



Web Engineering & Design 1

HTML

Basics

Tobias Blaser



Content

- Content model
- General document structure
- Content structuring
- Navigation
- Lists
- Browser
- Coding style
- Development tools

Learning goals

You are able to...

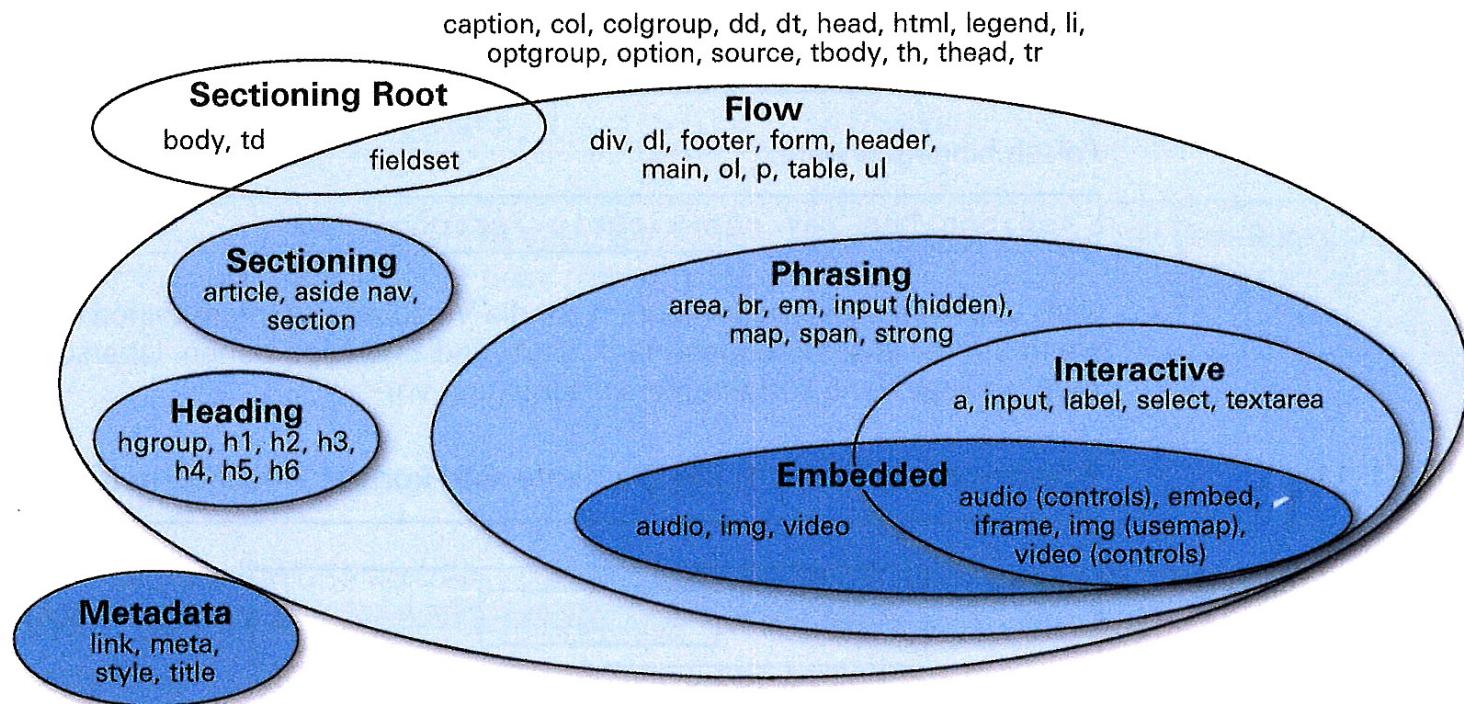
- ...explain the content model and identify nestable tags.
- ...explain the concept of quirks mode and in which case it will be applied.
- ...detect and correct errors in an invalid HTML document.
- ...explain the effect of the document type declaration.

You are able to..

- ...validate or markup an HTML document using head tags (meta, link, title, robots, ...) and body tags (document structuring, text markup, navigations, lists, ...)
- ...use the developer tools of your browser to inspect and manipulate HTML documents.
- ...use the W3C validator to validate documents.
- ...determine in which browsers an html tag/attribute is properly supported.

Content model

Which tags can be nested?



4.6.1 The `a` element

Categories

[Flow content](#).

When the element only contains [phrasing content](#): [phrasing content](#).

[Interactive content](#).

Contexts in which this element can be used:

When the element only contains [phrasing content](#): where [phrasing content](#) is expected.

Otherwise: where [flow content](#) is expected.

Content model:

[Transparent](#), but there must be no [interactive content](#) descendant.

Content attributes:

[Global attributes](#)

[href](#)

[target](#)

[rel](#)

[media](#)

[hreflang](#)

[type](#)

DOM interface:

```
interface HTMLAnchorElement : HTMLElement {  
    stringifier attribute DOMString href;  
    attribute DOMString target;  
    attribute DOMString rel;  
    readonly attribute DOMTokenList relList;  
    attribute DOMString media;  
    attribute DOMString hreflang;  
    attribute DOMString type;  
  
    attribute DOMString text;  
  
    // URL decomposition IDL attributes  
    attribute DOMString protocol;  
    attribute DOMString host;
```

4.4.2 The `section` element

Categories

[Flow content](#).
[Sectioning content](#).
[formatBlock candidate](#).

Contexts in which this element can be used:

Where [flow content](#) is expected.

Content model:

[Flow content](#).

Content attributes:

[Global attributes](#)

DOM interface:

Uses [HTMLElement](#).

The `section` element [represents](#) a generic section of a document or application. A section, in this context, is a thematic grouping of content, typically with a heading.

Examples of sections would be chapters, the various tabbed pages in a tabbed dialog box, or the numbered sections of a thesis. A Web site's home page could be split into sections for an introduction, news items, and contact information.

Note: Authors are encouraged to use the `article` element instead of the `section` element when it would make sense to syndicate the contents of the element.

Note: The `section` element is not a generic container element. When an element is needed for styling purposes or as a convenience for scripting, authors are encouraged to use the `div` element instead. A general rule is that the `section` element is appropriate only if the element's contents would be listed explicitly in the document's outline.

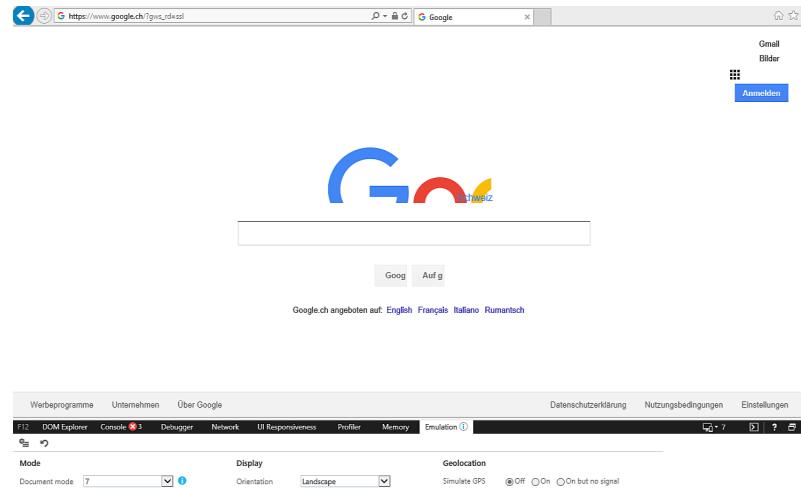
In the following example, we see an article (part of a larger Web page) about apples, containing two

General document structure

Document type (**DOCTYPE**)

```
<!DOCTYPE html>
```

- First line of an HTML document (before `<html>` tag)
- Standard mode ↔ quirks mode (**compatibility mode**)
- Doctype missing → Quirks mode
 - developer.mozilla.org/en-US/docs/Mozilla_Quirks_Mode_Behavior
 - Browser dependent

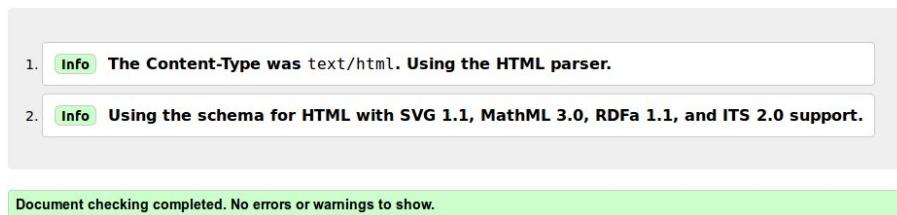


Document validation

```
<!DOCTYPE html>
<head>
    <title>First document</title>
    <meta charset="utf-8" />
<body>
    <h1>A first document</h1>
    <p>This is our first HTML-document.</p>
```

examples/minimal-document.html

- Validator: validator.w3.org/#validate_by_input



- Tag omitting: Parser inserts missing (end) tags
 - Valid but **not recommended**
 - **Correct indentation** recommended!

