1. Write a program to reverse a word using loop? (Not to use inbuilt functions)

Sample Input:

String: TEMPLE

Sample Output:

Reverse String: ELPMET

Sol:

public class ReverseWord {

public static void main(String[] args) {

String input = "TEMPLE";

String reversed = "";

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

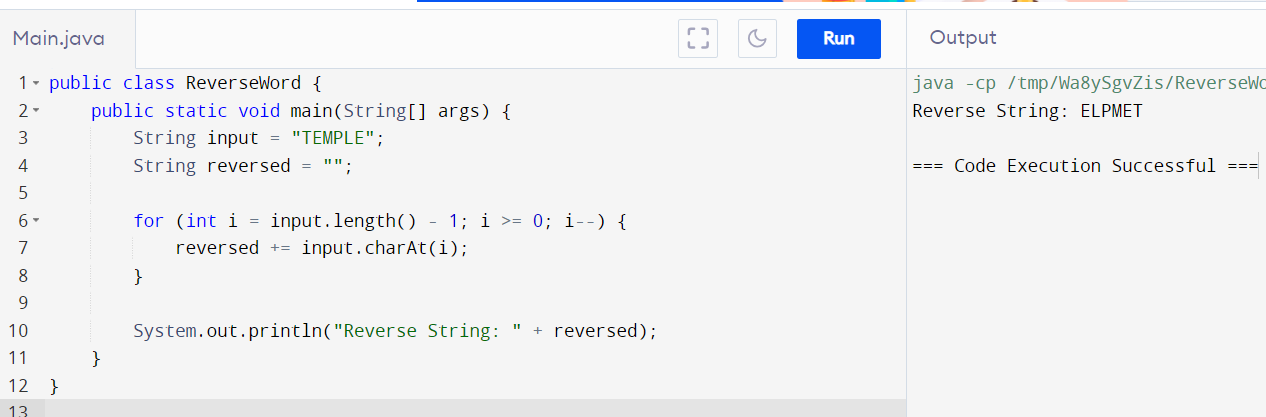
reversed += input.charAt(i);

}

System.out.println("Reverse String: " + reversed);

}

}



1. Write a program to convent the given string to integer?

Sample Input:

String: 1234

Sample Output:

Out put String: 1234

Sol:

public class StringToInteger {

public static void main(String[] args) {

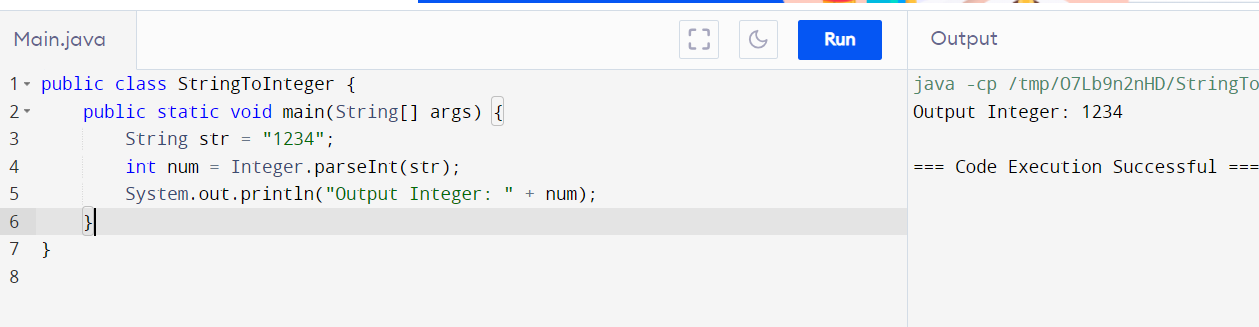
String str = "1234";

int num = Integer.parseInt(str);

System.out.println("Output Integer: " + num);

}

}



1. Write a program to check the entered user name is valid or not. Get both the inputs from the user.

Sol:

import java.util.regex.\*;

class GFG {

public static boolean isValidUsername(String name)

{

String regex = "^[A-Za-z]\\w{5,29}$";

Pattern p = Pattern.compile(regex);

if (name == null) {

return false;

}

Matcher m = p.matcher(name);

return m.matches();

}

public static void main(String[] args)

{

String str1 = "Geeksforgeeks";

System.out.println(isValidUsername(str1));

String str3 = "1Geeksforgeeks";

