Mini Project -1 (Number Guessing Game)

(Language PYTHON)

* Build a number Guessing Game in which the user selects a range.
* Assume the user selected a range from X to Y where both X and Y are integers.
* So a random number in that range is selected by the system where the user needs to guess the number in minimum number of guesses.

**Number Guessing Game Algorithm:**

1. **Initialization:**

* Print a welcome message and prompt the user to set the range for the numbers they want to guess**.**

1. **Input Validation for Range:**

* Prompt the user to enter the lower bound (lower\_bound) and upper bound (upper\_bound) of the range.
* Check if upper\_bound is greater than lower\_bound. If not, display an error message and repeat the input process.

1. **Game Setup:**

* Generate a random number (secret\_number) using random.randint(lower\_bound, upper\_bound).
* Initialize attempts counter to track the number of guesses made by the user.

1. **Game Loop**:

* Repeat until the user guesses the correct number:
* Prompt the user to enter a guess (guess) within the range [lower\_bound, upper\_bound].
* Validate the input:  
   If the input is not an integer or falls outside the range, display an error message and prompt again.
* Compare guess with secret\_number
* If guess is less than secret\_number, print "Too low! Try guessing a higher number."
* If guess is greater than secret\_number, print "Too high! Try guessing a lower number."
* If guess equals secret\_number, print "Congratulations! You guessed the number [secret\_number] correctly!" and display the number of attempts.
* Break out of the loop.

1. **Play Again Option:**

* After the game ends:
* Prompt the user if they want to play again (play\_again).
* If play\_again is "yes", go back to step 2 (Input Validation for Range).
* If play\_again is not "yes", print a thank you message and end the game.

1. **End of Program:**

* When the user decides not to play again, print a farewell message and terminate the program.

**Pseudo code:**

Procedure number\_guessing\_game():

secret\_number = Random number between 1 and 100

attempts = 0

Display "Welcome to the Number Guessing Game!"

Display "The guessed number is number between 1 and 100."

Repeat:

Input guess

If guess is not a valid number:

Display "Invalid input. Please enter a valid number."

Else:

attempts = attempts + 1

If guess < secret\_number:

Display "Too low! Try guessing a higher number."

Else If guess > secret\_number:

Display "Too high! Try guessing a lower number."

Else:

Display "Congratulations! You guessed the number secret\_number correctly!"

Display "It took you attempts attempts to guess the correct number."

Break

End Procedure

**Code for Number Guessing Game:**

1. import random

2. def guess\_number():

3. print(" Welcome to the Number Guessing Game!")

4. while True:

5. try:

6. lower\_bound = int(input("Enter the lower bound of the range (an integer): "))

7. upper\_bound = int(input("Enter the upper bound of the range (an integer): "))

8. if lower\_bound >= upper\_bound:

9. print("Upper bound must be greater than lower bound. Please try again.")

10. continue

11. secret\_number = random.randint(lower\_bound, upper\_bound)

12. attempts = 0

13. print(f"the guessed number is between {lower\_bound} and {upper\_bound}.")

14. while True:

15. try:

16. guess = int(input(f"Enter your guess (between {lower\_bound} and {upper\_bound}): "))

17. attempts += 1

18. if guess < lower\_bound or guess > upper\_bound:

19. print(f"Please enter a number within the range {lower\_bound} and{upper\_bound}.")

20. continue

21. if guess < secret\_number:

22. print("Too low! Try guessing a higher number.")

23. elif guess > secret\_number:

24. print("Too high! Try guessing a lower number.")

25. else:

26. print(f"Congratulations! You guessed the number {secret\_number} correctly!")

27. print(f"It took you {attempts} attempts to guess the correct number.")

28. break

29. except ValueError:

30. print("Invalid input. Please enter a valid number.")

31. play\_again = input("Do you want to play again? (yes/no): ").lower()

32. if play\_again != "yes":

33. print("Thank you for playing!")

34. break

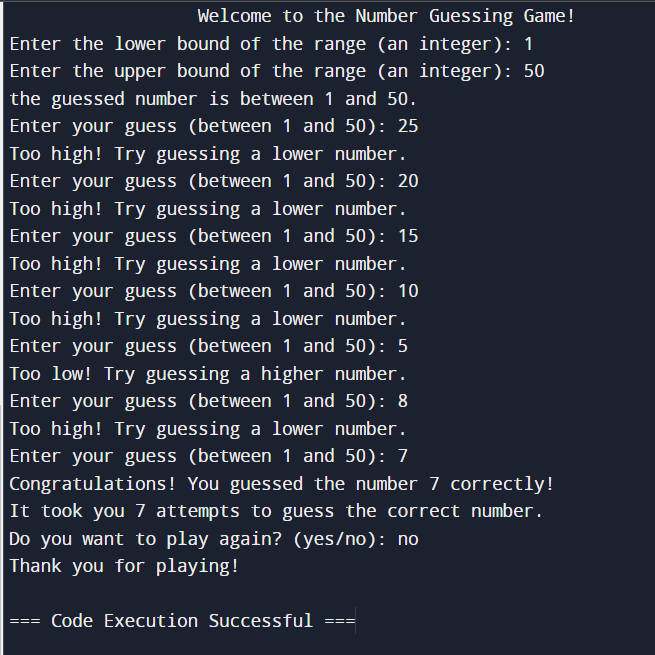
35. except ValueError:

36. print("Invalid input. Please enter valid integers for the range.")

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

38. guess\_number()

**Output:**

****