NAME:Gundupalli Srujan Deep

REG:192125009

COURSE:CSA0978-Programming in Java

1.

import java.io.BufferedReader;

import java.io.FileReader;

import java.io.IOException;

public class FileStats {

public static void main(String[] args) {

String fileName = "File1.txt";

int lines = 0, words = 0, chars = 0;

try (BufferedReader br = new BufferedReader(new FileReader(fileName))) {

String line;

while ((line = br.readLine()) != null) {

lines++;

chars += line.length();

words += line.split("\\s+").length;

}

} catch (IOException e) {

e.printStackTrace();

}

System.*out*.println("Number of lines: " + lines);

System.*out*.println("Number of words: " + words);

System.*out*.println("Number of characters: " + chars);

}

}

2.

class Customer {

private int accountNo;

private String accName;

private double balance;

public Customer(int accountNo, String accName, double balance) {

this.accountNo = accountNo;

this.accName = accName;

this.balance = balance;

}

public synchronized void deposit(double amount) {

balance += amount;

System.*out*.println("Deposited " + amount + " into account " + accountNo);

notify();

}

public synchronized void withdraw(double amount) {

if (amount > balance) {

try {

System.*out*.println("Requested amount " + amount + " is not available in account " + accountNo + ". Waiting for deposit.");

wait();

} catch (InterruptedException e) {

e.printStackTrace();

}

}

balance -= amount;

System.*out*.println("Withdrawn " + amount + " from account " + accountNo);

}

}

public class customerclass {

public static void main(String[] args) {

Customer customer = new Customer(123456789, "John Doe", 1000.0);

double withdrawAmount = 1500.0;

Thread t1 = new Thread(() -> {

customer.withdraw(withdrawAmount);

});

Thread t2 = new Thread(() -> {

customer.deposit(500.0);

});

t1.start();

t2.start();

}

}

3.

import java.util.\*;

public class fizzbuzz {

public static String[] fizzBuzz(int n) {

String[] answer = new String[n];

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

if (i % 3 == 0 && i % 5 == 0) {

answer[i - 1] = "FizzBuzz";

} else if (i % 3 == 0) {

answer[i - 1] = "Fizz";

} else if (i % 5 == 0) {

answer[i - 1] = "Buzz";

} else {

answer[i - 1] = String.*valueOf*(i);

}

}

return answer;

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

System.*out*.print("Enter a positive integer: ");

int n = sc.nextInt();

String[] result = *fizzBuzz*(n);

for (String s : result) {

System.*out*.println(s);

}

}

}

4.

import java.util.Scanner;

public class goal

{

public static boolean canShift(String s, String goal) {

if (s.length() != goal.length()) {

return false;

}

String shifted = s + s;

return shifted.contains(goal);

}

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

System.*out*.print("Enter the first string: ");

String s = sc.nextLine();

System.*out*.print("Enter the second string: ");

String goal = sc.nextLine();

boolean result = *canShift*(s, goal);

System.*out*.println(result);

}

}

5.

class PrimeExample implements Runnable {

public void run() {

int m = 20;

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

boolean isPrime = true;

for (int j = 2; j <= Math.*sqrt*(i); j++) {

if (i % j == 0) {

isPrime = false;

break;

}

}

if (isPrime) {

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

} else {

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

}

}

}

}

class Prime {

public static void main(String args[]) {

try {

PrimeExample p1 = new PrimeExample();

Thread t1 = new Thread(p1);

t1.start();

} catch (Exception e) {

System.*out*.println(e.getMessage());

}

}

}

6.

import java.util.Scanner;

public class VoteEligibility {

public static void main(String[] args) {

Scanner input = new Scanner(System.*in*);

System.*out*.print("Enter your age: ");

int age = input.nextInt();

if (age < 18) {

int yearsLeft = 18 - age;

System.*out*.println("Sorry, you are not eligible to vote yet.");

System.*out*.println("You can vote after " + yearsLeft + " years.");

} else {

System.*out*.println("Congratulations, you are eligible to vote!");

}

}

}

7.

import java.util.Scanner;

public class LcmGcd {

public static void main(String[] args) {

Scanner input = new Scanner(System.*in*);

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

int n = input.nextInt();

int[] numbers = new int[n];

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

System.*out*.print("Enter integer " + (i+1) + ": ");

numbers[i] = input.nextInt();

}

int lcm = *getLcm*(numbers);

int gcd = *getGcd*(numbers);

System.*out*.println("LCM: " + lcm);

System.*out*.println("GCD: " + gcd);

}

public static int getLcm(int[] numbers) {

int lcm = numbers[0];

for (int i = 1; i < numbers.length; i++) {

lcm = lcm \* numbers[i] / *getGcd*(lcm, numbers[i]);

}

return lcm;

}

public static int getGcd(int[] numbers) {

int gcd = numbers[0];

for (int i = 1; i < numbers.length; i++) {

gcd = *getGcd*(gcd, numbers[i]);

}

return gcd;

}

public static int getGcd(int a, int b) {

if (b == 0) {

return a;

} else {8

return *getGcd*(b, a % b);

}

}

}

8.

import java.util.Scanner;

public class SimpleInterest {

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

System.*out*.print("Enter the amount invested: ");

double principal = sc.nextDouble();

System.*out*.print("Enter the duration of investment (in years): ");

double time = sc.nextDouble();

System.*out*.print("Enter the customer's age: ");

int age = sc.nextInt();

double roi = age >= 60 ? 0.12 : 0.10; // senior citizen gets 12% ROI, others get 10%

double interest = (principal \* roi \* time);

System.*out*.println("The simple interest is: " + interest);

sc.close();

}

}

9.

