

ING.Pavel Drgo,29.april 2017,15:21

1.Pozícia elementu

Pomocou CSS 3 vlastnosti *position* nastavujeme typ pozície elementu. Element môže byť vložený normálne do obsahu stránky, môže z neho byť však vytrhnutý a umiestnený na presné súradnice. Dokonca môže byť prichytený k niektorej z hrán okna prehliadača.

Hodnoty pozicie elementu

* **static** (predvolené) - Element je zobrazený na udanej pozíciu s ohľadom na ďalšie obsah dokumentu.
* **absolute** - Element je zobrazený na udanej pozíciu a je vytrhnutý z ostatného obsahu dokumentu. Zobrazí sa vždy presne na dané súradnice v rodičovskom elemente a to aj napriek okolité prvky.
* **fixed** - Element je umiestnený relatívne oproti oknu prehliadača. Môžeme ho tak prichytiť k nejakej hrane okna a pri skrolovaní stránkou bude stále na rovnakom mieste.
* **relative** - Element je umiestnený relatívne oproti svojej pozícii v dokumente. Pokiaľ mu teda nastavíme ľavú pozíciu na 20px, zobrazí sa 20px od svojho pôvodného miesta.
* **inherit** - Vlastnosť *position* bude zdedená od rodičovského elementu.

Pr.1.poziciovanie

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Absolútne poziciovanie obrázku na stránke **</title>**

**<link** href='images/mojelogo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

#vtacik

{

**position**: **absolute**;*/\*Zobrazenie elementu na nastavenej pozícií\*/*

**left**: 200px;*/\*pozícia vľavo\*/*

**top**: 20px;*/\*pozicia zhora\*/*

}

**p**{**line-height**: 2;}

**</style>**

**</head>**

**<body>**

**<img** src="images/bird.png" alt="vtak" id="vtacik" width ="200" height="200" **/>**

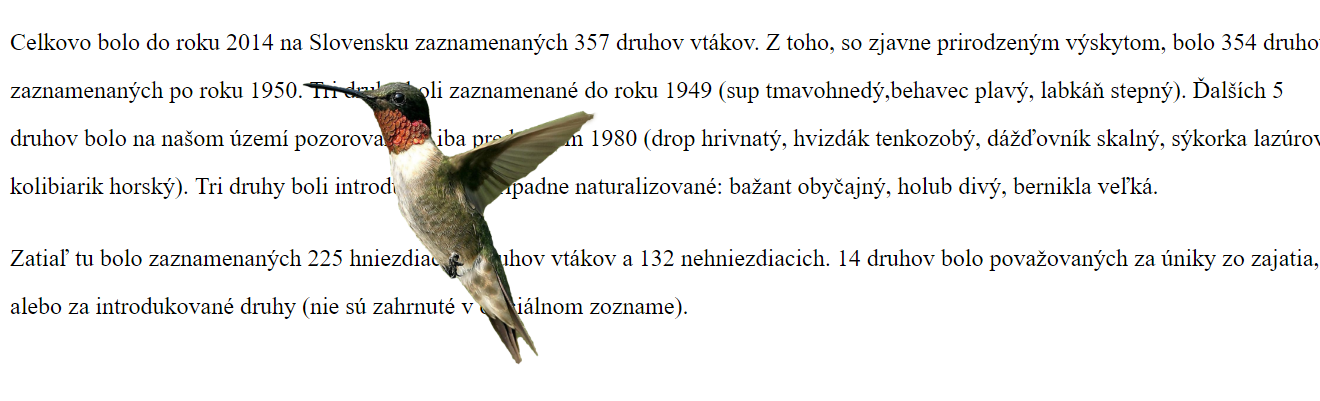
**<p>**Celkovo bolo do roku 2014 na Slovensku zaznamenaných 357 druhov vtákov. Z toho, so zjavne prirodzeným výskytom, bolo 354 druhov zaznamenaných po roku 1950. Tri druhy boli zaznamenané do roku 1949 (sup tmavohnedý,behavec plavý, labkáň stepný). Ďalších 5 druhov bolo na našom území pozorovaných iba pred rokom 1980 (drop hrivnatý, hvizdák tenkozobý, dážďovník skalný, sýkorka lazúrová, kolibiarik horský). Tri druhy boli introdukované, prípadne naturalizované: bažant obyčajný, holub divý, bernikla veľká.**</p>**

**<p>**Zatiaľ tu bolo zaznamenaných 225 hniezdiacich druhov vtákov a 132 nehniezdiacich. 14 druhov bolo považovaných za úniky zo zajatia, alebo za introdukované druhy (nie sú zahrnuté v oficiálnom zozname).**</p>**

