



Úvod HTML5

kurz Základy webových
technológií

Eduard Kuric

Motivácia

- **Koľko času denne strávite na Internete?**

Motivácia

- Koľko času denne strávite na Internete?
 - **priemerne dospelý človek 2h 45m**

Motivácia

- Koľko času denne strávite na Internete?
 - priemer dospelý človek 2h 45m
- Cez internet
 - **vzdelávame sa**
 - **zdieľame zážitky, socializujeme sa**
 - **nakupujeme**
 - **menežujeme svoje financie**
 - **hľadáme služby (opravára chladničky)**
 - **čítame noviny**
 - **hráme hry**
 - ...

Motivácia

- profesia.sk – najviac pracovných ponúk
programátor/dizajnér webov

Motivácia

- profesia.sk – najviac pracovných ponúk programátor/dizajnér webov
- **Microsoft IE, 2003 – 95% osobných počítačov**
 - **dominancia** (stabilné prostredie pre vývojárov), web stagnoval, **nedostatok konkurencie**

Motivácia

- profesia.sk – najviac pracovných ponúk programátor/dizajnér webov
- Microsoft, IE, 2003
 - dominancia (stabilné prostredie pre vývojárov), web stagnoval, nedostatok konkurencie
- Dnes?
 - **4 kľúčoví výrobcovia prehliadačov**
 - **Chrome, Firefox, Safari, Edge**
 - **skoro rovnaký počet operačných systémov**

Motivácia

- profesia.sk – najviac pracovných ponúk programátor/dizajnér webov
- Microsoft, IE, 2003
 - dominancia (stabilné prostredie pre vývojárov), web stagnoval, nedostatok konkurencie
- Dnes?
 - 4 kľúčoví výrobcovia prehliadačov, skoro rovnaký počet operačných systémov
- **Spôsob prístupu na Web sa zmenil**
 - mobily, tablety, rôzne inteligentné zariadenia - hodinky, televízory, herné konzoly

Motivácia

- **nároky na web a očakávania sa zmenili...**
- od “tohto” ;-)
 - <http://www.theworldsworstwebsiteever.com/>
- k moderným, použiteľným, funkčným, bezpečným, pekným, prenositeľným webovým stránkam/aplikáciám

Motivácia

- Kedysi, keď sme mali Žiguliaka, alebo Škodu 105
 - nikoho netrápilo, že pasažieri nie sú vzadu pripútaní
 - dnes, by auto nemohlo ísť na cestu bez zadných pásov
- Podobne je to aj s webom, posúvame sa...
 - vývojári chcú vytvárať krásne, funkčné, moderné, použiteľné a prístupné (naprieč rôznymi zariadeniami, pre používateľov s hendikepmi) webové aplikácie
 - používatelia chcú informácie, zábavu, zážitky, inovácie
 - všetko pri sebe - vo vrecku aktuálne a hneď
vyberať to čo je dôležité, zaujímavé, pre mňa, vo vhodný čas/v reálnom čase

Motivácia

- Kedysi, keď sme mali Žiguliaka, alebo Škodu 105
 - nikoho netrápilo, že pasažieri nie sú vzadu pripútaní
 - dnes, by auto nemohlo ísť na cestu bez zadných pásov
 - Podobne je to aj s webom, posúvame sa...
 - natívne aplikácie chcú ísť na web, tam to žije**
 - nie v Play Store
- ... všetko chce ísť na Web
- ... nároky na webové aplikácie sú väčšie a väčšie...

Pôjdeme postupne...

- Úvod, HTML(5) – živá špecifikácia, DOM
- CSS, SaSS, responzívny dizajn
- Architektúry webových aplikácií
- **Server Side Rendering**, REST API
- MVC Architektúra - Laravel rámec, zákl. koncepty
- Základy JavaScript, WEB API, XML technológie
- Prístupnosť, multimediálne formáty, základné nástroje na optimalizáciu

Prednášky/cvičenia

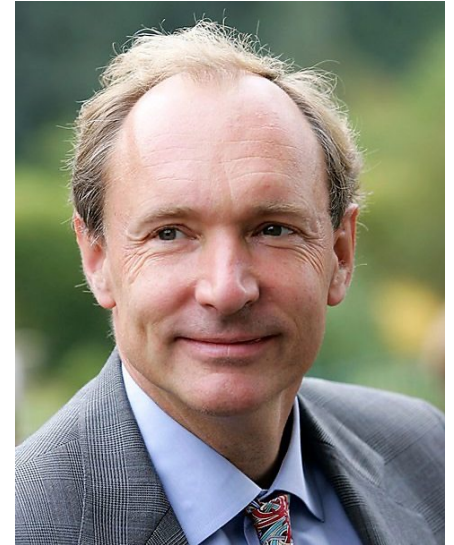
- **Ing. Eduard Kuric, PhD.**
- **eduard.kuric@stuba.sk**
- miestnosť 4.42

Všetky informácie:

- <https://github.com/kurice/wtech23>

WWW

- **World Wide Web** (celosvetová sieť)
- **informačný priestor rôznych zdrojov** (dokumentov) na Internete **prístupných** prostredníctvom protokolu **HTTP(S)**
- autorom Webu je **Tim Berners-Lee**
- dokumenty - zvyčajne HTML
 - uložené na webových serveroch
 - štrukturované v HTML jazyku
 - prezeráme ich pomocou webových prehliadačov



HTTP /RFC 2616

- **Hypertext Transfer Protocol**
- **internetový protokol na výmenu hypertextových dokumentov (HTML)**
- rozšírenie MIME (**M**ultipurpose **I**nternet **M**ail **E**xtensions) umožňuje prenášať akýkoľvek súbor
- implicitný port 80, https 443
- **používa URL** (Uniform Resource Locator) - jednoznačné umiestnenie zdroja

URL

- **Uniform Resource Locator**
- je referencia, ktorá určuje umiestnenie webového zdroja na Internete

scheme:[//[user[:password]@]host[:port]][/path][?query][#fragment]

