

mozilla

SEARCH AND FIND

How we made MDN discoverable

Jannis Leidel

JANNIS LEIDEL

Open Source software developer

Django core developer

Python enthusiast

PSF & DSF member, director of
German Django association

Mozillian working on **MDN**



WHAT IS MDN?

Mozilla Developer Network

WHAT IS MDN?

Web platform (HTML, CSS, JavaScript, etc)

Firefox **desktop**

Firefox for **Android**

Firefox **OS** (marketplace and apps)

WHAT IS MDN?

5.5 staff writers, **6** staff developers

900 live code demos by **500** developers

9M page views in **4.6M** visits by **2.1M** unique visitors *per month*

33K wiki documents with **375K** edits *in total*

14K community contributors *globally*

WHAT IS KUMA?

Django based wiki (Python)

Open Source software (MPL 2.0)

Public UI for writing/editing

Code on **Github** available

Kumascript scripting language
(server-side JavaScript via Node.js)



MDN HISTORY

DevEdge (AOL) 2000–2005

DevEdge

Home | Archive | ViewSource | Tech Central | Library | Toolbox | Community | Site Map

/web/20021202221020/http://devedge.netscape.com/library/manuals/2000/javascript/1.5/reference/regexp.html/

PreviousContentsIndexNext

Core JavaScript Reference 1.5

RegExp

A regular expression object contains the pattern of a regular expression. It has properties and methods for using that regular expression to find and replace matches in strings.

In addition to the properties of an individual regular expression object that you create using the `RegExp` constructor function, the predefined `RegExp` object has static properties that are set whenever any regular expression is used.

| | |
|----------------|--|
| Core object | |
| Implemented in | JavaScript 1.2, NES 3.0 JavaScript 1.3: added toSource method. JavaScript 1.5, NES 6.0: added <code>m</code> flag, non-greedy modifier, non-capturing parentheses, lookahead assertions. ECMA 262, Edition 3 |

Created by
A literal text format or the `RegExp` constructor function.

The literal format is used as follows:

```
/pattern/flags
```

The constructor function is used as follows:

```
new RegExp("pattern"[, "flags"])
```

Parameters

| | |
|---------|---|
| pattern | The text of the regular expression. |
| flags | If specified, flags can have any combination of the following values: <ul style="list-style-type: none">g: global matchi: ignore casem: match over multiple lines |

MDN HISTORY

MDC (MediaWiki) 2005–2008

mozilla

mozilla developer center beta

DocumentationDevNewsWebwatch

Create an account or log in

Hide Sidebar

Core JavaScript 1.5 Reference:Global Objects:RegExp:test

Main Page > Core JavaScript 1.5 Reference > Global Objects > RegExp > **test**

Contents

[\[hide\]](#)

[1 Summary](#)

[2 Syntax](#)

[3 Parameters](#)

[4 Description](#)

[5 Examples](#)

[5.1 Example: Using test](#)

Summary

[edit]

Executes the search for a match between a regular expression and a specified string. Returns true or false.

Method of **RegExp**

| | |
|-----------------|------------------------|
| Implemented in: | JavaScript 1.2, NES3.0 |
| ECMA Version: | ECMA-262, Edition 3 |

Syntax

[edit]

regex.test([str])

Parameters

[edit]

regex
The name of the regular expression. It can be a variable name or a literal.

str
The string against which to match the regular expression.

Description

[edit]

When you want to know whether a pattern is found in a string use the test method (similar to the [String.search](#) method); for more information (but slower execution) use the [exec](#) method (similar to the [String.match](#) method).

Examples

[edit]

Example: Using test

[edit]

The following example prints a message which depends on the success of the test:

```
function testinput(re, str){
```


MDN HISTORY

MDC (MindTouch Deki)
2008–2011

mozilla<developercenter/>

Site toolsHelpRegister | Log in

[Main Page](#)[Core JavaScript 1.5 Reference](#)[Global Objects](#)[RegExp](#)[test](#)

Print pageHistoryEdit pageMore options

Languages

test

Summary

Executes the search for a match between a regular expression and a specified string. Returns true or false.

Method of **RegExp**

| | |
|-----------------|------------------------|
| Implemented in: | JavaScript 1.2, NES3.0 |
| ECMA Version: | ECMA-262, Edition 3 |

Syntax

regexp.test([*str*])

Parameters

regexp

The name of the regular expression. It can be a variable name or a literal.

str

The string against which to match the regular expression.

Description

When you want to know whether a pattern is found in a string use the test method (similar to the [String.search](#) method); for more information (but slower execution) use the [exec](#) method (similar to the [String.match](#) method).

