

- Řádkové elementy (inline elements, phrasing content)
 - Neovlivňují tok textu (zalomení)
- Blokové elementy (block-level elements)
 - Způsobují zalomení řádku
 - Mohou způsobit odsazení, vynechání místa apod
 - V HTML 5 specifikaci rozděleny na více kategorií

a) Významové odlišení částí textu

<code></code>	<i>zvýraznění – důraz</i>
<code></code>	zvýraznění – důležitost
<code></code>	zvýraznění – obecné
<code><i></code>	<i>odlišený text (styl, jazyk, apod.)</i>
<code><u></code>	vizuální <u>označení části textu</u>
<code><small></code>	méně významná část textu (pod čarou)
<code><s></code>	část textu, která již neplatí
<code><abbr></code>	zkratka nebo akronym, např. OSN
<code></code>	bez významu, možné atributy
<code>
</code>	zalomení řádku (prázdný)

b) vzorce, programy apod.

<code><var></code>	<i>proměná v programu, vzorci</i>
<code><sup></code>	horní _{index}
<code><sub></code>	dolní _{index}
<code><code></code>	výpis kódu
<code><dfn></code>	nový <i>termín</i>
<code><cite></code>	<i>citace, odkaz na zdroj (název, jméno, apod.)</i>
<code><q></code>	„ocitovaný obsah“ (doslovně)

Relativita odkazů

- Na jiný server
 - `href="http://www.vutbr.cz/"`
 - `href="http://www.vutbr.cz/doc/info.html"`
- Na stejný server
 - Absolutní `href="/doc/info.html"`
 - Relativní k umístění souč. dokumentu
 - `href="next.html"`
 - `href="obsah/clanek.html"`
 - `href="../index.html"`

Základní struktura obsahu

- **article** – článek, příspěvek (samostatně použitelná část obsahu)
- **section** – kapitola článku, příspěvku, apod.
- **aside** – část, která se jen vzdáleně vztahuje k hlavnímu obsahu
- **header** – záhlaví dokumentu, článku, apod.
- **footer** – zápatí
- **nav** – část obsahující navigaci
- **h1** - **h6** – nadpisy
- **div** – obecný blok bez dalšího významu (pro skriptování, styly, apod.)

Strukturování elementů

- Řádkové elementy lze libovolně vnořovat
 - `y = <i>e</i>^{2<i>x</i>}`
 - $y = e^{2x}$
- Řádkový element **nesmí** obsahovat blokové
 - V ``tučném textu `<p>`odstavec`</p>` nemůže být``.
- Blokový element může obsahovat všechny řádkové
- Různé blokové elementy mohou obsahovat některé jiné blokové elementy
 - Určeno specifikací
 - Přehledně uvedeno [v příloze](#)

Odstavce

- Odstavec začíná značkou `<p>`
- Ukončující značka `</p>` není povinná
 - Pokud chybí, je odstavec ukončen začátkem dalšího blokového elementu
 - Doporučuje se odstavce ukončovat
- Uvnitř odstavce již mohou být pouze řádkové elementy
- Obvykle se zobrazí vertikálně odsazený od okolního textu

Vnořování seznamů

- Seznam `` a `` může obsahovat jen elementy ``
- Položka seznamu `` může obsahovat libovolné blokové i řádkové elementy
- Je tedy možné i vnořovat seznamy

```
<ul>
<li>Jméno může být
  <ul>
    <li>Jan</li>
    <li>Jana</li>
  </ul>
</li>
<li> ... </li>
</ul>
```

Položky definičního seznamu

- Seznam `<dl>` může obsahovat jen elementy `<dt>` a `<dd>` (v libovolné posloupnosti)
- Termín `<dt>` nemůže obsahovat sekce a nadpisy
- Definice `<dd>` může obsahovat libovolné blokové i řádkové elementy

Obrázky

```

```



- Obrázek je **řádkový** element
- Nepárová značka (element je vždy prázdný)

Vložené SVG

- Přímo v HTML dokumentu

```
<svg height="120"
  xmlns="http://www.w3.org/2000/svg">
  <circle cx="50" cy="50" r="50" fill="red" />
  <rect x="80" y="20" height="60" width="60"
    style="stroke:none; fill:#00ff00;
    fill-opacity: 0.5;" />
</svg>
```



