Convert Character Array to String in Java

KaashyapMSK

Strings are defined as an array of characters. The difference between a character array and a string is the string is terminated with a special character “\0”. A character array can be converted to a string and vice versa. In the previous article, we have already discussed how to convert a string to a character array. In this article, we will discuss how to convert a character array to a string.

Illustrations:

Input 1 : *char s[] = { ‘g’, ‘e’, ‘e’, ‘k’, ‘s’, ‘f’, ‘o’, ‘r’, ‘g’, ‘e’, ‘e’, ‘k’, ‘s’ }*

Output 1 : *“geeksforgeeks”*

Input 2 : *char s[] = { ‘c’, ‘o’, ‘d’, ‘i’, ‘n’, ‘g’ }*

Output 2 : *“coding”*

Methods:

3. Using valueOf() method of String class
4. Using copyValueOf() method of String class
5. Using Collectors in Streams

Now let us discuss each of the methods in detail alongside implementing them with help of a clean java program.

Method 1: Using copyOf() method of Array class

The given character can be passed into the String constructor. By default, the character array contents are copied using the Arrays.copyOf() method present in the Arrays class.

Example:

Java

```
// Java Program to Convert Character Array to String
// Using copyOf() method ofArrays() Class

// Importing required classes
import java.util.*;

// Main class
class GFG {

    // Method 1
    // To convert a character
    // array to a string using the constructor
    public static String toString(char[] a)
    {
        // Creating object of String class
        String string = new String(a);

        return string;
    }

    // Method 2
    // Main driver method
    public static void main(String args[])
    {

        // Declaring and initializing a character array
        char s[] = { 'g', 'e', 'e', 'k', 's', 'f', 'o',
                     'r', 'g', 'e', 'e', 'k', 's' };

        // Printing converted string from character array
        System.out.println(toString(s));
    }
}
```

Output:

```
geeksforgeeks
```

Method 2: Using StringBuilder class

Another way to convert a character array to a string is to use the StringBuilder class. Since a StringBuilder is a mutable class, therefore, the idea is to iterate through the character array and append each character at the end of the string. Finally, the string contains the string form of the characters.

Example:

Java

```java
// Java Program to Convert Character Array to String
// Using StringBuilder Class

// Importing required classes
import java.util.*;

// Main class
public class GFG {

    // Method 1
    // To convert a character array to a string
    // using the StringBuilder class
    public static String toString(char[] a)
    {
        // Creating object of String class
        StringBuilder sb = new StringBuilder();

        // Creating a string using append() method
        for (int i = 0; i < a.length; i++) {
            sb.append(a[i]);
```

```
        return sb.toString();
    }

    // Method 2
    // Main driver method
    public static void main(String args[])
    {

        // Declaring and initialzaing input character array
        char s[] = { 'g', 'e', 'e', 'k', 's', 'f', 'o',
                     'r', 'g', 'e', 'e', 'k', 's' };

        // Printing the string
        // corresponding to character array
        System.out.println(toString(s));
    }
}
```

Output

```
geeksforgeeks
```

Method 3: Using valueOf() method of String class

Another way to convert a character array to a string is to use the **valueOf()** method present in the String class. This method inherently converts the character array to a format where the entire value of the characters present in the array is displayed. This method generally converts int, float, double, char, boolean, and even object to a string. Here we will achieve the goal by converting our character array to string.

Example:

Java

```
// Java Program to Convert Character Array to String
// Using valueOf() method of String Class

// Importing required classes
import java.util.*;

// Main class
class GFG {

    // Method 1
    // To convert a character array to string
    // using the valueOf() method
    public static String toString(char[] a)
    {
        // Creating an object of String class
        String string = String.valueOf(a);
```

```
    }

    // Method 2
    // Main driver method
    public static void main(String args[])
    {

        // Declaring and initializing input character array
        char s[] = { 'g', 'e', 'e', 'k', 's', 'f', 'o',
                     'r', 'g', 'e', 'e', 'k', 's' };

        // Print the corresponding string to
        // character array
        System.out.println(toString(s));
    }
}
```

Output

```
geeksforgeeks
```

Method 4: Using copyValueOf() method of String class

The contents from the character array are copied and subsequently modified without affecting the string to be returned, hence this method also enables us to convert the character array to a string which can be perceived even better from the example provided below as follows.

Example:

Java

```
// Java Program to Convert Character Array to String
// Using copyValueOf() method of String Class

// Importing String class
import java.util.*;

// Main class
class GFG {

    // Main driver method
    public static void main(String[] args)
    {
        // Declaring and initializing input character array
        char[] arr = { 'g', 'e', 'e', 'k', 's', 'f', 'o',
                       'r', 'g', 'e', 'e', 'k', 's' };

        // Storing it in a string
        // using copyValueOf() over string
```

```
        // Printing the converted string corresponding
        // character array
        System.out.print(str);
    }
}
```

Output

```
geeksforgeeks
```

Method 5: Using Collectors in Streams

With the introduction of streams in java8, we straight away use Collectors in streams to modify our character input array elements and later uses joining() method and return a single string and print it.

