



CSS manuál

Ing.Drgo Pavel

5.august 2016

OBSAH

1.CSS Úvod.....	3
2.Ako a kam umiestniť kaskádové štýly.....	4
3.CSS návod 03 Vyberáme elementy.....	6
4.CSS návod 04 CLASS a ID.....	13
5.CSS návod 05 Dalšie možnosti výberu elementov.....	18
6. CSS návod 06 Odkazy a pseudo triedy.....	26
7.CSS návod 07 Pseudo triedy pokračovanie.....	32
8.CSS návod 08 Základné práce s písmom.....	39
9.CSS návod 09 Nastavujeme vlastnosti textu.....	49
10.CSS návod10 Nastavujeme vlastnosti textu pokračovanie.....	53
11.CSS návod 11 farby a obrázok na pozadí.....	61
12.CSS návod 12 Obrázok na pozadí pokračovanie.....	68
13.CSS návod 13 Rozmery a okraje elementu.....	73
14.CSS návod 14 Okraje elementu PADDING, MARGIN a BORDER....	78
15.CSS návod 15 Rámeček BORDER detailněji.....	83
16.CSS návod 16 Zagulatený rámeček BORDER RADIUS.....	87
17.CSS návod 17 Jednoduché menu pomocí CSS.....	94

1.CSS Úvod



CSS je skratka: Cascade styles sheet

Kaskádové štýly nám umožňujú nastavenie vlastností elementov html dokumentov. Element môže byť:paragraf,nadpis,kontajner,odkaz,obrázok ,span,zoznam,div.

Ako napríklad farbu textu,,veľkosť písma,farbu pozadia.Umožňujú nám nastaviť vlastnosti ,ktoré sa v HTML dokumente nastaviť nedajú.Práca s kakádovými štýlmi je omnoho efektívnejšia než nastavovanie vlastností elementu priamo v HTML dokumente,

V CSS je možné nastavovať vlastnosti skupine elementov.Dokument HTML môžeme celý formátovať a môžeme sa tak vyhnúť formátovaniu pomocou tabuliek.

Výhodou je použitie jedného externého súboru pre viacero súborov html, prípadne pre celý web..

Je možné definovať len niektoré prvky v jednotlivých html dokumentoch.

Pre zmenu celého webu je treba zmeniť niektorý súbor css , bez toho aby sme zasahovali do jednotlivých html dokumentov.

Priklad jednoduchého zoznamu:

- [Úvod](#)
- [Portfolio](#)
- [O nás](#)
- [Kontakt](#)

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">

<style>
ul {
  list-style-type: none;
  margin: 0;
```

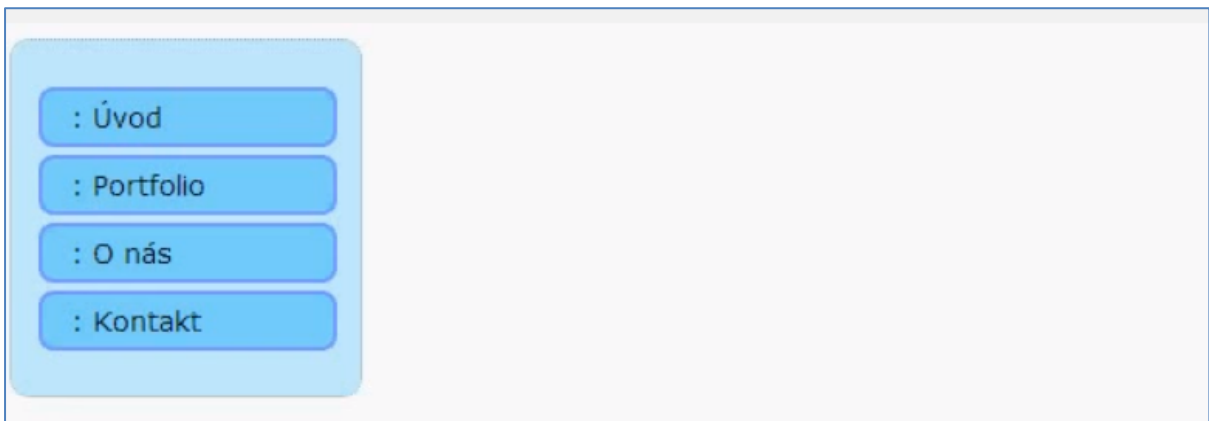
```
padding: 0;
}
</style>
</head>
<body>
```

<p> V tomto príklade sme odstránili guľky zo zoznamu , a jeho východiskovú výplň a okraj sme nastavili na nulu. </p>