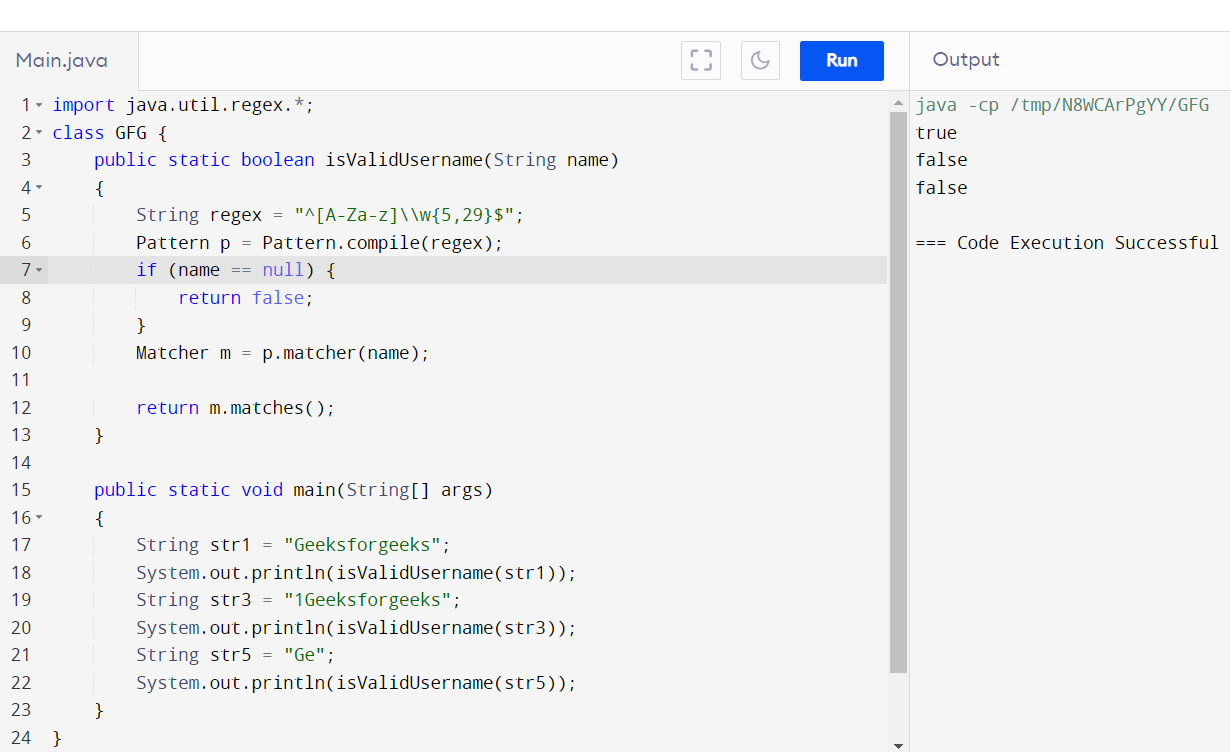
System.out.println(isValidUsername(str3));

String str5 = "Ge";

System.out.println(isValidUsername(str5));

}

}



1. Write a program that would sort a list of names in alphabetical order Ascending or Descending, choice get from the user?

Sample Input:

Banana

Carrot

Radish

Apple

Jack

Order(A/D) : A

Sample Output:

Apple

Banana

Carrot

Jack

Radish

Sol:

import java.io.\*;

class GFG {

public static void main(String[] args)

{

int n = 4;

String names[]

= { "Rahul", "Ajay", "Gourav", "Riya" };

String temp;

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

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

if (names[i].compareTo(names[j]) > 0) {

// swapping

temp = names[i];

names[i] = names[j];

names[j] = temp;

}

}

}

System.out.println(

"The names in alphabetical order are: ");

