**Exercise 7: Financial Forecasting**

1. **Understanding Recursive Algorithm:**

* Recursion takes place when a function calls itself to solve a subproblem of the original task. It is useful when the problem has repeating structure or can be broken down into subproblems. Few examples where we can use it are:
  + Fibonacci Series
  + Factorial Calculation
  + Tree Traversal

**2 and 3. Setup and Implementation:**Shown in code.

**4. Analysis:**

* The time complexity of my program is O(n) where n = years.
* The best way to carry out this function would be by using exponentiation (Math.pow).  
  We can just ` return presentValue \* Math.pow(1 + growthRate, years) ` from the function and get our value in one go.  
  The time complexity for this would be O(1). It is the most optimal, although not recursive in nature.