```
<ul>
  <li><a href="#home">Úvod</a></li>
  <li><a href="#news">Portfolio</a></li>
  <li><a href="#contact">O nás</a></li>
  <li><a href="#about">Kontakt</a></li>
</ul>

</body>
</html>
```

A zmena na menu s použitím CSS v súbore:



Zoznam sa zmenil na ponuku ,ktorá je dynamická,meni sa podľa polohy kurzora.

Kaskádové štýly môžeme vytvárať v akomkoľvek textovom editore.Výhodnejší je editor so zvýraznenou syntaxou:NotePad++,Netbeans,PSPAD,Sublime Text3.

2.Ako a kam umiestniť kaskádové štýly

Kaskádové štýly môžeme zapisovať:

1. Priamo do hlavičky HTML dokumentu
2. Priamo do elementu(prvku)
3. Do externého súboru-a potom sa v HTML na ne odkazovať
4. Skombinovať obidva spôsoby

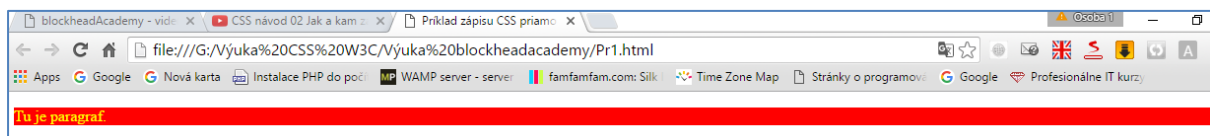
1.Priamo do HTML dokumentu

Pr2.Nastavenie farby a pozadia pre prvok paragraf

subor2.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad zápisu CSS priamo do HTML dokumentu</title>
    <style>
    <!--
    p {
      color:yellow;
      background-color:red;
    }
    </style>
  </head>
  <body>
    <p>Tu je paragraf.</p>
  </body>
</html>
```

Výstup:



Štýly označíme <style>.....</style> a zápis umiestníme do hlavičky medzi <head>....a </head>

2.Zápis priamo do elementu(prvku)

subor3.html

```
<!DOCTYPE html>
<html lang="sk">
```

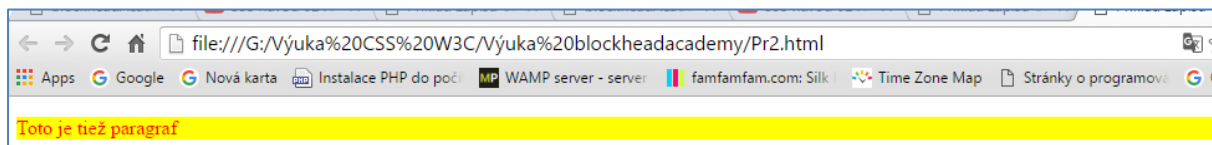
```

<head>
  <meta charset="UTF-8">
  <title>Príklad zápisu CSS priamo do elementu p</title>

</head>
<body>
  <p style="color:red;background-color:yellow;">Toto je tiež
paragraf</p>
</body>
</html>

```

Výstup:



V tomto prípade je definícia štýlu priamo v zápise vlastností elementu

3.Zápis do externého súboru

Najskôr musíme externý súbor pripojiť k súboru HTML.

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

Link vložíme do hlavičky.

