Python Fundamentals (Assignment2)

Note - To solve some of the assignment Qs we'll have to learn some additional logic (already mentioned as hints). The purpose of these questions is to help build programming logic so even if we have to take hint for mathematical part of the problem & we write the Python code on our own, it's alright.

Assignment Problems

Q1. Write a program that takes `salary` as input. Using conditional statements, calculate the **final tax rate** based on these rules:

- If salary < 30,000 → **5%**

- If salary is 30,000–70,000 → **15%**

- If salary > 70,000 → **25%**

Q2. Write a function that takes two integers `a` and `b` and prints all even numbers between them (inclusive).

Q3. Write a function that prints the **digits** of a number, `n`.

  For eg: `n = 312`, there are 3 digits in it 3, 1 and 2 & we need to print them.

  [**Hint** - The right most digit of a number N is N%10.

  And to remove the right most digit from a number, we can do N = N / 10.]

Q4. Write a function to return the **count** the number of digits in a number, `n`.

Q5. Write a function to return the **sum of digits** of a number, `n`.

Q6. Write a program to print all numbers from 1 to 100 that are divisible by both 3 and 5.

Q7. Design a program to continuously input a number `n` from user & print if it is positive or negative until the user enters "Quit".

Q8. Let's create a Simple **Calculator** that performs arithmetic operations. Create a function `calculator(a, b, operation)` that performs addition, subtraction, multiplication, or division based on the `operation` parameter.

[`operation` parameter can have values `'+'` , `'-'` , `'*'` & `'/'` .

Q9. Write a function `is_prime(n)` that returns `True` if `n` is a prime number and `False` otherwise, using a loop.

[**Hint** –

    1. We only check prime for 2 or numbers greater than 2. `2` is the smallest prime number.

    2. A <u>non-Prime</u> number, `n` , will always get divided by atleast one number in range [2, n-1].

Eg – For number `9` we'll check in range (2, 8) & it'll get divided by 3. So `9` is non-prime & we'll return false for it.

For number `7` we'll check in range (2, 6) & it won't get divided by any. So `7` is prime & we'll return true for it.]

Q10. Let's create a "**Number Guessing Game**". Given a secret number (already decided by you), write a program that asks the user to guess it and prints:

- `"Too high"` if the guess is above the number
- `"Too low"` if the guess is below
- `"Correct!"` if the guess matches