**</body>**

**</html>**

Chrome:



2.Viditeľnosť elementu

Pomocou CSS 3 vlastnosti *visibility* môžeme skryť HTML element. Skrytý element nie je vykresľovaný, ale je v stránke prítomný a je miesto neho vykreslené prázdne miesto. Pre úplné vybratie elementu zo stránky použite vlastnosť [display](https://www.itnetwork.cz/html-css/css-manual/pozicovani/display-css-3-vlastnost-cesky-manual) , element tak ale bude neviditeľný napr. Aj pre JavaScript.

hodnoty

* **visible** (predvolené) - Element je vykresľovaný.
* **hidden** - Element nie je vykreslovaný a je skrytý. Je však na stránke stále prítomný a namiesto neho je vykresľovanie prázdne miesto.
* **collapse** - Vlastnosť slúži len pre tabuľkové elementy, po ich skrytie nie je narušené rozloženie tabuľky.
* **inherit** - Vlastnosť *visibility* bude zdedená od rodičovského elementu.

Pr.2.skrývanie elementu

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Skrývanie prvku na stránke **</title>**

**<link** href='images/mojelogo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**background-color**: #59A4FF;

**border**: 1px **solid** blue;

**font-size**: 20px;

}

#kontajner

{

**visibility**: **hidden**;*/\*kontajner je skrytý\*/*

}

**</style>**

**</head>**

**<body>**

**<div>**Tento oddiel je bežne viditeľný.**</div>**

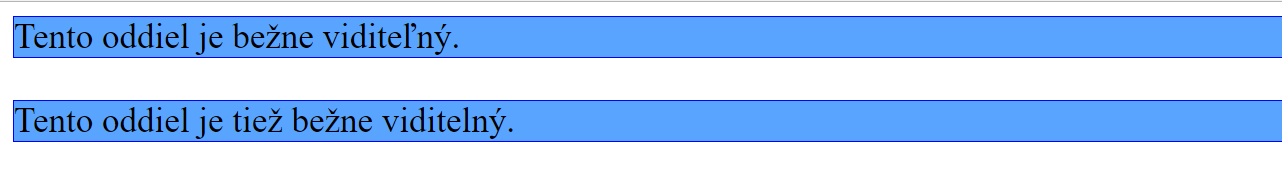
**<div** id="kontajner"**>**Tento kontejner je skrytý, ale stále zabiera miesto na stránke.**</div>**

**<div>**Tento oddiel je tiež bežne viditelný.**</div>**

**</body>**

**</html>**

Chrome:



Druhý prvok je skrytý

# 3.Nastavenie šírky prvku: width

Pomocou CSS 3 vlastnosti *width* nastavujeme šírku HTML elementu, môžeme tak meniť jeho veľkosť / rozmery.Vlastnosť sa dedí.

hodnoty

* **auto** (predvolené) - Šírku spočíta prehliadač.
* **dĺžka** - Šírka špecifikovaná v ľubovoľných dĺžkových jednotkách, napr. px, pt, cm ...
* **%** - Šírka vyjadrená ako časť šírky rodičovského elementu. Toto môže byť mätúce, keď nastavíme obrázku šírku na 50%, nebude polovičná, ale bude cez polovicu stránky alebo elementu, v ktorom je vložený.
* **inherit** - Zdedí vlastnosť *width* z rodičovského elementu.

Šírkou elementu sa myslí šírka bez akýchkoľvek rámčekov alebo okrajov (bez border, padding i margin). Tie budú k výslednej šírke ešte navyše pripočítané.

Pr.3.nastavenie šírky elementu

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Nastavenie šírky kontajneru **</title>**

**<link** href='images/mojelogo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

#kontajner

{

**width**: 300px;

**height**: 100px;

**border**: 1px **solid** blue;

}

**</style>**

**</head>**

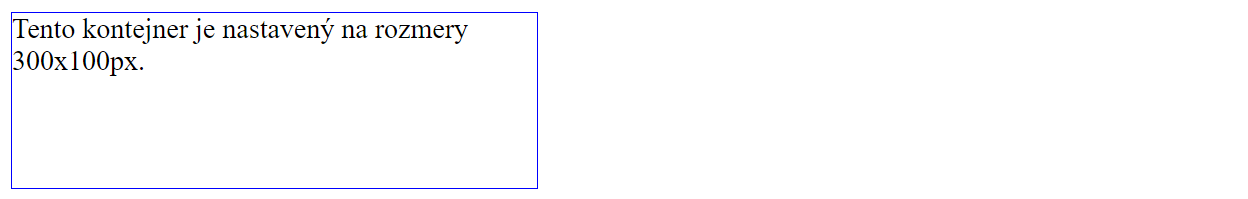
**<body>**

**<div** id="kontajner"**>**Tento kontejner je nastavený na rozmery 300x100px.**</div>**

**</body>**

**</html>**

Chrome:



Ak nastavíme iba šírku alebo výšku obrázku, druhý rozmer prehliadač dopočíta tak, aby zostal zachovaný pomer strán.