Document root ([html](#))

```
<!DOCTYPE html>
<html lang="en"><!-- document language: english -->
    <head><!-- head & meta data --></head>
        <body><!--      content      --></body>
</html>
```

Document head ([head](#))

```
<head>
    <!-- Charachter encoding, place in first 1024 bytes -->
    <meta charset="utf-8" />
    <title>Title of the document</title>
    <!-- Tab / bookmark icon -->
    <link rel="icon" href="/icon.png" type="image/png" />
    <!-- Link base url, used for relative links -->
    <base href="http://www.my-page.tld" />
</head>
```

Meta Data (**meta**)

```
<head>
  <meta name="author" content="Max Muster" />
  <meta name="description" content="HTML Presentation">

  <meta name="keywords" lang="en" content="example, html, head" />
  <meta name="keywords" lang="de" content="Beispiel, HTML, Kopf" />

  <meta name="date" content="2010-05-15T08:49:37+02:00">
  <meta name="creator" content="Hanna Hansen" />
  <meta name="publisher" content="HSR" />
  <meta name="application-name" content="Reveal.js">
  <meta name="generator" content="no-generator V1.0">
</head>
```

- Search engines
- Printer / PDF exporter
- Accessibility tools

Device & Robots meta data

```
<!-- Device behaviour, see later -->
<meta name="viewport"
      content="width=device-width, initial-scale=1.0" />
<!-- Search engine behaviour (do not index page) -->
<meta name="robots" content="noindex,nofollow" />
```

Transport protocol (*HTTP*) equivalent

```
<!-- Document forwarding -->
<meta http-equiv="refresh" content="5; URL=http://other.html/" />
```

Content structuring

Skyguide newspaper

Local news

Daily flights to Paris overbooked

The daily flights between ...

Worldwide news

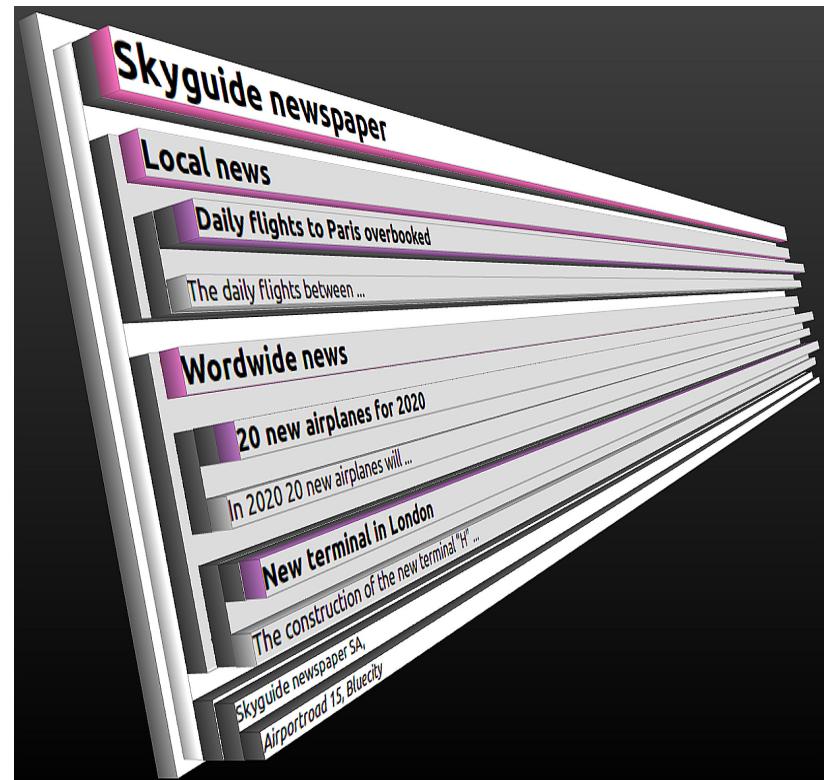
20 new airplanes for 2020

In 2020 20 new airplanes will ...

New terminal in London

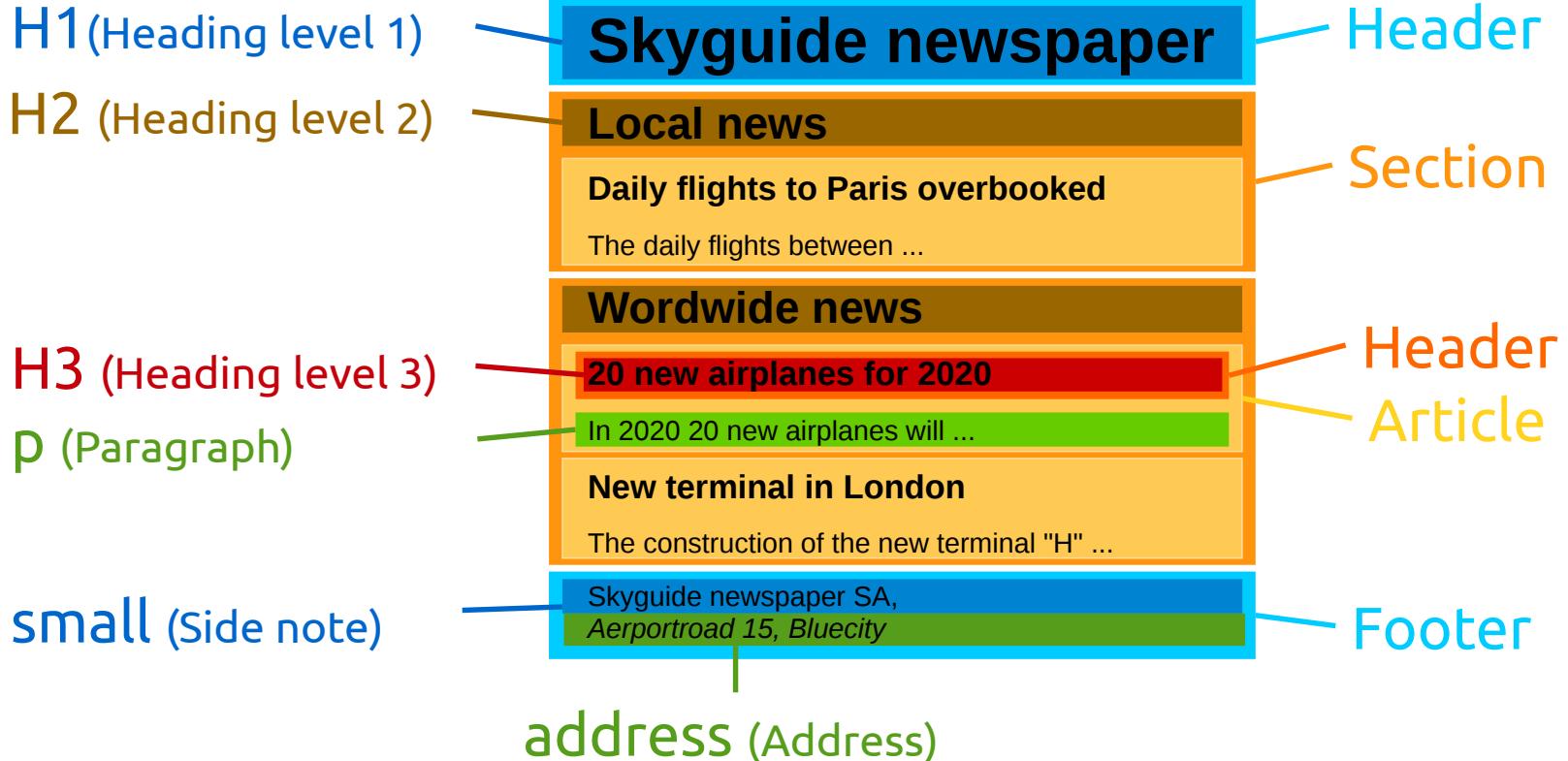
The construction of the new terminal "H" ...

