String Built-in Functions

1. charAt():

The charAt() method returns the character at the specified index in a string. The index of the first character is 0, the second character is 1, and so on.

1. codePointAt():

The codePointAt() method returns the Unicode value of the character at the specified index in a string. The index of the first character is 0, the second character is 1, and so on.

1. codePointBefore():

The codePointBefore() method returns the Unicode value of the character before the specified index in a string. The index of the first character is 1, the second character is 2, and so on.

1. codePointCount():

The codePointCount() method returns the number of Unicode values found in a string. Use the startIndex and endIndex parameters to specify where to begin and end the search. The index of the first character is 0, the second character is 1, and so on.

1. concat():

The concat() method appends (concatenate) a string to the end of another string.

1. contains():

The contains() method checks whether a string contains a sequence of characters. Returns **true** if the characters exist and **false** if not.

1. contentEquals():

The contentEquals() method searches a string to find out if it contains the exact same sequence of characters in the specified string or StringBuffer. Returns **true** if the characters exist and **false** if not.

1. copyValueOf():

The copyValueOf() method returns a String that represents the characters of a char array. This method returns a new String array and copies the characters into it.

1. endsWith():

The endsWith() method checks whether a string **ends** with the specified character(s).

1. startWith():

The [startsWith()](https://www.w3schools.com/java/ref_string_startswith.asp) method to check whether a string **starts** with the specified character(s).

1. equals():

The equals() method compares two strings, and returns **true** if the strings are equal, and **false** if not.

1. equalsIngoreCase():

The equalsIgnoreCase() method compares two strings, ignoring lower case and upper case differences. This method returns **true** if the strings are equal, and **false** if not.

1. format():

The Java string format() method is used to format strings, integers, decimal values, and so on, by using different format specifiers.

1. indexOf():

The indexOf() method returns the position of the first occurrence of specified character(s) in a string.

1. lastIndexOf():

The lastIndexOf() method returns the position of the last occurrence of specified character(s) in a string.

1. isEmpty():

The isEmpty() method checks whether a string is empty or not. This method returns **true** if the string is empty ([length()](https://www.w3schools.com/java/ref_string_length.asp) is 0), and **false** if not.

1. length():

The length() method returns the length of a specified string.

1. replace():

The replace() method searches a string for a specified character, and returns a new string where the specified character(s) are replaced.

1. replaceFirst():

replaces the first matching substring in a string with the specified replacement string.

1. replaceAll():

Replaces all occurrences of a String in another String matched by regex.

1. matches():

The method of String class that checks whether a string matches a given regular expression or not.

1. substring():

Substring in Java is a commonly used method of java. lang. String class that is used to create smaller strings from the bigger one.

1. toCharArray():

The toCharArray() method in Java is a built-in function that converts a string to a sequence of characters.

1. trim():

Join method in Java allows one thread to wait until another thread completes its execution.

1. join():

It allows one thread to wait until another thread completes its execution.

1. valueOf():

The java string valueOf() method converts different types of values into string.

1. toString():

A toString() is an in-built method in Java that returns the value given to it in string format.

1. split():

split() method takes a string and splits it into an array of substrings based on a pattern delimiter.

1. toUpperCase():

The method toUpperCase() converts all characters of a String to upper case.

1. toLowerCase():

The toLowerCase() method converts a string to lowercase letters.

Source Code for String Built-in Functions

class StringBuilt

{

public static void main(String args[])

{

//String function using charAt()

String str1 = "Java Programming";

System.out.println("The value of charAt(): " +str1.charAt(2));

System.out.println("\n");

//codePointAt()

String str2 = "Hello";

int result1 = str2.codePointAt(0);

System.out.println("The value of codePointAt(): " +result1);

System.out.println("\n");

//codePointBefore

String str3 = "Hello";

int result2 = str3.codePointBefore(1);

System.out.println("The value of codePointBefore(): " +result2);

System.out.println("\n");

//codePointCount

String str4 = "Hello";

int result3 = str4.codePointCount(0, 5);

System.out.println("The value of codePointCount(): " +result3);

System.out.println("\n");

//concat()

String firstName = "Sujith ";

String lastName = "Ashok";

String name = firstName.concat(lastName);

System.out.println("The value of concat(): " +name);

System.out.println("\n");

//contains()

String str5 = "Hello";

System.out.println("The value of contains(): ");

System.out.println(str5.contains("Hel"));

System.out.println(str5.contentEquals("e"));

System.out.println(str5.contains("Hi"));

System.out.println("\n");

//contentEquals

String str6 = "Hello";

System.out.println("The value of contentEquals(): ");

System.out.println(str6.contentEquals("Hello"));

System.out.println(str6.contentEquals("Hi"));

System.out.println("\n");

//copyValueOf()

char[] str7 = {'H', 'e', 'l', 'l', 'o'};

