7TH AUGUST RECURSION

Q1: Given an integer, find out the sum of its digits using recursion.

```java
public class DigitSum {
    public static void main(String[] args) {
        int n = 1234;
        int sum = sumOfDigits(n);
        System.out.println("Sum of digits: " + sum);
    }

    public static int sumOfDigits(int n) {
        if (n == 0) {
            return 0;
        }
        return n % 10 + sumOfDigits(n / 10);
    }
}
```

Q2: Given a number n, find the sum of natural numbers till n but with alternate signs.

```java
public class AlternateSum {
    public static void main(String[] args) {
        int n = 10;
        int result = alternateSignSum(n);
        System.out.println("Alternate sign sum: " + result);
    }

    public static int alternateSignSum(int n) {
        if (n == 0) {
            return 0;
        }
        if (n % 2 == 0) {
            return -n + alternateSignSum(n - 1);
        } else {
            return n + alternateSignSum(n - 1);
        }
    }
}
```

Q3: Print the max value of the array [13, 1, -3, 22, 5].

```java
public class MaxValueArray {
    public static void main(String[] args) {
        int[] array = {13, 1, -3, 22, 5};
        int max = Integer.MIN_VALUE;

        for (int num : array) {
            if (num > max) {
                max = num;
            }
        }

        System.out.println("Max value: " + max);
    }
}
```

Q4: Find the sum of the values of the array [92, 23, 15, -20, 10].

```java
public class ArraySum {
    public static void main(String[] args) {
        int[] array = {92, 23, 15, -20, 10};
        int sum = 0;

        for (int num : array) {
            sum += num;
        }

        System.out.println("Sum of array values: " + sum);
    }
}
```

Q5: Given a number n, print if it is an Armstrong number or not.

```java
public class ArmstrongNumber {
    public static void main(String[] args) {
        int n = 153;
        boolean isArmstrong = isArmstrongNumber(n);
        if (isArmstrong) {
            System.out.println("Yes");
        } else {
            System.out.println("No");
        }
    }

    public static boolean isArmstrongNumber(int n) {
        int originalNumber = n;
        int sum = 0;
        int totalDigits = String.valueOf(n).length();

        while (n > 0) {
            int digit = n % 10;
            sum += Math.pow(digit, totalDigits);
            n /= 10;
        }

        return sum == originalNumber;
    }
}
```