Skyguide newspaper SA
Airportroad 15, Bluecity



examples/sections.html

jsfiddle.net/m2btvnqt/



Skyguide newspaper

Local news

Daily flights to Paris overbooked

The daily flights between ...

```
<body>
  <header>
    <h1>Skyguide newspaper</h1>
  </header>

  <section>
    <header><h2>Local news</h2></header>
    <article>
      <header>
        <h3>Daily flights to Paris overbooked</h3>
      </header>
      <p>The daily flights between ...</p>
    </article>
  </section>
```

jsfiddle.net/m2btvnqt/1/

Wordwide news

20 new airplanes for 2020

In 2020 20 new airplanes will ...

New terminal in London

The construction of the new terminal "H" ...

```
<section>
    <header><h2>Wordwide news</h2></header>
    <article>
        <header>
            <h3>20 new airplanes for 2020</h3>
        </header>
        <p>In 2020 20 new airplanes will ...</p>
    </article>
    <article>
        <header>
            <h3>New terminal in London</h3>
        </header>
        <p>The construction of the terminal "H" ...</p>
    </article>
</section>
```

jsfiddle.net/m2btvnqt/2/

Content areas

section

Semantic section (e.g. chapter, topic)

aside

Side content, (additional information)

article

Complete article, post, ...

→ containing header, footer, text, author, ...

header, footer

Header or footer of document, section, article, ...

main

Main section of the document (**Only use once!**)



h1-h6 (heading level 1 - 6)

Heading of document, section, aside, article, ...

Heading structure variants

HTML4

h1 Skyguide newspaper

h2 Local news

h3 Daily flights ...

h2 Worldwide news

h3 20 new airplanes ...

h3 New terminal in London

HTML5

h1 Skyguide newspaper
section

h1 Local news
article

h1 Daily flights ...
section

h1 Worldwide news
article

h1 20 new airplanes ...
article

h1 New terminal in London

HTML5 compatible to HTML4

h1 Skyguide newspaper
section

h2 Local news
article

h3 Daily flights ...
section

h2 Worldwide news
article

h3 20 new airplanes ...
article

h3 New terminal in Londo

Text

The construction of the new terminal "H" ...

Skyguide newspaper SA
Airportroad 15, Bluecity

```
<article>
    <header><h3>...</h3></header>
    <p>The construction of ...</p>
</article>
</section>

<footer>
    <small>Skyguide newspaper SA,</small>
    <address>Airportroad 15, Bluecity</address>
</footer>
</body>
```

p (Paragraph)

Text paragraph, may contain line breaks

small (Small annotations)

Small prints, legal notice

time, address, abbr, blockquote, cite, pre, code

Time, address (contact only), abbreviation, blockquote, cite, plain text and code markup

strong, em

Important content / emphasized content

hr

Paragraph-level thematic break

Tags providing non-semantic markup

i (names, terms)

Other content without additional meaning

b

Stylistisch different content

div

General tag to group other *flow content*.

span

General tag to group *phrasing content*, used for styling.

See Selfhtml wiki: wiki.selfhtml.org/wiki/HTML

Navigation, Link (anchor)

- [Home](#)
- [About](#)
- [Home](#)
- [Skyguide international](#)

examples/navigations.html

```
<nav>
    <ul>
        <li><a href="/home">Home</a></li>
        <li><a href="/products">About</a></li>
    </ul>
</nav>
```

```
<aside>
    <ul>
        <li><a href="/sitemap">Home</a></li>
        <li><a href="http://www.skyguide-glob.com"
            target="_blank">Skyguide international</a></li>
    </ul>
</aside>
```

Navigation rules

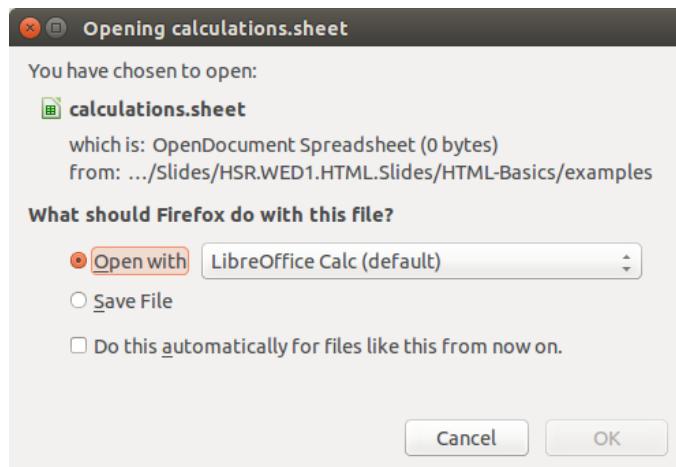
- **nav: Main navigation section**
→ not for aside menus, language menus or footer menus
Links (a)
- **May contain flow content**
→ containers, sections, **titles, paragraphs, images, lists, tables, ...**
- **Must not contain interactive tags**
→ other **links**, , , ...

■ *target*:

```
<!-- open in current window (default) -->
<a href="/about" target="_self">About</a>

<!-- open in new window (if browser allows) -->
<a href="www.ebay.com/my-shop" target="_blank">Shop</a>
```

■ *download, type*: Download linked document



```
<a href="calculations.sheet"
    type="text/comma-separated-values"
    download />
    Calculations
</a>
```

→ [Mime types SELFHTML.org](#)
→ [Mime types FREEFORMATER.com](#)

- User likes to view or download? (Images, PDF's, ...)
- *title* Tooltip: For all tags, not only for a-tags

■ *tabindex*: tabulator ordering

```
<a href="/about" tabindex="2">About</a>
<a href="/shop" tabindex="1">Shop</a>
```

■ Link naming:

```
<!-- bad -->
Download <a href="../download/pdf">here</a>
<!-- good -->
Download <a href="../download/pdf">AG Business Report 2015</a>
```

■ Call & mail links: [call +41 044 555 66 77](#), [info@iana.com](#)

```
<a href="tel:+410445556677" rel="nofollow">call +41 044 555 66 77</a>
<a href="sms:+410445556677" rel="nofollow">send sms to +41 044 555 66 77</a>
<a href="mailto:info@iana.com?subject=Hello">info@iana.com</a>
```

■ *callto*: / *skype*: Skype call (proprietary)

■ Anchors: [Product specs](#specs)

```
<h3 id="specs">Product specs</h3>
```

Lists

Unordered list (**ul**)

- London
- Berlin
 - East
 - West
- Paris

[examples/lists.html](#)

```
<ul>
    <li>London</li>
    <li>Berlin
        <ul>
            <li>East</li>
            <li>West
        </ul>
    </li>
    <li>Paris</li>
</ul>
```

[jsfiddle.net/3zhxwqct/](#)

Ordered List (**ol**)

```
1. Press clutch  
2. Turn key
```

```
3. GC  
2. FCZ  
1. FCB
```

[examples/lists.html](#)

```
<ol id="ol">  
    <li>Press clutch</li>  
    <li>Turn key</li>  
</ol>  
<ol start="3" reversed>  
    <li>GC</li>  
    <li>FCZ</li>  
    <li>FCB</li>  
</ol>
```

- **start:** Numbering start value
- **reversed:** Reverse ordering
- **value:** List point value (cont. numbering)

Definition list (**dl**)

Airport	Place where airplanes start and land
Airline	
Airplane	Vehicle to fly Also called <i>Aircraft</i>

[examples/lists.html](#)