Pr.4.Nastavenie rozmerov obrázku cez CSS

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Nastavanie rozmerov obrázku cez CSS **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**.nahlad**

{

**width**:400px;/\*Obrázku\*/

}

**</style>**

**</head>**

**<body>**

**<img** src="images/vtaci.png" alt="Vtáky" class="nahlad" **/>**

**</body>**

**</html>**

Chrome:



4.Nastavenie tretieho rozmeru elementu z-indexu

Pomocou CSS 3 vlastnosti *z-index* nastavujeme hĺbku (z-súradnicu) HTML elementu na stránke. Ak sa viac elementov prekrýva, môžeme pomocou tejto vlastnosti nastaviť poradie v akom sa majú vykresľovať. Inými slovami ktoré budú vpredu a ktoré vzadu. Vlastnosť funguje len pre pozicovanie elementy, teda s position: absolute, fixed a relative.

hodnoty

* **auto** (predvolené) - Poradie vykresľovanie prvkov je založené na rodičovskej elementu.
* **číslo** - Číslo, určujúci poradie, v ktorom sa má element vykresliť. Čím vyššie číslo, tým je element vyššie.
* **inherit** - Vlastnosť *z-index* bude zdedená od rodičovského elementu.

Pr.5.Použitie z-indexu na prekrývanie elementov

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Prekrývanie elementov v CSS **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**opacity**: 0.8;

**width**: 100px;

**height**: 100px;

}

#zeleny

{

**z-index**: 3;

**position**: **relative**;*/\*Element je umiestnený relatívne oproti svojej pozícii v dokumente. \*/*

**background-color**: #01EB4B;

**border**: 1px **solid** green;

}

#modry

{

**z-index**: 2;

**position**: **absolute**;*/\*Element je zobrazený na udanej pozíciu a je vytrhnutý z ostatného obsahu dokumentu. \*/*

**left**: 50px;

**top**: 50px;

**background-color**: #4B89FF;

**border**: 1px **solid** blue;

}

#cerveny

{

**z-index**: 1;

**position**: **fixed**;*/\*pri skrolovaní stránkou bude stále na rovnakom mieste.\*/*

**top**: 40px;

**left**: 80px;

**background-color**: #FF004B;

**border**: 1px **solid** red;

}

**</style>**

**</head>**

**<body>**

**<div** id="modry"**>**PROSTREDNÝ**</div>**

**<div** id="cerveny"**>**NAJNIŽŠŠÍ**</div>**

**<div** id="zeleny"**>**NAJVYŠŠÍ**</div>**

**</body>**

**</html>**

Chrome:



5.Nastavenie pretekania obsahu prvku

Pomocou CSS 3 vlastnosti *overflow-x* nastavujeme správanie HTML elementu v prípade, že jeho obsah pretečie šírku elementu. Takto môžeme obsah skryť, nechať pretiecť alebo vytvoriť elementu scrollbar.

hodnoty

* **visible** (predvolené) - Pretekajúca obsah je vykreslený mimo hraníc elementu.
* **hidden** - Pretekajúca obsah bude skrytý.
* **scroll** - element je pridaný scrollbar a to aj v prípade, že obsah nepreteká element.
* **auto** - V prípade pretekaniu obsahu bude elementu pridaný scrollbar.

Ak by sme chceli nastaviť, čo sa stane, keď obsah pretečie výšku elementu (os *y* ), môžeme využiť vlastnosti [overflow-y](http://www.itnetwork.cz/html-css/css-manual/pozicovani/overflow-y-css-3-vlastnost-cesky-manual) . Pre nastavenie oboch strán naraz môžeme využiť atribútu [overflow](http://www.itnetwork.cz/html-css/css-manual/pozicovani/overflow-css-3-vlastnost-cesky-manual) .

### Priklad 6

Vytvorme si div s rozmermi 200px × 150px a modrým, plným rámčekom o šírke 1px. Do tohto divu umiestnime odstavec s šírkou 300px a textom Lorem ipsum, ktorý necháme pretekať.

