



JavaScript Basics #03



JavaScript is a programming language that drives the web: from front-end user interface design to server-side backend programming, you'll find JavaScript at every stage of a web site and web application. In this course, you'll learn the fundamental programming concepts and syntax of the JavaScript programming language.

Working with Strings

Introducing Strings

Strings

Transform and Manipulate Strings

length property

toLowerCase()

toUpperCase()

Capture Input

prompt()

Combine Strings

Challenge 01

Template Literals

Display the Value of a String on a Page

document.querySelector

Variables and Strings Challenge

Challenge 02

Working with Strings

Learn how to create, transform, and manipulate strings, as well as a more intuitive way to work with strings with template literals.

Introducing Strings

Learn the different ways you can create and work with strings.

Strings

The string data type is used to represent textual data. Strings are created using single or double quotes surrounding one or more characters, as shown below:

```
var carName1 = "Volvo XC60"; // Using double quotes
var carName2 = 'Volvo XC60'; // Using single quotes
```

Transform and Manipulate Strings

JavaScript provides additional functionality to use with strings via "properties" and "methods." Learn how to use the '.length' property, as well as the '.toUpperCase()' and '.toLowerCase()' methods.

length property

The length property returns the length of a string (number of characters).

The length of an empty string is 0.

```
let city = "Barcelona";
console.log(city.length); // 9

let favEvenNumber = 7;
let favOddNumber = 11;
console.log(favEvenNumber); // 1
console.log(favOddNumber); // 2
```

toLowerCase()

method converts a string to lowercase letters.

```
let str = "Hello World";
let str2 = str.toLowerCase();

console.log(str); // Hello World
console.log(str2 ); // hello world
```

toUpperCase()

method converts a string to lowercase letters.

```
let str = "Hello World";
let str2 = str.toUpperCase();

console.log(str); // Hello World
console.log(str2 ); // HELLO WORLD
```

Capture Input

Every application that you interact with provides ways for its users to give it information. Learn how to use 'prompt()' to capture dynamic input from your site users.

prompt()

Display a prompt box which ask the user for her/his name, and output a message:

```
const name = prompt("What is your name");
console.log(name);
```

Combine Strings

There are many occasions where you may need to join two or more strings to create a single value. Programmers call the process of joining together two or more strings to create one new string **concatenation**.

For example, you might have a first and last name in two separate variables and want to join them to show a full name. In this example, the variable called full Name would hold the string 'Ivy Stone'.

```
var firstName = 'Ivy ' ;
var lastName = ' Stone' ;
var fullName = firstName + lastName;

console.log(fullName);
// expected output: Ivy Stone
```

Another Example

```
// Create variables for the welcome message
var greeting = 'Howdy ';
var name = 'Molly';
var message= ', please check your order: ' ;

// Concatenate the three variables above to create the welcome message
var welcome = greeting + name + message;
// expected output: "Howdy Molly, please check your order: "
```

Challenge 01

Assign your first name to the variable firstName and your last name to lastName.

Below role, create a new variable named msg that combines the firstName, lastName and role variables to create a string like "Carlos Salgado: developer".

Finally, convert the string stored in role to uppercase letters. The final msg string should look similar to this: "Carlos Salgado: DEVELOPER".

```
let firstName = "Marlon";
let lastName = "Martins";
let role = 'developer';

let msg = firstName + " " + lastName + ": " + role.toUpperCase();
```

Template Literals

JavaScript provides a more intuitive way to work with strings with a feature called "template literals." Template literals not only make your code more usable and readable, but they also enhance how you work with strings.

To create a template literal, instead of single quotes (') or double quotes (") quotes we use the backtick (`) character. This will produce a new string, and we can use it in any way we want.

Examples:

```
let name = `Harry Potter`;

console.log(`Hi my name is ${name}`); // Hi my name is Harry Potter
```

```
let price = 22;
let tax = 1.50;

let total = `The total prices is ${price * tax}`;

console.log(total); // The total prices is 33
```

Display the Value of a String on a Page

You learned how to collect information from users with the `prompt()` method, and you displayed that information in an alert dialog, as well as in the console. In this video, you'll learn a common way to display information on a webpage using JavaScript.

`document.querySelector`

The Document method `querySelector()` returns the first Element within the document that matches the specified selector, or group of selectors. If no matches are found, null is returned.

```
const stringToShout = prompt("What do you want to shout?");
const shout = stringToShout.toUpperCase();
const shoutMessage = `

## The message to shout is: ${shout}!!</h2>`; document.querySelector('main').innerHTML = shoutMessage;


```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>JavaScript Basics</title>
    <link href="css/style.css" rel="stylesheet">
  </head>
  <body>
    <main></main>
    <script src="js/shout.js"></script>
  </body>
</html>
```

Variables and Strings Challenge

Create a program that retrieves user feedback and combines that feedback to create a string that displays on the page.

```
// 1. Declare variables and capture input.
const adjective = prompt('Please type an adjective.');
```

```
const verb = prompt('Please type a verb.');
```

```
const noun = prompt('Please type a noun.');
```



```
// 2. Combine the input with other words to create a story.
const sentence = `

There once was a ${adjective} programmer who wanted to use JavaScript
to ${verb} the ${noun}.

`;
```



```
// 3. Display the story as a <p> inside the <main> element.
document.querySelector('main').innerHTML = sentence;
```

Challenge 02

Convert the set of concatenated strings assigned to the drink variable to a template literal. The final string should be "Blueberry Smoothie: \$4.99".

```
const flavor = "Blueberry";
const type = "Smoothie";
const price = 4.99;
```



```
const drink = `${flavor} ${type}: ${price}`;
```