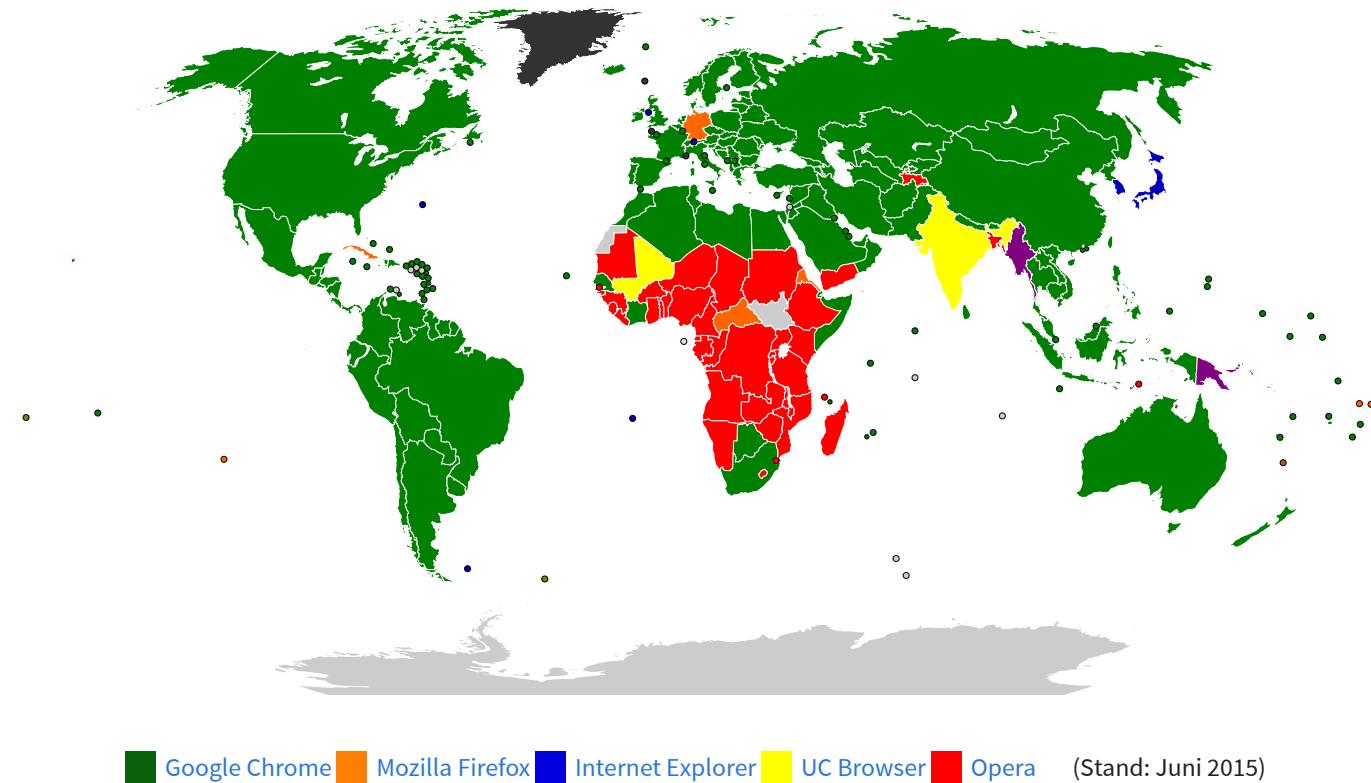
```
<dl>
    <dt>Airport</dt>
    <dd>Place where airplanes start and land</dd>

    <dt>Airline</dt>

    <dt>Airplane</dt>
    <dd>Vehicle to fly</dd>
    <dd>Also called <i>Aircraft</i></dd>
</dl>
```

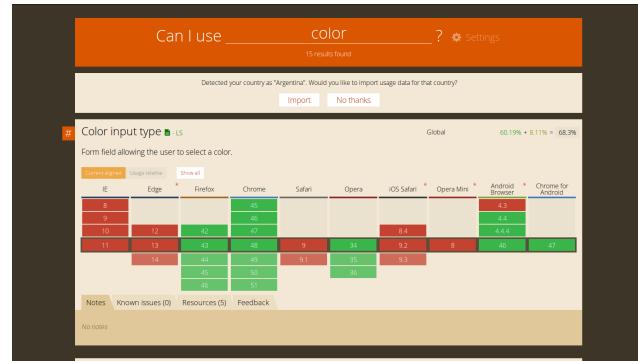
- Multiple *dt* or *dd* in-order allowed

Renderers / Browsers

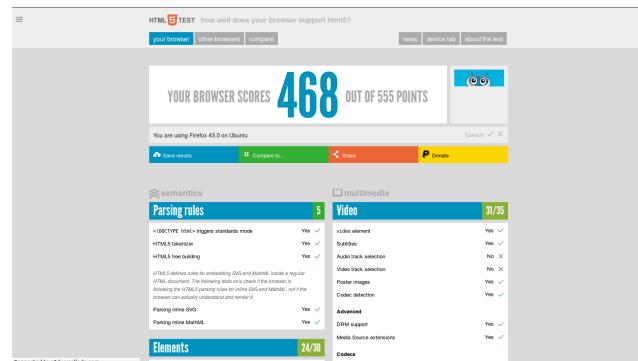


Von [Wilbysuffolk](#) - Stat Counter, CC-BY-SA 4.0, commons.wikimedia.org/w/index.php?curid=41797862

caniuse.com

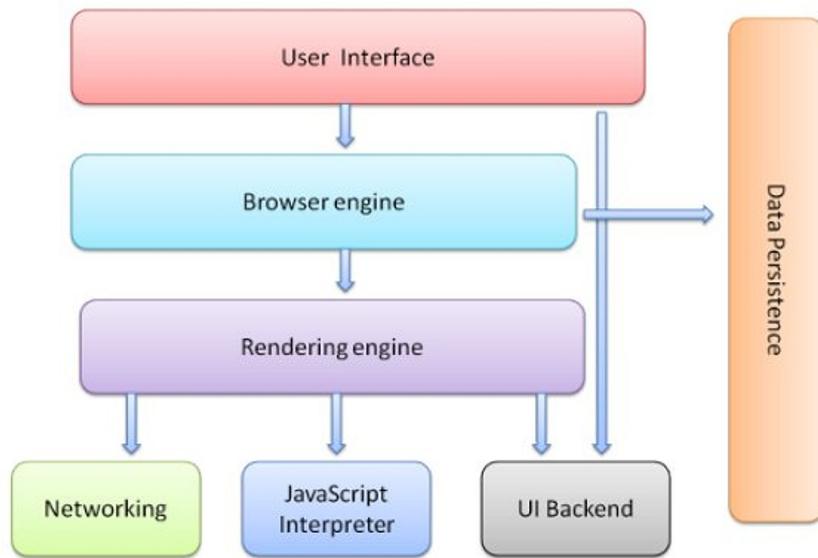


html5test.com

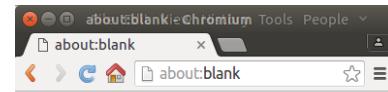


30 / 52

Internal structure of browsers



User interface



Browser engine & rendering engine

HTML parsing, DOM visualization

Data persistence

Cookies, local storage, ...

Networking

Request handling (HTTP)

UI backend

Operating system API

Webstorm codecheck

The screenshot shows the WebStorm IDE interface. The top navigation bar includes tabs for 'File', 'Edit', 'View', 'Run', 'Tools', 'Help', and a 'File Structure' icon. Below the bar, there are tabs for 'badjs1.js', 'badjs2.js', 'badjs3.js', 'SamplePage.html', and 'eslint.ini'. The left sidebar displays a project tree with 'WED1-FS15-LE05' expanded, showing files like 'badjs1.js', 'badjs2.js', 'badjs3.js', 'jquery.js', 'SamplePage.html', 'style.css', and 'style.stylesheet'. The main code editor window contains the following HTML and JavaScript code:

```
1 <!doctype HTML>
2 <HTML lang="en">
3   <head>
4     <META charset="utf-8">
5     <title>WED1 Coding Style Sample Page</title>
6     <link rel="stylesheet" href="style.stylesheet">
7   <style>
8     body {font-size: 10em;}</style>
9   <script src='jquery.js'></script>
10
11  <script>
12    $(function() {
13      Global_Variable = 'HTML'
14      $("P").append( Global_Variable ) })
15    <br>
16  <body>
17
18  <P>This is valid
19
```

The code editor has syntax highlighting and line numbers. Below the editor is a toolbar with icons for file operations. At the bottom of the screen is a status bar.

The bottom half of the image shows the 'Inspection' tool window. It has tabs for 'Results for Inspection Profile "Project Default"', 'Results for Inspection Profile "Project Default"', and 'Select inspection rule'. The 'Code style issues' section is expanded, showing one item under 'Unterminated statement' at line 13 of 'SamplePage.html'. The 'General' section shows one item under 'Implicitly declared global JavaScript variable' at line 13 of 'SamplePage.html'. The 'HTML' section shows one item under 'File reference problems' at line 13 of 'SamplePage.html', indicating 'Cannot resolve file "style.stylesheet"'.

