

```

// ===== 1. Import Required Packages =====
import java.util.*;
import java.io.*;
import java.math.*;

// ===== 2. Main Class and Method =====
public class Main {
    public static void main(String[] args) {
        // ===== 3. Input Handling =====
        Scanner scanner = new Scanner(System.in);
        // Read the number of test cases
        int T = scanner.nextInt();
        // Loop through each test case
        for (int t = 0; t < T; t++) {
            // ===== 4. Problem Logic =====
            // Read the input values for this test case
            int N = scanner.nextInt();
            // Calculate the result based on the problem logic
            long result = calculateResult(N);
            // ===== 5. Output Handling =====
            // Print the result for this test case
            System.out.println(result);
        }
    }

    // ===== 6. Helper Methods =====
    // Method to calculate the result based on the problem logic
    private static long calculateResult(int N) {
        // Logic for calculating the result
        // Example: Sum of squares from 1 to N
        long sum = 0;
        for (int i = 1; i <= N; i++) {
            sum += i * i;
        }
        return sum;
    }
}

```

Problem Statement (Simplified):

Given a positive integer N , calculate the sum of the squares of all integers from 1 to N .

Input: The first line contains the number of test cases T . Each of the next T lines contains a positive integer N .

Output: For each test case, output the sum of the squares of integers from 1 to N .

Example:

Input: $T = 2$, $N_1 = 3$, $N_2 = 4$

Output: 14 (for $N_1 = 3$), 30 (for $N_2 = 4$)

Constraints:

- $1 \leq T \leq 100$
- $1 \leq N \leq 10^5$

Time Complexity: $O(N)$ per test case.

Space Complexity: $O(1)$.