Day-3

**1.Perfect Number**

public class perfect

{

public static void main(String args[])

{

int sum = 0,n=10,i;

for(i=1;i<=n/2;i++)

{

if(n % i==0)

{

sum = sum + i;

}

}

if(sum == n && n!=0)

{

System.out.println(sum + " is Perfect number");

}

else

{

System.out.println(sum + " is not a Perfect number ");

}

}

}

**Output:**

6 is Perfect number

**2. Composite Number**

public class compositeChecker{

public static void main(String[] args)

{

int num = 12,i;

if(num <= 1)

{

System.out.println("invalid");

}

else{

boolean isComposite = false;

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

{

if(num % i == 0)

{

isComposite = true;

break;

}

}

if(isComposite)

{

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

}

else{

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

}

}

}

}

**Output:**

4 is a composite number

**3. Prime Number**

public class PrimeNumberChecker {

public static void main(String[] args) {

int number = 29;

boolean isPrime = true;

// Edge cases

if (number <= 1) {

isPrime = false;

} else {

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

if (number % i == 0) {

isPrime = false;

break;

}

}

}

if (isPrime) {

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

} else {

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

}

}

}

**Output:**

7 is a Prime number

**4. Vowels And Consonants**

import java.util.Scanner;

public class vowels{

public static void main(String args[])

{

char ch;

String input;

int count = 0,vowels=0,consonants=0;

Scanner scanner = new Scanner(System.in);

input = scanner.nextLine();

for(int i=0;i<input.length();i++)

{

ch = input.charAt(i);

input = input.toLowerCase();

if(ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u')

{

vowels++;

}

else

{

consonants++;

}

}

System.out.println("NUMBER OF VOWELS IS :" + vowels);

System.out.println("NUMBER OF CONSONANTS IS :" + consonants);

}

}

**Output:**

Vamsi

Number of vowels is : 2

Number of consonants is : 3

**5.Palindrome number**

public class PalindromeChecker {

public static void main(String[] args) {

String str = "madam";

boolean isPalindrome = true;

int length = str.length();

for (int i = 0; i < length / 2; i++) {

if (str.charAt(i) != str.charAt(length - 1 - i)) {

isPalindrome = false;

break;

}

}

if (isPalindrome) {

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

} else {

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

}

}

}

**Output:**

Madam

**6.** **Fibonacci Series**

public class FibonacciSeries {

public static void main(String[] args) {

int n = 10;

int firstTerm = 0, secondTerm = 1;

System.out.println("Fibonacci Series up to " + n + " terms:");

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

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

// Compute the next term

int nextTerm = firstTerm + secondTerm;

firstTerm = secondTerm;

secondTerm = nextTerm;

}

}

}

**Output:**

Fibonacci Series up to 10 terms:

0 1 1 2 3 5 8 13 21 34

**7. Descending Order of String**

public class ascAlpha

{

public static void main(String args[])

{

String[] arr = {"vamsi","bobby","nani","sudheer","raghu"};

int n = arr.length,i,j;

String temp;

for(i=0;i<n-1;i++)

{

for(j=0;j<n-1-i;j++)

{

if(arr[j].compareTo(arr[j + 1]) < 0)

{

temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

}

System.out.println("Descending order");

for(String array : arr)

{

System.out.println(array);

}

}

}

**Output:**

Vamsi,Sudheer,Raghu,nani,bobby

**8. Special Characters**

public class special

{

public static void main(String args[])

{

String num = "123@#$";

char ch;

StringBuilder specialcharacter = new StringBuilder();

int count=0;

for(int i=0;i<num.length();i++)

{

ch = num.charAt(i);

if(!Character.isLetterOrDigit(ch) && !Character.isWhitespace(ch))

{

specialCharacter.append(ch);

count++;

}

}

System.out.println(specialCharacter.toString() + "Special Characters");

System.out.println(count + "Special Characters count");

}

}

**Output:**

vamsi@#$%

@#$% Special Characters

4 Special characters count

**9. Pyramid Pattern**

public class practice{

public static void main(String args[]){

int i,j;

int n=5;

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

{

for(j=1;j<=n-i;j++)

{

System.out.print(" ");

}

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

{

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

}

System.out.println();

}

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

{

for(j=1;j<=n-i;j++)

{

System.out.print(" ");

}

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

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

}

System.out.println();

}

}

}

**10. Matrix multiplication**

public class mulMatrix

{

public static void main(String[] args)

{

int i,j;

int[][] A={

{2, 3},

{4, 5}

};

int[][] B={

{6, 7},

{8, 9}

};

int[][] C = new int[2][2];

for(i=0;i<2;i++)

{

for(j=0;j<2;j++)

{

C[i][j] = A[i][j] + B[i][j];

}

}

for(i=0;i<2;i++)

{

for(j=0;j<2;j++)

{

System.out.println(C[i][j] + " ");

}

}

System.out.println();

}

}

**Output:**

(2+6) (3+7) -> 8 10

(4+8) (5+9) -> 12 14