The screenshot shows a code editor window titled "scratch". The code is an HTML document with the following structure:

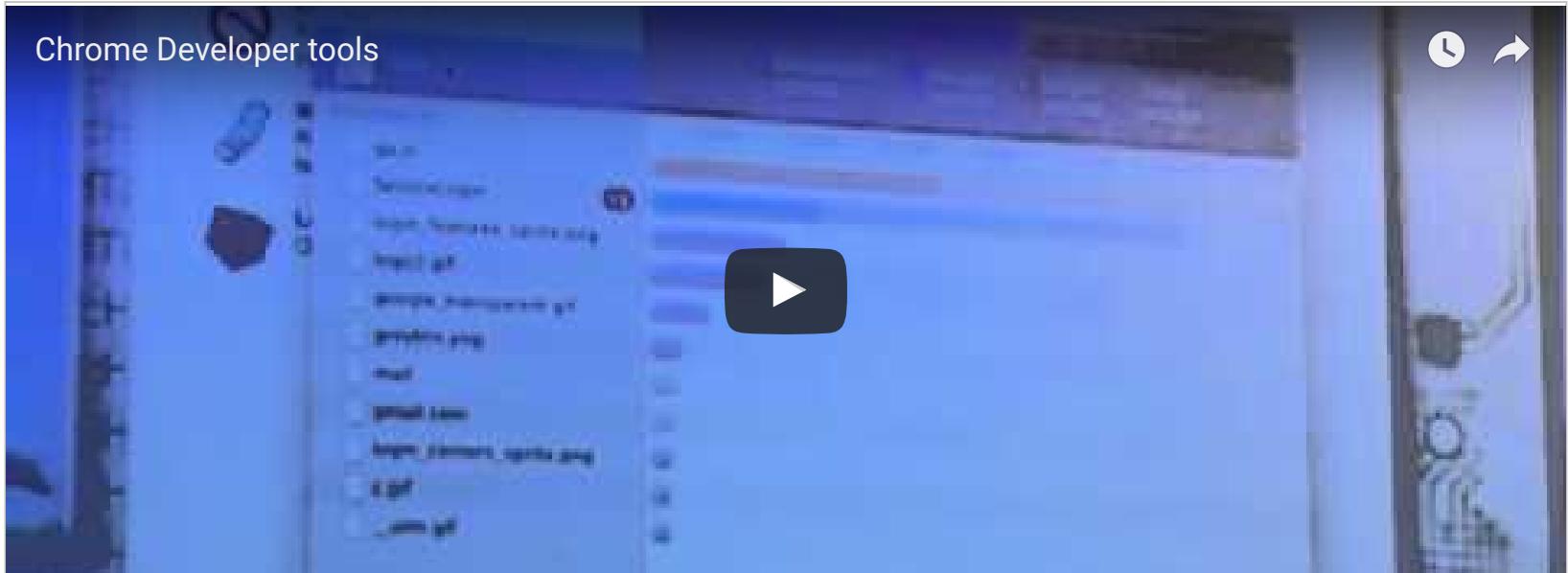
```
1 <!-- documenttype: HTML -->
2 <html lang="en">
3   <header>
4     <meta charset="uft-8" />
5     <title>Cars</title>
6   </header>
7   <body>
8     <div font="Arial" id="theUniqueContainer">
9       </img>
10      <p id="theUniqueContainer">
11        </div>
12      </p>
13      <form action=".submit" method="POST">
14        <textarea name="comment" size=5></textarea>
15      </form>
16    <body>
17    <foot>
18      <script src=".jquery.js">...
19      </foot>
20    </html>
```

Create scratch (temp file): Ctrl+Shift+Alt+Insert

Style guidelines

- google-styleguide.googlecode.com
- <http://codingstyleguide.com/lang/8/html>
- <https://github.com/styleguide/templates>

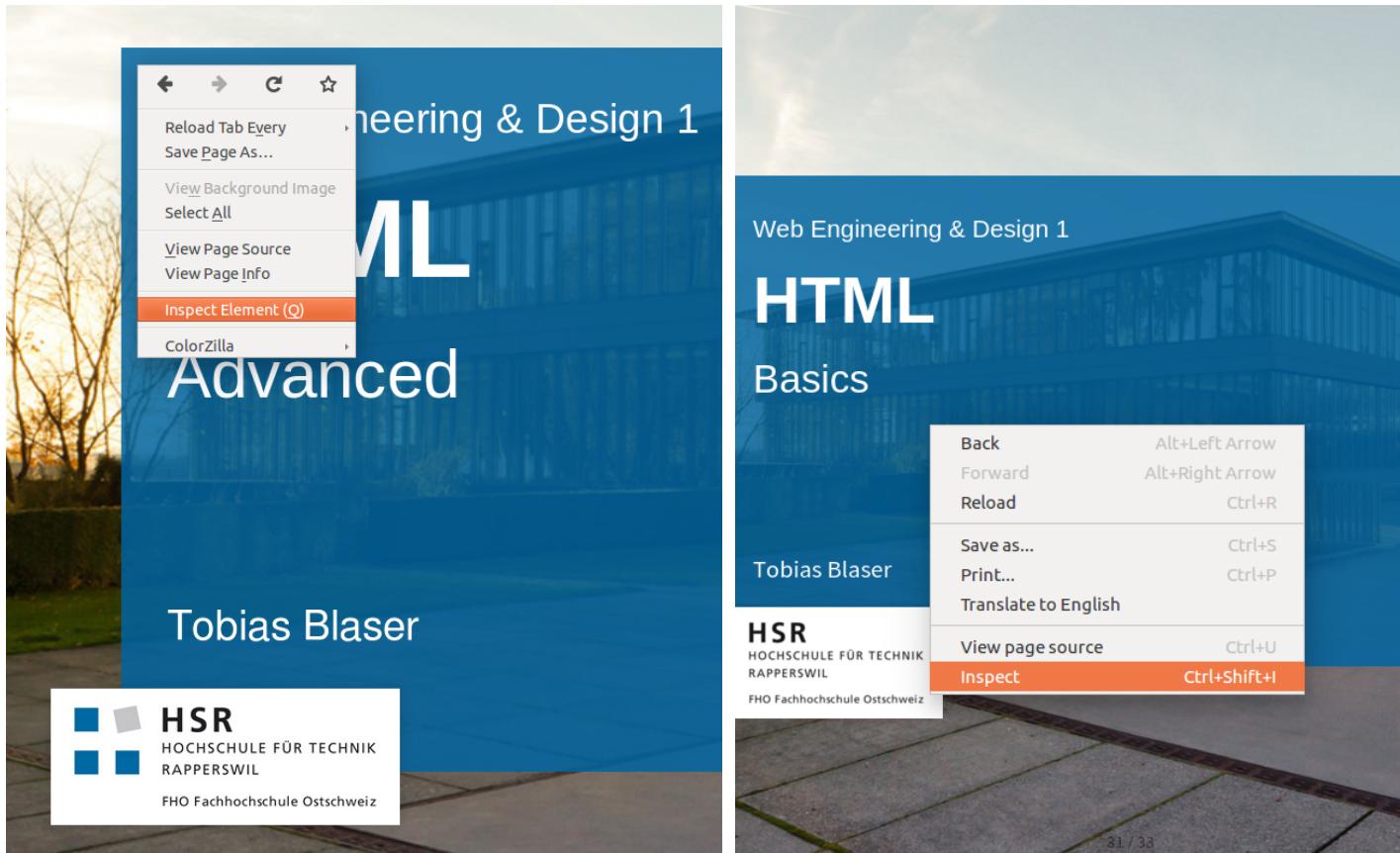
Plattform Independent Development Tools



Chrome DevTools (36:22)

