**Exercise 7: Financial Forecasting**

**Scenario:**

You are developing a financial forecasting tool that predicts future values based on past data.

**What is a recursion?**

**Recursion** is a method in programming where a function calls itself to solve a smaller version of the original problem.

It is helpful in problems that have a repeating structure, such as:

* Calculating factorial
* Fibonacci series
* Tree traversal

In financial forecasting, we can recursively calculate the future value using the compound interest formula:

FV=PV×(1+r)n

Where:

* FV = Future Value
* PV = Present Value
* r = Growth rate per year
* n = Number of years

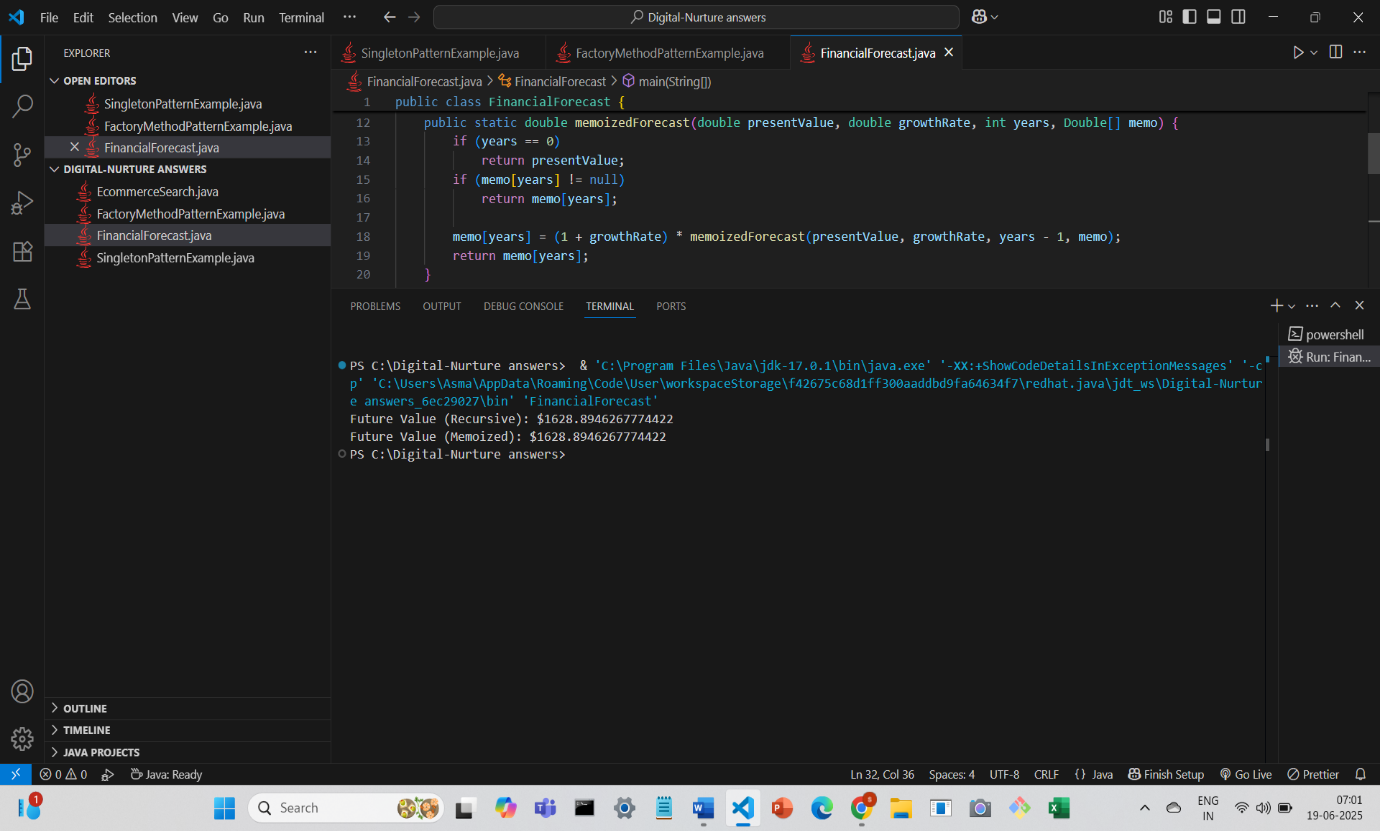
**Time Complexity:**

* **Basic recursion:**
  + Time Complexity = O(n)
  + Each year depends on the previous year, resulting in n recursive calls.
* **With memoization:**
  + Still O(n), but significantly faster by avoiding repeated calculations.
  + Memoization stores the result for each year so it's only computed once.

**Optimization Strategy:**

* Use a memoization array (e.g., Double[] memo) to store previously calculated values.
* Check if the result for the year already exists before computing.

**Output Screenshot:**



**Conclusion:**

we successfully implemented a recursive algorithm to forecast future financial values based on a given growth rate and initial investment.