**Week-1: (Module 2)**

**Date Structures and Algorithms**

**Exercise 7: Financial Forecasting**

**FinancialForecast.java**

public class FinancialForecast {

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

if (years == 0) return initialValue;

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

}

public static void main(String[] args) {

double initialValue = 10000;

double growthRate = 0.08;

int years = 5;

double result = forecast(initialValue, growthRate, years);

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

}

}

Output:



Conclusion:

The recursive function modeled the growth process by repeatedly applying the growth rate for each year until the base case (year 0) was reached. The time complexity of the solution was **O(n)**, where n represents the number of years.