



EPUB/PDF



The Modern JavaScript Tutorial

How it's done now. From the basics to advanced topics with simple, but detailed explanations.

Search

Last updated at 1st April 2022

 [Buy EPUB/PDF](#)Share:  16626 ★ [github](#) [Discord Chat](#)

Table of contents

Main course contains 2 parts which cover JavaScript as a programming language and working with a browser. There are also additional series of thematic articles.

PART 1

[The JavaScript language](#)

PART 2

[Browser: Document, Events, Interfaces](#)

PART 3

[Additional articles](#)

The JavaScript language

Here we learn JavaScript, starting from scratch and go on to advanced concepts like OOP.

We concentrate on the language itself here, with the minimum of environment-specific notes.

An introduction

- 1.1 [An Introduction to JavaScript](#)
- 1.2 [Manuals and specifications](#)

- 1.3 Code editors
- 1.4 Developer console

JavaScript Fundamentals

- 2.1 Hello, world!
- 2.2 Code structure
- 2.3 The modern mode, "use strict"
- 2.4 Variables
- 2.5 Data types
- 2.6 Interaction: alert, prompt, confirm
More...

Code quality

- 3.1 Debugging in the browser
- 3.2 Coding Style
- 3.3 Comments
- 3.4 Ninja code
- 3.5 Automated testing with Mocha
- 3.6 Polyfills and transpilers

Objects: the basics

- 4.1 Objects
- 4.2 Object references and copying
- 4.3 Garbage collection
- 4.4 Object methods, "this"
- 4.5 Constructor, operator "new"
- 4.6 Optional chaining '?.'
- 4.7 Symbol type
- 4.8 Object to primitive conversion

Data types

- 5.1 Methods of primitives
- 5.2 Numbers
- 5.3 Strings
- 5.4 Arrays
- 5.5 Array methods
- 5.6 Iterables

[More...](#)

Advanced working with functions

- [6.1 Recursion and stack](#)
- [6.2 Rest parameters and spread syntax](#)
- [6.3 Variable scope, closure](#)
- [6.4 The old "var"](#)
- [6.5 Global object](#)
- [6.6 Function object, NFE](#)

[More...](#)

Object properties configuration

- [7.1 Property flags and descriptors](#)
- [7.2 Property getters and setters](#)

Prototypes, inheritance

- [8.1 Prototypal inheritance](#)
- [8.2 F.prototype](#)
- [8.3 Native prototypes](#)
- [8.4 Prototype methods, objects without __proto__](#)

Classes

- [9.1 Class basic syntax](#)
- [9.2 Class inheritance](#)
- [9.3 Static properties and methods](#)
- [9.4 Private and protected properties and methods](#)
- [9.5 Extending built-in classes](#)
- [9.6 Class checking: "instanceof"](#)
- [9.7 Mixins](#)

Error handling

- [10.1 Error handling, "try...catch"](#)
- [10.2 Custom errors, extending Error](#)

Promises, async/await

- [11.1 Introduction: callbacks](#)

- 11.2 Promise
- 11.3 Promises chaining
- 11.4 Error handling with promises
- 11.5 Promise API
- 11.6 Promisification
- 11.7 Microtasks
- 11.8 Async/await

Generators, advanced iteration

- 12.1 Generators
- 12.2 Async iteration and generators

Modules

- 13.1 Modules, introduction
- 13.2 Export and Import
- 13.3 Dynamic imports

Miscellaneous

- 14.1 Proxy and Reflect
- 14.2 Eval: run a code string
- 14.3 Currying
- 14.4 Reference Type
- 14.5 BigInt

Browser: Document, Events, Interfaces

Learning how to manage the browser page: add elements, manipulate their size and position, dynamically create interfaces and interact with the visitor.

Document

- 1.1 Browser environment, specs
- 1.2 DOM tree
- 1.3 Walking the DOM
- 1.4 Searching: getElement*, querySelector*
- 1.5 Node properties: type, tag and contents

1.6 Attributes and properties

More...

Introduction to Events

2.1 Introduction to browser events

2.2 Bubbling and capturing

2.3 Event delegation

2.4 Browser default actions

2.5 Dispatching custom events

UI Events

3.1 Mouse events

3.2 Moving the mouse: mouseover/out, mouseenter/leave

3.3 Drag'n'Drop with mouse events

3.4 Pointer events

3.5 Keyboard: keydown and keyup

3.6 Scrolling

Forms, controls

4.1 Form properties and methods

4.2 Focusing: focus/blur

4.3 Events: change, input, cut, copy, paste

4.4 Forms: event and method submit

Document and resource loading

5.1 Page: DOMContentLoaded, load, beforeunload, unload

5.2 Scripts: async, defer

5.3 Resource loading: onload and onerror

Miscellaneous

6.1 Mutation observer

6.2 Selection and Range

6.3 Event loop: microtasks and macrotasks

Additional articles

List of extra topics that assume you've covered the first two parts of tutorial. There is no clear hierarchy here, you can read articles in the order you want.

Frames and windows

- 1.1 [Popups and window methods](#)
- 1.2 [Cross-window communication](#)
- 1.3 [The clickjacking attack](#)

Binary data, files

- 2.1 [ArrayBuffer, binary arrays](#)
- 2.2 [TextDecoder and TextEncoder](#)
- 2.3 [Blob](#)
- 2.4 [File and FileReader](#)

Network requests

- 3.1 [Fetch](#)
- 3.2 [FormData](#)
- 3.3 [Fetch: Download progress](#)
- 3.4 [Fetch: Abort](#)
- 3.5 [Fetch: Cross-Origin Requests](#)
- 3.6 [Fetch API](#)
- [More...](#)

Storing data in the browser

- 4.1 [Cookies, document.cookie](#)
- 4.2 [LocalStorage, sessionStorage](#)
- 4.3 [IndexedDB](#)

Animation

- 5.1 [Bezier curve](#)
- 5.2 [CSS-animations](#)
- 5.3 [JavaScript animations](#)

Web components

- 6.1 From the orbital height
- 6.2 Custom elements
- 6.3 Shadow DOM
- 6.4 Template element
- 6.5 Shadow DOM slots, composition
- 6.6 Shadow DOM styling
- 6.7 Shadow DOM and events

Regular expressions

- 7.1 Patterns and flags
 - 7.2 Character classes
 - 7.3 Unicode: flag "u" and class \p{...}
 - 7.4 Anchors: string start ^ and end \$
 - 7.5 Multiline mode of anchors ^ \$, flag "m"
 - 7.6 Word boundary: \b
- More...

Share  

 [Tutorial map](#)

Comments

- If you have suggestions what to improve - please submit a [GitHub issue](#) or a pull request instead of commenting.
- If you can't understand something in the article – please elaborate.
- To insert few words of code, use the `<code>` tag, for several lines – wrap them in `<pre>` tag, for more than 10 lines – use a sandbox ([plnkr](#), [jsbin](#), [codepen](#)...)