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, c;

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

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

c = a + b;

System.*out*.print(" " + c);

a = b;

b = c;

}

sc.close();

}

}

10.

import java.util.Scanner;

public class EvenSumFibonacci {

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

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

int n = sc.nextInt();

int a = 0, b = 1, c, sum = 0;

while (b <= n) {

c = a + b;

a = b;

b = c;

if (a % 2 == 0) {

sum += a;

}

}

System.*out*.println("The sum of even Fibonacci numbers up to " + n + " is: " + sum);

sc.close();

}

}

11.

import java.util.Scanner;

public class SkipNumbers {

public static void main(String[] args) {

Scanner sc = new Scanner(System.*in*);

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

int m = sc.nextInt();

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

int n = sc.nextInt();

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

int k = sc.nextInt();

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

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

}

sc.close();

}

}

12.

import java.util.Scanner;

public class CompositeNumbers {

public static void main(String[] args) {

Scanner input = new Scanner(System.*in*);

System.*out*.print("Enter the lower limit: ");

int a = input.nextInt();

System.*out*.print("Enter the upper limit: ");

int b = input.nextInt();

System.*out*.print("Composite numbers between " + a + " and " + b + ": ");

for (int i = a; i <= b; i++) {

if (*isComposite*(i)) {

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

}

}

}

public static boolean isComposite(int n) {

if (n <= 1) {

return false;

}

for (int i = 2; i <= Math.*sqrt*(n); i++) {

if (n % i == 0) {

return true;

}

}

return false;

}

}

13.

import java.util.Scanner;

public class Factorial {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a positive integer: ");

int n = scanner.nextInt();

int factorial = 1;

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

factorial \*= i;

}

System.out.println("Factorial of " + n + " is: " + factorial);

}

}

14.

import java.util.Scanner;

public class LeapYear {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

int year = scanner.nextInt();

boolean isLeapYear = false;

if (year % 4 == 0) {

if (year % 100 == 0) {

if (year % 400 == 0) {

isLeapYear = true;

}

} else {

isLeapYear = true;

}

}

if (isLeapYear) {

System.out.println(year + " is a leap year");

} else {

System.out.println(year + " is not a leap year");

}

}

}

15.

import java.util.Scanner;

public class FactorCount {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

int num = input.nextInt();

int count = 0;

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

if (num % i == 0) {

count++;

}

}

System.out.println("Number of factors of " + num + " is " + count);

}

}

16.

import java.util.Scanner;

public class PerfectNumber {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

int num = input.nextInt();

input.close();

int sum = 0;

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

if (num % i == 0) {

sum += i;

}

}

if (sum == num) {

System.out.println(num + " is a Perfect Number.");

} else {

System.out.println(num + " is not a Perfect Number.");

}

}

}

17.

import java.util.Scanner;

public class SquareAndCube {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

double num = input.nextDouble();

double square = num \* num;

double cube = num \* num \* num;

System.out.println("Square of " + num + " is: " + square);

System.out.println("Cube of " + num + " is: " + cube);

input.close();

}

}

18.

import java.util.Scanner;

public class NthOddNumber {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

int n = input.nextInt();

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

int m = input.nextInt();

int nthOddNumber = 2 \* n + (m - 1) \* 2;

System.out.println("The " + m + "th odd number after " + n + " odd numbers is: " + nthOddNumber);

}

}

19.

import java.util.\*;

public class FrequencyOfElements {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

System.out.print("Enter the size of the array: ");

int size = input.nextInt();

int[] arr = new int[size];

System.out.println("Enter the elements of the array:");

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

arr[i] = input.nextInt();

}

Map<Integer, Integer> frequencyMap = new HashMap<>();

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

if (frequencyMap.containsKey(arr[i])) {

frequencyMap.put(arr[i], frequencyMap.get(arr[i]) + 1);

} else {

frequencyMap.put(arr[i], 1);

}

}

System.out.println("Frequency of each element:");

for (Map.Entry<Integer, Integer> entry : frequencyMap.entrySet()) {

System.out.println(entry.getKey() + " occurs " + entry.getValue() + " times.");

}

}

}

20.

import java.util.Scanner;

public class ArmstrongNumber {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

int num = input.nextInt();

int temp = num;

int sum = 0;

int count = 0;

while(temp != 0) {

count++;

temp /= 10;

}

temp = num;

while(temp != 0) {

int digit = temp % 10;

sum += Math.pow(digit, count);

temp /= 10;

}

if(num == sum)

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

else

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

input.close();

}

}

21.

import java.util.Scanner;

public class SumOfDigits {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.print("Enter an N digit number: ");

int n = sc.nextInt();

int sum = 0;

while(n > 0) {

int digit = n % 10;

sum += digit;

n /= 10;

}

t

while(sum > 9) {

int tempSum = 0;

while(sum > 0) {

int digit = sum % 10;

tempSum += digit;

sum /= 10;

}

sum = tempSum;

}

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

}

}

22.

import java.util.Scanner;

public class SquareRoot {

public static void main(String[] args) {

Scanner input = new Scanner(System.in);

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

int n = input.nextInt();

double sqrt = Math.sqrt(n);

System.out.println("Positive square root: " + sqrt);

System.out.println("Negative square root: " + (-1 \* sqrt));

}

}