![JS JavaScript TUTORIAL]

String.prototype.concat()

Summary: in this tutorial, you'll learn how to use the JavaScript `concat()` method to concatenate strings.

Introduction to the JavaScript String concat() method

The `String.prototype.concat()` method accepts a list of strings and returns a new string that contains the combined strings:

```
string.concat(str1, [...strN]);
```

If the arguments are not strings, the `concat()` converts them to strings before carrying the concatenation.

> It's recommended that you use the `+` or `+=` operator for string concatenation to get better performance.

JavaScript String concat() examples

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

1) Concatenating strings

The following example uses the `concat()` method to concatenate strings:

```javascript
let greeting = 'Hi';
let message = greeting.concat(' ', 'John');

console.log(message);
```

Output:

```
Hi John
```

2) Concatenating an array of strings

The following example uses the `concat()` method to concatenate strings in an array:

```
let colors = ['Blue',' ','Green',' ','Teal'];
let result = ''.concat(...colors);

console.log(result);
```

Output:

```
Blue Green Teal
```

Note that the `...` before the `colors` array argument is the spread operator that unpacks elements of an array.

3) Concatenating non-string arguments

This example concatenates numbers into a string:

```
let str = ''.concat(1,2,3);

console.log(str);
```

Output:

```
123
```

In this example, the `concat()` method converts the numbers 1, 2, and 3 to the strings before concatenating.

Summary

- The `concat()` method concatenates a string list and returns a new string that contains the combined strings.

- Use `+` or `+=` operator to concatenate strings for better performance.