**Exercise 2: E-commerce Platform Search Function – Theoretical**

**1. Understanding Recursive Algorithms**

* Recursion is a programming concept where a method calls itself in order to solve a smaller version of the same problem. It breaks down complex problems into simpler sub-problems, making the code more elegant and easier to understand.
* A recursive function needs two main parts:
  1. A basecase that stops the recursion when a certain condition is met.
  2. A recursivecase that calls the function itself with modified inputs.
* Recursion is especially useful when the problem has a naturally recursive structure, such as tree traversals, factorial calculation, or predicting values over time.

**4. Analysis of the Recursive Algorithm**

* The timecomplexity of the recursive financial forecasting solution is **O(n)**, where n is the number of years. This is because the function makes one recursive call per year, processing each year exactly once until it reaches year 0.
* Although O(n) is acceptable for small to medium input sizes, recursive functions can still be inefficient if not handled properly, especially if they involve repeated calculations or deep recursion.
* To optimize the recursive approach and avoid unnecessary computation:

Memoization can be used to store the results of already computed years, so that the function doesn't calculate the same values repeatedly.

The recursive solution can be rewritten as an iterative one using a simple loop. This improves performance and avoids issues like stackoverflow that can happen with deep recursion.