**ASHISH JADHAV\_JUHU**

**ASSGNMENT NO.3**

**1.Write a program to calculate the sum of the first 50 natural numbers.**

public class SumOfNaturalNumbers

{

public static void main(String[] args)

{

int sum = 0; // Variable to store the sum

// Loop to calculate the sum of first 50 natural numbers

for (int i = 1; i <= 50; i++) {

sum += i; // Add the current number (i) to the sum

}

// Print the result

System.out.println("The sum of the first 50 natural numbers is: " + sum);

}

}

**O/P**

The sum of the first 50 natural numbers is: 1275

**------------------------------------------------------------------------------------------------**

**2.Write a program to compute the factorial of the number 10.**

public class Factorial

{

public static void main(String[] args)

{

int num = 10; // The number for which we need to calculate the factorial

long factorial = 1; // Initialize the result as 1 (we use long to handle large values)

// Loop to calculate factorial of the number

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

factorial \*= i; // Multiply current value of i to factorial

}

// Print the result

System.out.println("The factorial of " + num + " is: " + factorial);

}

}

**O/P**

**The factorial of 10 is: 3628800**

**------------------------------------------------------------------------------------------------**

**3. Write a program to print all multiples of 7 between 1 and 100.**

public class MultiplesOfSeven

{

public static void main(String[] args)

{

// Loop through numbers from 1 to 100

for (int i = 1; i <= 100; i++)

{

// Check if the number is a multiple of 7

if (i % 7 == 0)

{

System.out.println(i); // Print the number if it's a multiple of 7

}

}

}

}

**O/P**

7

14

21

28

35

42

49

56

63

70

77

84

91

98

**------------------------------------------------------------------------------------------------**

**4. Write a program to reverse the digits of the number 1234. The output should be 4321.**

public class ReverseNumber

{

public static void main(String[] args)

{

int number = 1234; // Number to be reversed

int reversedNumber = 0; // Variable to store the reversed number

// Loop until the number becomes 0

while (number != 0) {

int digit = number % 10; // Get the last digit

reversedNumber = reversedNumber \* 10 + digit; // Append the digit to the reversed number

number = number / 10; // Remove the last digit from the original number

}

// Print the reversed number

System.out.println("Reversed Number: " + reversedNumber);

}

}

**O/P**

Reversed Number: 4321

**------------------------------------------------------------------------------------------------**

**5. Write a program to print the Fibonacci sequence up to the number 21.**

public class FibonacciSequence

{

public static void main(String[] args)

{

int limit = 21; // The upper limit for the Fibonacci sequence

int a = 0; // The first number in the Fibonacci sequence

int b = 1; // The second number in the Fibonacci sequence

// Print the initial numbers of the Fibonacci sequence

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

// Generate the Fibonacci sequence up to the specified limit

while (true) {

int next = a + b; // Calculate the next number in the sequence

if (next > limit) {

break; // Exit the loop if the next number exceeds the limit

}

System.out.print(" " + next); // Print the next number in the sequence

a = b; // Update a to the previous b

b = next; // Update b to the next number

}

System.out.println(); // Move to the next line after printing the sequence

}

}

**O/P**

0 1 1 2 3 5 8 13 21

**------------------------------------------------------------------------------------------------**

**6. Write a program to find and print the first 5 prime numbers.**

public class FirstFivePrimes

{

public static void main(String[] args)

{

int count = 0; // Counter for the number of prime numbers found

int num = 2; // The number to check for primality

// Loop until we find the first 5 prime numbers

while (count < 5) {

if (isPrime(num)) {

System.out.print(num + " "); // Print the prime number

count++; // Increment the counter

}

num++; // Move to the next number

}

System.out.println(); // Move to the next line after printing all prime numbers

}

**---------------------------------------------------------------------------------------------**

**7. Write a program to calculate the sum of the digits of the number 9876. The output should be**

**30 (9 + 8 + 7 + 6).**

public class SumOfDigits

{

public static void main(String[] args)

{

int number = 9876; // The number whose digits will be summed

int sum = 0; // Variable to store the sum of the digits

// Loop to process each digit of the number

while (number > 0)

{

int digit = number % 10; // Extract the last digit of the number

sum += digit; // Add the digit to the sum

number /= 10; // Remove the last digit from the number

}

// Print the result

System.out.println("The sum of the digits is: " + sum);

}

}

**------------------------------------------------------------------------------------------------**

**8. Write a program to count down from 10 to 0, printing each number.**

public class Countdown

{

public static void main(String[] args)

{

// Loop to count down from 10 to 0

for (int i = 10; i >= 0; i--) {

System.out.println(i); // Print the current value of i

}

}

}

**------------------------------------------------------------------------------------------------**

**9. Write a program to find and print the largest digit in the number 4825.**

public class LargestDigit

{

public static void main(String[] args)