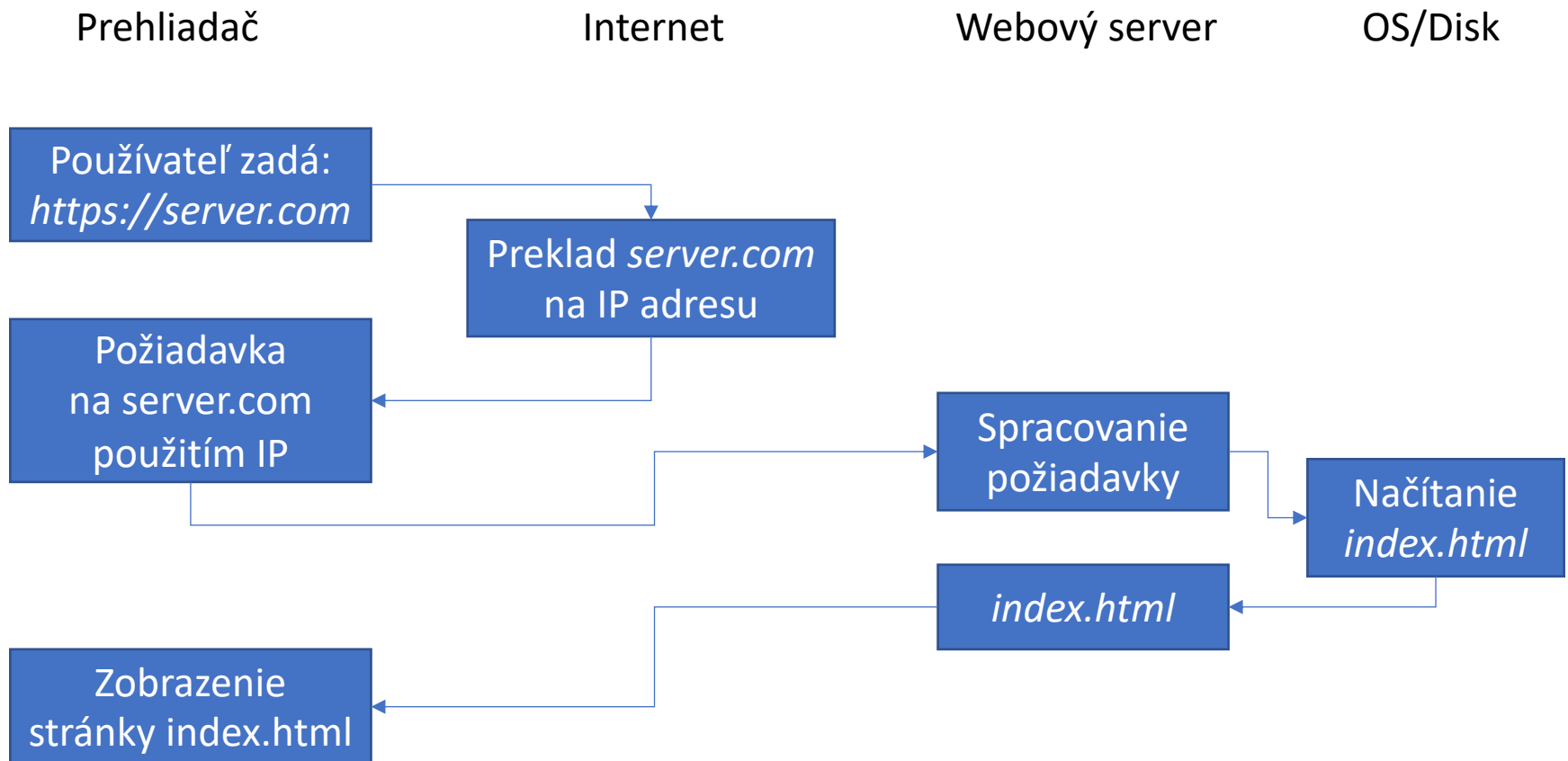
<https://www.eshop.com/smartphones/apple/list?order=ASC#pagination>

Webový server

- počítač, ktorý vykonáva HTTP požiadavky od klientov (najčastejšie webový prehliadač)
- odpoveď: dokument, najčastejšie HTML
- [Apache HTTP](#)
 - open-source, cross-platform
- [Internet Information Services](#)
 - Microsoft, Windows
- [Node.js](#)
 - open-source, cross-platform

Statické stránky

klient/server požiadavka/odpoveď



HTML

- HTML – **H**yper **T**ext **M**arkup **L**anguage
- **vyzerá ako text, je to text**
- **zjednodušené - obsah** (nadpisy, odseky, zoznamy, tabuľky...)
- nie je to programovací jazyk, je to **opisný značkovací jazyk**
- popisuje štruktúru stránky, slúži na štruktúrovanie stránok
- prehliadače nezobrazujú značky HTML, používajú ich na vykreslenie/zostavenie obsahu stránky
- samo o sebe je HTML fádny

CSS

- CSS - **Cascading Style Sheets**
- stará sa o **výzor stránky - prezentáciu, formátovanie**
 - sú to štýly, to, čomu zjednodušene hovoríme dizajn
 - písmo, farby, orámovanie, umiestnenie, pozadie,...
- HTML nebolo nikdy určené na to, aby formátovalo obsah
- HTML 3.2 špecifikácia priniesla element `` a atribút `color`
 - začala sa nočná mora pre vývojárov, každá stránka obsahovala písmo, farby – dlhý, zložitý, drahý vývoj
- CSS odstránilo formátovanie z HTML

JavaScript

- je to programovací jazyk HTML a Webu
- **na programovanie správania webových stránok**
- nemá nič s Javou

DOM

- **Document Object Model**
- **OO reprezentácia XML alebo HTML dokumentu**
- **je to API umožňujúce prístup/modifikáciu obsahu, štruktúry, alebo štýlu dokumentu**
- pôvodne mali prehliadače vlastné špecifické rozhranie na manipuláciu s HTML elementami
- W3C štandardizácia
- DOM umožňuje prístup k dokumentu, ako ku stromovej reprezentácii

