22 October 2024 16:22

Grading System Program

```
public class GradingSystem {
   public static void main(String[] args) {
      System.out.println("Grading System:");
      System.out.println("Input: 85 -> Grade: " + gradingSystem(85));
      System.out.println("Input: 70 -> Grade: " + gradingSystem(70));
      System.out.println("Input: 50 -> Grade: " + gradingSystem(50));
      System.out.println("Input: 30 -> Grade: " + gradingSystem(30));
   }

   public static String gradingSystem(int score) {
      if (score >= 80) return "A";
      else if (score >= 60) return "B";
      else if (score >= 40) return "C";
      else if (score >= 20) return "D";
      else return "F";
   }
}
```

Prime Number Check Program

```
public class PrimeNumberCheck {
   public static void main(String[] args) {
      System.out.println("Prime Number Check:");
      System.out.println("Input: 29 -> Is Prime? " + isPrime(29));
      System.out.println("Input: 15 -> Is Prime? " + isPrime(15));
      System.out.println("Input: 2 -> Is Prime? " + isPrime(2));
      System.out.println("Input: -7 -> Is Prime? " + isPrime(-7));
   }

   public static boolean isPrime(int num) {
      if (num <= 1) return false;
      for (int i = 2; i <= Math.sqrt(num); i++) {
         if (num % i == 0) return false;
      }
      return true;
   }
}
```

Sum of Natural Numbers Program

```
public class SumOfNaturalNumbers {
   public static void main(String[] args) {
      System.out.println("Sum of Natural Numbers:");
      System.out.println("Input: 5 -> Sum: " + sumOfNaturalNumbers(5));
      System.out.println("Input: -3 -> Sum: " + sumOfNaturalNumbers(-3));
```

```
        System.out.println("Input: 0 -> Sum: " + sumOfNaturalNumbers(0));
    }

    public static int sumOfNaturalNumbers(int n) {
        if (n < 1) return 0;
        return n * (n + 1) / 2;
    }
}
```

Count Digits Program

```
public class CountDigits {
    public static void main(String[] args) {
        System.out.println("Count Digits:");
        System.out.println("Input: 12345 -> Count: " + countDigits(12345));
        System.out.println("Input: -9876 -> Count: " + countDigits(-9876));
        System.out.println("Input: 0 -> Count: " + countDigits(0));
    }

    public static int countDigits(int num) {
        return String.valueOf(Math.abs(num)).length();
    }
}
```

Sum of All Numbers Program

```
import java.util.ArrayList;
import java.util.List;

public class SumOfAllNumbers {
    public static void main(String[] args) {
        List<Integer> numbers = new ArrayList<>();
        numbers.add(10);
        numbers.add(-5);
        numbers.add(20);
        System.out.println("Input List: [10, -5, 20] -> Sum: " + sumOfAllNumbers(numbers));

        numbers.clear();
        numbers.add(-10);
        numbers.add(-20);
        System.out.println("Input List: [-10, -20] -> Sum: " + sumOfAllNumbers(numbers));

        numbers.clear();
        numbers.add(0);
        System.out.println("Input List: [0] -> Sum: " + sumOfAllNumbers(numbers));
    }

    public static int sumOfAllNumbers(List<Integer> numbers) {
        int total = 0;
        for (int number : numbers) {
            total += number;
        }
        return total;
    }
}
```

MERGED ARRAY WITHOUT DUPLICATES

```
import java.util.ArrayList;
import java.util.Arrays;
import java.util.Scanner;
public class Main
{
   public static ArrayList<Integer> Main(int arr1[],int arr2[])
   {
   ArrayList<Integer> merged = new ArrayList();
   int i=0;
   int j=0;
   while(arr1[i]<arr1.length && arr2[j]>arr2.length)

   {
      if(arr1[i]>arr2[j])
      {
      if(!merged.contains(arr1[i]))
      {
      merged.add(arr1[i]);
      }
      i++;
      }
      else if(arr1[i]<arr2[j])
      {
      if(!merged.contains(arr2[i]))
      {
      merged.add(arr2[i]);
      }
      j++;
      }
      else
      {
         if(!merged.contains(arr1[i]))
      {
      merged.add(arr1[i]);
      }
      i++;
      j++;
      }
   }
      while (i < arr1.length) {
         if (!merged.contains(arr1[i])) {
            merged.add(arr1[i]);
         }
         i++;
      }

      while (j < arr2.length) {
         if (!merged.contains(arr2[j])) {
            merged.add(arr2[j]);
         }
         j++;
```

```
        }
        return merged;

    }
        public static void main(String[] args) {
        Scanner scanner = new Scanner(System.in);

        // Get input for the first array
        System.out.print("Enter the size of the first array: ");
        int size1 = scanner.nextInt();
        int[] arr1 = new int[size1];

        System.out.println("Enter the elements of the first array:");
        for (int i = 0; i < size1; i++) {
            System.out.print("Element " + (i + 1) + ": ");
            arr1[i] = scanner.nextInt();
        }

        // Get input for the second array
        System.out.print("Enter the size of the second array: ");
        int size2 = scanner.nextInt();
        int[] arr2 = new int[size2];

        System.out.println("Enter the elements of the second array:");
        for (int i = 0; i < size2; i++) {
            System.out.print("Element " + (i + 1) + ": ");
            arr2[i] = scanner.nextInt();
        }

        // Merge the two arrays
        ArrayList<Integer> mergedArray = Main(arr1, arr2);

        // Print the merged array
        System.out.println("Merged Array without duplicates: " + mergedArray);

        // Close the scanner to free resources
        scanner.close();
    }
}
```