Example:

Java

```
// Java Program to Convert a Character array to String
// Using Collectors in Streams in Java8

// Importing Collectos and Stream classes
// from java.util.stream package
import java.util.stream.Collectors;
import java.util.stream.Stream;

// Main class
class GFG {

    // Main driver method
    public static void main(String[] args)
    {

        // Custom input character array
        char[] charr = { 'g', 'e', 'e', 'k', 's', 'f', 'o',
                         'r', 'g', 'e', 'e', 'k', 's' };

        // Using collectors to collect array elements and
        // later using joining method to return a single
        // string
        String str = Stream.of(charr)
                         .map(arr -> new String(arr))
                         .collect(Collectors.joining());

        // Printing the stream received from Collectors
        System.out.println(str);
    }
}
```

Output

```
geeksforgeeks
```

Last Updated : 01 May, 2022

13

Similar Reads

1. Check if frequency of character in one string is a factor or multiple of frequency of same character in other string
2. Count of index pairs (i, j) such that string after deleting ith character is equal to string after deleting jth character
3. Convert a String to Character Array in Java
4. Convert the string into palindrome string by changing only one character
5. Convert character array to string in C++
6. Find a string such that every character is lexicographically greater than its immediate next character
7. Last remaining character after repeated removal of the first character and flipping of characters of a Binary String
8. Replace every character of a string by a different character
9. Map every character of one string to another such that all occurrences are mapped to the same character
10. Modify the string such that every character gets replaced with the next character in the keyboard

Related Tutorials

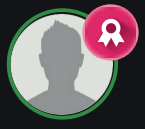
1. Mathematical and Geometric Algorithms - Data Structure and Algorithm Tutorials
2. Spring MVC Tutorial
3. Spring Tutorial
4. Spring Boot Tutorial

5. Java 8 Features - Complete Tutorial

Previous
Learn Java on Your Own in 20 Days - Free!

Next
How to Convert HashMap to ArrayList in Java?

Article Contributed By :



KaashyapMSK
KaashyapMSK
+ Follow

Vote for difficulty

Current difficulty : Expert

Easy | Normal | Medium | Hard | Expert

Improved By : solankimayank

Article Tags : Java-Arrays, Java-Strings, Arrays, DSA, Java, Strings

Practice Tags : Arrays, Java, Java-Strings, Strings

Improve Article | Report Issue

In Media
Contact Us
Advertise with us

Offline Classes (Delhi/NCR)
DSA in JAVA/C++
Master System Design
Master CP

Languages

Python
Java
C++
PHP
GoLang
SQL
R Language
Android Tutorial

Data Structures

Array
String
Linked List
Stack
Queue
Tree
Graph

Algorithms

Sorting
Searching
Greedy
Dynamic Programming
Pattern Searching
Recursion
Backtracking

Web Development

HTML
CSS
JavaScript
Bootstrap
ReactJS
AngularJS
NodeJS

Computer Science

GATE CS Notes
Operating Systems
Computer Network
Database Management System
Software Engineering
Digital Logic Design
Engineering Maths

Python

Python Programming Examples
Django Tutorial
Python Projects
Python Tkinter
OpenCV Python Tutorial
Python Interview Question

Data Science & ML

Data Science With Python
Data Science For Beginner
Machine Learning Tutorial
Maths For Machine Learning
Pandas Tutorial

DevOps

Git
AWS
Docker
Kubernetes
Azure

NLP Tutorial

Deep Learning Tutorial

Competitive Programming

Top DSA for CP

Top 50 Tree Problems

Top 50 Graph Problems

Top 50 Array Problems

Top 50 String Problems

Top 50 DP Problems

Top 15 Websites for CP

System Design

What is System Design

Monolithic and Distributed SD

Scalability in SD

Databases in SD

High Level Design or HLD

Low Level Design or LLD

Top SD Interview Questions

Interview Corner

Company Wise Preparation

Preparation for SDE

Experienced Interviews

Internship Interviews

Competitive Programming

Aptitude Preparation

GfG School

CBSE Notes for Class 8

CBSE Notes for Class 9

CBSE Notes for Class 10

CBSE Notes for Class 11

CBSE Notes for Class 12

English Grammar

Commerce

Accountancy

Business Studies

Economics

Management

Income Tax

Finance

UPSC

Polity Notes

Geography Notes

History Notes

Science and Technology Notes

Economics Notes

Important Topics in Ethics

UPSC Previous Year Papers

SSC/ BANKING

SSC CGL Syllabus

SBI PO Syllabus

SBI Clerk Syllabus

IBPS PO Syllabus

IBPS Clerk Syllabus

Aptitude Questions

SSC CGL Practice Papers

Write & Earn

Write an Article

Improve an Article

Pick Topics to Write

Write Interview Experience

Internships

Video Internship