Lesson 2 - Guess the Number

What we gonna learn?

- Control Flow (While, For loop)
- String concatenation
- Type Conversion
- Counter
- New Line

Now let's start coding!!!

```python
#Lesson 3 - Guess the Number
#Understanding the Basic & Control Flow Statements (While & For Loop)

#Import Library (Random)
import random

#Input Name (String)
Name = input("Enter Your Name : ")
Lower_limit = int(input("Enter Lower Limit :"))
Upper_limit = int(input("Enter Upper Limit :"))
#Range of random number to be generated (0-5)
#random.randint(start, stop)
random_number_generator = random.randint(Lower_limit,Upper_limit)
#help(random.randint)

#Number of try allowed
Number_of_try = 3

while Number_of_try > 0:

    #Input only a number(Integer), replace the Lower_limit and Upper_limit with your value
    print('Enter only number range from "Lower_limit" to "Upper_Limit" ')
    guess_a_number = int(input("Guess a Number : "))

    #Control Flow : If & else statements
    if random_number_generator == guess_a_number:
        print("Your guess correctly!!!")
    elif random_number_generator < guess_a_number:
        print("The number generated is smaller than number you guess")
    else:
        print("The number generated is bigger than number you guess")

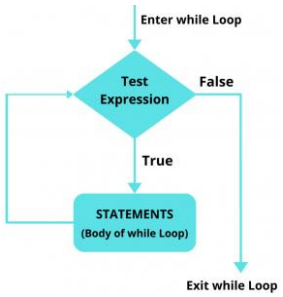
    #Number of try counter
    Number_of_try = Number_of_try - 1
```

Expected Output :

```
Enter Your Name : Tommy
Enter Lower Limit :1
Enter Upper Limit :10
Enter only number range from "Lower_limit" to "Upper_Limit"
Guess a Number : 1
The number generated is bigger than number you guess
Enter only number range from "Lower_limit" to "Upper_Limit"
Guess a Number : 2
The number generated is bigger than number you guess
Enter only number range from "Lower_limit" to "Upper_Limit"
Guess a Number : 3
The number generated is bigger than number you guess
```

Code Explanation:

1. Comments are written with # at the front (Line 1,2,4,7,11,12,14,16,21,25,33)
2. Line 5 – to import random library (python built in library)
3. Line 8 – Requesting user to input name (the datatype will be string by default)
4. Line 9 & 10 – Requesting the game master to enter lower limit(a) and upper limit(b) of value in the random.randint library.
5. Line 13 – Here is where we use the random library. Other than random.randint(a,b) ,(*a is the lower limit, b is the upper limit, and both will be included in the number generated*) there are more to it. For details Python Random Module (w3schools.com)
6. Line 17 – Counter for number of try available
7. Line 19 – While loop (One of the control flow available) Python While Loops (w3schools.com)



8. Line 22 & 23 – Guide the user on the number that can be entered and ask user to enter the guess number.
9. Line 26 to 31- If *guess_a_number* is the same as *random_number_generator* (It will output line 27), else if *random_number_generated* less than *guess_a_number* (line 29 will be outputted), else if *random_number_generated* greater than *guess_a_number* (line 31 will be outputted)
10. Line 34 – Number of try available (Maximum 3)

Let's try….

1. After Line 34, try to display the output of Number_of_try available (*Tips: to display an output, we need to use print. For statement, we will use print("Statement"), but for variable we can just print(variable) without the apostrophe*) (*Try to display statement+variable in single line, use "+" in between [Check the type of Number_of_try variable and change it to string by using str(variable)*)
2. After few trial, you will realize that the program continues to ask you input number to guess although you answered correctly, how should we make it stop if the answer is correct? (*Tips: Line 26 & 27 is for when the answer is correct, add new line after line 27, use either break, continue or pass based on the statement below*)

   break, continue and pass in Python
   - Break statement. The break statement is used to terminate the loop or statement in which it is present. After that, the...
   - Continue statement. Continue is also a loop control statement just like the break statement. continue statement is...
   - Pass statement. As the name suggests pass statement simply does nothing. The pass statement in...
3. After done with no 1 & 2, you will realize that there is no space between your output (after number_of_try) and it is hard to see. Try to make one empty space between each guess. (Tips: use \n at the end of your print(statement+variable(put \n here)). \n mean newline, you can also use print(" ") at the last line.)
4. Even after you give instruction on

   ```
   print('Enter only number range from "Lower_limit" to "Upper_Limit" ')
   ```

   There are still people entered number that is out of range. To prevent this, use not in range statement with if and see the difference.

   (*Tips:*

   1. ***Syntax:*** *range(start, stop, step)*
   2. ```
      if number not in range(start, stop[, step]):
          statement(s)
      ```

)

   ```
   if guess_a_number not in range(Lower_limit, Upper_limit):
       print("Please enter only number in range of " + str(Lower_limit) + " to " + str(Upper_limit))
       guess_a_number = int(input("Guess a Number : "))
   ```
   ```

Things to ponder…..

1. What if you print different type of item in one line? Eg. Print(integer+string+string), what will happen?
2. What will happen if in line 34, your put the “Number_of_Try = …..” in the same line as “While”?