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Pretekanie obsahu mino širku elementu **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**overflow-x**: **visible**;

**width**: 200px;

**height**: 150px;

**border**: 1px **solid** blue;

}

**p**

{

**width**: 300px;*/\*Ak nezdáme druhý rozmer,dopočita tento prehliadač\*/*

}

**</style>**

**<title>**overflow-x**</title>**

**</head>**

**<body>**

**<div>**

**<p>**

Je dávno známe, že ak je zrozumiteľný obsah stránky, na ktorej rozloženie sa čitateľ díva, jeho pozornosť je rozptýlená. Dôvodom použitia Lorem Ipsum je fakt, že má viacmenej normálne rozloženie písmen, takže oproti použitiu 'Sem príde text, sem príde text' sa obsah vypĺňanej oblasti na stránke viac podobá na skutočný text.

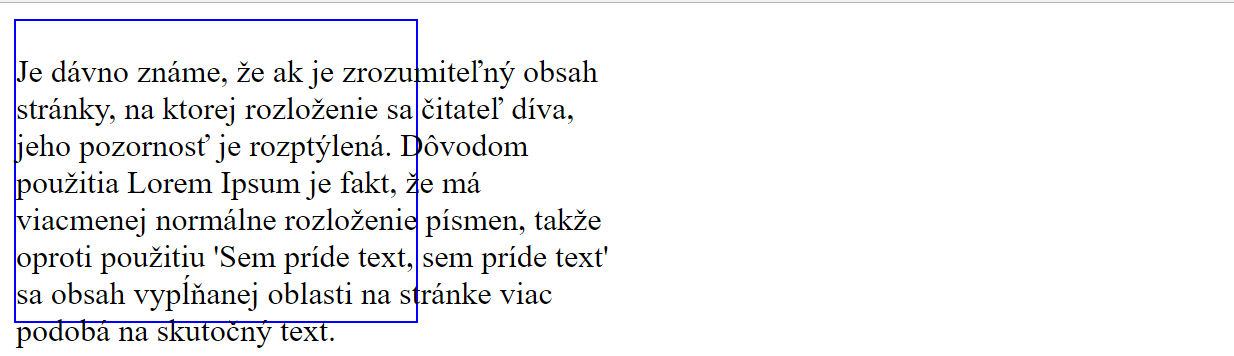
**</p>**

**</div>**

**</body>**

**</html>**

Chrome:



6.Plynulé nastavenie veľkosti prvku-resize

Pomocou CSS 3 vlastnosti *resize* môžeme užívateľovi dovoliť rozťahovať element. V predvolenom nastavení sa môže napr. [Textarea](http://www.itnetwork.cz/html-css/html-manual/formulare/html-textarea-textove-pole-formulare-cesky-manual) (vstupný formulár pre zadanie viac riadkového textu) ľubovoľne rozťahovať. **Tento atribút nepodporuje webový prehliadač IE!**

hodnoty

* **none** (predvolené) - Používateľ nemôže zmeniť veľkosť elementu.
* **both** - Užívateľ môže element roztiahnuť horizontálne aj vertikálne.
* **horizontal** - Element sa môže roztiahnuť iba horizontálne.
* **vertical** - Element sa môže roztiahnuť iba vertikálne.
* **initial** - Nastaví vlastnosť na predvolenú hodnotu (teda na none).
* **inherit** - Hodnota sa zdedí z rodičovského elementu.

**Poznámka:** pre využitie vlastnosti *resize* musíme nastaviť atribút [overflow](http://www.itnetwork.cz/html-css/css-manual/pozicovani/overflow-css-3-vlastnost-cesky-manual) na inú hodnotu, než je hodnota visible.

PR.6.resize

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Zmena veľkosti prvku pomocou myši. **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**width**: 300px;

**height**: 100px;

**border**: 1px **solid** blue;

resize: vertical;