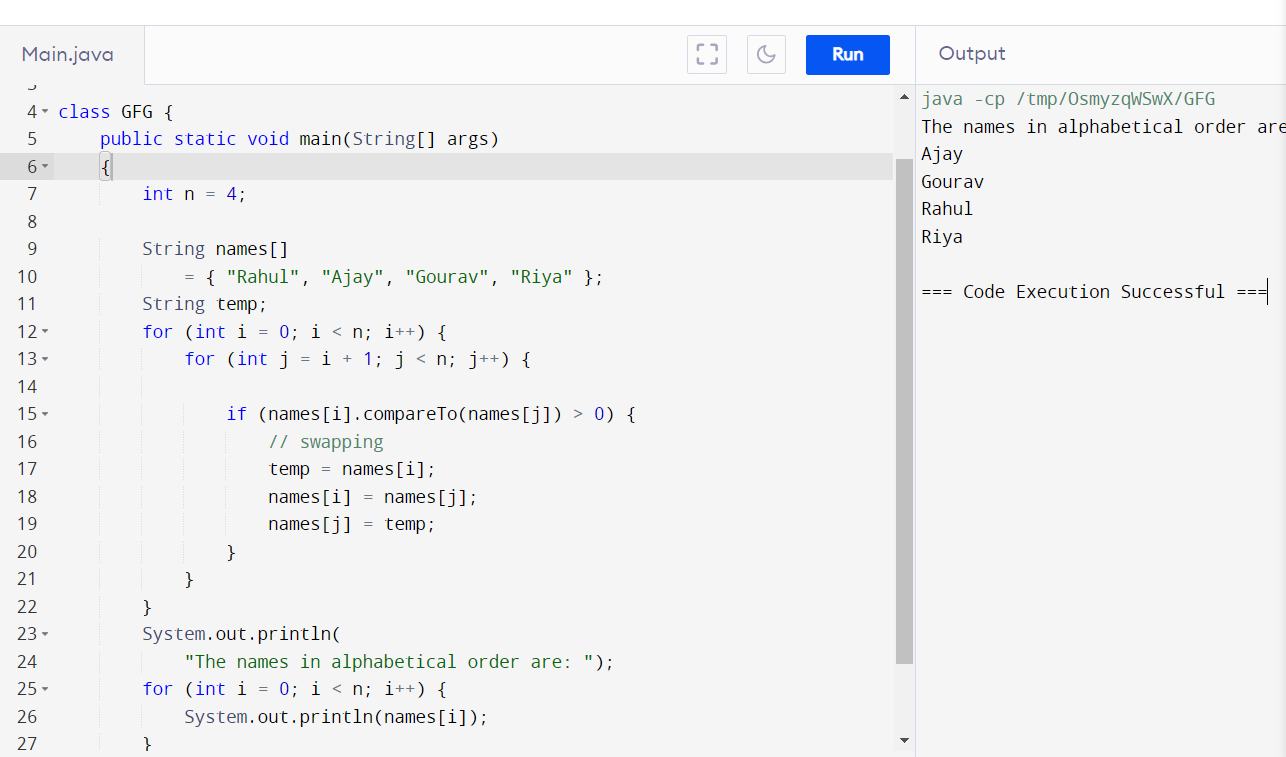
for (int i = 0; i < n; i++) {

System.out.println(names[i]);

}

}

}



1. Write a program to print the special characters separately and print number of Special characters in the line?

Sol:

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class SpecialCharactersCounter {

public static void main(String[] args) {

String line = "Hello! How are you? 123 #$%^&\*";

Pattern pattern = Pattern.compile("[^a-zA-Z0-9\\s]");

Matcher matcher = pattern.matcher(line);

int count = 0;

System.out.println("Special Characters in the Line:");

while (matcher.find()) {

System.out.println(matcher.group());