Examples

Example: Using test

The following example prints a message which depends on the success of the test:

```
function testinput(re, str){
  if (re.test(str))
    midstring = " contains ";
  else
    midstring = " does not contain ";
  document.write (str + midstring + re.source);
}
```

Table of contents

[Summary](#)[Syntax](#)[Parameters](#)[Description](#)[Examples](#)[Example: Using test](#)

Tags

Files

MDN HISTORY

MDN (MindTouch Deki) 2011–2012

MDN

TOPICS

DOCS

DEMOS

LEARNING

COMMUNITY

Join MDN

Log in

Search MDN

MDN

Docs

JavaScript

JavaScript Reference

Global Objects

RegExp

test

Languages

This page

Site tools

test

WATCH

EDIT

Summary

Executes the search for a match between a regular expression and a specified string. Returns true or false.

| Method of RegExp | |
|----------------------------------|------------------------|
| Implemented in | JavaScript 1.2 |
| ECMAScript Edition | ECMAScript 3rd Edition |

Syntax

```
regex.test([str])
```

Parameters

regex

The name of the regular expression. It can be a variable name or a literal.

str

The string against which to match the regular expression.

Description

When you want to know whether a pattern is found in a string use the `test` method (similar to the [String.search](#) method); for more information (but slower execution) use the `exec` method (similar to the [String.match](#) method). As with `exec` (or in combination with it), `test` called multiple times on the same global regular expression instance will advance past the previous match.

Gecko 8.0 note

(Firefox 8.0 / Thunderbird 8.0 / SeaMonkey 2.5)

Prior to Gecko 8.0 (Firefox 8.0 / Thunderbird 8.0 / SeaMonkey 2.5) , `test()` was implemented incorrectly; when it was called with no parameters, it would match against the value of the previous input (`RegExp.input` property) instead of against the string `"undefined"`. This is fixed; now `/undefined/.test()` correctly results in `true`, instead of an error.

Examples

Example: Using `test`

The following example prints a message which depends on the success of the test:

```
function testinput(re, str){
  if (re.test(str))
    midstring = " contains ";
  else
    midstring = " does not contain ";
  document.write (str + midstring + re.source);
}
```

TABLE OF CONTENTS

Summary

Syntax

Parameters

Description

Examples

Example: Using test


TAGS

FILES

Page Notifications **Off**

MDN HISTORY

MDN (Kuma)
2012–2013



MOZILLA DEVELOPER NETWORK

TOPICS DOCS DEMOS LEARNING COMMUNITY

Sign in

mozilla

Search MDN

MDN JavaScript JavaScript Reference Global Objects RegExp test Languages This page

test

HISTORYEDIT

Summary

Executes the search for a match between a regular expression and a specified string. Returns true or false.

| | |
|----------------------------------|------------------------|
| Method of RegExp | |
| Implemented in | JavaScript 1.2 |
| ECMAScript Edition | ECMAScript 3rd Edition |

Syntax

```
regex.test([str])
```

Parameters

regex
The name of the regular expression. It can be a variable name or a literal.

str
The string against which to match the regular expression.

Description

When you want to know whether a pattern is found in a string use the test method (similar to the [String.search](#) method); for more information (but slower execution) use the [exec](#) method (similar to the [String.match](#) method). As with [exec](#) (or in combination with it), test called multiple times on the same global regular expression instance will advance past the previous match.

Gecko 8.0 note
(Firefox 8.0 / Thunderbird 8.0 / SeaMonkey 2.5)

Prior to Gecko 8.0 (Firefox 8.0 / Thunderbird 8.0 / SeaMonkey 2.5), test() was implemented incorrectly; when it was called with no parameters, it would match against the value of the previous input (RegExp.input property) instead of against the string "undefined". This is fixed; now /undefined/.test() correctly results in true, instead of an error.

Examples

Example: Using test

The following example prints a message which depends on the success of the test:

```
function testinput(re, str){
  if (re.test(str)) {
    midstring = " contains ";
  } else {
    midstring = " does not contain ";
  }
}
```

TABLE OF CONTENTS

[Summary](#)
[Syntax](#)
[Parameters](#)
[Description](#)
[Examples](#)
Example: Using test

TAGS

FILES

MDN HISTORY

MDN (Kuma) 2013–present

Welcome to the new, improved MDN design. We hope you like it! If you have any suggestions or comments, [please share!](#)

fscholz | [Sign out](#)

mozilla

MDN

MOZILLA DEVELOPER NETWORK

ZONES

WEB PLATFORM

DEVELOPER PROGRAM

TOOLS

DEMOS

Q

MDN > [Web technology for developers](#) > [JavaScript](#) > [JavaScript reference](#) > [Standard built-in objects](#) > [RegExp](#) > RegExp.prototype.test()

EDIT PAGE

RegExp.prototype.test()

▲ HIDE SIDEBAR

SEE ALSO

Standard built-in objects

RegExp

▼ Properties

RegExp.lastIndex

RegExp.prototype

RegExp.prototype.global

RegExp.prototype.ignoreCase

RegExp.prototype.multiline

RegExp.prototype.source

⚠️ RegExp.prototype.sticky

▼ Methods

RegExp.prototype.exec()

RegExp.prototype.test()

⚠️ RegExp.prototype.toSource()

RegExp.prototype.toString()

Inheritance:

Function

► Properties

► Methods

Object

► Properties

► Methods

Summary

The **test()** method executes a search for a match between a regular expression and a specified string. Returns true or false.