DOM /2

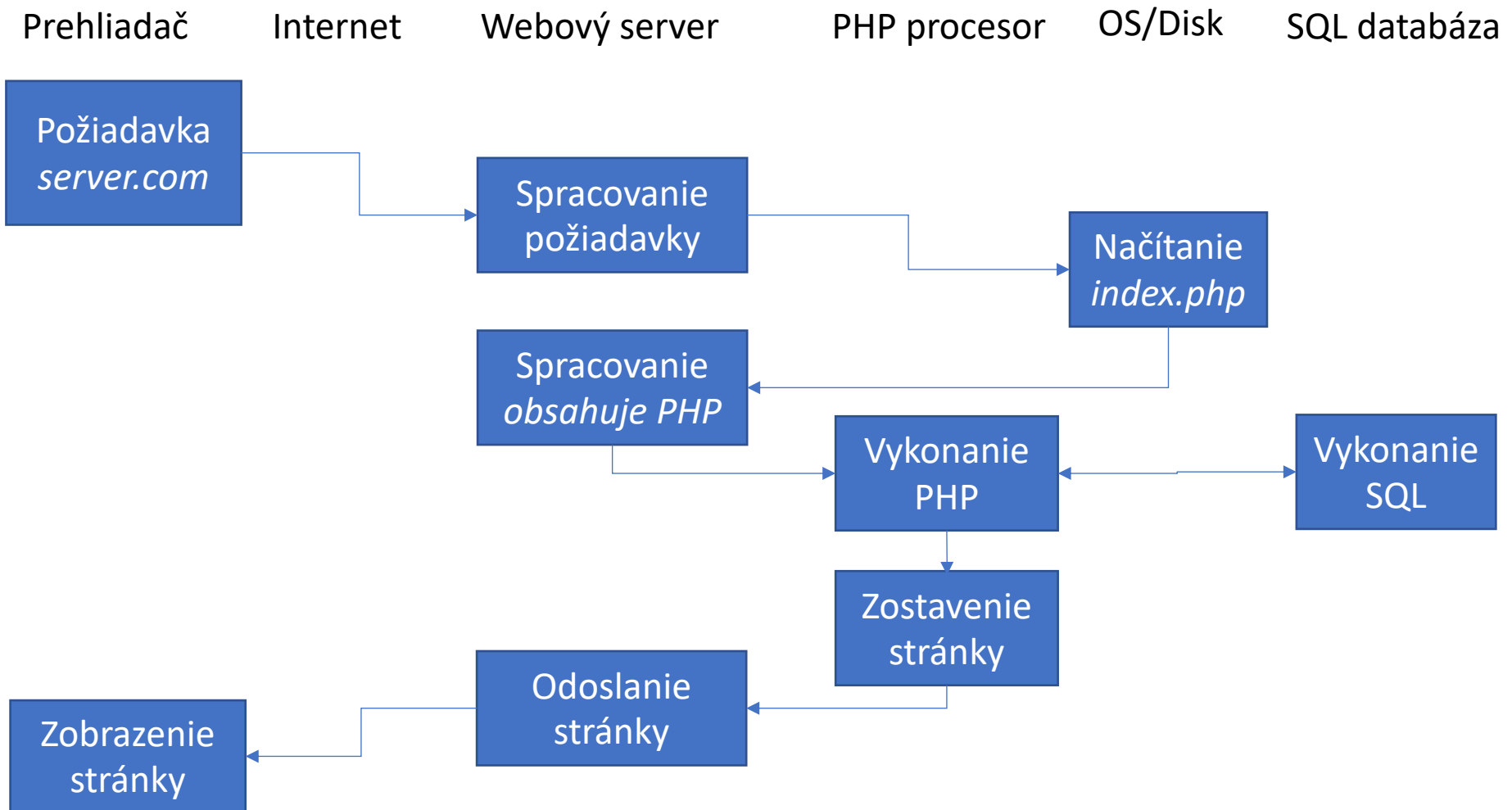
- na vykreslenie stránky používa väčšina **prehliadačov** interný **model podobný DOM**
- uzly v strome sú usporiadané v stromovej štruktúre – **DOM tree**
- **koreň** sa nazýva **document object**
- keď je stránka načítaná, prehliadač vytvorí DOM, ktorý funguje ako rozhranie medzi JavaScriptom a dokumentom
 - umožňuje vytvárať dynamické webové stránky
- JavaScript môže **pridávať, meniť, odstraňovať HTML elementy** a atribúty v stránke
- JavaScript môže **meniť** všetky **CSS štýly**
- JavaScript môže **reagovať** na všetky existujúce **udalosti** na stránke
- JavaScript môže **vytvárať** nové **udalosti** v rámci stránky

HTML+CSS+JavaScript

- **tvoria chrbticu**, neodlučiteľná trojica
- postačujú na vytváranie plnohodnotných webových aplikácií (+ nejaké tie rámce):
 - moderné
 - interaktívne
 - fungujúce v reálnom čase
- svet je pestrofarebný, treba poznať výhody/nevýhody technológií

Dynamické stránky

klient/server požiadavka/odpoveď



Web server - jazyky

- Apache + interpretované jazyky:
 - PHP
 - Perl
 - Ruby
 - Python
- IIS najmä ASP.NET
 - ASP.NET rámec (angl. framework)
 - kompilované programovacie jazyky C#, VB
- Node.js
 - JavaScript – interpretovaný/kompilovaný

Zostavenie obsahu

- na serveri (*angl. server side rendering*)
- na klientovi (*angl. client side rendering*)
- Na serveri
 - Apache
 - PHP (*Hypertext Preprocessor*)
 - Laravel rámec (*angl. framework*)
 - SQL databáza
- Na klientovi
 - Vue.js
 - [shadow DOM](#)

HTML – história

- HTML 0.9-1.2 (1991 – 1993)
 - Tim Berners-Lee, Daniel Connolly
 - na základe SGML, nepodporuje grafické rozhranie
- HTML 2.0 (1993)
 - IETF (vyvíja internetové štandardy)
 - pridané interaktívne formuláre a podpora grafiky
- HTML 3.0? (1995)
 - neujal sa, komplikovaný, vývojári ho neboli schopní implementovať
- HTML 3.2 (1997)
 - W3C
 - **tabuľky**, zarovnanie textu, štýly

W3 konzorcium (W3C)

- v 1994 založil Tim Berners-Lee (predseda W3C)
- medzinárodné konzorcium
 - vyvíja webové štandardy pre WWW
- pred založením ponúkali rôzne firmy rôzne upravené verzie jazyka HTML (nekompatibilita)
- zjednotilo verzie od výrobcov
 - definujú základné princípy a komponenty nových štandardov

HTML – história

- vytvoril v 1990/1991 Tim Berners-Lee
- HTML 1.1 (1992)
- HTML 2.0 (1993)
- HTML 3.0? (1995, neujal sa)
- HTML 3.2 (1997, W3C)

-
- HTML 4.01 (1999, 3 var.: strict, transitional, frameset)
 - do popredia sémantika, niektoré prezentačné prvky boli odobraté
 - XHTML 1.0 (2000, reformulácia HTML, tiež 3 var.)
 - XHTML 1.1 (2001)
 - XHTML 2.0 (vývoj ukončený 2009)

XHTML

- **dokumenty XHTML sú súčasne dokumentmi XML**
- prísnejšia syntax
- odstraňuje atribúty upravujúce vzhľad, rozmery a pozíciu elementov jazyka
- sebe menšia syntaktická chyba v dokumente viedla k odmietnutiu jeho spracovania
- žiadna nová funkcionálna oproti HTML 4.01
- špecifikácia znemožňovala pridávať nové elementy/atribúty, čo mala byť pôvodne jedna z výhod
- výrobcovia prehliadačov sa nehrnuli do impl., až IE 9 (10 rokov)
- pomalšie vykresľovanie oproti HTML
 - načítanie celej stránky do pamäte, vybudovanie stromu, vykreslenie
 - HTML sa vykresľuje postupne, podobný mechanizmus v XHTML až v 2006

