***Skill – Algorithms\_Data Structures***

**Exercise 7: Financial Forecasting**

**Purpose:**

**The purpose of this exercise is to implement a financial forecasting tool using recursion. It will calculate future values based on past data and a consistent growth rate, demonstrating the application of recursive algorithms in real-world financial computations.**

**Understanding Recursive Algorithms:**

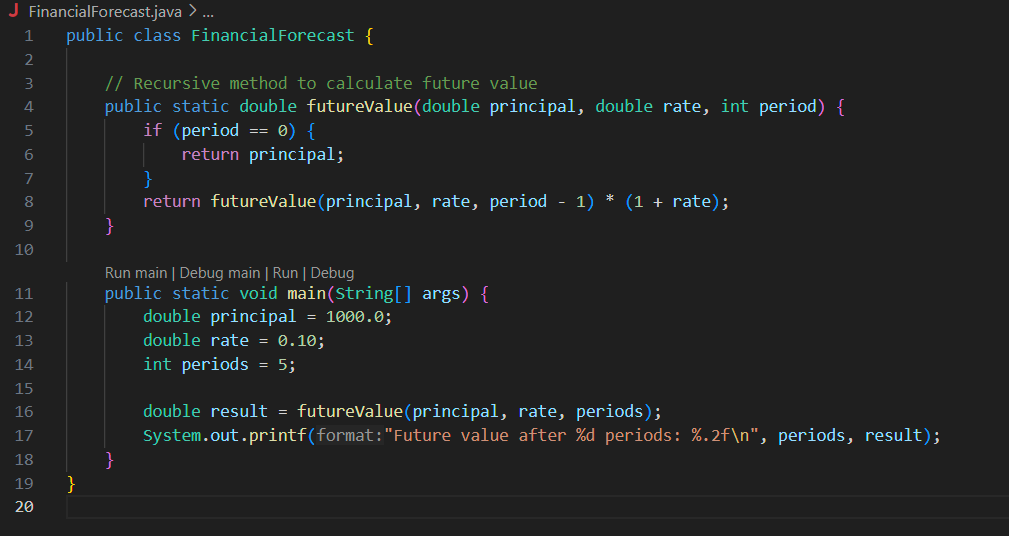
**Recursion is a programming technique where a method calls itself to solve smaller instances of the same problem. A recursive solution typically has a base case (to end the recursion) and a recursive case (to simplify the problem). It helps in solving problems that have repetitive substructures like calculating factorials, Fibonacci numbers, and in this case, forecasting financial values over a period.**

**Setup:**

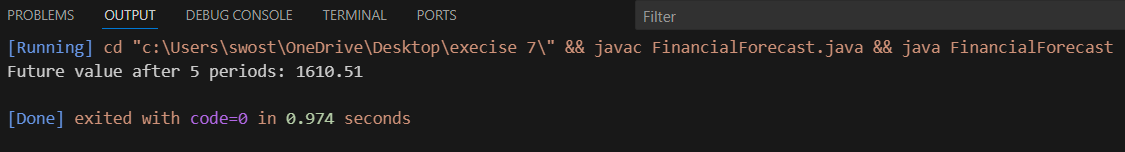
**To forecast future value recursively, we use the formula:  
  
FV(n) = FV(n-1) × (1 + r), where:  
- FV(n) is the future value after n periods,  
- r is the growth rate per period,  
- and the base case is FV(0) = P (initial principal).**

**Implementation:**

**Code:**



**Output:**



**Analysis:**

**Time Complexity: The recursive approach performs one call per period, so the time complexity is O(n).  
Space Complexity: The recursive stack grows linearly with the number of periods, resulting in O(n) space complexity.  
Optimization: Although recursion works, it can be optimized using an iterative approach or tail-recursion. Iteration avoids stack overhead and is generally more efficient for this use case.**

**Iterative efficient solution:**

