**What is Recursion?**

Recursion is a technique where a method calls itself to solve a problem.

**✅ When to Use It:**

* Problems that can be broken into smaller, similar subproblems.
* Examples: factorial, Fibonacci, tree traversal, financial projections, etc.

Java Code:

public class FinancialForecasting {

public static double forecast(int years, double currentValue, double growthRate) {

if (years == 0) {

return currentValue;

}

return forecast(years - 1, currentValue \* (1 + growthRate), growthRate);

}

public static void main(String[] args) {

int years = 5;

double currentValue = 10000;

double growthRate = 0.08;

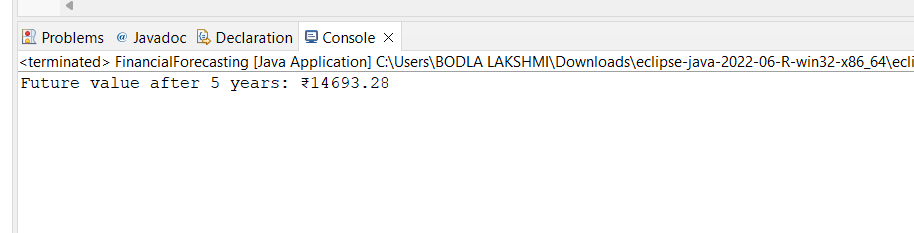
double futureValue = forecast(years, currentValue, growthRate);

System.out.printf("Future value after %d years: ₹%.2f%n", years, futureValue);

}

}

OUTPUT:



**Time Complexity**

* Recursive calls are made n times (for n years)
* **Time Complexity:** O(n)
* **Space Complexity:** O(n) → due to recursive call stack