Globální atributy

- Atributy použitelné pro všechny elementy

title= text	Titulek elementu
lang= cs en ...	Jazyk obsahu
tabindex= 0-n	Pořadí aktivace přes TAB
dir= rtl ltr	Směr psaní textu
hidden	Skrytý prvek
id= jmeno	Jednoznačný identifikátor
class= jmeno	Použití pro kaskádové styly (CSS)
style= styl	

Generické elementy

- Dva obecné elementy:
 - `` - obecný řádkový element
 - `<div>` - obecný blokový element
- Vymezují část řádku nebo blok s blíže neurčeným speciálním významem
 - Bližší určení lze doplnit např. pomocí atributů **class** nebo **title**
- Obvykle pro definici vzhledu v kombinaci s CSS

Strukturované selektory

- Vychází ze stromové struktury HTML dokumentu
- Čteme nejlépe odzadu
- **s1 s2 { }**
 - Element odpovídající **s2**, který je **potomkem** elementu **s1**
 - Např. **.vyrazne strong { }**
- **s1 > s2 { }**
 - Element odpovídající **s2**, který je **synovským uzlem** elementu **s1**
 - Např. **a > em { }**
- **s1 + s2 { }**
 - Element odpovídající **s2**, který je **předcházen** elementem **s1**
 - Např. **table + p { }**

Selektory s atributy

- **značka[atribut] { }**
 - Značka s nastaveným atributem **atribut**
 - Např. **p[class] { }**
- **značka[atribut="hodnota"] { }**
 - Značka s atributem **atribut** nastaveným na hodnotu **hodnota**
 - Např. **p[class] { }**
- **značka[atribut~="hodnota"] { }**
 - Značka s atributem **atribut**, který obsahuje hodnoty oddělené mezerami, z nichž jedna je rovna **hodnota**
 - Např. **p[class~="red"] { }**
- **značka[atribut|="hodnota"] { }**
 - Hodnota začíná daným řetězcem nebo za řetězcem následuje pomlčka.

Délky v CSS

- U různých vlastností v CSS - šířky, délky, pozice, ...
- Zápis číslem a jednotkou, např. **10mm** (bez mezery)
- Absolutní jednotky:
 - pixely - **px**
 - milimetry - **mm**
 - centimetry - **cm**
 - palce - **in** (1in = 2.54cm)
 - typografické body, pica - **pt, pc** (1pc = 12pt, 1pt = 1/72in)
- Relativní jednotky
 - čtverčik (eM) - **em** - šířka velkého M
 - x-height - **ex** - výška malého x

Barvy v CSS

- Pojmenované barvy
 - | | | | | |
|--------|---------|--------|--------|-------|
| maroon | red | orange | yellow | olive |
| purple | fuchsia | white | lime | green |
| navy | blue | aqua | teal | |
| black | silver | gray | | |
 - CSS3 zavádí [rozšířenou sadu barev](#)
- RGB model - různé způsoby zápisu
 - `rgb(100%,50%,20%)` **Ukázka**
 - `rgb(255,10,127)` **Ukázka**
 - `#a0ff00` **Ukázka**
 - `#afb` **Ukázka**
- Barva textu:
 - `color: #aa7f5b`

Pozadí elementu

- Barva pozadí
 - `background-color: transparent`
 - `background-color: red`
- Obrázek na pozadí
 - `background-image: none`
 - `background-image: url('abstract.png')`
- Opakování obrázku
 - `background-repeat: repeat`
 - `background-repeat: no-repeat`
 - `background-repeat: repeat-x`
 - `background-repeat: repeat-y`

Pozadí elementu (II)

- Pozice obrázku na pozadí (v rámci elementu)
 - `background-position: 20% 5%` (hor., vert.)
 - `background-position: 2cm 1cm`
 - `background-position: top left | center right`
 - `background-position: center`
- Chování při scrollování
 - `background-attachment: scroll`
 - `background-attachment: fixed`

Vlastnost background

- Shrnuje všechny vlastnosti **background-xxxx**

```
background: red url('pozadi5.gif')
top left no-repeat;
```

- Neuvedené vlastnosti opět nastavuje na implicitní hodnoty

Druh písma (II)

- Je nutné vždy specifikovat alternativy
- Pokud název obsahuje mezery, musí být v úvozovkách
- Poslední alternativa musí být generická

```
p.c1 { font-family: Verdana, Arial, sans-serif; } p.c2 { font-family: 'Times New Roman', Times, serif; } p.c3 { font-family: 'Courier New', Courier, monospace; }
```