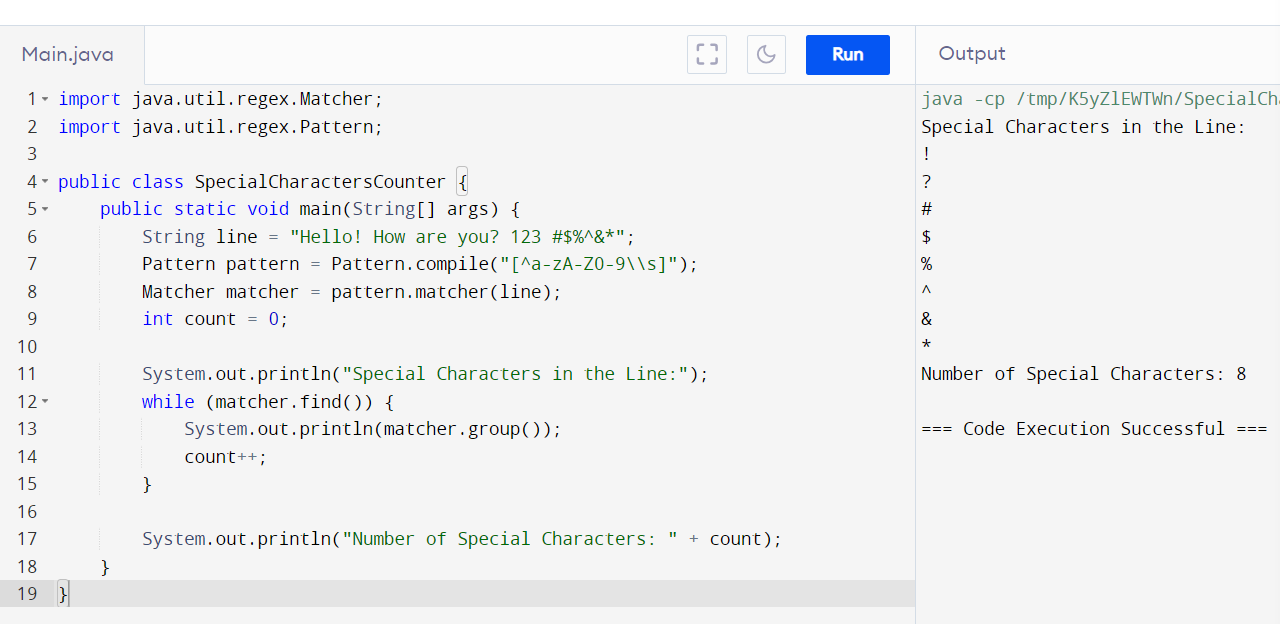
count++;

}

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

}

}



1. Write a program to print the number of vowels in the given statement?

Sample Input:

Saveetha School of Engineering

Sample Output:

Number o vowels = 12

Sol: import java.io.\*;

public class vowel {

public static void main(String[] args)

throws IOException

{

String str = "GeeksForGeeks";

str = str.toLowerCase();

int count = 0;

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

// check if char[i] is vowel

if (str.charAt(i) == 'a' || str.charAt(i) == 'e'

|| str.charAt(i) == 'i'

|| str.charAt(i) == 'o'

|| str.charAt(i) == 'u') {

// count increments if there is vowel in

// char[i]

count++;

}

}

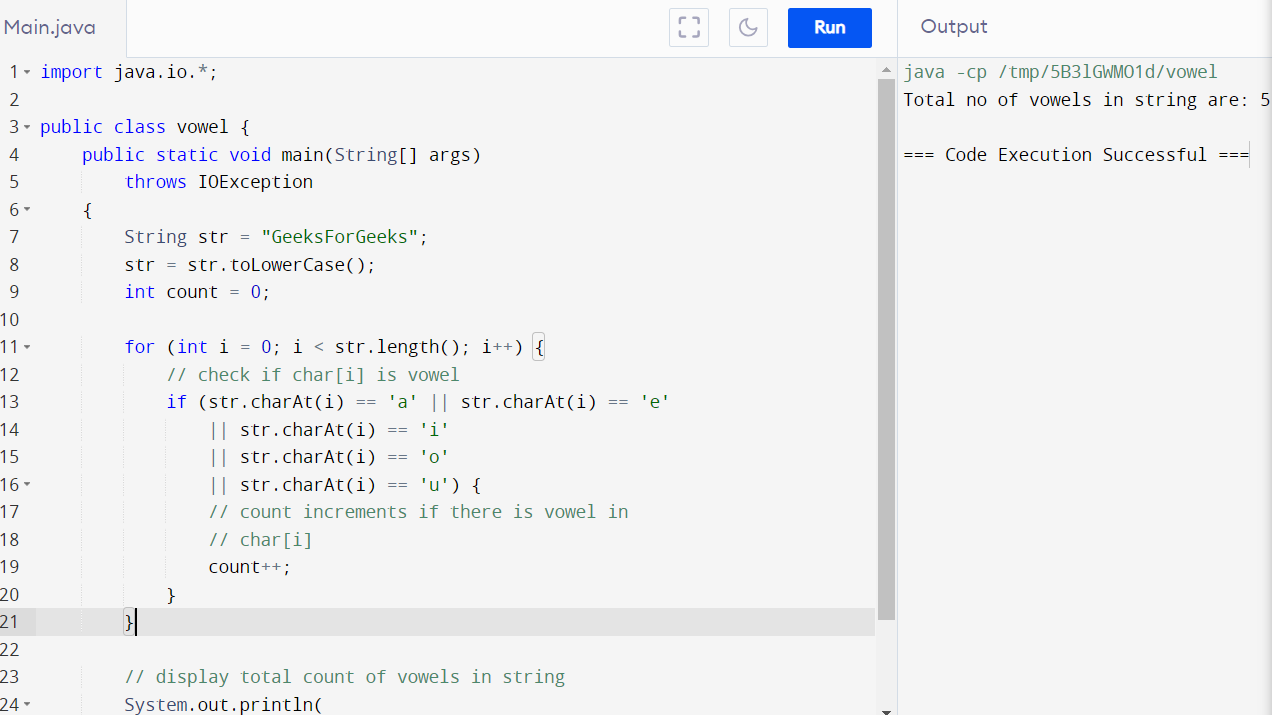
// display total count of vowels in string