String str8 = "";

str8 = str8.copyValueOf(str7, 0, 5);

System.out.println("The value of copyValueOf(): ");

System.out.println("Returned String: " + str8);

System.out.println("\n");

//endsWith()

String str9 = "Hello";

System.out.println("The value of endsWith(): ");

System.out.println(str9.endsWith("Hel"));

System.out.println(str9.endsWith("llo"));

System.out.println(str9.endsWith("o"));

System.out.println("\n");

//startWith()

String str10 = "Hello";

System.out.println("The value of startWith(): ");

System.out.println(str10.endsWith("Hel"));

System.out.println(str10.endsWith("llo"));

System.out.println(str10.endsWith("o"));

System.out.println("\n");

//equals()

String str11 = "Hello";

String str12 = "Hello";

String str13 = "Another String";

System.out.println("The value of equals(): ");

System.out.println(str11.equals(str12));

System.out.println(str11.equals(str13));

System.out.println("\n");

//equalsIngoreCase()

String str14 = "Hello";

String str15 = "HELLO";

String str16 = "Another String";

System.out.println("The value of equalsIgnoreCase(): ");

System.out.println(str14.equalsIgnoreCase(str15));

System.out.println(str14.equalsIgnoreCase(str16));

System.out.println("\n");

//format()

String str17 = "Java";

String formatStr = String.format("Language: %s", str17);

System.out.println("The value of format(): ");

System.out.println(formatStr);

System.out.println("\n");

//indexOf()

String str18 = "Hello planet earth, you are a great planet.";

System.out.println("The value of indexOf(): ");

System.out.println(str18.lastIndexOf("earth"));

System.out.println("\n");

//lastIndexOf()

String str19 = "Hello planet earth, you are a great planet.";

System.out.println("The value of lastIndexOf(): ");

System.out.println(str19.lastIndexOf("planet"));

System.out.println("\n");

//isEmpty()

String str20 = "Hello";

String str21 = "";

System.out.println("The value of isEmpty(): ");

System.out.println(str20.isEmpty());

System.out.println(str21.isEmpty());

System.out.println("\n");

//length()

String txt = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";

System.out.println("The value of length(): ");

System.out.println(txt.length());

System.out.println("\n");

//replace()

String str22 = "Hello";

System.out.println("The value of replace(): ");

System.out.println(str22.replace('l', 'p'));

System.out.println("\n");

//replaceFirst

String str23 = "aabbaaac";

String regex = "\\d+";

System.out.println("The value of replaceFirst(): ");

System.out.println(str23.replaceFirst("aa", "zz"));

System.out.println("\n");

//replaceAll()

String str24 = "Java123is456fun";

String regex1 = "\\d+";

System.out.println("The value of replaceAll(): ");

System.out.println(str24.replaceAll(regex, " "));

System.out.println("\n");

//matches()

String regex2 = "^J..a$"; // a regex pattern for four letter string that starts with 'J' and end with 'a'

System.out.println("The value of matches(): ");

System.out.println("Java".matches(regex));

System.out.println("\n");

//subString()

String str25 = "java is fun";

// extract substring from index 0 to 3

System.out.println("The value of subString(): ");

System.out.println(str25.substring(0, 4));

System.out.println("\n");

//toCharArray

String str26 = "Java Programming";

char[] res;

res = str26.toCharArray();

System.out.println("The value of toCharArray(): ");

System.out.println(res);

System.out.println("\n");

//trim()

String str27 = " Hello World! ";

System.out.println("The value of trim(): ");

System.out.println(str27);

System.out.println(str27.trim());

System.out.println("\n");

//join

String str28 = "I";

String str29 = "love";

String str30 = "Java";

// join strings with space between them

String joinedStr = String.join(" ", str28, str29, str30);

System.out.println("The value of join(): ");

System.out.println(joinedStr);

System.out.println("\n");

//valueOf()

double interest = 923.234d;

// convert double to string

System.out.println("The value of valueOf(): ");

System.out.println(String.valueOf(interest));

System.out.println("\n");

//toString

Integer number=10;

// Calling the toString() method as a function of the Integer variable

System.out.println("The value of toString(): ");

System.out.println( number.toString() );

System.out.println("\n");

//split()

String text = "Java is a fun programming language";

// split string from space

System.out.println("The value of split(): ");

String[] res1 = text.split(" ");

System.out.print("res1 = ");

for (String Str : res1) {

System.out.print(Str + ", "); }

System.out.println("\n");

//toUpperCase()

String txt1 = "Hello World";

System.out.println("The value of toUpperCase(): ");

System.out.println(txt1.toUpperCase());

System.out.println("\n");

//toLowerCase()

String txt2 = "Hello World";

System.out.println("The value of toLowerCase(): ");

System.out.println(txt2.toLowerCase());

}

}