Kaskáda pravidel

- Vezmeme všechna pravidla, jejichž selektor odpovídá danému elementu
- Pokud je některá vlastnost definována opakovaně, vybereme hodnotu podle následujících kritérií:
 1. Zdroj pravidla
 - Různé zdroje jsou různě významné
 2. Seřadíme podle specifičnosti selektoru
 - Čím konkrétnější selektor, tím větší prioritu má pravidlo
 3. Seřadíme podle pořadí specifikace
 - Má-li pravidlo stejný zdroj, prioritu a specifičnost pak vyhrává **později** uvedené pravidlo

Zdroj pravidla

- Význam jednotlivých pravidel podle zdroje od nejméně významného k nejvíce významnému je stanoven takto:
 1. Styly prohlížeče
 2. Normální styly uživatele
 3. Normální styly autora dokumentu
 4. Prioritní (!important) styly autora dokumentu
 5. Prioritní styly uživatele
- Autor stránek tedy může uvažovat pouze 3 a 4

Specifičnost

- Specifičnost je číslo **abcd**, kde
 - **a=1** pokud se jedná o inline styl, jinak **a=0**
 - **b** je počet atributů **id** v selektoru
 - **c** je počet tříd v selektoru
 - **d** je počet názvů elementů v selektoru
- `p.uvod span { }`: 0 - 0 - 1 - 2 ~ **12**
- `p span { }`: 0 - 0 - 0 - 2 ~ **2**

Hodnota **inherit** - Příklad

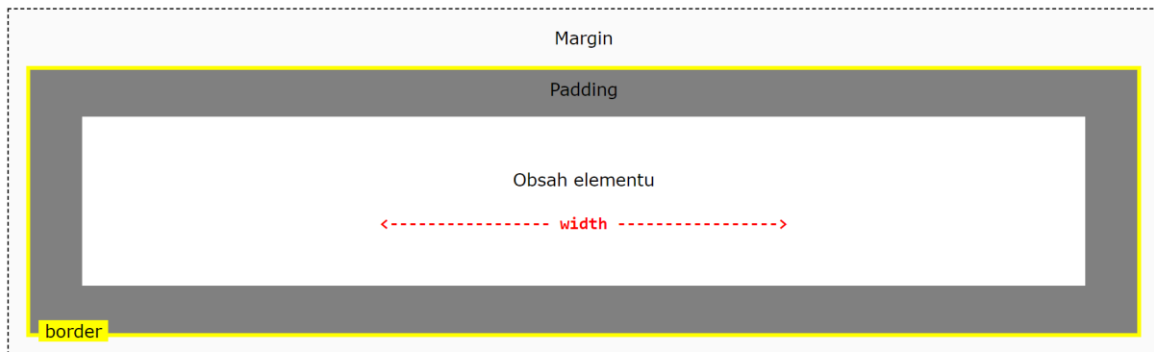
Menu

První odstavec

Druhý odstavec

```
#menu { border: 3px #99ff99 solid; }  
#menu p { border: inherit; }
```

Box model



Vnější okraj (margin)

- Po jednotlivých stranách
 - `margin-top:`, `margin-bottom:`
 - `margin-left:`, `margin-right:`
- Najednou
 - `margin: 2em 1em 3em 2em` - top, right, bottom, left
 - `margin: 2em 1em 1em` - top, right&left, bottom
 - `margin: 2em 1.5em` - top&bottom, right&left
 - `margin: 1em` - všechny

Vnitřní okraj (padding)

- Po jednotlivých stranách
 - **padding-top:**, **padding-bottom:**
 - **padding-left:**, **padding-right:**
- Najednou
 - **padding:** 2em 1em 3em 2em - top, right, bottom, left
 - **padding:** 2em 1em 1em - top, right&left, bottom
 - **padding:** 2em 1.5em - top&bottom, right&left
 - **padding:** 1em - všechny

Hodnoty auto

- Vlastnosti **margin**, **width** a **height** mohou mít zvláštní hodnotu **auto**
 - Pro **width** a **height** výchozí
 - U **margin** je výchozí hodnota 0
- Skutečné hodnoty vlastností s hodnotou **auto** se dopočítají podle ostatních hodnot
- Způsob výpočtu se liší podle způsobu rozmístění objektů

Způsob rozmístění objektů

- Normální tok (normal flow)
 - Tzv. **in-flow** elementy
 - Elementy se rozmísťují v pořadí jak jsou uvedeny v HTML kódu
 - Řádkové elementy na řádek, blokové pod sebe
- Plovoucí bloky (floating blocks)
 - Elementy obtékané in-flow obsahem následujícím po nich
 - Vlastnost **float**
- Pozicované bloky
 - Pozice bloku na stránce je explicitně určena
 - Vlastnost **position**

