What Can JavaScript Do?

JavaScript is the programming language of the web.
It can calculate, manipulate and validate data.
It can update and change both HTML and CSS.

The <script> Tag

In HTML, JavaScript code is inserted between <script> and </script> tags.

JavaScript Variables

Variables are containers for **storing data values**.
Variables must
be **identified** with **unique names**.

JavaScript Operators

JavaScript **assignment operators** (=) assign values to variables:

Example

```
let x = 5;
let y = 6;
let sum = x + y;
```

Try it Yourself »

JavaScript Comments

❮ PreviousNext ❯

JavaScript comments can be used to explain JavaScript code, and to make it more readable.

JavaScript comments can also be used to prevent execution, when

testing alternative code.

Example

```
// Change heading:
document.getElementById("myH").innerHTML = "My First Page";

// Change paragraph:
document.getElementById("myP").innerHTML = "My first paragraph.";
```

Try it Yourself »

Variables = Data Containers

JavaScript variables are containers for data.

JavaScript variables can be **declared** in 4 ways:

Modern JavaScript

- Using let
- Using const

Older JavaScript

- Using var (Not Recommended)
- Automatically (Not Recommende

HTML CSS JAVASCRIPT SQL PYTHON JAVA PHP HOW TO W3.CSS C C++ + C# BOOTSTRAP REACT MYSQL JQUERY EXCEL XML DJANGO NUMPY PANDAS NODEJS DSA TYPESCRIPT ANGULAR ANGULARJS GIT POSTGRESQL MONGODB ASP AI R GO KOTLIN SWIFT SASS VUE GEN
AI SCIPY AWS CYBERSECURITY DATA SCIENCE INTRO TO PROGRAMMING INTRO TO HTML & CSS BASH RUST

JavaScript Let

❮ PreviousNext ❯

The let keyword was introduced in ES6 (2015)

Variables declared with let have **Block Scope**

Variables declared with let must be **Declared** before use

Variables declared with let cannot be **Redeclared** in the same scope

SCIENCE INTRO TO PROGRAMMING INTRO TO HTML & CSS BASH RUST

JavaScript Const

❮ PreviousNext ❯

The const keyword was introduced in ES6 (2015)

Variables defined with const cannot be **Redeclared**

Variables defined with const cannot be **Reassigned**

Variables defined with const have **Block Scope**

JavaScript else

❮ PreviousNext ❯

The else Statement

Use the else statement to specify a **block of code** to be executed if a **condition** is false.

```
if (condition) {
  //  block of code to be executed if the condition is true
} else {
  //  block of code to be executed if the condition is false
}
```

The Boolean Data Type

In JavaScript, a Boolean is a primitive data type that can only have one of two values:

true or **false**

The Boolean value of an expression is the basis for all

JavaScript **comparisons** and **conditions**.

Logical Operators

Logical operators are used to **combine boolean expressions**. Logical operators can be used to modify the results of comparisons. Typically, you will use a comparison operator to check a condition, and a logical operator to combine conditions into more complex logic.

JavaScript Loops

Loops are handy, if you want to run the same code over and over again,

each time with a different value.

Often this is the case when working with arrays:

Instead of writing:

```
text += cars[0] + "<br>";
text += cars[1] + "<br>";
text += cars[2] + "<br>";
text += cars[3] + "<br>";
text += cars[4] + "<br>";
text += cars[5] + "<br>";
```

The For Loop

The for statement creates a loop with 3 optional expressions:

```
for (expr1; expr2; expr) {
  // code block to be executed
}
```

JavaScript For Loop

❮ PreviousNext ❯

For Loops can execute a block of code a number of times.

For Loops are fundamental for tasks like performing an action multiple times.

While Loops

While loops execute a block of code as long as a specified condition is true.

JavaScript have two types of while loops:

- The while loop
- The do while loop

The Break Statement

The break statement "jumps out" of loops and switches.
The break statement **terminates** the execution of a **loop** or
a **switch** statement.

What are Functions?

Functions are **fundamental building blocks** in all programming.
Functions are **reusable block of code** designed to perform a particular task.
Functions **are executed** when they are "**called**" or "**invoked**".

JavaScript Arrays

❮ PreviousNext ❯

Example

```
const cars = ["Saab", "Volvo", "BMW"];
```

Try it Yourself »

An Array is an object type designed for storing data collections.

Key characteristics of JavaScript arrays are:

- **Elements**: An array is a list of values, known as elements.
- **Ordered**: Array elements are ordered based on their index.
- **Zero indexed**: The first element is at index 0, the second at index 1, and so on.
- **Dynamic size**: Arrays can grow or

shrink as elements are added or removed.

- **Heterogeneous**: Arrays can store elements of different data types (numbers, strings, objects and other arrays).

Why Use Arrays?

If you have a list of items (a list of car names, for example), storing the names in single variables could look like this:

```
let car1 = "Saab";
let car2 = "Volvo";
let car3 = "BMW";
```

JavaScript Object

Definitions

Methods for Defining JavaScript Objects

- Using an Object Literal
- Using the new Keyword
- Using an Object Constructor
- Using Object.assign()
- Using Object.create()
- Using Object.fromEntries()