Syntax

```
regexObj.test(str)
```

Parameters

str
The string against which to match the regular expression.

Return value

Boolean. true or false.

Description

You can use `test()` whenever want to know whether a pattern is found in a string (similar to the [String.search](#) method); for more information (but slower execution) use the [exec](#) method (similar to the [String.match](#) method). As with `exec` (or in combination with it), `test` called multiple times on the same global regular expression instance will advance past the previous match.

Examples

Example: Using test

The following example prints a message which depends on the success of the test:

```
1 function testinput(re, str){
2   var midstring;
3   if (re.test(str)) {
4     midstring = " contains ";
5   } else {
6     midstring = " does not contain ";
```

IN THIS ARTICLE

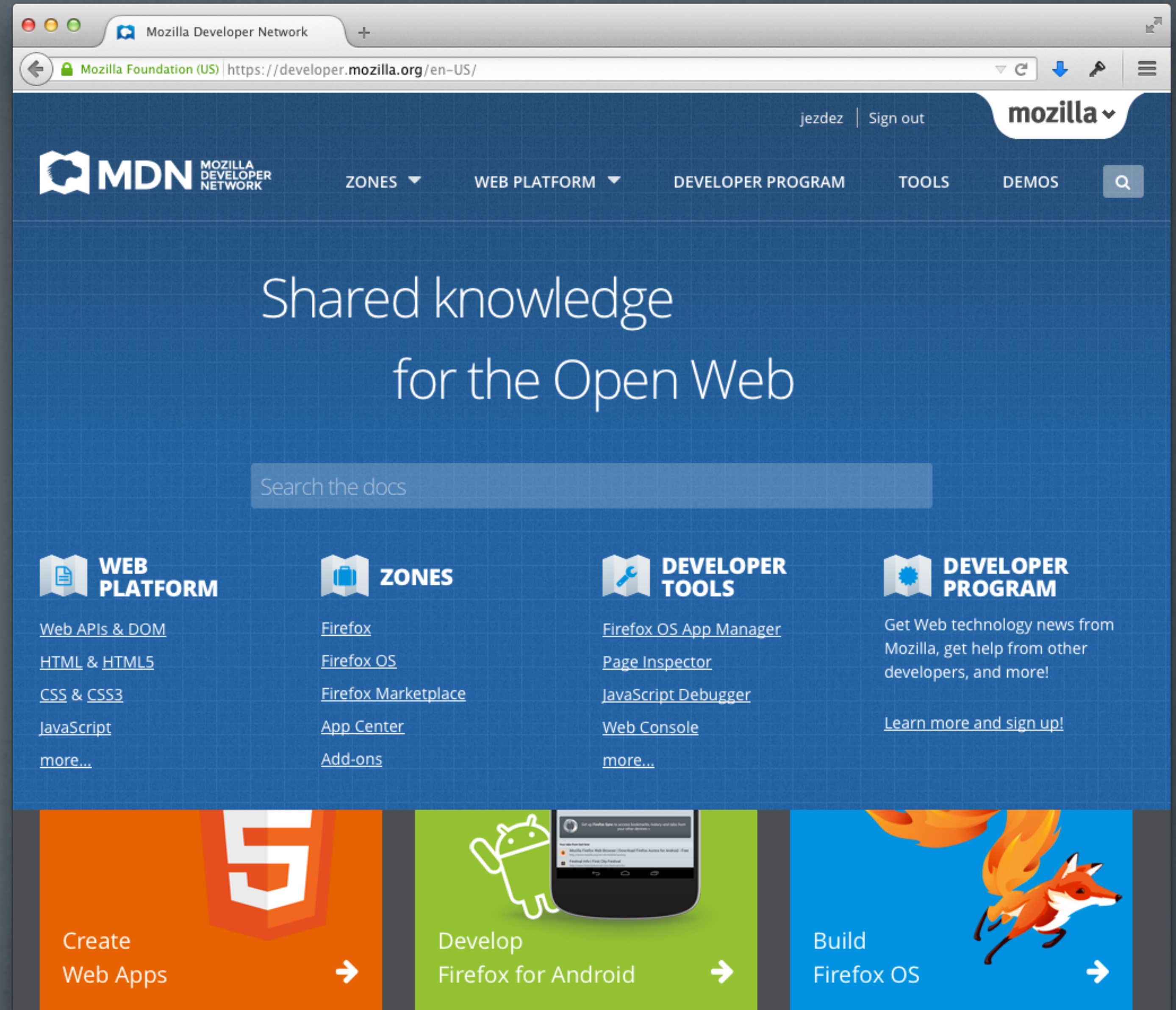
- [Summary](#)
- [Syntax](#)
 - [Parameters](#)
 - [Return value](#)
- [Description](#)
- [Examples](#)
 - [Example: Using test](#)
- [Specifications](#)
- [Browser compatibility](#)
 - [Gecko-specific notes](#)
- [See also](#)

