

class PrintHW100 {

    public static void main(String[] args) {

        int i = 1;

        while (i <= 100) {

            System.out.println("Hello World! " + i);

            i++;

        }

    }

}



class Print1to10 {

    public static void main(String[] args) {

        int i = 1;

        while(i <= 10){

            System.out.print(i+" ");

            i++;

        }

    }

}



import java.util.Scanner;

class Print1toN {

    public static void main(String[] args) {

        System.out.println("Enter the Range = ");

        Scanner input = new Scanner(System.in);

        int N = input.nextInt();

        System.out.println("Printing 1 to "+N);

        int i = 1;

        while (i <= N) {

            System.out.println(i);

            i++;

        }

        input.close();

    }

}



import java.util.Scanner;

class PrintSumofNNaturalNo {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter the Range to calculate the Sum of N Natural Numbers = ");

        int Range = input.nextInt();

        int sum = 0, i = 1;

        while(i <= Range){

            sum += i;

            i++;

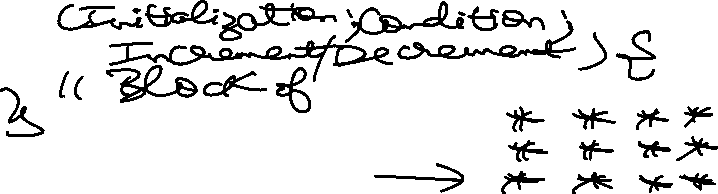
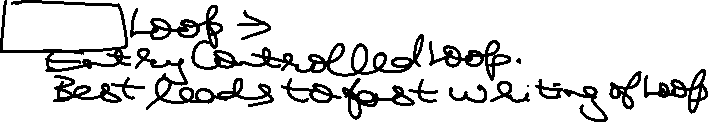
        }

        System.out.println("Sum of First N Natural Numbers is = "+sum);

        input.close();

    }

}



class Pattern {

    public static void main(String[] args) {

        for(int i = 1 ; i <= 4 ; i++){

            for (int j = 1; j<=4; j++){

                System.out.print("\* ");

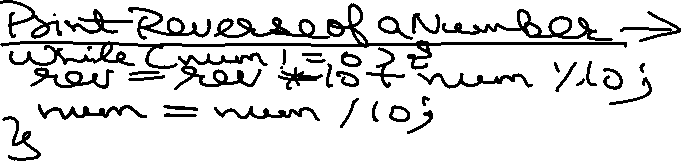
            }

            System.out.println("");

        }

    }

}



import java.util.Scanner;

class ReverseofNumber {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter the value of Number = ");

        int Number = input.nextInt();

        int rev = 0;

        while (Number != 0) {

            rev = rev \* 10 + Number % 10;

            Number = Number / 10;

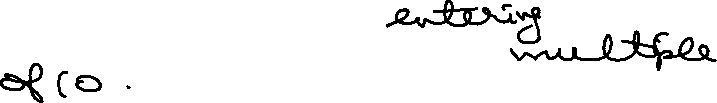
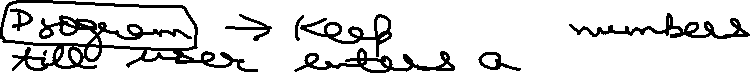
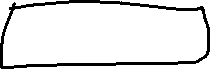
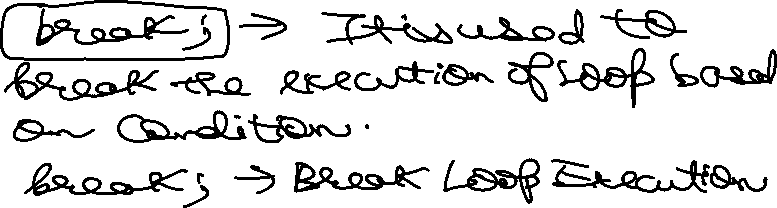
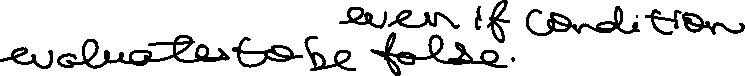
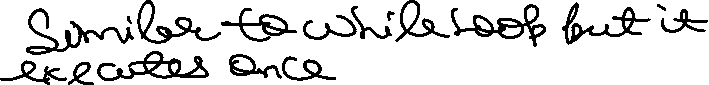
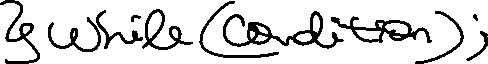
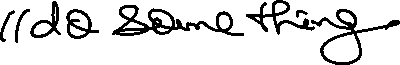
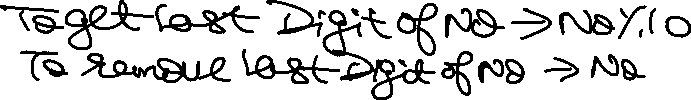
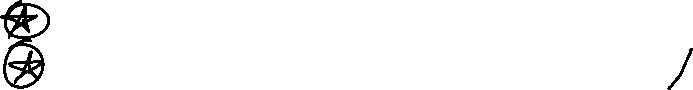
        }

        System.out.println("Reverse of Number is = "+rev);

        input.close();

    }

}



// Keep entering numbers till user enters a multiple of 10.

import java.util.Scanner;

class Multipleof10 {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter the Value = ");

        while(true){

            int N = input.nextInt();

            if(N % 10 == 0){

                System.out.println("Number "+N+" is a Multiple of 10.");

                break;