WHATWG a HTML5

- niektorým, najmä výrobcom prehliadačov sa nepáčilo **smerovanie XHTML**
- **Web Hypertext Application Technology Working Group** (Apple, Google, Opera, Mozilla + ďalší)
- príprava špecifikácie, ktorú by schválilo W3C
- 2007 W3C založilo novú pracovnú skupinu
- dohoda, nová verzia bude založená na špecifikácii od WHATWG, ponesie označenie HTML5, XHTML 2.0 definitívne končí

HTML5 | W3C + WHATWG

- HTML5 (2014) je následníkom HTML 4.01
- prináša množstvo nových funkcií
- niektoré označuje za zastarané, alebo ich rovno ruší, alebo mení ich správanie
- štandardizuje veľa „hackov“ a návrhových vzorov, ktoré sa vývojári naučili používať
- rozširuje HTML pre potreby moderných webových aplikácií
- živá špecifikácia – „nie je číslovaná“
 - pribúdajú nové funkcie, menia sa existujúce
 - W3C si osvojuje špecifikácie a čísluje ich, upravuje

HTML 5.1, 5.2 ...

- je snaha o vývoj webu smerom, ktorý by pokrýval väčšinu súčasných požiadaviek
 - tie sa významne zmenili od jednoduchej siete vzájomne prepojených dokumentov
- HTML 5.1 (2016) – W3C odporúčanie
- HTML 5.2 (2017) – W3C odporúčanie
- [HTML 5.3 \(2018\)](#) – W3C odporúčanie

HTML - **Living Standard**

- <https://www.w3.org/blog/2019/05/w3c-and-whatwg-to-work-together-to-advance-the-open-web-platform/>
- [WHATWG](#)
- HTML už nie je ďalej číslované, **jednoducho jazyk HTML**

WHATWG <address>

HTML

Living Standard — Last Updated 20 September 2021

- <https://html.spec.whatwg.org/multipage/sections.html#the-address-element>

“The address element **must not be used** to represent arbitrary addresses (e.g. **postal addresses**), unless those addresses are in fact the relevant contact information.”

W3C <address>

- HTML 5.2
- <https://www.w3.org/TR/html52/grouping-content.html#the-address-element>

“The address element represents contact information for a person, people or organization. It **should include physical and/or digital location/contact information** and a means of identifying a person(s) or organization the information pertains to.”

W3C <address>

- HTML 5.2
- <https://www.w3.org/TR/html52/grouping-content.html#the-address-element>

<address>

<p>Name: Hament Dhanji

<p>House number: 1976

<p>Street: Meadowband Road

...

</address>

<https://developer.mozilla.org/en-US/docs/Web/HTML/Element/address>

HTML - kostra dokumentu

- značky (tagy) sú základné stavebné bloky
- značky reprezentujú typ obsahu

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>Moja prvá stránka</title>
  </head>
  <body>

    <h1>Môj prvý nadpis</h1>
    <p>Môj prvý odsek</p>

  </body>
</html>
```

HTML – značka/tag

- značky sú **názvy elementov** ohraničené/obklopené lomenými zátvorkami (angl. angle brackets)

`<p>Môj prvý odsek</p>`

- sú zvyčajne v pároch
 - začiatočná, resp. koncová značka (otváracia/uzatváracia)
- pred názvom koncovej značky je lomítko /
- značka: `<p>`, `</p>`, `<h1>`, ...
- element: `<p>Môj prvý odsek</p>`
 - začiatočná, koncová, obsah (voliteľný)

HTML <!DOCTYPE >

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>
      Moja prvá stránka
    </title>
  </head>
  <body>
    <h1>
      Môj prvý nadpis
    </h1>
    <p>Môj prvý odsek</p>
  </body>
</html>
```

deklarácia definuje typ dokumentu,
konkrétne HTML(5)

XHTML 1.1:

```
<!DOCTYPE html PUBLIC
"-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/D
TD/xhtml11.dtd">
```

pozn.: odstavec je nespisovne

HTML <html>

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>
      Moja prvá stránka
    </title>
  </head>
  <body>
    <h1>
      Môj prvý nadpis
    </h1>
    <p>Môj prvý odsek</p>
  </body>
</html>
```

koreňový element HTML stránky

lang je atribút s hodnotou "sk"

atribúty poskytujú ďalšie informácie o HTML elementoch, nachádzajú sa vždy v začiatkovej značke

v tomto prípade, jazyk html dokumentu (lang) je slovenský ("sk")

globálne atribúty, napr. class, id, lang, style, môžu byť použité so všetkými elementami HTML špecifikácie

špecifické atribúty. napr. charset

HTML <head> <meta> <title>

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>
      Moja prvá stránka
    </title>
  </head>
  <body>
    <h1>
      Môj prvý nadpis
    </h1>
    <p>Môj prvý odsek</p>
  </body>
</html>
```

head element obsahuje meta informácie o dokumente

element <meta>
s atribútom charset="utf-8"

- kódovanie/znaková sada dokumentu

element <title>

- názov dokumentu
- je povinný, nesmie byť prázdny

HTML <body> <h1> <p>

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>
      Moja prvá stránka
    </title>
  </head>
  <body>
    <h1>
      Môj prvý nadpis
    </h1>
    <p>Môj prvý odsek</p>
  </body>
</html>
```

body element obsahuje telo dokumentu (viditeľný obsah stránky)

h1 element definuje nadpis prvej úrovne – najväčší

p element definuje odsek

Prázdné elementy

- elementy nemusia mať obsah

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="utf-8">
    <title>
      Názov dokumentu je
      obsah elementu
      title
    </title>
  </head>
  <body>
    ...
</html>
```

element, ktorý nemá obsah, je prázdny element (angl. empty element) napr.:

```
<meta charset="utf-8">
  NIE JE TU ČO DAŤ
</meta>
```

```
<img src= "logo.png">
```

v HTML(5) nemusí mať koncovú značku

HTML5 vs. XHTML

- HTML5 nevyžaduje, aby mali prázdne elementy koncovú značku
 - ak chceme striktnejšiu validáciu alebo, aby bol dokument párovateľný XML parsermi, musíte uzatvoriť každý element
 - použijeme lomítko pred koncovou lomenou zátvorkou
`<meta charset="utf-8" />`
- HTML značky nie sú case-sensitive
 - `<p>` je to isté ako `<P>`
 - W3C odporúča malé písmená (angl. lower case)

