**Exercise 7: Financial Forecasting**

**Recursion** is a programming concept where a function calls **itself** in order to solve a problem.

Instead of solving a problem all at once, recursion breaks it down into **smaller sub-problems** of the same type — solving each one until it reaches a **base case** (the simplest version of the problem, which can be solved directly).

**FinancialForecast.java**

public class FinancialForecast {  
  
 public static double futureValue(double initialValue, double growthRate, int years) {  
 if (years == 0) {  
 return initialValue;  
 }  
 return *futureValue*(initialValue \* (1 + growthRate), growthRate, years - 1);  
 }  
  
 public static void main(String[] args) {  
 double initialInvestment = 10000;  
 double annualGrowthRate = 0.05;  
 int forecastYears = 10;  
  
 double predictedValue = *futureValue*(initialInvestment, annualGrowthRate, forecastYears);  
 System.*out*.format("Predicted Value after %d years: %.2f rs", forecastYears, predictedValue);  
 }  
}

**OUTPUT SCREENSHOT:**