REDESIGN 2013

Responsive design

Content zones

Search front and center

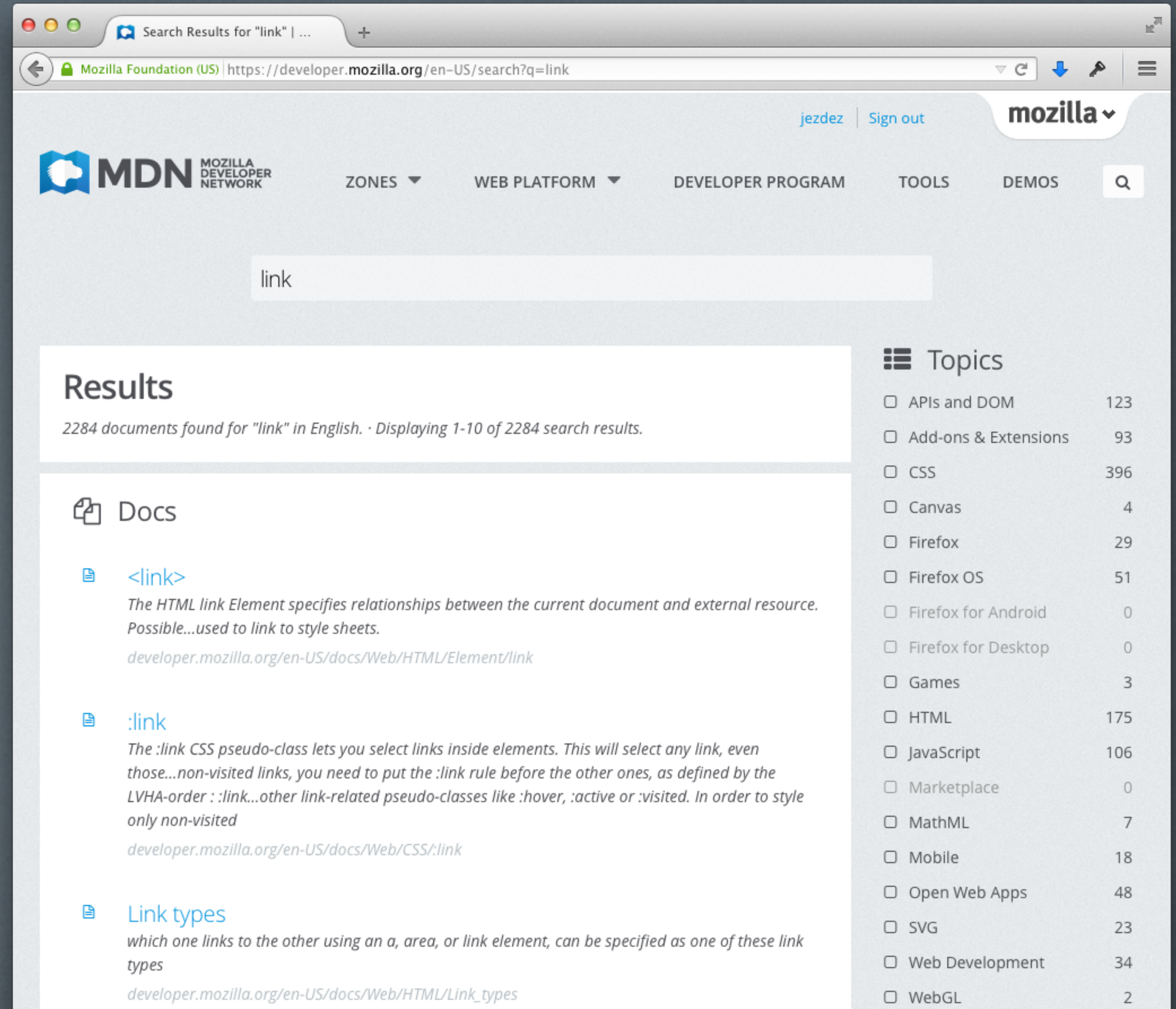


SEARCH

From custom Google search
to own implementation

Full text search, multi-lingual

Faceting, filters, pagination

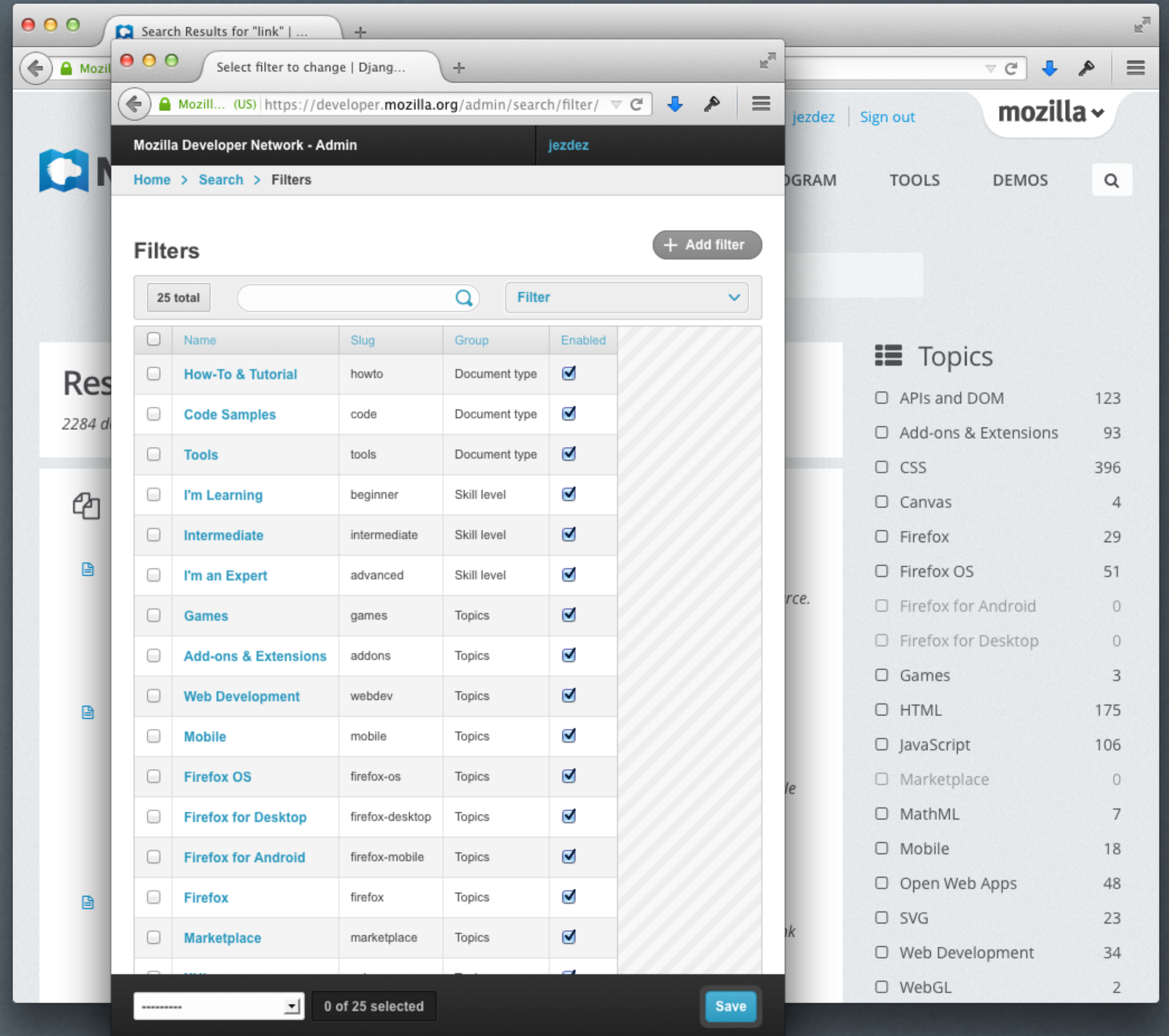


SEARCH FILTERS

Extensible via admin UI

Based on document tags

Topics, skill level and
document type currently



SEARCH API

Every search page is also available as JSON

Includes are metadata to programmatically use the search

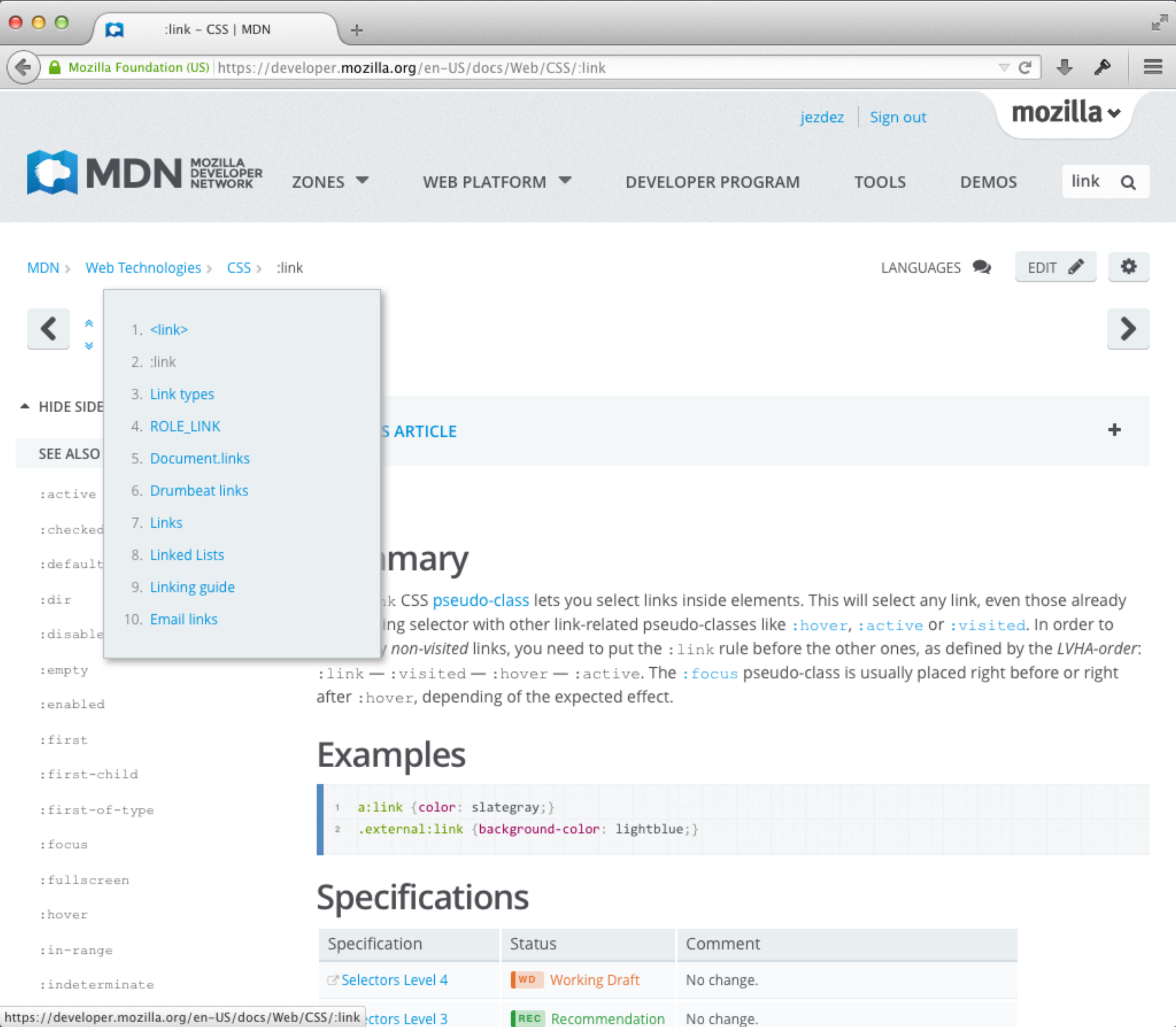
The screenshot shows a web browser displaying the search results for the query "link" on the Mozilla Developer website. The search results are presented as a JSON object. The JSON structure includes metadata such as the total count of results (2284), the current page (1), and the total number of pages (229). It also lists the search query, locale, and filters. The main content is a list of documents, with the first document being the "link" element in HTML. The document metadata includes its ID (2012), title, slug, locale, excerpt, URL, and