HTML5 má tolerantnú syntax

- tieto elementy sú ekvivalentné

```
<img src=logo.png alt="">
```

```

```

```
<IMG src="logo.png" alt="" />
```

- hodnoty atribútov musia byť v **úvodzovkách**, pokiaľ sa skladajú z **viac slov oddelených medzerou** (napr. názvy tried), alebo **obsahujú určité špeciálne znaky**

Atribúty - pravda/nepravda

- niektoré atribúty môžu mať hodnoty iba pravda/nepravda

```
<input type="checkbox" checked>
```

- v XHTML je potrebné písať

```
<input type="checkbox" checked="checked" />
```

- v HTML5 sú tieto elementy ekvivalentné

Atribúty - typ úvodzoviek

- dvojité úvodzovky sú v HTML zaužívané, ale môžu byť použité aj jednoduché
- niekedy je nutné použiť jednoduché úvodzovky:

```

```

- alebo naopak:

```

```

Nezabúdajme na koncovú značku

```
<body>  
  <p>This is a paragraph  
  <p>This is a paragraph  
</body>
```

- toto je **OK** vo všetkých moderných prehliadačoch, **koncová značka je voliteľná**
- **nespoliehajte sa**, môže to spôsobiť neočakávané chyby
- **OSOBNÉ PREFERUJEM** malé písmená, úvodzovky, neuzatváram prázdne elementy
- **zvoľte si systém**, ktorý vám vyhovuje, **pevne sa ho držte** – buďte konzistentní

W3C validátor

- umožňuje skontrolovať, či sú HTML (XHTML) dokumenty správne vytvorené
- dôležitý je `<!DOCTYPE`
 - ten určuje typ dokumentu
- <https://validator.w3.org/>

[UKÁŽKA]

Prehliadače – režimy zobrazovania

- **quirks** – emuluje neštandardné zobrazovanie prehliadača IE (potrebné pre zachovanie spätnej kompatibility)
- **štandardný režim** – moderné správanie, v súlade so štandardami
`<!DOCTYPE html>`
- **takmer štandardný režim** – štandardný režim s niekoľkými odchýlkami (tabuľky)

Zapína sa na základe DOCTYPE

- **budeme používať HTML5, a teda štandardný režim**

`<!DOCTYPE html>`

[Activating Browser Modes with Doctype](#)

Priklady: <http://strictquirks.nl/quirks/?mode=c#emptycel>

Prehliadače – vykreslovacie jadrá

- Chrome – Blink, na iOS WebKit
- Opera – Blink, predtým Presto
- Firefox – Mozilla Gecko, na iOS WebKit
- Safari – WebKit
- ~~• Edge – EdgeHTML, na iOS WebKit, na Adroide Blink~~
- Edge2019 – Blink, na iOS WebKit
- ~~• IE – Trident~~

Pozn. Blink fork Webkitu

- <http://gs.statcounter.com/browser-market-share>

Testovať, testovať, testovať...

- množstvo zariadení, veľa prehliadačov, veľa aktívnych verzií prehliadačov
- na začiatku, počas, na konci, kedykoľvek bude príležitosť
- testovanie zaberie 40-50% času
- najlepšie reálne zariadenia
 - keď sa už nedá aj emulátory sú lepšie ako nič

Prehliadače - predvolené štýly

- prehliadače majú rôzne predvolené štýly
 - [CSS reset](#), [types of CSS resets](#)
- [nemôžete si byť istí, ako bude HTML vykreslené](#)
 - napr. prehliadače pridávajú biele miesto pred a za odsek
- malé, veľké obrazovky, veľkosť okna (bloku) budú produkovať rôzne výstupy

```
<p>Toto je prvý odsek.</p>
```

Toto je prvý odsek.

```
<p>Toto je druhý odsek.</p>
```

```
<p>
```

Toto je druhý odsek.

```
  Toto je tretí odsek,
```

```
  <br/>ktorý obsahuje aj
```

```
  zlom riadku.
```

Toto je tretí odsek,

ktorý obsahuje aj zlom riadku.

```
</p>
```

Prehliadače - extra oddelovače

- **nemôžete ovplyvniť výstup pridaním viacerých medzier alebo riadkov**
- **odstraňujú extra medzery a riadky, odsadenie cez tabulátor**

```
<p>
Toto je odsek,
ktorý obsahuje
v html dokumente
niekoľko riadkov,
ale prehliadač ich
ignoruje.
</p>
```

```
<p>
Toto je      odsek, ktorý
obsahuje
v            html
dokumente   veľa medzier,
ale         prehliadač
ich        ignoruje.
</p>
```


Prehliadače - extra oddelovače

- **nemôžete ovplyvniť výstup pridaním viacerých medzier alebo riadkov**
- **odstraňujú extra medzery a riadky, odsadenie cez tabulátor**

Toto je odsek, ktorý obsahuje v html dokumente niekoľko riadkov, ale prehliadač ich ignoruje.

Toto je odsek, ktorý obsahuje v html dokumente veľa medzier, ale prehliadač ich ignoruje.

Sémantika

- náuka, ktorá študuje význam slov a fráz jazyka (angl. semantics)
- sémantické elementy – elementy s významom
- sémantické elementy jednoznačne určujú ich význam/obsah (stroju-prehliadaču, človeku)
 - `<p>` odsek (angl. paragraph)
 - `<h1>` nadpis (angl. heading)
- nesémantické elementy nehovoria nič o ich obsahu
 - `<div>` blok (angl. document division)
 - ``

Skupiny HTML elementov

- **HTML elementy je možné rozdeliť do skupín:**
 - Hlavný koreň HTML dokumentu
 - Metadáta dokumentu
 - Koreň tela dokumentu
 - Elementy rozdeľujúce obsah do logických sekcií
 - Elementy textového obsahu určujúce účel alebo štruktúru
 - Elementy určujúce sémantiku textu na úrovni riadkov (slov, fráz)
 - Obrázky a multimédiá
 - Elementy umožňujúce vložiť ďalší obsah
 - Elementy umožňujúce vložiť skripty
 - Elementy vyznačujúce zmeny v dokumente
 - Tabuľky
 - Formuláre
 - Interaktívne elementy
 - Webové komponenty

