**Module 2**

**Exercise 7: Financial Forecasting**

**Understand Recursive Algorithms**

* Recursion is when a function **calls itself** to solve smaller subproblems of a larger problem.
* Simplifies problems that have **repetitive or nested structure**, like tree traversal or forecasting where current value depends on previous ones.

**Setup - Recursive Financial Forecasting Method**

FutureValue(n) = FutureValue(n-1) \* (1 + rate)

Where n is number of years,rate is annual growth rate

**Program:**

public class FinancialForecast {

public static double futureValue(double initial, double rate, int n) {

if (n == 0) {

return initial;

}

return (1 + rate) \* futureValue(initial, rate, n - 1);

}

public static void main(String[] args) {

double initial = 1000.0;

double rate = 0.05;

int years = 10;

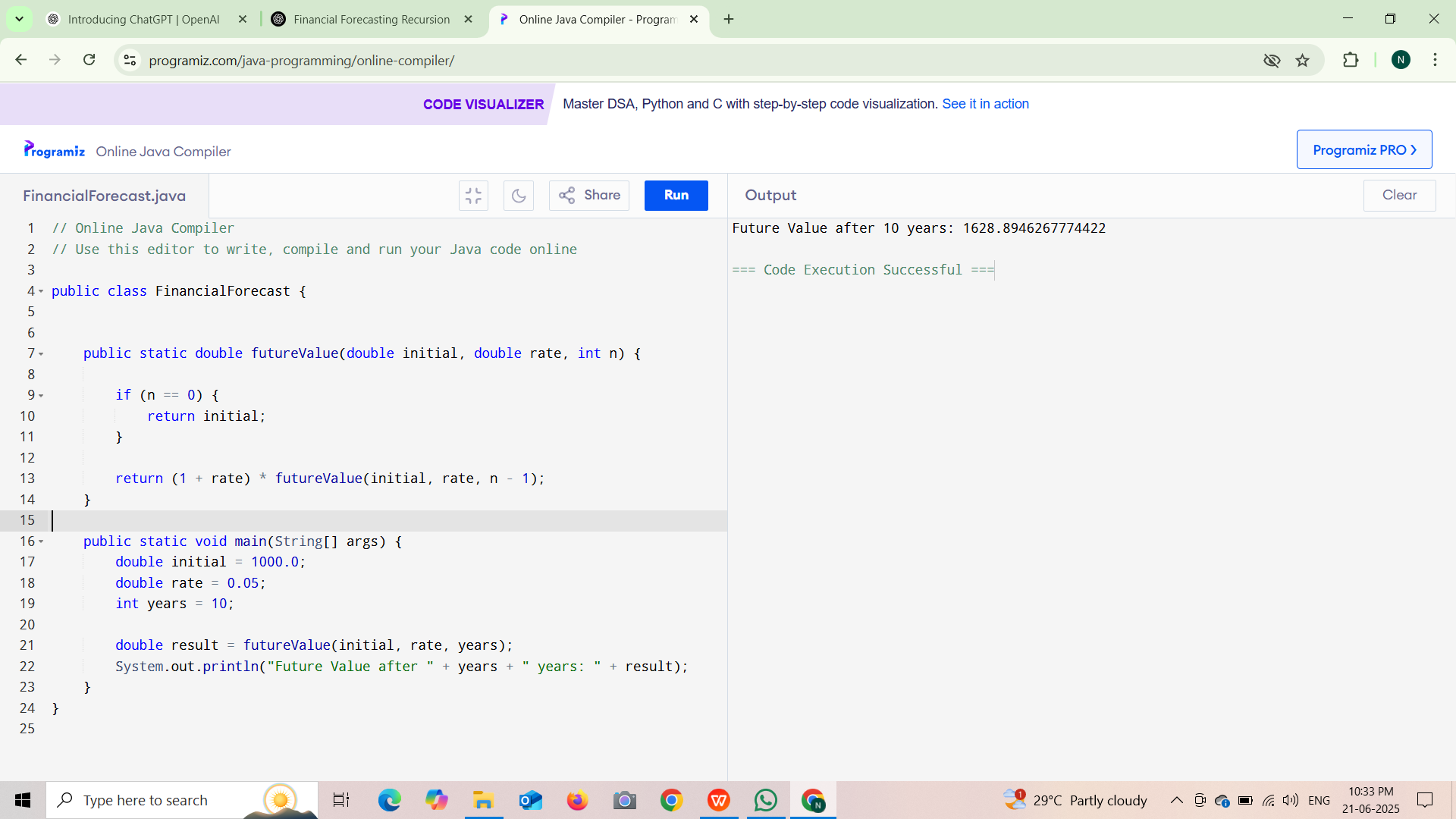
double result = futureValue(initial, rate, years);

System.out.println("Future Value after " + years + " years: " + result);

}

}

**OUTPUT:**



**Analysis:**

**Time Complexity**

Recursive Forecasting-O(n)

**How to Optimize:**

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

for (int i = 0; i < years; i++) {

presentValue \*= (1 + rate);

}

return presentValue;

}

More Efficient in memory