Šířka bloku – normální tok

- **Blok vždy zabírá celou šířku obsahu bloku, ve kterém je obsažen**
 - U kořenového elementu celou šířku obrazovky
- Platí tedy:
 $\text{margin-left} + \text{border-left-width} + \text{padding-left} + \text{width} + \text{padding-right} + \text{border-right-width} + \text{margin-right} = \text{šířka_obsahujícího_bloku}$
- Z toho lze dopočítat případné hodnoty **auto**
- Pokud žádná hodnota není **auto**, je jedna hodnota nadbytečná => ignoruje se hodnota **pravého okraje**

Výpočet hodnot auto

- Pokud je pouze jedna hodnota **auto**, dopočte se z předchozí rovnice
- Pokud šířka není zadána (**width=auto**), hodnoty **auto** u okrajů se interpretují jako 0
- Pokud levý i pravý okraj mají hodnotu **auto**, jejich výsledné velikosti jsou stejné => **centrování bloku v obsahujícím bloku**

Výška bloku

- Je-li **height=auto**, určí se výška tak, aby se vešly všechny obsažené elementy
 - Standardně se uvažují jen in-flow elementy
 - Lze ovlivnit nastavením **overflow**
- U horního a spodního okraje se hodnota **auto** považuje za 0

Spojování okrajů

- **Horizontální** okraje se **nikdy nespojují**
- **Vertikální** okraje se spojují
 - U dvou pod sebou ležících **neplovoucích** bloků
 - U dvou vzájemně vnořených bloků (horní nebo dolní okraj)
 - Horní okraj pouze pokud vnitřní objekt má **clear: none**
 - Výsledný okraj je **maximem** spojovaných okrajů

Šířka a výška

- Vlastnosti **width** a **height** určují šířku/výšku boxu
 - Hodnota **auto** nebo zadaná délka
- Vlastnost **box-sizing** určuje, co daný rozměr zahrnuje
 - **box-sizing: content-box** – **width** a **height** nastavuje rozměry **obsahu**
 - **box-sizing: border-box** – **width** a **height** nastavuje rozměry **včetně border a padding**
- **box-sizing** se nedědí, je třeba nastavit pro všechny dotčené boxy. Např:

```
.column div {  
  box-sizing: border-box;  
}
```

Plovoucí objekty

- Objekty obtékané okolním obsahem (zleva nebo zprava)
- Vytvoření polovoucího objektu:
 - **float: left**
 - **float: right**
- Plovoucí elementy se stávají automaticky **blokovými**
- Plovoucí objekt je
 - Objekt je vyjmut z normálního toku (nijak se neprojeví na původním místě)
 - Odsunut ke kraji rodičovského prvku
- Mění se postup výpočtu šířky
 - Nejlépe je mít nastavenou šířku
 - Pokud není, určí se automaticky

Pozicování

- Ruční nastavení polohy objektu na stránce
- Absolutní
 - Umístění objektu na konkrétní místo stránky nezávisle na ostatním textu
- Relativní
 - Posunutí oproti normální poloze objektu
- Vlastnost **position**
 - **position: static** – normální stav
 - **position: absolute** – absolutní pozicování
 - **position: relative** – relativní pozicování
 - **position: fixed** – fixní pozice

Rozměry pozicovaných elementů

- Platí rovnice:
 $\text{left} + \text{margin-left} + \text{border-left-width} + \text{padding-left} + \text{width} + \text{padding-right} + \text{border-right-width} + \text{margin-right} + \text{right} = \text{šířka_obsahujícího_bloku}$
- Hodnoty **auto** pro **left** a **top** vycházejí ze **statické pozice**
- Pravidla pro dopočítávání hodnot ve [specifikaci](#)
- Analogicky pro výšku

Překrývání objektů

- Dva objekty se mohou překrývat (relativní i absolutní pozicování)
- Normální stav: později uvedený objekt překrývá dříve uvedený objekt
- Lze změnit pomocí **z-index**
 - **z-index: 10**
 - Číslo udává Z souřadnici objektu (čím vyšší tím „výše“ se objekt nachází)
 - Pokud není uvedeno, bere se **0**
 - Platí jen pro pozicované elementy

