

# String.prototype.trimEnd()

**Summary:** in this tutorial, you'll learn how to use the JavaScript String `trimEnd()` method to remove whitespace from the end of a string.

## Introduction to JavaScript String trimEnd() method

To remove the whitespace from the end of a string, you use the `trimEnd()` method:

```
let newString = str.trimEnd();
```

The `trimEnd()` method returns a new string ( `newString` ) from the original string ( `str` ) with the trailing whitespace stripped.

Note that the `trimEnd()` method returns a new string and doesn't change the original string.



To remove whitespace from the beginning of a string, you use the `trimStart()` method.

The following characters are whitespace characters in JavaScript:

- A space character ( `' '` )
- A tab character ( `\t` )
- A carriage return character ( `\r` )
- A new line character. ( `\n` )
- A vertical tab character. ( `\v` )

- A form feed character. ( `\f` )

## JavaScript String trimEnd() method examples

Let's take some examples of using the JavaScript `trimEnd()` method.

### Basic JavaScript String trimEnd() method example

The following example shows how to use the `trimEnd()` to remove the whitespace from the end of a string:

```
const str = '   JavaScript   ';
const result = str.trimEnd();

console.log({ str });
console.log({ result });
```

Output:

```
{ str: '   JavaScript   ' }
{ result: '   JavaScript' }
```

### Removing trailing newlines

The following example uses the `trimEnd()` method to remove the trailing newline characters from a string:

```
const str = 'Hi\n\n'.trimEnd();
console.log({str});
```

Output:

```
{str: 'Hi'}
```

In this example, the `trimEnd()` method removes two newline characters ( `\n\n` ) from the end of the string.

## Chaining with other string methods

Since the `trimEnd()` method returns a string, you can chain it with other [string methods](#).

For example, the following uses the `trimEnd()` method to remove whitespace and then use the `replace()` method to replace specific characters in a string:

```
let message = 'Hello, World! ';  
let greeting = message.trimEnd().replace('World', 'JavaScript');  
console.log(greeting);
```

Output:

```
Hello, JavaScript!
```

In this example, we use the `trimEnd()` to remove the trailing spaces, and `replace()` changes "World" to "JavaScript" .

## Alias

The `trimRight()` method is an alias for the `trimEnd()` method. This means that both methods refer to the same function object. Therefore, the `trimRight()` provides the same functionality as the `trimRight()` method.

However, it's recommended that you use the `trimEnd()` method for the following reasons:

- **Standardization:** The `trimEnd()` method is part of ECMAScript whereas the `trimRight()` is a non-standard method. This means that the `trimEnd()` is more likely supported across different JavaScript environments.
- **Consistency:** The `trimEnd()` method is consistent with other string methods like `padEnd()`, making the code more readable and easier to understand.

## Summary

- The `trimEnd()` method returns a new string from an original string with the ending whitespace characters stripped. The `trimEnd()` method doesn't change the original string.
- The `trimRight()` method is an alias for the `trimEnd()` method.