**overflow**: **auto**;

}

**</style>**

**</head>**

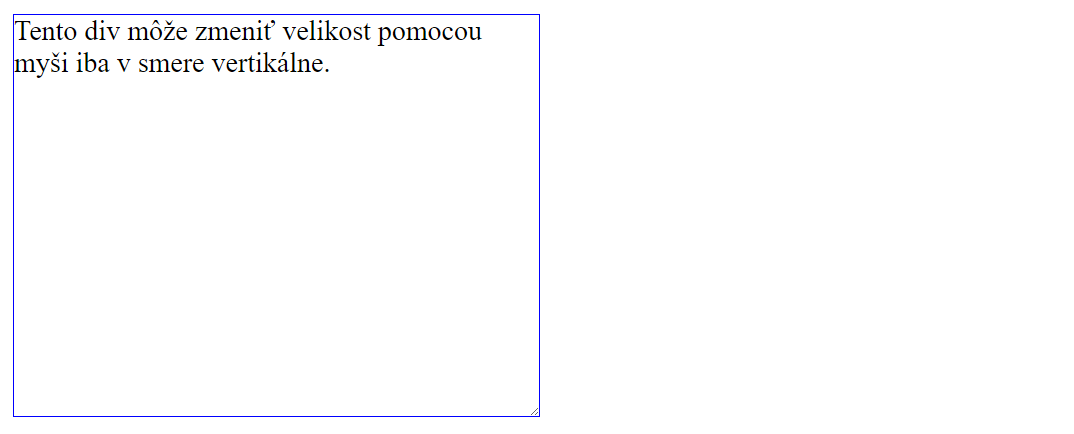
**<body>**

**<div>**Tento div môže zmeniť velikosť pomocou myši iba v smere vertikálne.**</div>**

**</body>**

**</html>**

Chrome:



7.Definovanie veľkosti elementu box-sizing

Pomocou CSS 3 vlastnosti *box-sizing* nastavujeme čo bude šírka a výška elementu zahŕňať. Vlastnosť sa dedí.

hodnoty

* **content-box** (predvolené) - Výška a šírka elementu zahŕňa iba obsah. Vlastnosti [padding](http://www.itnetwork.cz/html-css/css-manual/okraje/padding-css-3-vlastnost-cesky-manual) , [border](http://www.itnetwork.cz/html-css/css-manual/ramecek/border-css-3-vlastnost-cesky-manual) alebo [margin](http://www.itnetwork.cz/html-css/css-manual/okraje/margin-css-3-vlastnost-cesky-manual) nie sú zahrnuté.
* **border-box** - Výška a šírka elementu zahŕňa obsah a vlastnosti *padding* a *border* . Vlastnosť *margin* nie je zahrnutá do týchto rozmerov.
* **initial** - Nastaví vlastnosť na predvolenú hodnotu (teda na content-box).
* **inherit** - Hodnota sa zdedí z rodičovského elementu.

Priklad 7.1.: Vytvorme si dva elementy div s rozmermi 300px × 100px s modrým, úzkym a plným rámčekom a s výplňou 50px. Oba elementy budú mať okraj veľký 10px (vlastnosť margin ). Prvý div nebude nastavovať vlastnosť box-sizing a druhý div bude mať u nej hodnotu border-box.

Riešenie:

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Zmena veľkosti prvku pomocou myši. **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**width**: 300px;

**height**: 100px;

**padding**: 50px;

**border**: #3B94E0 1px **solid**;

**margin**: 10px;

}

**.box-sizing**

{

box-sizing: **border**-box;

}

**</style>**

**<title>**box-sizing**</title>**

**</head>**

**<body>**

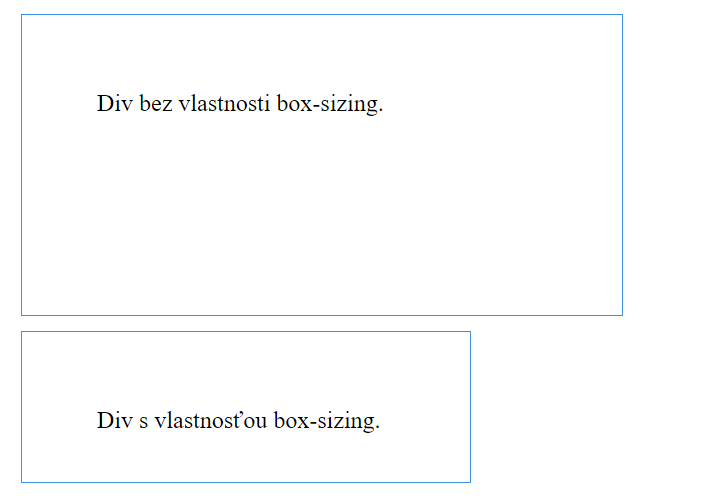
**<div>**Div bez vlastnosti box-sizing.**</div>**

**<div** class="box-sizing"**>**Div s vlastnosťou box-sizing.**</div>**

**</body>**

**</html>**

chrome:



# 8. Borders (rámčeky)

  Nástupom CSS3 môžete využívať zaoblených rámčekov, pridať vlastný tieň alebo aj ako rámček použiť obrázok bez použitia grafických programov.