Skupiny HTML elementov

- **HTML elementy je možné rozdeliť do skupín:**
 - Hlavný koreň HTML dokumentu
 - **Metadáta dokumentu**
 - Koreň tela dokumentu
 - Elementy rozdeľujúce obsah do logických sekcií
 - Elementy textového obsahu určujúce účel alebo štruktúru
 - Elementy určujúce sémantiku textu na úrovni riadkov (slov, fráz)
 - Obrázky a multimédiá
 - Elementy umožňujúce vložiť ďalší obsah
 - Elementy umožňujúce vložiť skripty
 - Elementy vyznačujúce zmeny v dokumente
 - Tabuľky
 - Formuláre
 - Interaktívne elementy
 - Webové komponenty

Metadáta dokumentu

HTML <link>

- k aktuálnemu dokumentu pripája a externý zdroj
 - napr. odkaz na externé štýly (CSS)
- dôležité atribúty
 - rel – vzťah aktuálneho dokumentu a pripojeného zdroja
 - href – URL pripojeného zdroja
 - type – MIME typ obsahu, v HTML5 prednastavený css

```
<link rel="stylesheet" type="text/css" href="style.css">
```

```
<link rel="stylesheet" href="style.css">
```

HTML <meta>

- reprezentuje ďalšie metadáta
 - také, ktoré nie sú reprezentované inými elementami definujúcimi metadáta dokumentu, napr. <title>, <link>, <style>, ...
- dôležité atribúty:
 - name – názov metadát
 - content – hodnota pre atribút určený name

```
<meta name="description" content="Webové  
technológie, kurz zameraný na...">
```

```
<meta name="keywords" content="HTML,CSS,Sass">
```

```
<meta name="author" content="Eduard Kuric">
```

HTML <style>

- obsahuje štýly dokumentu
 - ak nie je určený typ, očakáva sa CSS

```
<style type="text/css">  
  body {  
    color: red;  
  }  
</style>
```


HTML <title>

- definuje názov HTML dokumentu
- ak nebude dokument obsahovať element `title` bude nevalidný
- prehliadače v názve karty, obľúbené
- vyhľadávače, indexovanie stránok, výsledky vyhľadávania

Skupiny HTML elementov

- **HTML elementy je možné rozdeliť do skupín:**
 - Hlavný koreň HTML dokumentu
 - Metadáta dokumentu
 - Koreň tela dokumentu
 - **Elementy rozdeľujúce obsah do logických sekcií**
 - Elementy textového obsahu určujúce účel alebo štruktúru
 - Elementy určujúce sémantiku textu na úrovni riadkov (slov, fráz)
 - Obrázky a multimédiá
 - Elementy umožňujúce vložiť ďalší obsah
 - Elementy umožňujúce vložiť skripty
 - Elementy vyznačujúce zmeny v dokumente
 - Tabuľky
 - Formuláre
 - Interaktívne elementy
 - Webové komponenty

Elementy rozdeľujúce
obsah do logických sekcií

HTML nadpisy

- headings

`<h1>Nadpis 1</h1>`

`<h2>Nadpis 2</h2>`

`<h3>Nadpis 3</h3>`

`<h4>Nadpis 4</h4>`

`<h5>Nadpis 5</h5>`

`<h6>Nadpis 6</h6>`

Nadpis 1

Nadpis 2

Nadpis 3

Nadpis 4

Nadpis 5

Nadpis 6

pozn.: `<head>` element nemá nič s nadpismi, zapúzdruje metadáta dokumentu, je umiestnený medzi `<html>` a `<body>`

HTML nadpisy /2

- slúžia na štruktúrovanie html stránok/html dokumentov
- vyhľadávače (Google) používajú nadpisy na indexovanie štruktúry a obsahu stránok
- poskytujú vodítka, používateľom umožňujú rýchlejšie skenovanie obsahu
- **používajte nadpisy pre nadpisy, nie na to, aby bol text väčší, alebo tučnejší**

HTML4 vs. HTML5

- veľa webov obsahuje:
 - `<div id="nav">`
 - `<div class="header">`
 - `<div id="footer">`
 - na indikovanie navigácie, hlavičky, pätičky
- vývojári nemali možnosti, ako lepšie štruktúrovať a označovať obsah
- HTML5 prináša **nové rozdeľujúce elementy**, ktoré definujú rôzne oblasti stránky

HTML5 rozdeľujúce elementy

HTML4

`<div id="header">`

`<div id="menu">`

`<div id="content">`

`<div class="article">`

`<div id="footer">`

HTML5

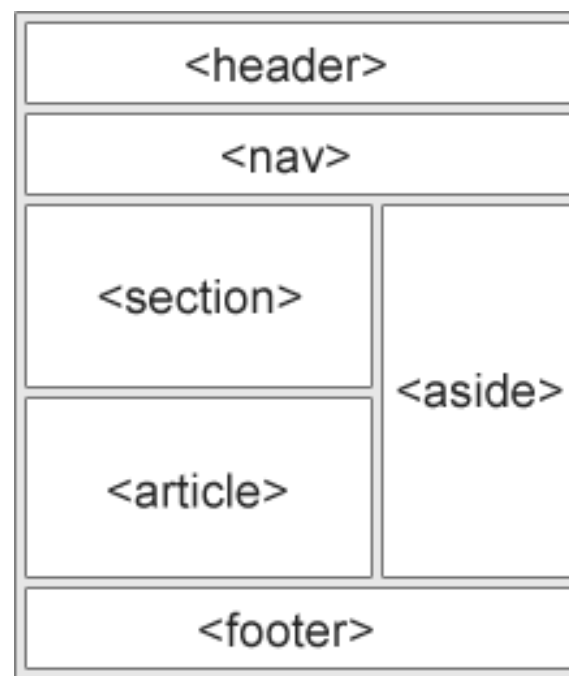
`<header>`

`<nav>`

`<section>`

`<article>`

`<footer>`



HTML `<article>` `<section>`

- `<article>` **nezávislá časť dokumentu, samostatný obsah**
 - napr. príspevok blogu, príspevok diskusného fóra, komentár používateľa, novinový článok
- `<section>` **tématické zoskupenie obsahu, zvyčajne s nadpisom**
 - napr. kapitola knihy, karta v dialógovom okne

<https://www.hongkiat.com/blog/html5-section-article-elements/>

HTML<nav> <aside>

- **<nav> hlavný blok navigačných odkazov**
 - obsahuje odkazy na ďalšie dokumenty, alebo v rámci dokumentu, napr. menu, TOC
- **<aside> oblasť stránky, ktorá okrajovo súvisí s okolitým obsahom**
 - napr. postranný stĺpček v novinovom článku

```
<p>
```

```
S rodinou sme cez leto navštívili Minieurópu  
v Bruseli.
```

```
</p>
```

```
<aside>
```

```
<h4>Minieurópa</h4>
```

```
<p>Minieurópa je ...</p>
```

```
</aside>
```

