REVERSE OF STRING 2

import java.util.\*;

public class main{

public static void main(String[] args){

Scanner sc=new Scanner(System.in);

String st=sc.next();

char[] ch=st.toCharArray();

String rev="";

for(int i=ch.length-1;i>=0;i--){

rev+=ch[i];

}

System.out.println(rev);

}

}

DELETE THE BLANK SAPCE 1

import java.util.Scanner;

public class RemoveSpaces {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.nextLine();

String output = input.replace(" ", "");

System.out.println(output);

scanner.close();

}

}

DELETE THE VOWELS 1

import java.util.Scanner;

public class DeleteVowels {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.nextLine();

String output = input.replaceAll("[aeiouAEIOU]", "");

System.out.println(output);

scanner.close();

}

}

CONCATENATE THE STRING 1

import java.util.Scanner;

public class ConcatenateStrings {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String first = scanner.nextLine().trim();

String second = scanner.nextLine().trim();

System.out.println(first + second);

scanner.close();

}

}

COUNT THE VOWELS 1

import java.util.Scanner;

public class CountVowels {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.nextLine();

input = input.toLowerCase();

int count = 0;

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

char ch = input.charAt(i);

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

count++;

}

}

System.out.println(count);

scanner.close();

}

}

COUNT VOWELS, CONSONANTS, DIGITS,

SPECIAL CHARACTERS

import java.util.Scanner;

public class CharacterTypeCounter {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.nextLine();

scanner.close();

int vowels = 0, consonants = 0, digits = 0, specialChars = 0;

for (char c : input.toCharArray()) {

if (Character.isLetter(c)) {

char lower = Character.toLowerCase(c);

if (lower == 'a' || lower == 'e' || lower == 'i' || lower == 'o' || lower == 'u') {

vowels++;

} else {

consonants++;

}

} else if (Character.isDigit(c)) {

digits++;

} else {

specialChars++;

}

}

System.out.println("vowels:" + vowels);

System.out.println("consonants:" + consonants);

System.out.println("digits:" + digits);

System.out.println("special characters:" + specialChars);

}

}

CHECK IF STRING CONTAINS ONLY DIGITS 2

import java.util.Scanner;

public class DigitChecker {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String input = scanner.nextLine();

scanner.close();

if (input.matches("\\d+")) {

System.out.println("only digits");

} else {

System.out.println("no");

}

}

}

STRING ANAGRAM 6

import java.util.Arrays;

import java.util.Scanner;

public class AnagramChecker {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String str1 = scanner.nextLine();

String str2 = scanner.nextLine();

scanner.close();

if (str1.length() != str2.length()) {

System.out.println("The given strings are not an anagram");

return;

}

char[] arr1 = str1.toCharArray();

char[] arr2 = str2.toCharArray();

Arrays.sort(arr1);

Arrays.sort(arr2);

if (Arrays.equals(arr1, arr2)) {

System.out.println("The given strings are an anagram");

} else {

System.out.println("The given strings are not an anagram");

}

}

}

ALTERNATING CODE 3

import java.util.Scanner;

import java.util.HashSet;

public class CSKWinChecker {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

String code = scanner.nextLine();

scanner.close();

HashSet<Character> uniqueChars = new HashSet<>();

for (char c : code.toCharArray()) {

uniqueChars.add(c);

}

if (uniqueChars.size() != 2) {

System.out.println("No");

return;

}

char[] chars = code.toCharArray();

char first = chars[0];

char second = chars[1];

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

if (i % 2 == 0 && chars[i] != first) {

System.out.println("No");

return;

} else if (i % 2 == 1 && chars[i] != second) {

System.out.println("No");

return;

}

}

System.out.println("Yes");

}

}

RECURSION – NATURAL NUMBERS 1

import java.util.Scanner;

public class NaturalNumberPrinter {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int num = scanner.nextInt();

scanner.close();

if (num >= 1 && num <= 50) {

System.out.println("The first " + num + " Natural Numbers are:");

printNumbers(1, num);

} else {

System.out.println("Enter a Valid Input!!!!!!!!!!!!!!!!!!!!!!!!");

}

}

public static void printNumbers(int current, int target) {

if (current > target) {

return;

}

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

printNumbers(current + 1, target);

}

}

RECURSION – SUM OF NUMBERS 1

import java.util.Scanner;

public class SumCalculator {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int n = scanner.nextInt();

scanner.close();

if (n >= 1 && n <= 50) {

int sum = recursiveSum(n);

System.out.printf("The sum of numbers from 1 to %d : %04d\n", n, sum);

} else {

System.out.println("Enter a Valid Input!!!!!!!!!!!!!!!!!!");

}

}

public static int recursiveSum(int n) {

if (n == 1) {

return 1;

}

return n + recursiveSum(n - 1);

}

}

RECURSION – SUM OF DIGITS 2

import java.util.Scanner;

public class DigitSum {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int number = scanner.nextInt();

scanner.close();

if (number >= 0 && number < 1000) {

int sum = sumOfDigits(number);

System.out.printf("Sum of Digit:%04d\n", sum);

} else {

System.out.println("Enter a Valid Input!!!!!!!");

}

}

public static int sumOfDigits(int n) {

if (n == 0) {

return 0;

}

return (n % 10) + sumOfDigits(n / 10);

}

}

RECURSION COUNT THE DIGITS 1

import java.util.Scanner;

public class DigitCounter {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int number = scanner.nextInt();

scanner.close();

int count = countDigits(Math.abs(number)); // Handle negative numbers if needed

System.out.println("The Count the digits: " + count);

}

public static int countDigits(int n) {

if (n < 10) {

return 1;

}

return 1 + countDigits(n / 10);

}

}

ARMSTRONG NUMBER OR NOT 14

import java.util.Scanner;

public class ArmstrongChecker {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

int number = scanner.nextInt();

scanner.close();

if (number <= 0 || number >= 100000) {

System.out.println("Invalid Input");

} else if (isArmstrong(number)) {

System.out.println("yes");

} else {

System.out.println("no");

}

}

public static boolean isArmstrong(int num) {

int original = num;

int sum = 0;

int digits = countDigits(num);

while (num > 0) {

int digit = num % 10;

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

num /= 10;

}

return sum == original;

}

public static int countDigits(int num) {

if (num == 0) return 1;

int count = 0;

while (num > 0) {

count++;

num /= 10;

}

return count;

}

}