www.youtube.com/embed/a7k6q_4TG4o

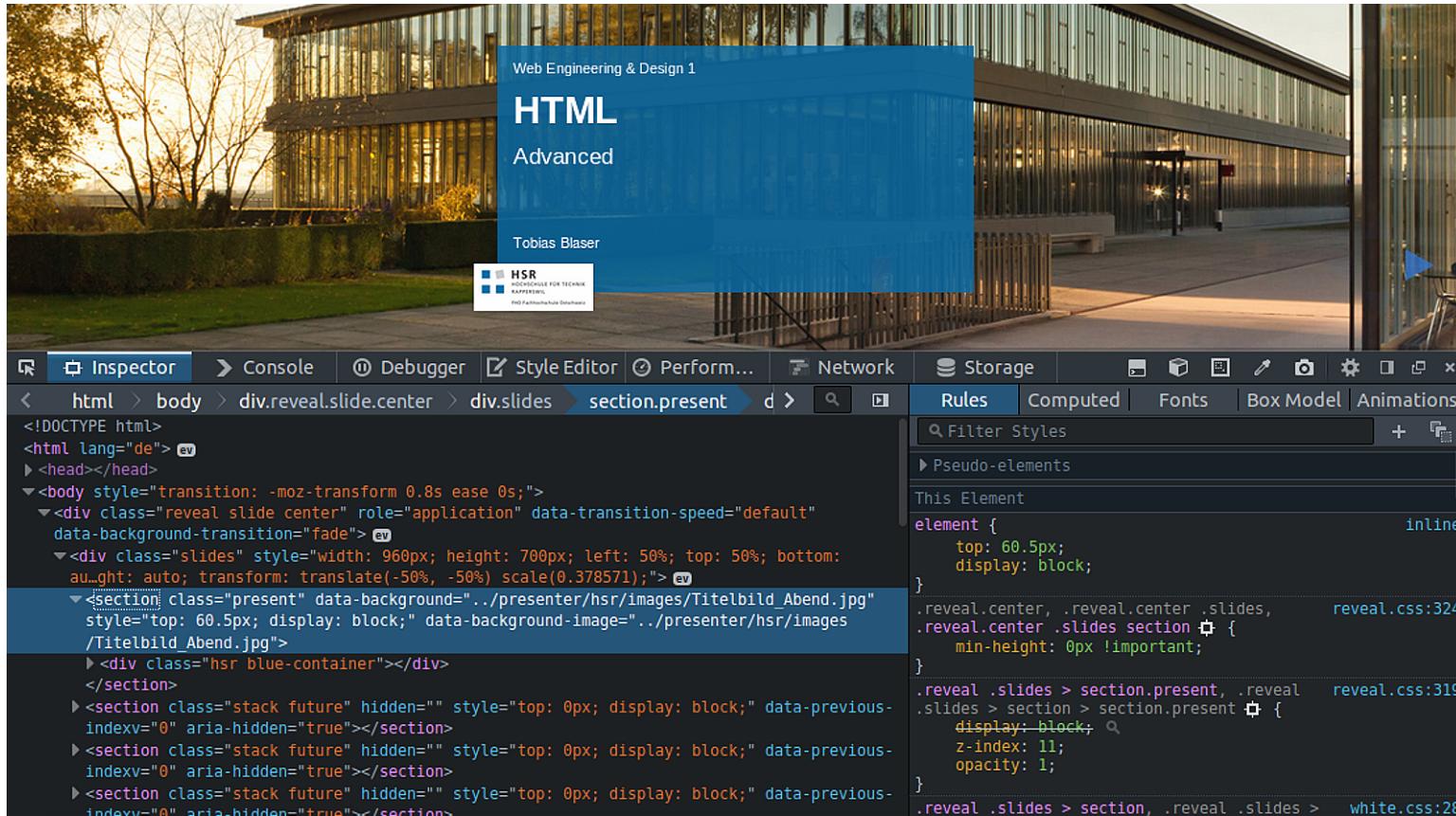
→ Switch your browser to english!



Firefox / Chrome

36 / 52

Inspector



DOM Inspector Firefox



Inspector Console Debugger Style Editor Performance

```
< div.reveal.slide.center > div.slides > section.present > div.hsr.blue-container > h1 translate(-50%, -50%) scale(0.494286);> ev
  <section class="present" data-background="..../presenter/hsr/images/Titelbild_Aben
  image=..../presenter/hsr/images/Titelbild_Abend.jpg" style="top: 15px; display: b
  <div class="hsr blue-container">
    <hgroup>
      <h3></h3>
        <h1>HTML - Hype <span>&copy;</span></h1>
        <h2>Advanced</h2>
    </hgroup>
```

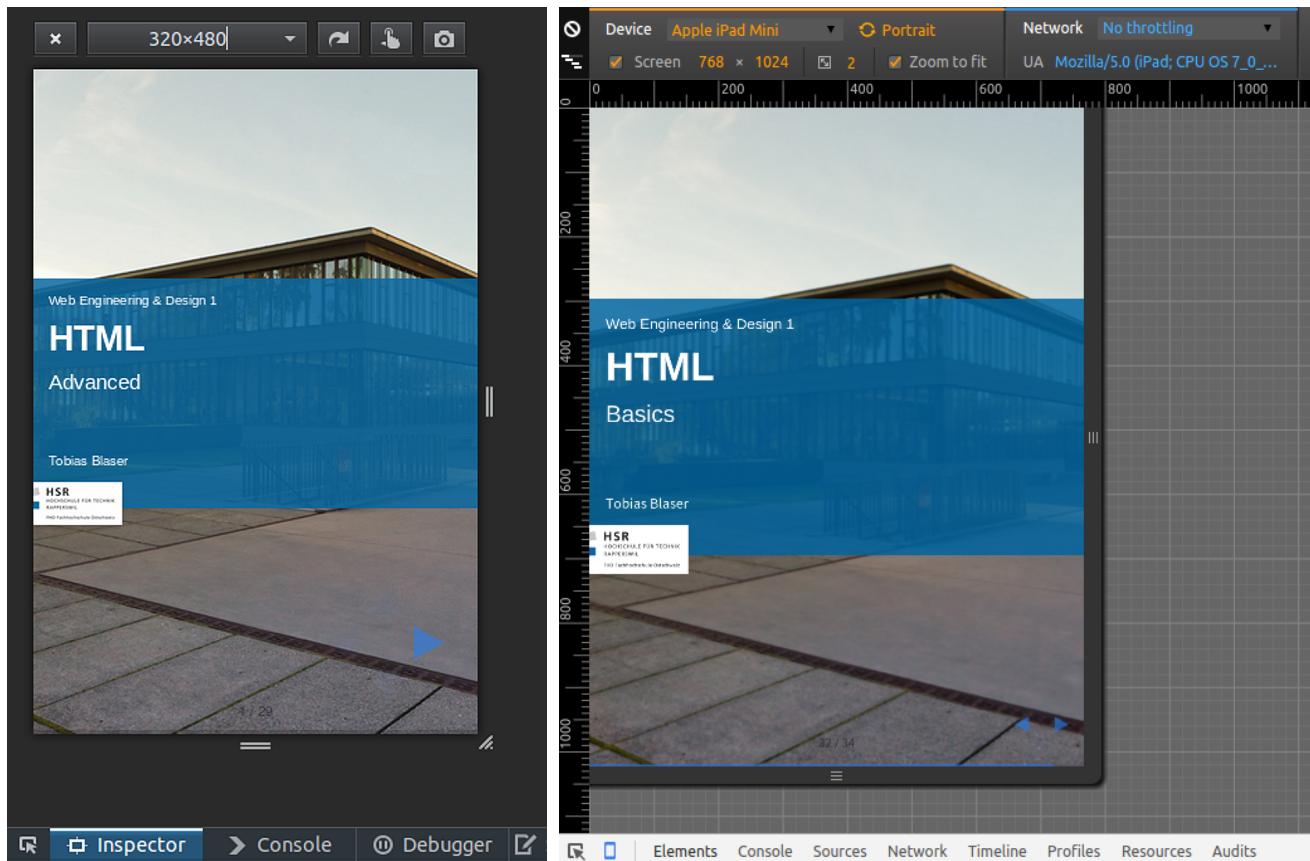
Inspector Console Debugger Style Editor Performance

```
< div.reveal.slide.center > div.slides > section.present > div.hsr.blue-conta
  > h3></h3>
    <h1>HTML
      <span>&copy;</span>
    </h1>
```

3D View (Firefox)



Responsive Design View



Firefox / Chrome

40 / 52

Network Speed Simulation (Chrome)

The screenshot shows the Chrome DevTools Network tab with a list of requests and a throttling dropdown menu open.