Všeobecne - Podpora prehliadačov

* IE9 podporuje len veľmi málo nových vlastností
* Pri ostatných prehliadačov je vyžadovaný prefix
  + Firefox **-moz-** vlastnosť
  + Chrome, Safari **-webkit-** vlastnosť
  + Opera **-o-** vlastnosť
  + IE9 **-ms-** vlastnosť

Pr.8.1 Radius 25px

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Nastavenie zaokruhlenia rohov prvku. **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**width**: 600px;

**height**: 100px;

**color**:white;

**background-color**: gray;

**font-size**: 25px;

**border**: 2px **solid**;

**border**-radius: 25px;

}

**</style>**

**</head>**

**<body>**

**<div>**

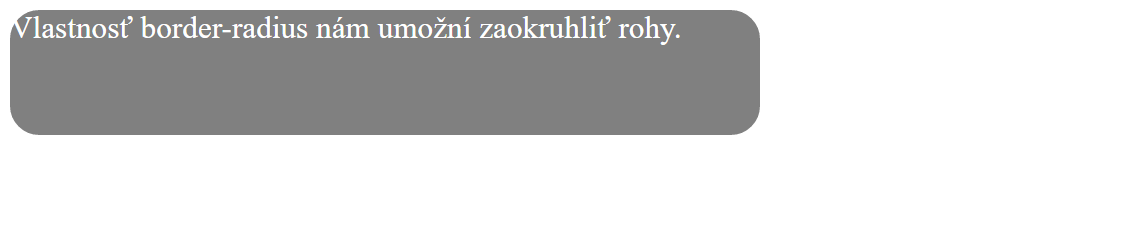
Vlastnosť border-radius nám umožní zaokruhliť rohy.

**</div>**

**</body>**

**</html>**

Chrome:



### PR.8.2 CSS3 vrhnutý tieň (box-shadow)

!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Vrhnutý tieň. **</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**

{

**width**: 600px;

**height**: 100px;

**color**:white;

**background-color**: lightblue;

**border**: 2px **solid** black;

box-shadow: 10px 10px #888888;*/\*tieň vpravo a dole\*/*

}

**</style>**

**</head>**

**<body>**

**<div>**

Tento prvok vrhá tieň.

**</div>**

**</body>**

**</html>**

### Chrome:

### 

9.2D transformácie.

Za pomocou CSS3 môžeme dané elementy ľubovoľne natáčať, škálovať, posúvať, naťahovať ai ... A to za použitia 2D alebo 3D transformáciou.

V nasledujúcom tutoriálu si popíšeme 2D transformácie.

Pre daný prehliadač je potrebné použiť daný prefix:

* IE9 **-ms-**
* Firefox **-moz-**
* Chrome a Safari **-webkit-**
* Opera **-o-**

#### translate ()

Posunie element na dané súradnice.

PR.9.1.posun na súradnice 50,100

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**POSUN z pôvodnej polohy o hodnotu ,ktorú zadame**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**{

**width**:140px;

**height**:140px;

**background-color**: lightblue;

**border**: **solid** 2px black;

transform: translate(50px,100px);

-ms-transform: translate(50px,100px); */\* IE 9 \*/*

-webkit-transform: translate(50px,100px); */\* Safari and Chrome \*/*

-o-transform: translate(50px,100px); */\* Opera \*/*

-moz-transform: translate(50px,100px); */\* Firefox \*/*\*/

}

**</style>**

**</head>**

**<body>**

**<div>** Toto je posunuté o 50px vpravo a 100px dolu**</div>**

**</body>**

**</html>**

Chrome:



#### rotate ()

Orotujeme element o daný uhol.

PR.9.2 rotácia oddielu o 90 stupňov

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Rotácia z pôvodnej polohy o hodnotu ,ktorú zadame**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**{

*/\*Veľkosť prvku div\*/*

**width**:140px;

**height**:140px;

**background-color**: lightblue;

**border**: **solid** 2px black;

*/\*rotácia o 30 stupńov\*/*

transform: rotate(90deg);

-ms-transform: rotate(90deg); */\* IE 9 \*/*

-webkit-transform: rotate(90deg); */\* Safari and Chrome \*/*

-o-transform: rotate(90deg); */\* Opera \*/*

-moz-transform: rotate(90deg); */\* Firefox \*/*

}

**</style>**

**</head>**

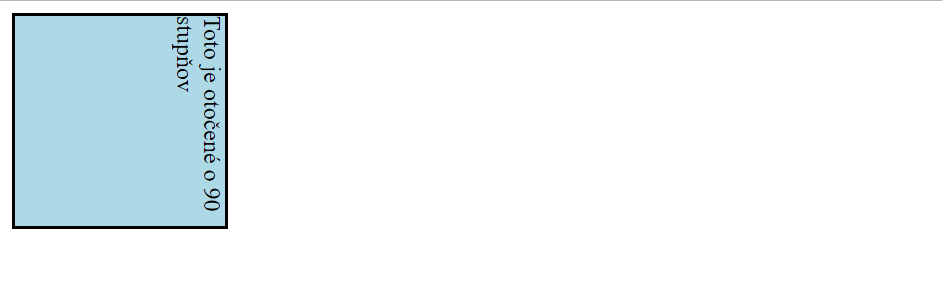
**<body>**

**<div>** Toto je otočené o 90 stupňov**</div>**

**</body>**

**</html>**

Chrome:



