

DOM in JavaScript II: Advanced Techniques

Course code: JS_DOM2

Only the javascript APIs built into modern browsers are what turn a web page into a full-fledged web application, whether it's graphics, interactivity, data handling, or reading inputs from the real world. The possibilities are limited only by security and privacy or hardware performance.

Affiliate	Duration	Course price	ITB
Praha	3	15 300 Kč	30
Brno	3	15 300 Kč	30
Bratislava	3	612 €	30

The prices are without VAT.

Course terms

Date	Duration	Course price	Type	Course language	Location
📅 25.05.2026	3	15 300 Kč	Telepresence	CZ/SK	GOPAS Praha
📅 25.05.2026	3	612 €	Telepresence	CZ/SK	GOPAS Bratislava
📅 25.05.2026	3	15 300 Kč	Telepresence	CZ/SK	GOPAS Brno

The prices are without VAT.

What we will teach you

- Working with user accessed files
- Storing data in the browser
- Improved form validation in HTML5
- Webcam and microphone input
- Javascript and CSS animations
- Working with graphics (canvas, video)
- Manipulating browser history for SPA
- Working with data from other domains (CORS)
- Running more complex calculations in separate threads (Worker)
- Service worker (resource caching, offline applications)
- Fetch API basics (modern successor to AJAX)
- Uploading files to server
- Responding to page scrolling, lazy-loading

Required entry-level skills

- Knowledge of JavaScript at the level of the JS_PROG2 course and DOM at the level of the JS_DOM1 course.

Course Outline

Modules

- Module Types
- Lazy loading

Cookies and Web Storage Basics

- Cookies
- Creating and Reading Cookies
- Cookie Options and Limitations
- Security Considerations and Limitations
- Local Storage

GOPAS Praha

Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved

DOM in JavaScript II: Advanced Techniques

- Data Storage
- Data Loading
- Data Deletion
- Limitations and Security Considerations
- Session Storage
- Differences between Local Storage and Session Storage
- Usage Examples

Working with User Accessed Files

- <input type="file">, Native drag&drop in HTML5
- `FileList`, `File`, `FileReader` and `URL API`
- Previewing an image before uploading
- `FormData API` (uploading data to the server)

Session History Management ("History API")

- How to preserve user experience when changing page content
- Back button treatment in AJAX applications
- `History.pushState`, `history.replaceState`
- `popstate` event on `window` object

Storing data in browser

- treatment of page refresh button [F5]
- `WebStorage` (`localStorage`)
- `IndexedDB`

Working with animations

- API for working with CSS3 animations
- `requestAnimationFrame`

Working with graphics

- <canvas> in 2d context - Programmable image

Retrieving input from webcam and microphone

- `getUserMedia` and related APIs

Improved form validation in HTML5

- new form element and attribute types
- Constraint and Validation API

Working with data from other domains (CORS)

- Cross-Origin Resource Sharing
- use for AJAX, canvas and external files

Fetch API basics (modern successor to AJAX)

JavaScript in separate threads (Worker)

Service worker (resource caching, offline applications)

Response to page scrolling, lazy-loading

- `Intersection Observer`