edit URL. The document tags are listed as "HTML document metadata", "Élément", "Element", "Reference", "Référence", "Web", and "HTML". The score for this document is 10.331484. The second document in the list is the "link" pseudo-class in CSS, with ID 4972, title ":link", slug "Web/CSS/:link", and a score of 10.331484. The background of the screenshot shows the Mozilla Developer website interface, including the search bar, navigation links, and a list of topics.

```
{
  count: 2284,
  next: https://developer.mozilla.org/en-US/search.json?q=link&page=2,
  previous: null,
  query: "link",
  page: 1,
  pages: 229,
  start: 1,
  end: 10,
  locale: "en-US",
  filters: [
    + { ... },
    + { ... },
    + { ... }
  ],
  documents: [
    - {
      id: 2012,
      title: "<link>",
      slug: "Web/HTML/Element/link",
      locale: "en-US",
      excerpt: "The HTML <em>link</em> Element specifies relationships between the current document and external resource. Possible...used to <em>link</em> to style sheets.",
      url: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/link,
      edit_url: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/link$edit,
      tags: [
        "HTML document metadata",
        "Élément",
        "Element",
        "Reference",
        "Référence",
        "Web",
        "HTML"
      ],
      score: 10.331484,
      explanation: { }
    },
    - {
      id: 4972,
      title: ":link",
      slug: "Web/CSS/:link",
      locale: "en-US",
      excerpt: "The :<em>link</em> CSS pseudo-class lets you select <em>links</em> inside elements. This will select any <em>link</em>, even those...non-visited <em>links</em>, you need to put the :<em>link</em> rule before the other ones, as defined by the LVHA-order : :<em>link</em>...other <em>link</em>-related pseudo-classes like :hover, :active or :visited. In order to style only non-visited",
      score: 10.331484,
      explanation: { }
    }
  ]
}
```


SEARCH API

Search result navigator
on document pages

Coming from search page



SEARCH API

Documentation status

Processed through
Kumascript

JavaScript documentation ...