System.out.println(

"Total no of vowels in string are: " + count);

}

}



1. Write a program to print consonants and vowels separately in the given word

Sample Input:

Given Word: Engineering

Sample Output:

Consonants: n g n r n g

Vowels: e i e ei

Sol:

public class SeparateConsonantsAndVowels {

public static void main(String[] args) {

String word = "Engineering";

separateConsonantsAndVowels(word);

}

public static void separateConsonantsAndVowels(String word) {

String vowels = "aeiouAEIOU";

String consonants = "bcdfghjklmnpqrstvwxyzBCDFGHJKLMNPQRSTVWXYZ";

System.out.print("Consonants: ");

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

char c = word.charAt(i);

if (consonants.indexOf(c) != -1) {

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

}

}

System.out.print("\nVowels: ");

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

char c = word.charAt(i);

if (vowels.indexOf(c) != -1) {

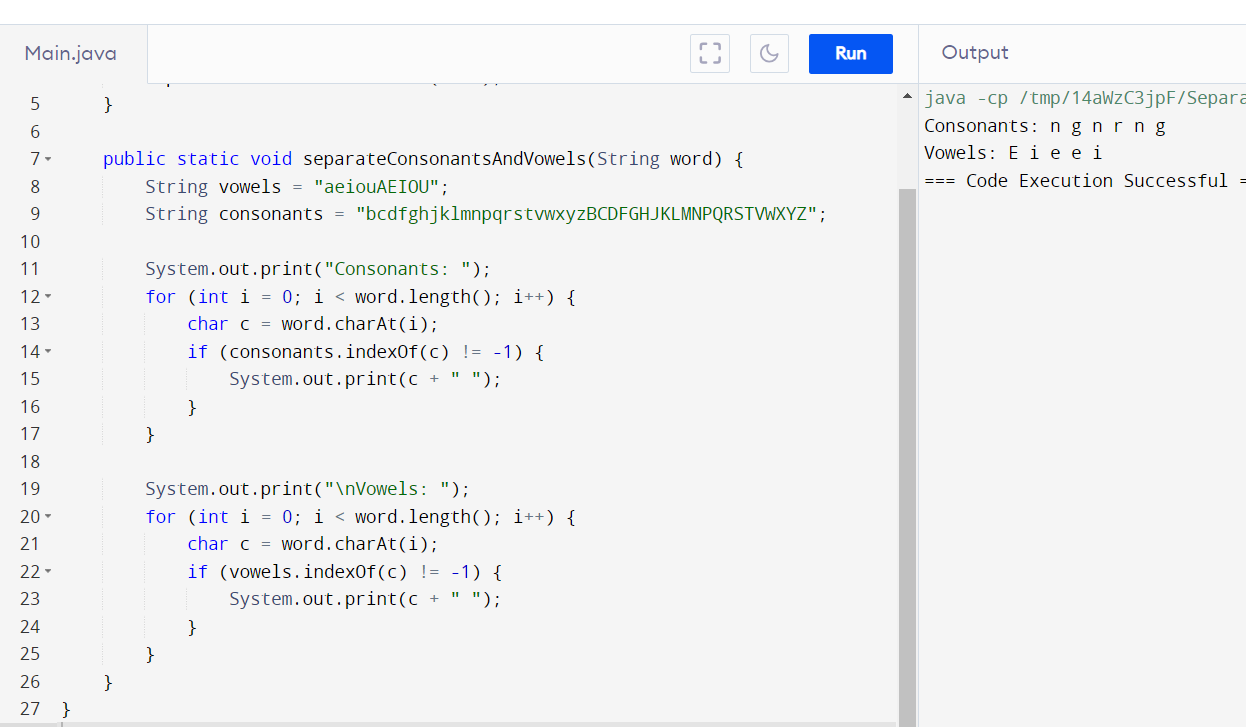
System.out.print(c + " ");