{

int number = 4825; // The number from which we want to find the largest digit

int largestDigit = 0; // Variable to store the largest digit found

// Loop to process each digit of the number

while (number > 0) {

int digit = number % 10; // Extract the last digit of the number

if (digit > largestDigit) {

largestDigit = digit; // Update the largest digit if the current digit is larger

}

number /= 10; // Remove the last digit from the number

}

// Print the largest digit

System.out.println("The largest digit is: " + largestDigit);

}

}

**------------------------------------------------------------------------------------------------**

**10. Write a program to print all even numbers between 1 and 50.**

public class EvenNumbers

{

public static void main(String[] args)

{

// Loop through numbers from 1 to 50

for (int i = 1; i <= 50; i++)

{

// Check if the number is even

if (i % 2 == 0)

{

// Print the even number

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

}

}

}

}

**------------------------------------------------------------------------------------------------**

**11. Write a Java program to demonstrate the use of both pre-increment and post-decrement operators in a single expression**

public class IncrementDecrementDemo

{

public static void main(String[] args)

{

int i = 5; // Initialize variable i

// Demonstrate pre-increment and post-decrement in a single expression

int result = ++i + i--;

// Print the results

System.out.println("Value of i after operations: " + i);

System.out.println("Result of the expression ++i + i--: " + result);

}

}

**------------------------------------------------------------------------------------------------**

**12. Write a program to draw the following pattern:**

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

**\*\*\*\*\***

public class SquareStarPattern {

public static void main(String[] args) {

int rows = 5; // Number of rows in the pattern

int columns = 5; // Number of columns in the pattern

// Loop through each row

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

// Loop through each column

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

// Print a star

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

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**13. Write a program to print the following pattern:**

**1**

**2\*2**

**3\*3\*3**

**4\*4\*4\*4**

**5\*5\*5\*5\*5**

**5\*5\*5\*5\*5**

**4\*4\*4\*4**

**3\*3\*3**

**2\*2**

**1**

public class NumberPattern

{

public static void main(String[] args)

{

int n = 5; // Maximum number for the pattern

// Print the top half of the pattern

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

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

if (j > 1) {

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

}

System.out.print(i);

}

System.out.println(); // Move to the next line after each row

}

// Print the bottom half of the pattern

for (int i = n; i >= 1; i--) {

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

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

System.out.print(i);

}

System.out.println(); // Move to the next line after each row

}

}

}

**------------------------------------------------------------------------------------------------**

**14. Write a program to print the following pattern:**

**\***

**\*\***

**\*\*\***

**\*\*\*\*\***

**\*\*\*\*\*\*\***

**\*\*\*\*\*\*\*\*\***

public class TriangleStarPattern

{

public static void main(String[] args)

{

int rows = 6; // Number of rows in the pattern

// Loop through each row

for (int i = 1; i <= rows; i++)

{

// Print leading spaces

for (int j = 1; j < i; j++)

{

System.out.print(" ");

}

// Print stars

for (int j = 1; j <= (2 \* i - 1); j++)

{

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

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**15. Write a program to print the following pattern:**

**\***

**\*\***

**\*\*\***

**\*\*\*\***

**\*\*\*\*\***

public class TriangleStarPattern {

public static void main(String[] args) {

int rows = 5; // Number of rows in the pattern

// Loop through each row

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

// Print leading spaces

for (int j = rows; j > i; j--) {

System.out.print(" ");

}

// Print stars

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

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

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**16. Write a program to print the following pattern:**

**\***

**\*\*\***

**\*\*\*\*\***

**\*\*\*\*\*\*\***

**\*\*\*\*\*\*\*\*\***

public class StarPyramidPattern

{

public static void main(String[] args)

{

int rows = 5; // Number of rows in the pattern

// Loop through each row

for (int i = 1; i <= rows; i++)

{

// Print leading spaces

for (int j = rows; j > i; j--)

{

System.out.print(" ");

}

// Print stars

for (int j = 1; j <= (2 \* i - 1); j++)

{

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

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**17. Write a program to print the following pattern:**

**\*\*\*\*\***

**\*\*\*\***

**\*\*\***

**\*\***

**\***

public class InvertedTrianglePattern

{

public static void main(String[] args)

{

int rows = 5; // Number of rows in the pattern

// Loop through each row

for (int i = 0; i < rows; i++)

{

// Print leading spaces

for (int j = 0; j < i; j++)

{

System.out.print(" ");

}

// Print stars

for (int j = rows - i; j > 0; j--)

{

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

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**18. Write a program to print the following pattern:**

**\***

**\*\*\***

**\*\*\*\*\***

**\*\*\*\*\*\*\***

**\*\*\*\*\***

**\*\*\***

**\***

public class StarDiamondPattern

{

public static void main(String[] args)

{

int n = 4; // Number of rows in the top half of the pattern

// Print the top half of the pattern

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

{

// Print leading spaces

for (int j = n; j > i; j--)

