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)

Code:-

```
package Bit;
public class Bitcal {
  public static void main(String[] args) {
    int totalSum = 0;
                for (int i = 1; i <= 5; i++) {
      int Number = i;
      String binaryString = toBinaryString(Number);
      System.out.println("Binary representation of " + Number + " is = " +
binaryString);
      int countOnes = 0;
      int n = Number;
      while (n != 0) {
        countOnes += n & 1;
        n >>= 1;
      }
      System.out.println("Number of 1s in binary representation: " + countOnes);
      totalSum += countOnes;
    }
    System.out.println("Total sum of 1s in binary representations from 1 to 5: " +
totalSum);
  }
  public static String toBinaryString(int decimalNumber) {
    return Integer.toBinaryString(decimalNumber);
  }
}
```

OUTPUT:-

```
Binary representation of 1 is = 1
Number of 1s in binary representation: 1
Binary representation of 2 is = 10
Number of 1s in binary representation: 1
Binary representation of 3 is = 11
Number of 1s in binary representation: 2
Binary representation of 4 is = 100
Number of 1s in binary representation: 1
Binary representation of 5 is = 101
Number of 1s in binary representation: 2
Total sum of 1s in binary representations from 1 to 5: 7
```

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)
Code:-

```
package Bit;
public class NonRepeat {
  public static void findTwoUniqueElements(int[] arr) {
     int xor = 0;
     for (int num : arr) {
        xor ^= num;
     }
     int setBit = xor & ~(xor - 1);
     int num1 = 0, num2 = 0;
     for (int num : arr) {
        if ((num & setBit) == 0) {
           num1 ^= num;
        } else {
           num2 ^= num;
        }
     }
     System.out.println("The two non-repeating elements are " + num1 + " and " +
num2);
  }
  public static void main(String[] args) {
     int[] arr = {2, 4, 7, 9, 2, 4, 5, 7};
     findTwoUniqueElements(arr);
  }
}
```

OUTPUT:-

```
The two non-repeating elements are 9 and 5
```