Viditelnost objektů

- Dvě možnosti
 1. **display: none**
 - Prvek se při renderování zcela ignoruje
 2. **visibility: hidden**
 - Prvek zabírá místo na stránce, jako by byl zobrazen, ale nezobrazí se

```
table { border-collapse: separate }
```

Jméno	Příjmení
Jan	Novák
Josef	Dvořák

```
table { border-collapse: collapse }
```

Jméno	Příjmení
Jan	Novák
Josef	Dvořák

Typ odrážky

- Vlastnost `list-style-type`
- Aplikace na element ``

```
ul li { list-style-type: square; }
```

- Nečíslované seznamy:
 - `list-style-type: disc`
 - `list-style-type: circle`
 - `list-style-type: square`

Typy kurzoru

auto	crosshair	default
hand	pointer	move
text	wait	help

nw-resize	n-resize	ne-resize
w-resize		e-resize
sw-resize	s-resize	se-resize

- Ukazovací kurzor spolehlivě:
 - `cursor: pointer;` `cursor: hand`

Pseudoprvky

- Abstraktní označení některých prvků dokumentu nebo jejich stavu
- Definují vzhled částí dokumentu
 - **Pseudotřídy**: doplňkové charakteristiky elementů
 - Např. linky navštívené, nenavštívené
 - Specifičnost jako pro třídy
 - **Pseudoelementy**: části elementů nad rámec značkování
 - První řádek, první písmeno, ...
 - Specifičnost jako pro jména elementů
- Zápis pomocí :
 - `p:first-letter { font-size: 20px; }`

Pseudotřídy: stylování odkazů

- Statické
 - Nenavštívený odkaz:
`a:link { color: green; }`
 - Navštívený odkaz:
`a:visited { color: red; }`
- Dynamické
 - Přejetí myší:
`a:hover { ... }`
 - Aktivace odkazu:
`a:active { ... }`
 - Zaměření klávesnice:
`a:focus { ... }`

Pseudoelementy: první písmena a řádky

- Element **:first-line**
 - Označuje první řádek libovolného elementu
 - Např. `p:first-line` - první řádek odstavce
 - Nelze předem určit, záleží na šířce okna, velikosti písma, ...
- Element **:first-letter**
 - Označuje první písmeno libovolného elementu
 - Např. `p:first-letter` - první písmeno odstavce

Absolutní pozicování (II)

- Absolutní pozicování k nadřazenému prvku
 - Nadřazený prvek musí mít nastaveno *position: relative*
 - [Příklad](#)

```
#content {  
    width: 80%; margin: auto;  
    position: relative;  
}  
#menu {  
    width: 140px;  
    position: absolute;  
    right: -70px; top: 30px;  
}
```

Obtékání

- Levý prvek je plovoucí vlevo
- Pravý prvek obtéká
- [Příklad](#)

```
#levy {  
    width: 5em;  
    float: left;  
}  
#pravy {  
    width: 10em;  
}
```

- Potíže při rozdílných výškách: [příklad](#)

Obtékání (II)

- Řešení: levý okraj
- [Příklad](#)

```
#levy {  
    width: 5em;  
    float: left;  
}  
#pravy {  
    width: 10em;  
    margin-left: 5em;  
}
```

Speciální hodnoty display

- Inline block **display: inline-block**
 - Bloky rozmísťované jako řádkové elementy
- Prvky tabulek **display: table-cell**
 - Vytvoří **anonymní tabulku**
 - Obdobně **table-row, table, inline-table**
- [Příklady](#)

Flexbox

- Nástroj pro rozmísťování obsahu v lokální části stránky
 - Formuláře, položky seznamů, ...
- Rodičovský element slouží jako **flex kontejner**
 - Nastavením **display: flex**
 - Definuje se **hlavní osa** a **vedlejší osa**
- Synovské elementy se stávají **flex položkami**

Grid layout

- Rodičovský element slouží jako **grid kontejner**
 - Nastavením **display: grid**
 - Nastaví se počet řádků a sloupců mřížky
 - Explicitní (template) nebo implicitní (auto)
 - Délková jednotka **fr** (fraction)
 - Možnost vytvoření pojmenovaných oblastí (více buněk mřížky)
- Pro synovské elementy se nastaví poloha v mřížce
 - **grid-row-start**, **grid-row-end** (**grid-row**)
 - **grid-column-start**, **grid-column-end** (**grid-column**)
 - nebo **grid-area** ([demo](#))
- [Popis a příklad na MDN](#)
- [Podrobnější popis](#)