{

System.out.print(" ");

}

// Print stars

for (int j = 1; j <= (2 \* i - 1); j++)

{

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

}

System.out.println(); // Move to the next line

}

// Print the bottom half of the pattern

for (int i = n - 1; i >= 1; i--) {

// Print leading spaces

for (int j = n; j > i; j--) {

System.out.print(" ");

}

// Print stars

for (int j = 1; j <= (2 \* i - 1); j++) {

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

}

System.out.println(); // Move to the next line

}

}

}

**------------------------------------------------------------------------------------------------**

**19. Write a program to print the following pattern:**

**1**

**1\*2**

**1\*2\*3**

**1\*2\*3\*4**

**1\*2\*3\*4\*5**

public class NumberStarPatternAscending

{

public static void main(String[] args)

{

int rows = 5; // Number of rows in the pattern

// Loop through each row

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

// Loop to print numbers and stars

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

// Print the ascending number (j)

System.out.print(j);

// Print a star if it's not the last number in the row

if (j < i) {

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

}

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**20. Write a program to print the following pattern:**

**5**

**5\*4**

**5\*4\*3**

**5\*4\*3\*2**

**5\*4\*3\*2\*1**

public class NumberStarPatternDescending

{

public static void main(String[] args)

{

int rows = 5; // Number of rows in the pattern

// Loop through each row

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

// Loop to print numbers and stars

for (int j = 0; j < i; j++) {

// Print the descending number (5 - j)

System.out.print((5 - j));

// Print a star if it's not the last number in the row

if (j < i - 1) {

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

}

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**21. Write a program to print the following pattern:**

**1**

**1\*3**

**1\*3\*5**

**1\*3\*5\*7**

**1\*3\*5\*7\*9**

public class NumberStarPattern

{

public static void main(String[] args)

{

int rows = 5; // Number of rows in the pattern

// Loop through each row

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

// Loop to print numbers and stars

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

// Print the odd number (2 \* j - 1)

System.out.print((2 \* j - 1));

// Print a star if it's not the last number in the row

if (j < i) {

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

}

}

// Move to the next line after each row

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**22. Write a program to print the following pattern:**

**\*\*\*\*\*\*\*\*\***

**\*\*\*\*\*\*\***

**\*\*\*\*\***

**\*\*\***

**\***

**\*\*\***

**\*\*\*\*\***

**\*\*\*\*\*\*\***

**\*\*\*\*\*\*\*\*\***

public class StarPattern

{

public static void main(String[] args)

{

int n = 5; // Number of rows for the upper half (including the middle row)

// Upper half of the diamond (including the middle row)

for (int i = n; i >= 1; i--) {

// Print leading spaces

for (int j = n; j > i; j--) {

System.out.print(" ");

}

// Print stars

for (int k = 1; k <= (2 \* i - 1); k++) {

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

}

System.out.println();

}

// Lower half of the diamond (excluding the middle row)

for (int i = 2; i <= n; i++) {

// Print leading spaces

for (int j = n; j > i; j--) {

System.out.print(" ");

}

// Print stars

for (int k = 1; k <= (2 \* i - 1); k++) {

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

}

System.out.println();

}

}

}

**------------------------------------------------------------------------------------------------**

**23. Write a program to print the following pattern:**

**11111**

**22222**

**33333**

**44444**

**55555**

public class NumberPattern

{

public static void main(String[] args)

{

// Outer loop to handle the rows

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

// Inner loop to print the same number in each row

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

System.out.print(i); // Print the value of i (same number for the entire row)

}

System.out.println(); // Move to the next line after each row

}

}

}

**------------------------------------------------------------------------------------------------**

**24. Write a program to print the following pattern:**

**1**

**22**

**333**

**4444**

**55555**

public class NumberPattern

{

public static void main(String[] args)

{

// Outer loop to handle the rows

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

// Inner loop to print numbers in each row

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

System.out.print(i); // Print the value of i (same number multiple times)

}

System.out.println(); // Move to the next line after each row

}

}

}

**------------------------------------------------------------------------------------------------**

**25. Write a program to print the following pattern:**

**1**

**12**

**123**

**1234**

**12345**

public class NumberPattern

{

public static void main(String[] args)

{

// Outer loop to handle the rows

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

// Inner loop to print numbers in each row

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

System.out.print(j); // Print the current value of j

}

System.out.println(); // Move to the next line after each row

}

}

}

**------------------------------------------------------------------------------------------------**

**26. Write a program to print the following pattern:**

**1**

**2 3**

**4 5 6**

**7 8 9 10**

**11 12 13 14 15**

public class NumberPattern

{

public static void main(String[] args)

{

int num = 1;

// Loop to print the pattern

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

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

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

num++;

}

System.out.println(); // Move to the next line after each row

}

}

}