#### scale ()

Roztiahne element v pomere k pôvodným rozmerom

PR9.3zväčšenie rozmeru oddielu v pomere ktorý je uvedený v príkaze

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Zmena pôvodných rozmerov div na hodnotu ,ktorú zadame**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**{

*/\*Veľkosť prvku div\*/*

**width**:140px;

**height**:140px;

**background-color**: lightblue;

**border**: **solid** 2px black;

transform: scale(2,4);

-ms-transform: scale(2,4); */\* IE 9 \*/*

-webkit-transform: scale(2,4); */\* Safari and Chrome \*/*

-o-transform: scale(2,4); */\* Opera \*/*

-moz-transform: scale(2,4); */\* Firefox \*/*

}

**</style>**

**</head>**

**<body>**

**<div>** Toto je zväčšené 2x v šírke a 4x v výske.**</div>**

**</body>**

**</html>**

Chrome:



#### Skew ()

Orotujeme element o zadaný uhol v daných osiach.

PR.9.4.

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Skosenie oddielu div**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div**{

*/\*Veľkosť prvku div\*/*

**width**:140px;

**height**:140px;

**background-color**: lightblue;

**border**: **solid** 2px black;

*/\*Skosenie\*/*

transform: skew(30deg,20deg);

-ms-transform: skew(30deg,20deg); */\* IE 9 \*/*

-webkit-transform: skew(30deg,20deg); */\* Safari and Chrome \*/*

-o-transform: skew(30deg,20deg); */\* Opera \*/*

-moz-transform: skew(30deg,20deg); */\* Firefox \*/*

}

**</style>**

**</head>**

**<body>**

**<div>** Toto je v x-ovej otočené o 30 stupňov a y-ovej otočené 20 stupňov.**</div>**

**</body>**

**</html>**

Chrome:



#### Matrix (n,n,n,n,n,n)

Kombinuje všetky predchádzajúce metódy  Rotácia, škála a posun v jednom príkaze.

PR.9.5

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Použitie metódy matrix()**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div** {

**width**: 300px;

**height**: 100px;

**background-color**: yellow;

**border**: 1px **solid** black;

}

**div**#myDiv1 {

-ms-transform: matrix(1, -0.3, 0, 1, 0, 0); */\* IE 9 \*/*

-webkit-transform: matrix(1, -0.3, 0, 1, 0, 0); */\* Safari \*/*

transform: matrix(1, -0.3, 0, 1, 0, 0); */\* Standard syntax \*/*

}

**div**#myDiv2 {

-ms-transform: matrix(1, 0, 0.5, 1, 150, 0); */\* IE 9 \*/*

-webkit-transform: matrix(1, 0, 0.5, 1, 150, 0); */\* Safari \*/*

transform: matrix(1, 0, 0.5, 1, 150, 0); */\* Standard syntax \*/*

}

**</style>**

**</style>**

**</head>**

**<body>**

**<p>** matrix() metóda kombinuje všetky 2D transformačné metódy v jednom.**</p>**

**<div>**

Toto je normálny div element bez použitia transformačnej metódy.

**</div>**

**<div** id="myDiv1"**>**

S použitín metody matrix().

**</div>**

**<div** id="myDiv2"**>**

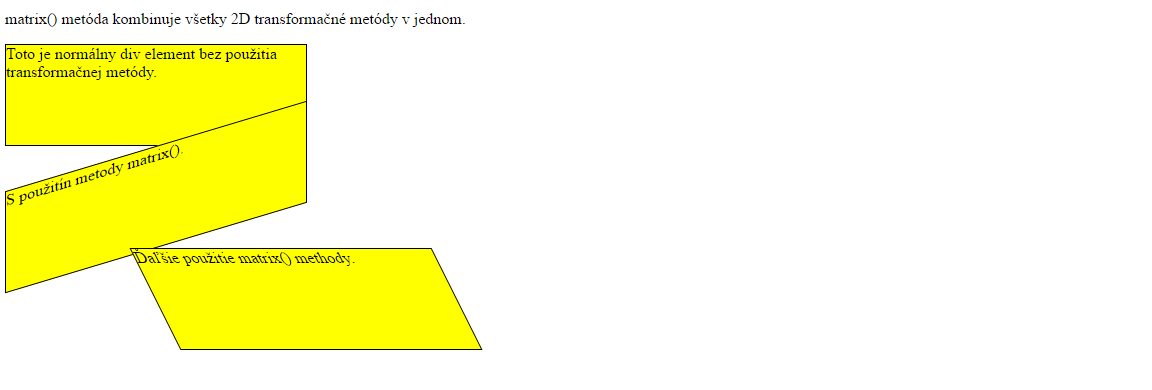
Ďaľšie použitie matrix() methody.

