Daily Assignment for Day 17: Recursion in Python

Practice Exercises, and MCQs

📌 Daily Assignment Questions (2 Questions)

1. Implement a Recursive Function for Exponentiation

Write a recursive function called power(base, exponent) that takes two arguments:

- base (integer)
- exponent (non-negative integer)

The function should return base raised to the power of exponent using recursion.

Example:

```
print(power(2, 3))  # Output: 8  (2^3 = 2 * 2 * 2)

print(power(5, 0))  # Output: 1  (Any number raised to 0 is 1)
```

---

2. Implement a Recursive Function for Finding the Greatest Common Divisor (GCD)

Write a recursive function gcd(a, b) to compute the **Greatest Common Divisor (GCD)** of two numbers using **Euclidean Algorithm**.

Example:

```
print(gcd(48, 18))  # Output: 6

print(gcd(56, 98))  # Output: 14
```

📌 Hint:

The formula for GCD using recursion is:

GCD(a, b) = GCD(b, a \mod b) \quad \text{(if b ≠ 0)}

Base case: When b == 0, return a.