Mozilla Foundation (US) | https://developer.mozilla.org/en-US/docs/MDN/Doc_status/JavaScript

jezdez | Sign out

mozilla

MDN MOZILLA DEVELOPER NETWORK

ZONES

WEB PLATFORM

DEVELOPER PROGRAM

TOOLS

DEMOS

The MDN project

MDN > The MDN project > Documentation status > JavaScript documentation status

LANGUAGES

EDIT

JavaScript documentation status

HIDE SIDEBAR

This page tries to track ongoing documentation work in the MDN JavaScript section. Feel free to contribute!

ABOUT MDN

CONTRIBUTING TO MDN

DOCUMENTATION STATUS

Documentation status

API documentation status overview

Accessibility documentation status

Addons

Apps Documentation Status

Summary

| Pages | No tags | Needs* tags | Editorial reviews | Technical reviews | Outdated pages | Dev-doc-needed bugs | Documentation requests | To be cleaned up |
|-------|---------|-------------|-------------------|-------------------|----------------|---------------------|------------------------|------------------|
| 491 | 0 (0%) | 45 (10%) | 8 (2%) | 27 (6%) | 20 (5%) | 31 (7%) | 8 (2%) | 141 (29%) |

See also [localization status](#) of this section.

Needs* tags

Found 45 pages:

Base64 encoding and decoding

NeedsUpdate

Compatibility

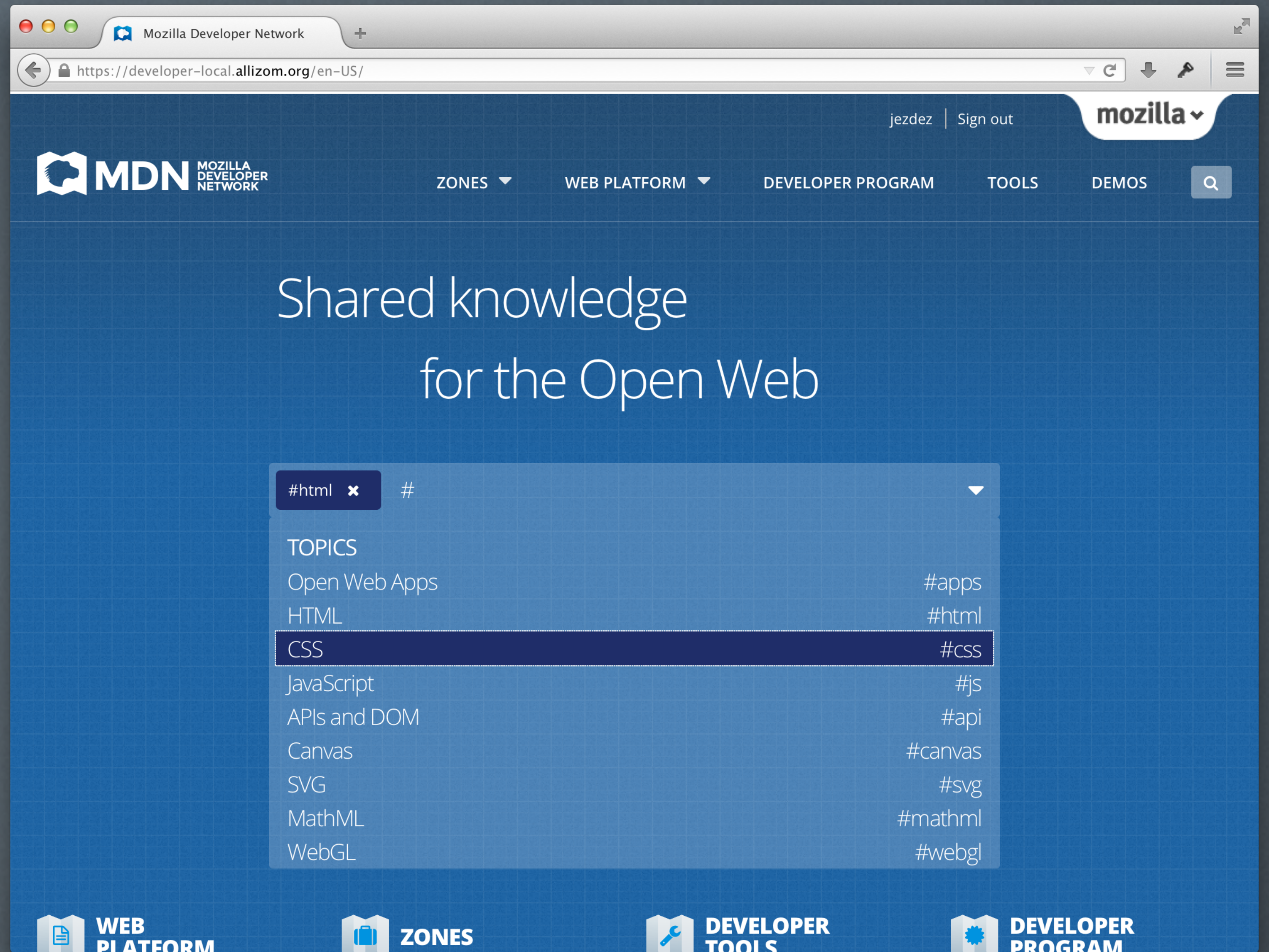
NeedsContent, NeedsUpdate

SEARCH UI

Command and query

Quick access to filters
with shortcuts

Extensible UI for
suggestions

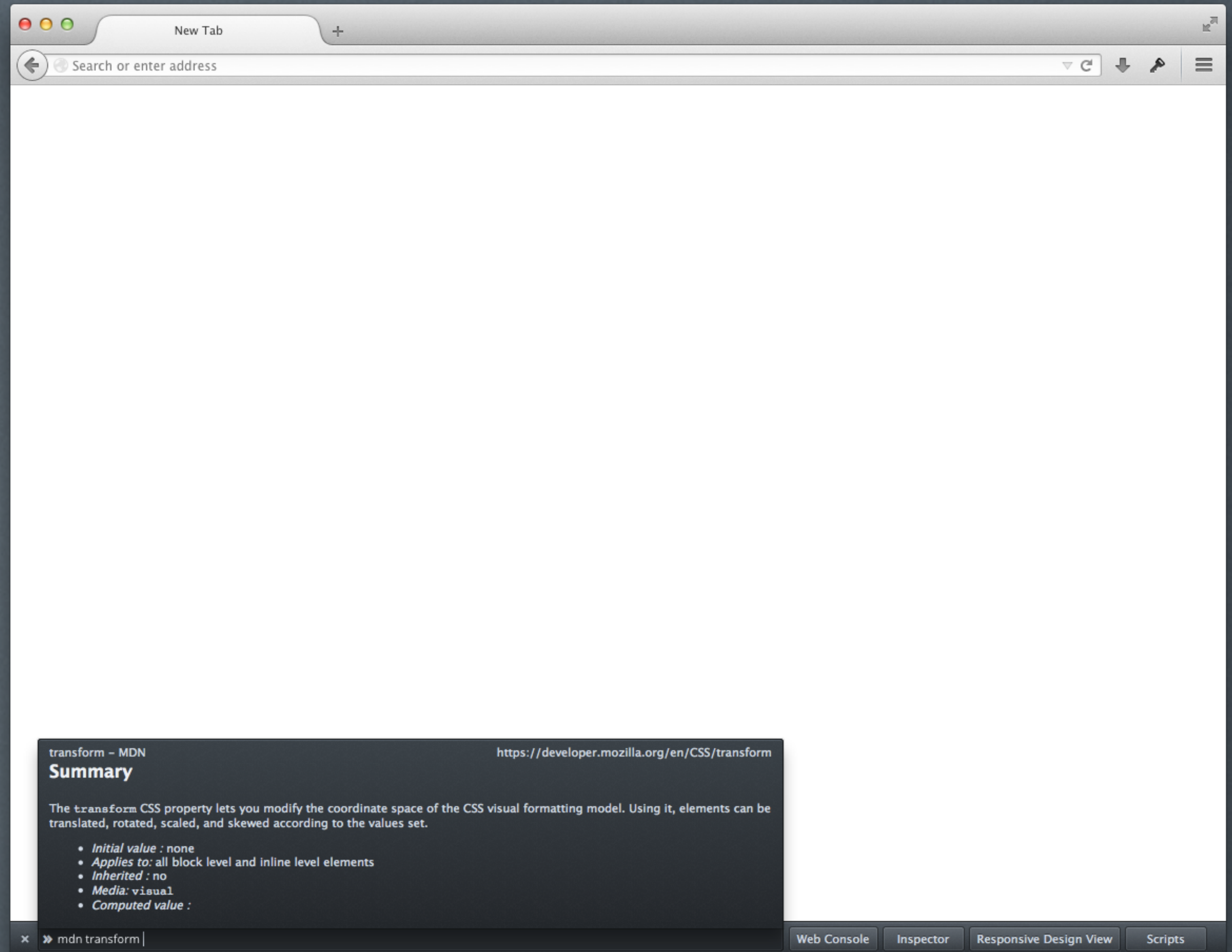


SEARCH CLI

Developer toolbar
integration

Command line

Source code inspector



SEARCH FUTURE

Code editor integration via **plugins**

User profile search with **@user** syntax

docs.json integration

QUESTIONS?

Jannis Leidel
@jezdez

github.com/mozilla/kuma