            }

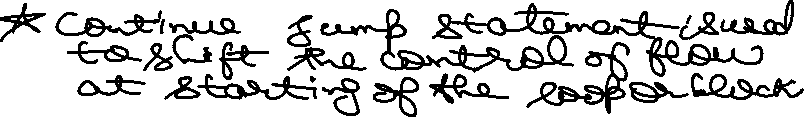
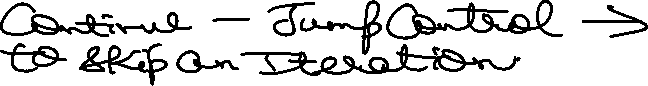
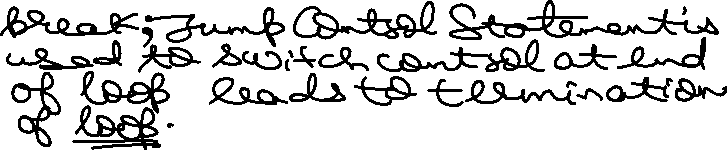
            System.out.println("Number you entered is = "+N);

        }

        input.close();

    }

}



//Display all numbers entered by user except multipes of 10

import java.util.Scanner;

class DisplayExceptMultipleof10 {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        while (true) {

            System.out.print("Enter the Number = ");

            int N = input.nextInt();

            if (N % 10 == 0) {

                continue;

            }

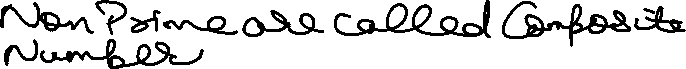
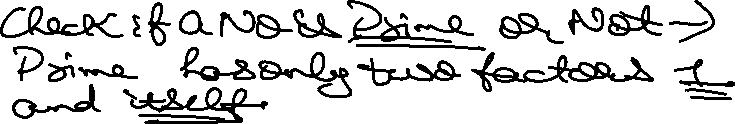
            System.out.println("Number is = "+N);

        }

    }

}

Jump statements in the C programming language allow programmers to alter the normal flow of execution in their code. These statements enable efficient control flow by providing options to break out of loops, skip iterations, transfer control to specific points, and terminate functions.



//Check if a number is Prime or Not!

import java.util.Scanner;

class PrimeorNot {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Program to check if a Number is Prime or Not!");

        System.out.print("Enter the Value of Number = ");

        boolean isPrime = true;

        int N = input.nextInt();

        //Prime Logic ->

        for(int i = 2 ; i < N ; i++){

            if(N % i == 0){

                isPrime = false;

                break;

            }

        }

        // Printing Prime Logic ->

        if (isPrime) {

            System.out.println("Number is a Prime Number = "+N);

        }else{

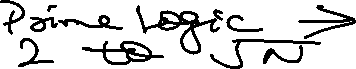
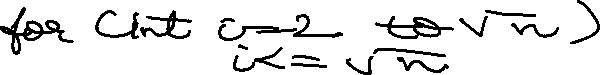
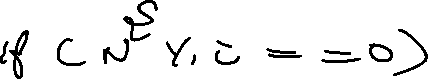
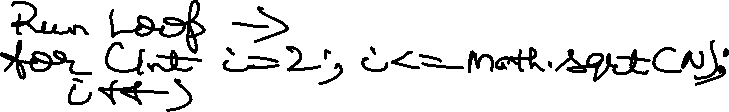
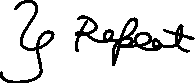
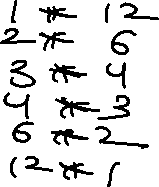
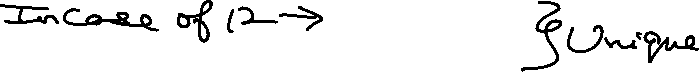
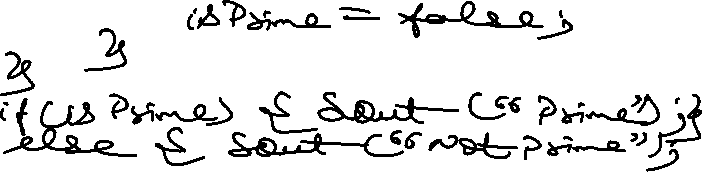
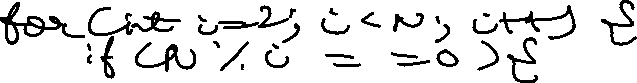
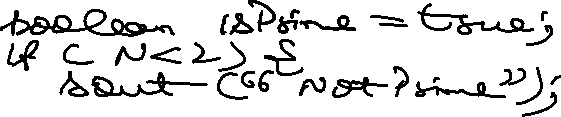
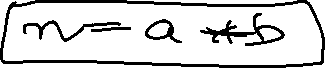
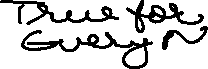
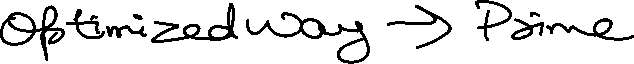
            System.out.println("Number is Not a Prime Number = "+N);

        }

        input.close();

    }

}



//Java Program to use Optimized Approach to solve Prime Number Problem

import java.util.Scanner;

class OptimizedPrime {

    public static void main(String[] args) {

        Scanner input = new Scanner(System.in);

        System.out.println("Enter the Value of Number = ");

        int N = input.nextInt();

        boolean isPrime = true;

        // Edge Cases

        if(N < 2){

            isPrime = false;

        }

        // N > 2 Cases

        for(int i = 2; i <= Math.sqrt(N) + 1; i++){

            if(N % i == 0){

                isPrime = false;

                break; // No need to continue checking

            }

        }

        // Printing the Output ->

        if (isPrime) {

            System.out.println("Number is a Prime Number = "+N);

        }else{

            System.out.println("Number is Not a Prime Number = "+N);

        }

        input.close();