **endTime** and **startTime** is calculated and stored in the **totalTime** variable.
12. The user's score and the time taken to complete the quiz are printed.

13. The user is asked if they want to retake the quiz. If the user chooses not to retake the quiz, the break statement is executed, which exits the outer while loop and the game ends.
14. Finally, the program asks the user if they want to retake the quiz, and if the user inputs 'N' or 'n', the program ends. If the user inputs anything else, the program will continue to run, and the quiz will start again.

CONCLUSION:

The simple quiz game is a basic implementation that can be used to practice exponentiation problems. The game can be modified and extended by adding more questions and introducing more complex math concepts.