}

}

}

}



1. Write a program that finds whether a given character is present in a string or not. In case it is present it prints the index at which it is present. Do not use built-in find functions to search the character.

Sample Input:

Enter the string: I am a programmer

Enter the character to be searched: p

Sample Output:

P is found in string at index: 8

Note: Check for non available Character in the given statement as Hidden Test case.

Sol:

import java.util.Scanner;

public class CharacterFinder {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

String inputString = scanner.nextLine();

System.out.print("Enter the character to be searched: ");

char searchChar = scanner.next().charAt(0);

int index = findCharacter(inputString, searchChar);

if (index != -1) {

System.out.println(searchChar + " is found in string at index: " + index);

} else {

System.out.println(searchChar + " is not found in the string.");

}

}

public static int findCharacter(String str, char ch) {

char[] charArray = str.toCharArray();

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

if (charArray[i] == ch) {

return i;

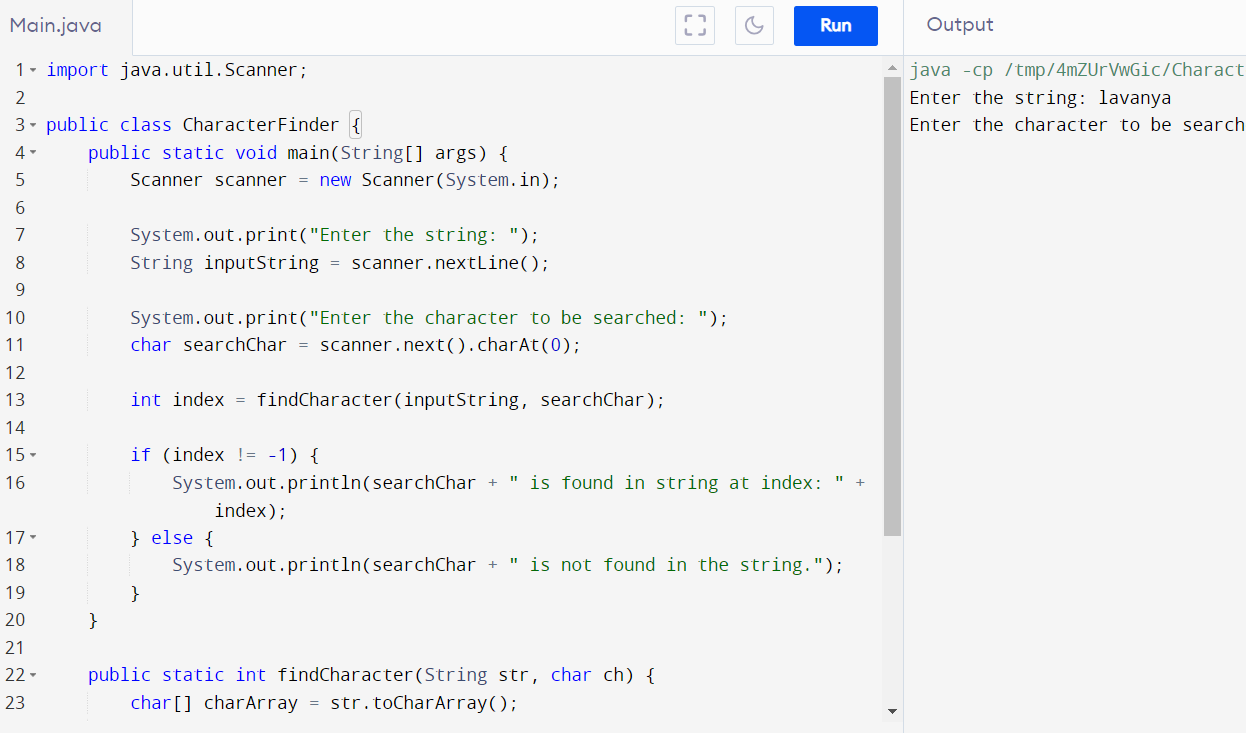
}

}

return -1;