HTML<footer> <header> <hgroup>

- **definujú oblasti vo vnútri rozdeleného obsahu**
- <footer> pätička dokumentu, alebo oblasti
 - zvyčajne metadáta o sekcii, v ktorej sa nachádza, napr. informácie o autorovi
- <header> hlavička dokumentu, alebo oblasti
 - zvyčajne obsahuje nadpisy
- <hgroup> zoskupuje viacero nadpisov, napr. nadpis s podnadpisom

HTML<hgroup>

- ~~bol odstránený z HTML5 (W3C) špecifikácie~~
 - ale je vo verzii od WHATWG
- je čiastočne implementovaný vo väčšine prehliadačov
~~(asi tak ľahko neodíde)~~
- jeho význam
 - ovplyvniť ako budú nadpisy zobrazené [algoritmom na vytvorenie osnovy dokumentu](#)
 - ale HTML5 špecifikácia algoritmu nie je implementovaná
 - význam `hgroup` je preto v súčasnosti skôr teoretický, ako praktický
- ~~W3C špecifikácia poskytuje rady, ako označiť podnadpisy bez použitia `hgroup`~~

Osnova dokumentu v HTML4

- nové **rozdeľujúce elementy vytvárajú oblasti dokumentu**, odhaľujú hierarchiu vnútorného obsahu
- v HTML 4 túto úlohu preberali nadpisy
 - element `<h1>` bol jedinečný pre celú stránku

```
<h1>Automobily</h1>  
  <h2>Volkswagen</h2>  
    <h3>Golf</h3>  
    <h3>Passat</h3>  
    ...  
  <h2>Audi</h2>  
  ...
```

```
1. Automobily  
  1.1 Volkswagen  
    1.1.1 Golf  
    1.1.2 Passat  
    ...  
  1.2 Audi  
  ...
```

implicitné rozčleňovanie pomocou nadpisov

Osnova dokumentu v HTML5

- v HTML 5 vytvárajú osnovu rozdeľujúce elementy

```
<h1>Automobily</h1>
<section>
  <h2>Volkswagen</h2>
  <article>
    <h3>Golf</h3>
    ...
  </article>
  <article>
    <h1>Passat</h1>
    ...
  </article>
</section>
<section>
  <h1>Audi</h1>
  ...
</section>
```

```
1. Automobily
  1.1 Volkswagen
    1.1.1 Golf
    1.1.2 Passat
    ...
  1.2 Audi
  ...
```

explicitné rozčleňovanie
pomocou rozdeľujúcich elementov
<section> a <article>

Osnova dokumentu v HTML5 /2

- **každý rozdeľujúci element vytvára novú oblasť**
- každá oblasť môže mať **svoju vlastnú hierarchiu nadpisov**
- **každá oblasť môže obsahovať** - okrem `<hgroup>`, `<header>`, `<footer>`, rozdeľujúcich koreňov (napr. `<blockquote>`) – **d ďalšie oblasti**
- **HTML5 nedostatočne definuje rozdiely medzi** `<section>` a `<article>`

HTML `<article>` vs. `<section>`

- `<article>` samostatná (nezávislá) oblasť súvisiacich elementov (a newspaper article, a blog entry, a comment)
- `<section>` oblasť (tématicky) súvisiacich elementov

`<article>`

```
  <h1>Volkswagen</h1>
```

```
  <article>
```

```
    <h1>Golf</h1>
```

```
    ...
```

```
  </article>
```

```
  <article>
```

```
    <h1>Passat</h1>
```

```
    ...
```

```
  </article>
```

```
</article>
```

`<section>`

```
  <h1>Volkswagen</h1>
```

```
  <article>
```

```
    <h1>Golf</h1>
```

```
    ...
```

```
  </article>
```

```
  <article>
```

```
    <h1>Passat</h1>
```

```
    ...
```

```
  </article>
```

```
</section>
```

