**Exercise 7: Financial Forecasting**

**Explain the concept of recursion and how it can simplify certain problems**

* Recursion is a programming technique where a method calls itself to solve smaller instances of the same problem.
* It is especially useful for problems that can be broken into subproblems, such as factorials, Fibonacci series, and some tree/graph traversals.
* Recursion helps in writing clean, readable, and concise code, but care must be taken to avoid performance issues due to repeated calls and stack overflows.

**Time Complexity:**

* The recursive function runs once per year, so:
* Time Complexity: O(n) where n is the number of years.
* Space Complexity: O(n) due to the call stack (if not optimized).

**Optimization Tip:**

* To avoid excessive recursive computation:
* Use memoization to store intermediate results, or

**Replace with iterative approach to reduce stack usage:**

public static double calculateFutureValueIterative(double presentValue, double growthRate, int years) {

for (int i = 0; i < years; i++) {

presentValue \*= (1 + growthRate);

}

return presentValue;

}

**Iterative Version:**

* Time Complexity: O(n)
* Space Complexity: O(1) → more efficient for larger n.