**FinancialForecast.java**

public class FinancialForecast {

    // Recursive method to calculate future value

    public static double calculateFutureValue(double presentValue, double rate, int years) {

        if (years == 0)

            return presentValue;

        else

            return (1 + rate) \* calculateFutureValue(presentValue, rate, years - 1);

    }

    // Optimized version using memoization

    public static double calculateFutureValueMemo(double presentValue, double rate, int years, double[] memo) {

        if (years == 0)

            return presentValue;

        if (memo[years] != 0)

            return memo[years];

        memo[years] = (1 + rate) \* calculateFutureValueMemo(presentValue, rate, years - 1, memo);

        return memo[years];

    }

}

**Main.java**

public class Main {

    public static void main(String[] args) {

        double presentValue = 10000;

        double rate = 0.05; // 5% growth rate

        int years = 5;

        // Simple recursion

        double futureValue = FinancialForecast.calculateFutureValue(presentValue, rate, years);

        System.out.printf("📈 Predicted Future Value (Recursive): ₹%.2f\n", futureValue);

        // Optimized with memoization

        double[] memo = new double[years + 1];

        double futureValueMemo = FinancialForecast.calculateFutureValueMemo(presentValue, rate, years, memo);

        System.out.printf("⚡ Optimized Future Value (Memoized): ₹%.2f\n", futureValueMemo);

    }

}

**Output:**