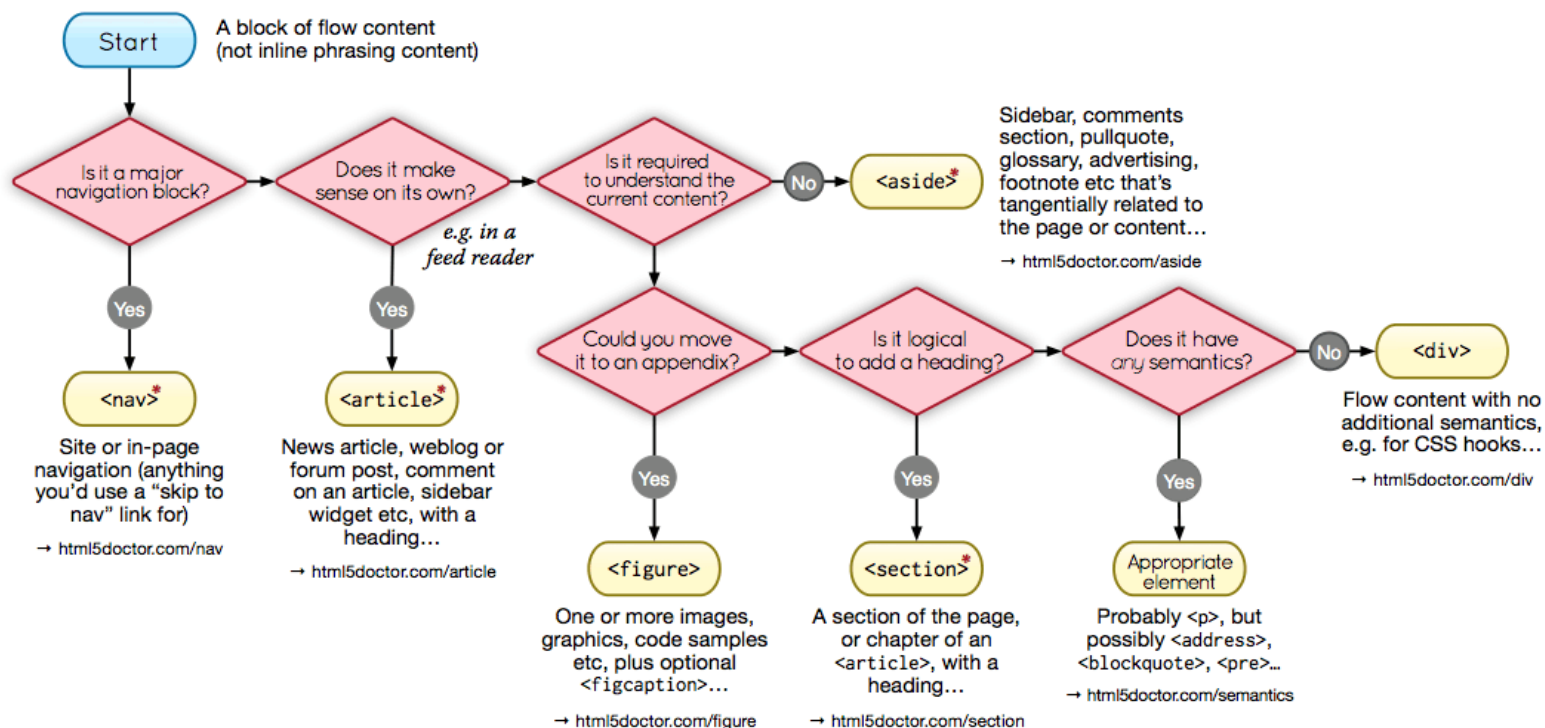
Diagram s HTML5Doctor napovie



HTML5 Element Flowchart

Sectioning content elements and friends

By @riddle & @boblet
www.html5doctor.com



* Sectioning content element

These four elements (and their headings) are used by HTML5's outlining algorithm to make the document's outline
→ html5doctor.com/outline

2011-07-22 v1.5

For more information:
www.html5doctor.com/semantics

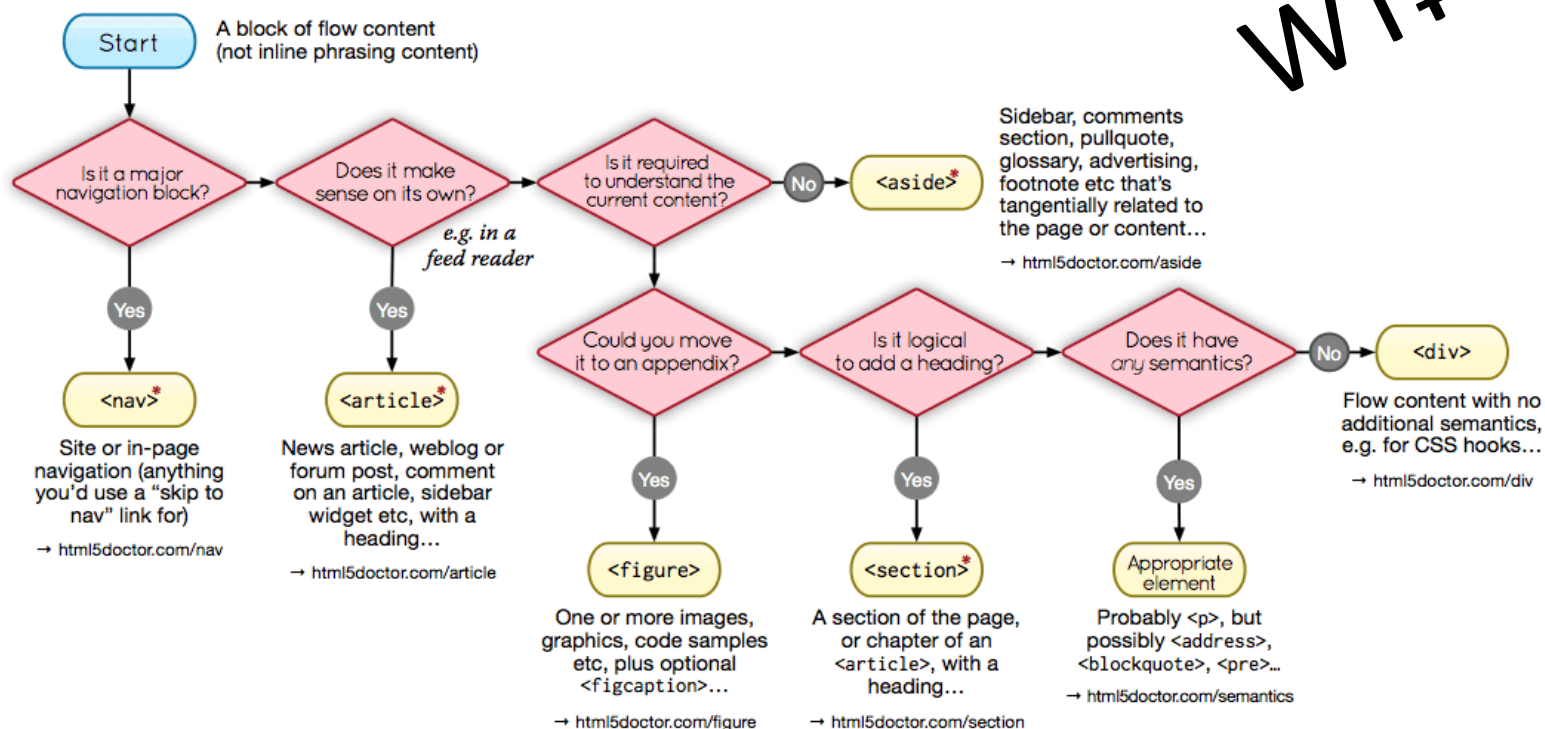
Diagram s HTML5Doctor napovie



HTML5 Element Flowchart Sectioning content elements and friends

By @riddle & @boblet
www.html5doctor.com

WTF?



* Sectioning content element

These four elements (and their headings) are used by HTML5's outlining algorithm to make the document's outline
→ html5doctor.com/outline

2011-07-22 v1.5
For more information:
www.html5doctor.com/semantics

HTML <article> vs. <section>

- uvažujme článok, ktorý má na konci komentáre používateľov

```
<article>
  <h1>Článok</h1>
  <p>...</p>
  <article>
    <article>
      <h1>Príspevok1</h1>
      <p></p>
    </article>
  </article>
</article>
```

```
<article>
  <h1>Článok</h1>
  <p>...</p>
  <section> // comments
    <section>
      <h1>Príspevok1</h1>
      <p></p>
    </section>
  </section>
</article>
```

Nejasná definícia <article>

- Luke Stevens vo svojej knihe *The Truth About HTML5* píše:

Špecifikácia zlyhá, pokiaľ nechá na vás, aby ste si niečo domýšľali. Podstatou špecifikácie je presne špecifikovať, čo by ste mali robiť. V tomto prípade je ju ale možné interpretovať rôzne, nemá jasný prínos a opakuje existujúcu funkčnosť.

- **Ľudia budú tieto elementy používať zle, pokiaľ nebude presnejšia definícia**

HTML5 online zdroje

- HTML5 je “živá” špecifikácia
- <https://html.spec.whatwg.org/>
- <https://www.w3schools.com/>
- <https://developer.mozilla.org/en-US/docs/Web>