# **object oriented programming and design**

**Looping Statements**

1. **Display the first N natural numbers**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter N: ");

int N = sc.nextInt();

System.out.println("First " + N + " natural numbers:");

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

System.out.print(i + " ");

}

}

}

1. **Display the multiplication table of a given integer**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

System.out.println("Multiplication table of " + num + ":");

for (int i = 1; i <= 10; i++) {

System.out.println(num + " x " + i + " = " + (num \* i));

}

}

}

1. **Find the cube of numbers up to a given integer**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

System.out.println("Cubes of numbers up to " + num + ":");

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

System.out.println("Cube of " + i + " is " + (i \* i \* i));

}

}

}

1. **Display N terms of odd natural numbers and their sum**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of terms: ");

int N = sc.nextInt();

int sum = 0;

System.out.println(N + " odd numbers are:");

for (int i = 1, count = 0; count < N; i += 2) {

System.out.print(i + " ");

sum += i;

count++;

}

System.out.println("\nSum: " + sum);

}

}

1. **Calculate the factorial of a given number**

import java.util.Scanner;

public class Main {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int fact = 1;

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

fact \*= i;

}

System.out.println("Factorial of " + num + " is " + fact);

}

}

1. **Display N terms of even natural numbers and their sum**

import java.util.Scanner;

public class EvenNumbers {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of terms: ");

int N = sc.nextInt();

int sum = 0;

System.out.println(N + " even numbers are:");

for (int i = 2, count = 0; count < N; i += 2) {

System.out.print(i + " ");

sum += i;

count++;

}

System.out.println("\nSum: " + sum);

}

}

1. **Find the sum of digits of a given number**

import java.util.Scanner;

public class SumOfDigits {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int sum = 0;

while (num > 0) {

sum += num % 10;

num /= 10;

}

System.out.println("Sum of digits: " + sum);

}

}

1. **Determine whether a given number is prime or not**

import java.util.Scanner;

public class PrimeCheck {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

boolean isPrime = true;

if (num < 2) {

isPrime = false;

} else {

for (int i = 2; i <= num / 2; i++) {

if (num % i == 0) {

isPrime = false;

break;

}

}

}

if (isPrime) {

System.out.println(num + " is a prime number.");

} else {

System.out.println(num + " is not a prime number.");

}

}

}

1. **Check whether a given number is an Armstrong number or not**

import java.util.Scanner;

public class ArmstrongCheck {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int originalNum = num, result = 0, digits = 0;

// Count digits

while (originalNum != 0) {

originalNum /= 10;

digits++;

}

originalNum = num;

// Check Armstrong

while (originalNum != 0) {

int remainder = originalNum % 10;

result += Math.pow(remainder, digits);

originalNum /= 10;

}

if (result == num) {

System.out.println(num + " is an Armstrong number.");

} else {

System.out.println(num + " is not an Armstrong number.");

}

}

}

1. **Display the sum of the series [ 9 + 99 + 999 + 9999 ...]**

import java.util.Scanner;

public class SumOfSeries {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of terms: ");

int N = sc.nextInt();

int sum = 0, term = 9;

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

sum += term;

System.out.print(term + " ");

term = term \* 10 + 9;

}

System.out.println("\nSum: " + sum);

}

}

1. **Display the first N terms of Fibonacci series**

import java.util.Scanner;

public class FibonacciSeries {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of terms: ");

int N = sc.nextInt();

int a = 0, b = 1;

System.out.println("Fibonacci series:");

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

System.out.print(a + " ");

int next = a + b;

a = b;

b = next;

}

}

}

1. **Display a number in reverse order**

import java.util.Scanner;

public class ReverseNumber {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int reversed = 0;

while (num != 0) {

int digit = num % 10;

reversed = reversed \* 10 + digit;

num /= 10;

}

System.out.println("Reversed number: " + reversed);

}

}

1. **Check whether a number is a palindrome or not**

import java.util.Scanner;

public class PalindromeCheck {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter a number: ");

int num = sc.nextInt();

int originalNum = num;

int reversed = 0;

while (num != 0) {

int digit = num % 10;

reversed = reversed \* 10 + digit;

num /= 10;

}

if (originalNum == reversed) {

System.out.println(originalNum + " is a palindrome.");

} else {

System.out.println(originalNum + " is not a palindrome.");

}

}

}

1. **Find the sum of the series 1 + 1/2 + 1/3 + ... + 1/n**

import java.util.Scanner;

public class HarmonicSeries {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the number of terms: ");

int N = sc.nextInt();

double sum = 0.0;

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

sum += 1.0 / i;

}

System.out.println("Sum of the series: " + sum);

}

}

1. **Find the sum of the series 1 + 1/(2*2) + 1/(3*3) + ... + 1/(n\*n)**

import java.util.Scanner;

public class SquareSeriesSum {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter the value of n: ");

int n = sc.nextInt();

double sum = 0.0;

// Calculate the sum of the series

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

sum += 1.0 / (i \* i); // Add 1/(i^2) to the sum

}

// Display the result

System.out.println("The sum of the series 1 + 1/(2^2) + 1/(3^2) + ... + 1/(" + n + "^2) is: " + sum);

}

}

1. **break Statement Example**

public class BreakExample {

public static void main(String[] args) {

System.out.println("Example of 'break' statement:");

for (int i = 1; i <= 5; i++) {

if (i == 3) {

break; // Exits the loop when i equals 3

}

System.out.println("i = " + i);

}

}

}

1. **Labelled break Statement Example**

public class LabelledBreakExample {

public static void main(String[] args) {

System.out.println("Example of 'labelled break' statement:");

outerLoop:

for (int i = 1; i <= 3; i++) {

for (int j = 1; j <= 3; j++) {

if (i == 2 && j == 2) {

break outerLoop; // Breaks out of both loops when i == 2 and j == 2

}

System.out.println("i = " + i + ", j = " + j);

}

}

}

}

1. **continue Statement Example**

public class ContinueExample {

public static void main(String[] args) {

System.out.println("Example of 'continue' statement:");

for (int i = 1; i <= 5; i++) {

if (i == 3) {

continue; // Skips the iteration when i equals 3

}

System.out.println("i = " + i);

}

}

}

1. **Labelled continue Statement Example**

public class LabelledContinueExample {

public static void main(String[] args) {

System.out.println("Example of 'labelled continue' statement:");

outerLoop:

for (int i = 1; i <= 3; i++) {

for (int j = 1; j <= 3; j++) {

if (i == 2 && j == 2) {

continue outerLoop; // Skips to the next iteration of the outer loop

}

System.out.println("i = " + i + ", j = " + j);

}

}

}

}

1. **return Statement Example**

public class ReturnExample {

public static void main(String[] args) {

System.out.println("Example of 'return' statement:");

for (int i = 1; i <= 5; i++) {

if (i == 3) {

return; // Exits the method when i equals 3

}

System.out.println("i = " + i);

}

System.out.println("This line will not be executed.");

}

}