}

}



1. Write a program to arrange the letters of the word alphabetically in reverse order

Sample Input:

Enter the word: MOSQUE

Sample Output:

Alphabetical Order: U S Q O M E

Test Case:

1. HYPOTHECATION
2. MATRICULATION
3. MANIPULATION

Sol:

import java.io.\*;

class GFG {

public static void main(String[] args)

{

int n = 4;

String names[]

= { "Rahul", "Ajay", "Gourav", "Riya" };

String temp;

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

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

if (names[i].compareTo(names[j]) > 0) {

temp = names[i];

names[i] = names[j];

names[j] = temp;

}

}

}

System.out.println(

"The names in alphabetical order are: ");

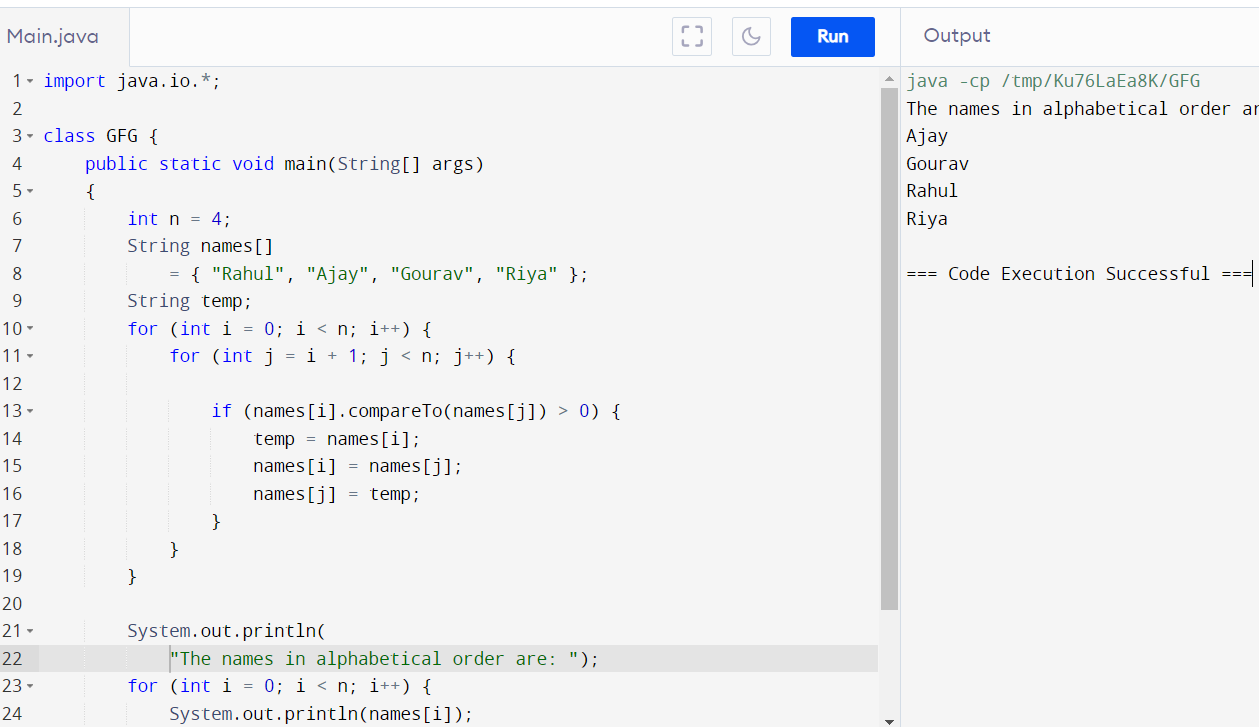
for (int i = 0; i < n; i++) {

System.out.println(names[i]);

}

}

}



1. Write a program that accepts a string from user and displays the same string after removing vowels from it.

Sample Input & Output:

Enter a string: we can play the game

The string without vowels is: w cn ply thgm

SOL:

import java.util.Scanner;

public class RemoveVowels {

public static void main(String[] args) {

Scanner scanner = new Scanner(System.in);

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

String input = scanner.nextLine();

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

System.out.println("The string without vowels is: " + result);

}

}