Network Tab Headers:

- Elements
- Console
- Sources
- Network** (selected)
- Timeline
- Profiles
- Resources
- Audits

Throttling Options:

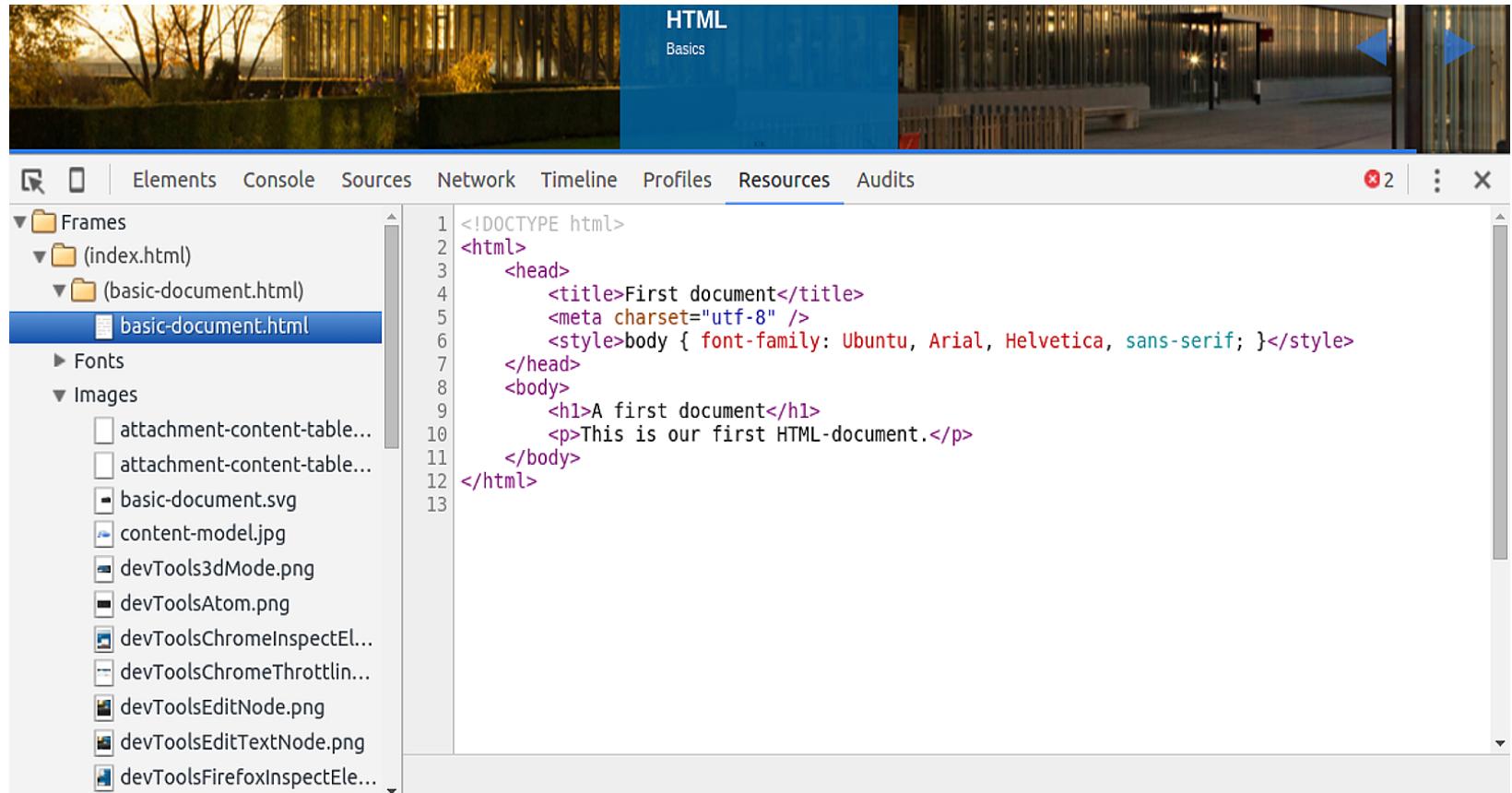
- View: **Preserve log** (checked), **Disable cache** (checked)
- No throttling
- Custom: Add...
- Presets:
 - Offline
 - GPRS (50 kb/s 500ms RTT)
 - Regular 2G (250 kb/s 300ms RTT)
 - Good 2G (450 kb/s 150ms RTT)
 - Regular 3G (750 kb/s 100ms RTT)
 - Good 3G (1 Mb/s 40ms RTT)
 - Regular 4G (4 Mb/s 20ms RTT)
 - DSL (2 Mb/s 5ms RTT)
 - WiFi (30 Mb/s 2ms RTT)
- Disabled: No throttling

Network Requests Table:

Name	Status	Type	Initiator	Size	Time	Content	Latency
Path	Text			Content	Downloaded	Time	Latency
/home/tblaser/Docu...	Finished	media	multimedia...	0 B	1 ms		
			Parser	50.0 KB	1 ms		
/home/tblaser/Docu...	Finished	media	multimedia...	0 B	6.29 s		
			Parser	8.0 MB	2 ms		
/home/tblaser/Docu...	Finished	media	Other	0 B	6.29 s		
				8.0 MB	2 ms		
/home/tblaser/Docu...	Finished	media	Other	0 B	6.27 s		
				8.0 MB	1 ms		

42 requests | 0 B transferred | Finish: 43.13 s | DOMContentLoaded: 151 ms | Load: 541 ms

Resources (Chrome)



The screenshot shows the Chrome DevTools interface with the "Resources" tab selected. The left sidebar displays a tree view of files and resources:

- Frames
 - (index.html)
 - (basic-document.html)
 - basic-document.html
- Fonts
- Images
 - attachment-content-table...
 - attachment-content-table...
 - basic-document.svg
 - content-model.jpg
 - devTools3dMode.png
 - devToolsAtom.png
 - devToolsChromeInspectEl...
 - devToolsChromeThrottlin...
 - devToolsEditNode.png
 - devToolsEditTextNode.png
 - devToolsFirefoxInspectEle...

The main content area shows the source code for "basic-document.html".

```
<!DOCTYPE html>
<html>
  <head>
    <title>First document</title>
    <meta charset="utf-8" />
    <style>body { font-family: Ubuntu, Arial, Helvetica, sans-serif; }</style>
  </head>
  <body>
    <h1>A first document</h1>
    <p>This is our first HTML-document.</p>
  </body>
</html>
```

HTML Validator

 Markup Validation Service
Check the markup (HTML, XHTML, ...) of Web documents

[Validate by URI](#) [Validate by File Upload](#) [Validate by Direct Input](#)

Validate by direct input

Enter the Markup to validate:

```
dependencies: [ ... ]  
!document.body.classList; } ],  
        { src: '../presenter/reveal/lib/js/classList.js', condition: function() { return  
return !document.querySelector( 'pre code' ); }, callback: function() { hljs.initHighlightingOnLoad(); } },  
        { src: '../presenter/reveal/plugin/highlight/highlight.js', async: true, condition: function() {  
        [ ... ]  
        } );  
    </script>  
</body>  
</html>
```

▼ More Options

Validate Full Document

Use Doctype: Only if Doctype is missing

Validate HTML fragment

Use Doctype: HTML 4.01 XHTML 1.0

List Messages Sequentially Group Error Messages by Type

Show Source Clean up Markup with HTML-Tidy

Show Outline Validate error pages Verbose Output

[Check](#)