**</div>**

**</body>**

**</html>**

Chrome:



PR.9.6 Ešte jedno pužitie metódy matrix

<!DOCTYPE html>

**<html** lang="sk"**>**

**<head>**

**<meta** charset="UTF-8"**>**

**<title>**Použitie metódy matrix()**</title>**

**<link** href='images/logo.png' rel='shortcut icon' type='image/png'**>**

**<style>**

**div** {

**width**: 300px;

**height**: 100px;

**background-color**: lightblue;

**border**: 1px **solid** black;

}

**div**#myDiv1 {

transform:matrix(0.866,0.5,-0.5,0.866,0,0);

-ms-transform:matrix(0.866,0.5,-0.5,0.866,0,0); */\* IE 9 \*/*

-moz-transform:matrix(0.866,0.5,-0.5,0.866,0,0); */\* Firefox \*/*

-webkit-transform:matrix(0.866,0.5,-0.5,0.866,0,0); */\* Safari and Chrome \*/*

-o-transform:matrix(0.866,0.5,-0.5,0.866,0,0); */\* Opera \*/*

}

**</style>**

**<body>**

**<div>**

Toto je normálny div element bez použitia transformačnej metódy.

**</div>**

**<div** id="myDiv1"**>**

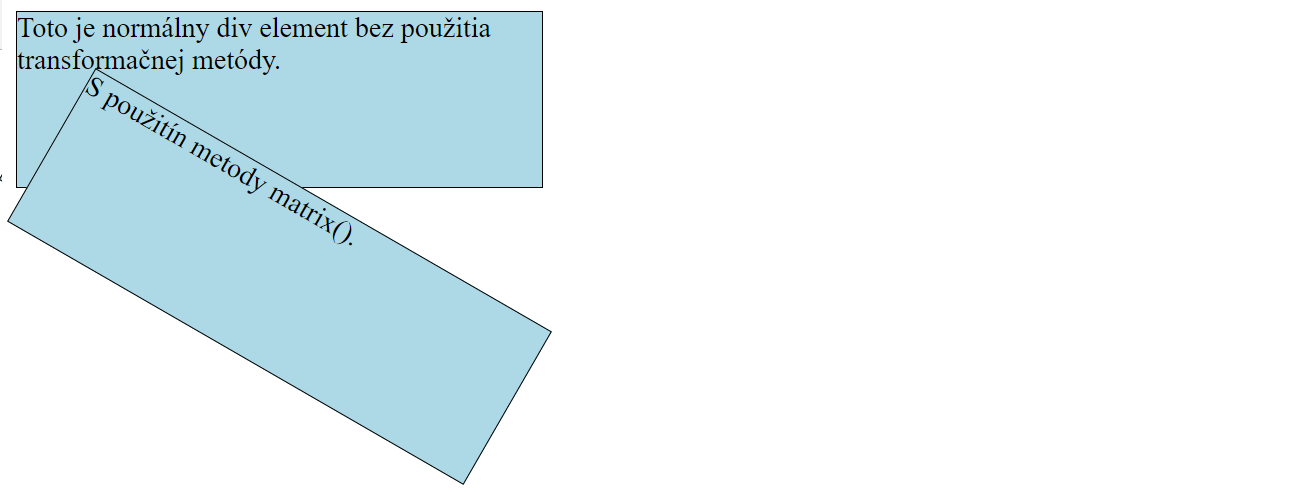
S použitín metody matrix().

**</div>**

**</body>**

**</html>**

Chrome:



# 10.Animations (animácie)

S CSS3 máme možnosť vytvárať animácie, ktoré môžu nahradiť napríklad aminované obrázky, Flash animácie alebo JavaScript, ktoré sa používajú na mnohých weboch.

Pre ovládnutie animácií je potrebné si osvojiť prácu s @keyframes.

Príklad nastaveenia svojho @keyframe: