ASSIGNMENT: Day_12

Task 1: Bit Manipulation Basics

Create a function that counts the number of set bits (1s) in the binary representation of an integer. Extend this to count the total number of set bits in all integers from 1 to n.

ANS:

```
public class BitManipulationBasics {



    public static int countSetBits(int n) {

        int count = 0;

        while (n > 0) {

            count += (n & 1);

            n >>= 1;

        }

        return count;

    }



    public static int countTotalSetBits(int n) {

        int totalCount = 0;

        for (int i = 1; i <= n; i++) {

            totalCount += countSetBits(i);

        }

        return totalCount;

    }
    public static void main(String[] args) {
        int n = 5;
        System.out.println("Number of set bits in " + n + ": " + countSetBits(n));
        System.out.println("Total number of set bits from 1 to " + n + ": " + countTotalSetBits(n));
    }
}
//code_by_RUBY
```

Task 2: Unique Elements Identification

Given an array of integers where every element appears twice except for two, write a function that efficiently finds these two non-repeating elements using bitwise XOR operations.

ANS:

```
public class UniqueElementsIdentification {

    public static int[] findTwoUniqueElements(int[] nums) {
        int xor = 0;

        for (int num : nums) {
            xor ^= num;
        }

        int setBit = xor & ~(xor - 1);

        int x = 0, y = 0;
        for (int num : nums) {
            if ((num & setBit) != 0) {
                x ^= num;
            } else {
                y ^= num;
            }
        }

        return new int[]{x, y};
    }

    public static void main(String[] args) {
        int[] nums = {1, 2, 3, 2, 1, 4};
        int[] uniqueElements = findTwoUniqueElements(nums);
        System.out.println("The two unique elements are: " + uniqueElements[0] + " and " + uniqueElements[1]);
    }
}
//code_by_RUBY
```