validator.w3.org/#validate_by_input+with_options

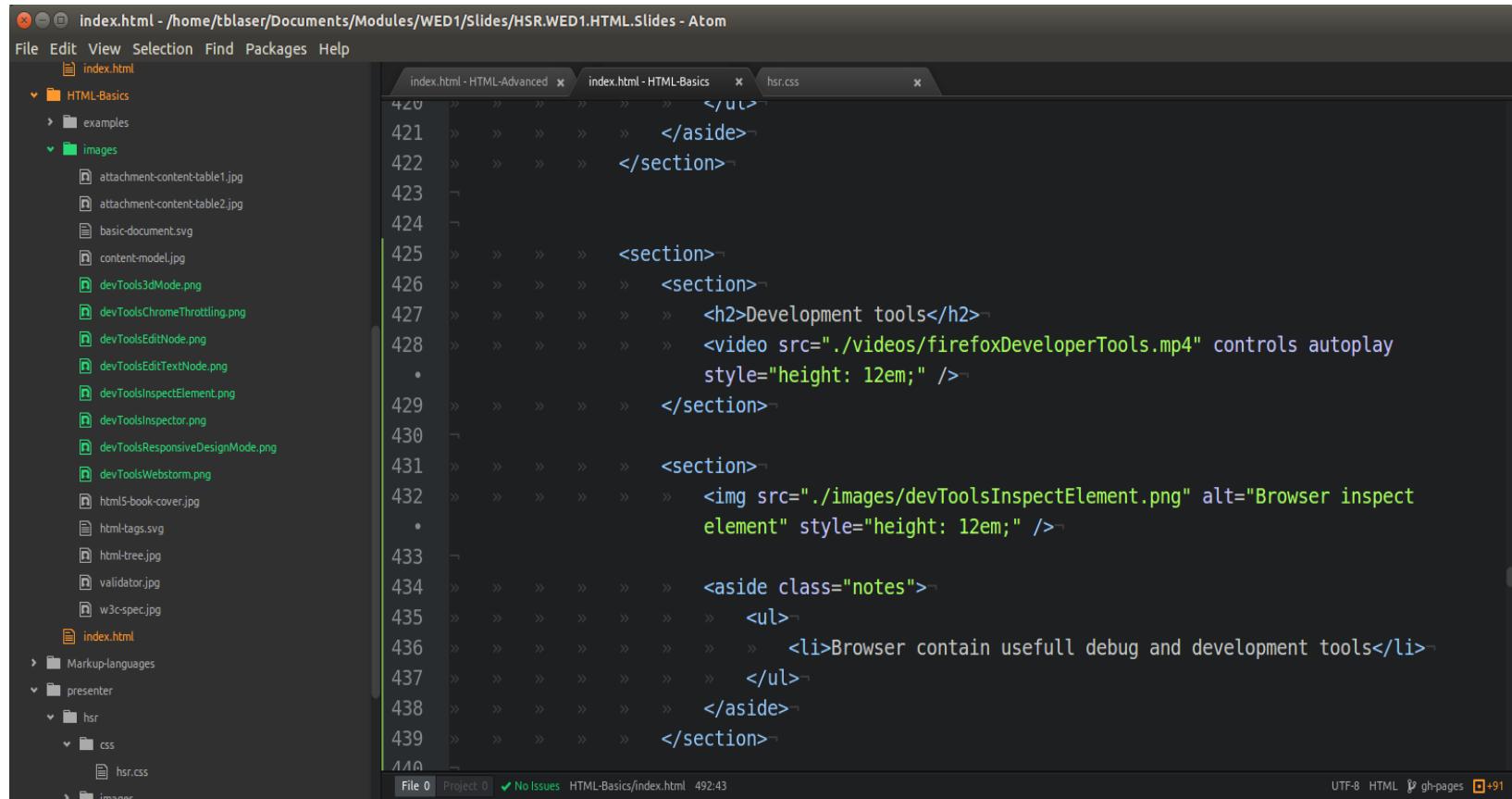
Webstorm (JetBrains IDE)

The screenshot shows the WebStorm IDE interface with the following details:

- Title Bar:** HSR.WED1.HTML.Slides - [/Documents/Modules/WED1/Slides/HSR.WED1.HTML.Slides] - .../HTML-Basics/index.html - WebStorm 10.0.4
- Menu Bar:** File Edit View Navigate Code Refactor Run Tools VCS Window Help
- Toolbar:** Includes icons for Run, Stop, Refresh, and various VCS and settings.
- Project Explorer:** Shows the project structure under "HSR.WED1.HTML.Slides".
 - HTML-Advanced
 - HTML-Basics
 - examples
 - images
 - attachment-content-table1.jpg
 - attachment-content-table2.jpg
 - basic-document.svg
 - content-model.jpg
 - devTools3dMode.png
 - devToolsChromeThrottling.png
 - devToolsEditNode.png
 - devToolsEditTextNode.png
 - devToolsInspectElement.png
 - devToolsInspector.png
 - devToolsResponsiveDesignMode.png
 - html5-book-cover.jpg
 - html-tags.svg
 - html-tree.jpg
 - validator.jpg
 - w3c-spec.jpg
 - index.html
 - Markup-languages
 - presenter
 - .gitignore
 - illustrations.odg
 - index.html
 - README.md
 - robots.txt
 - External Libraries
- Code Editor:** The "index.html" file is open, showing the following code:

```
<html>
  <head>
    ...
  </head>
  <body>
    ...
    <aside>
      
    </aside>
    <div>
      <section>
        <h2>Content</h2>
        <ul>
          <li><a href="#/5">A first HTML document</a></li>
          <li><a href="#/6">HTML Tags</a></li>
          <li><a href="#/7">Content model</a></li>
          <li><a href="#/8">Document validation</a></li>
          <li><a href="#/9">Development tools</a></li>
        </ul>
      </section>
      <section>
        <h2>Learning goals</h2>
        <ul>
          <li>You are able to create a basic document containing documents</li>
          <li>You are able to read the W3C spec of an element to determine its allowed content</li>
        </ul>
      </section>
    </div>
  </body>
</html>
```
- Status Bar:** 23:1 LF+ UTF-8 Git: gh-pages

Atom (Hackable Editor By GitHub)



The screenshot shows the Atom code editor interface. The top menu bar includes File, Edit, View, Selection, Find, Packages, and Help. The left sidebar displays a file tree with the following structure:

- index.html - /home/tblaser/Documents/Modules/WED1/Slides/HSR.WED1.HTML.Slides - Atom
- File
- Edit
- View
- Selection
- Find
- Packages
- Help
- index.html
- HTML-Basics
- examples
- images
 - attachment-content-table1.jpg
 - attachment-content-table2.jpg
 - basic-document.svg
 - content-model.jpg
 - devTools3dMode.png
 - devToolsChromeThrottling.png
 - devToolsEditNode.png
 - devToolsEditTextNode.png
 - devToolsInspectElement.png
 - devToolsInspector.png
 - devToolsResponsiveDesignMode.png
 - devToolsWebstorm.png
 - html5-book-cover.jpg
 - html-tags.svg
 - html-tree.jpg
 - validator.jpg
 - w3c-spec.jpg
- index.html
- Markup-languages
- presenter
 - hsr
 - css
 - hsr.css

The main editor area has three tabs open:

- index.html - HTML-Advanced
- index.html - HTML-Basics (active tab)
- hsr.css

The content of the active tab (index.html - HTML-Basics) is as follows:

```
421 >> >> >> >> >> </aside>
422 >> >> >> >> </section>
423
424
425 >> >> >> >> <section>
426 >> >> >> >> <section>
427 >> >> >> >> >> <h2>Development tools</h2>
428 >> >> >> >> >> <video src=".//videos/firefoxDeveloperTools.mp4" controls autoplay style="height: 12em;" />
429 >> >> >> >> </section>
430
431 >> >> >> >> <section>
432 >> >> >> >> >> 
433
434 >> >> >> >> >> <aside class="notes">
435 >> >> >> >> >> >> <ul>
436 >> >> >> >> >> >> >> <li>Browser contain usefull debug and development tools</li>
437 >> >> >> >> >> >> >> </ul>
438 >> >> >> >> >> >> </aside>
439 >> >> >> >> >> </section>
```

The status bar at the bottom shows: File 0 Project 0 No Issues HTML-Basics/index.html 492:43, followed by icons for UTF-8, HTML, gh-pages, and +91.

Other tools

- Ecplipse web IDE / Aptana
- NetBeans IDE
- Sublime
- Visual Studio Code (based on Atom)



Web Engineering & Design 1

HTML Basics

Tobias Blaser



Self Check

1. Which of the following tags may be nested inside a "p" tag: "div", "span", "img", "video", "table", "body"?
2. What is the Quirks mode? When will it be used?
3. How do you determine which tags are valid without a closing tag and which tags are void tags?
4. Why should you add meta data to your HTML document?
5. What are "http-equiv"-meta tags?

6. How do you use header, aside, small, headings and main?
7. Why does it make sense to add the attribute type, if you use the attribute download on links?
8. How do you choose link names and why?
9. Lists: What's the difference between *ul*, *ol* and *dl*?
10. How do you determine which browser supports a tag?

Solutions

1. "p"-tags should contain only phrasing content and subtypes: span, img, video.
2. It's a compatibility mode. Some rendering will be different. Browsers will use it if there's no or a legacy DocType defined.
3. W3C Spec
4. For search engines, accessibility tools and document export tools
5. HTTP header equivalent tags

6. See slide #/6/4
7. The browser knows the downloaded file type and may pass the file path directly to the desired application on the client machine.
8. Specific names like "Tesla S data sheet" and not "download here" → better search engine ranking and accessibility
9. ul: unordered list, ol: ordered list, dl: definition list for glossary
10. caniuse.com

A photograph of a modern, multi-story building with large glass windows and a dark facade. The sky is overcast and has a blue tint, suggesting dusk or dawn. The building is set against a backdrop of trees and a paved area.

Web Engineering & Design 1

HTML Basics

Tobias Blaser