Obsah HTML súboru,v ktorom sa odkazujeme na CSS súbor:

subor3.html

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad zápisu CSS do externého súboru s odkazom v HTML</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Toto je modrý paragraf.</p>
  </body>
</html>

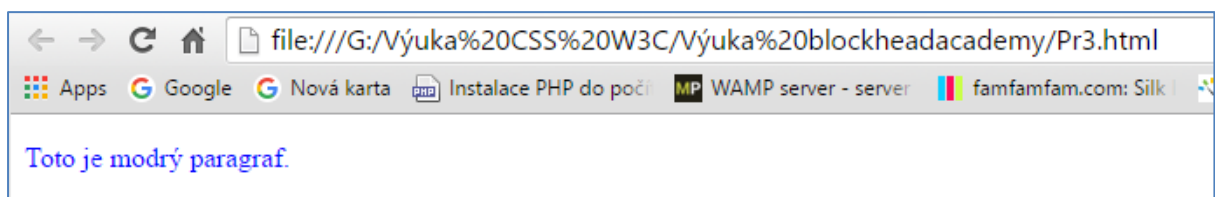
```

Obsah súboru main.css

main.css

```
/*súbor main.css*/  
  
p {  
    color: blue;  
}
```

Výstup:



Externých súborov css môže byť aj viac. Najväčšiu prioritu má ten súbor ktorý je najneskôr vložený do html.

Ak nastavujeme vlastnosti v rôznych súboroch css, tak sa nastavujú tie vlastnosti z posledného súboru CSS

Pr4. Použitie viacerých externých súborov

Subor4.html

```
<!DOCTYPE html>  
<html lang="sk">  
  <head>  
    <meta charset="UTF-8">  
    <title>Príklad zápisu CSS do externého súboru s odkazom v  
HTML</title>  
    <link rel="stylesheet" type="text/css" href="css/main.css"/>  
    <link rel="stylesheet" type="text/css" href="css/menu.css"/>  
    <link rel="stylesheet" type="text/css" href="css/zoznam.css"/>  
  </head>  
  <body>  
<p>Toto je modrý paragraf, ktorý bude nastavený podľa zoznam.css</p>  
  </body>  
</html>
```

3.CSS návod 03 Vyberáme elementy

Vyberáme elementy, ktorým budeme chcieť definovať vlastnosti. Definície štýlov budeme zapisovať do externých súborov.

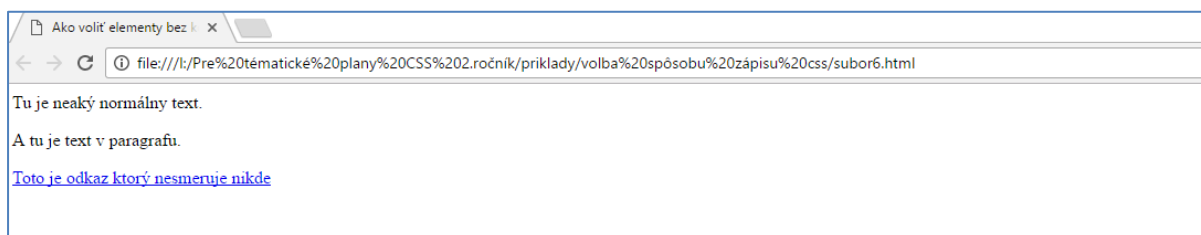
Pr6 Ako voliť elementy bez konkrétnej špecifikácie

Nepoužijeme kaskádové štýly. Prehliadač použije tzv. default- predvolené nastavenia elementov.

subor6.html:

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title> Ako voliť elementy bez konkrétnej špecifikácie </title>
    <!--<link rel="stylesheet" type="text/css" href="css/main.css"/>-->
  </head>
  <body>
    Tu je neaký normálny text.
    <p>A tu je text v paragrafu.</p>
    <a href="#">Toto je odkaz ktorý nesmeruje nikde</a>
  </body>
</html>
```

Výstup:



-----TU POKRAČOVAT NABUDÚCE-----

Pr7.Ako vyberieme všetky prvky , ktoré sa nachádzajú v tele dokumentu(v body)

subor7.html:

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
```



```

<title>Nastavenie vlastností viacerých elementov naraz.</title>
  <link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
Tu je neaký normálny text.
<p>A tu je text v odseku.</p>
<a href="#">Toto je neaký odkaz</a><!--Odkaz nikam-->
</body>
</html>

```

main.css

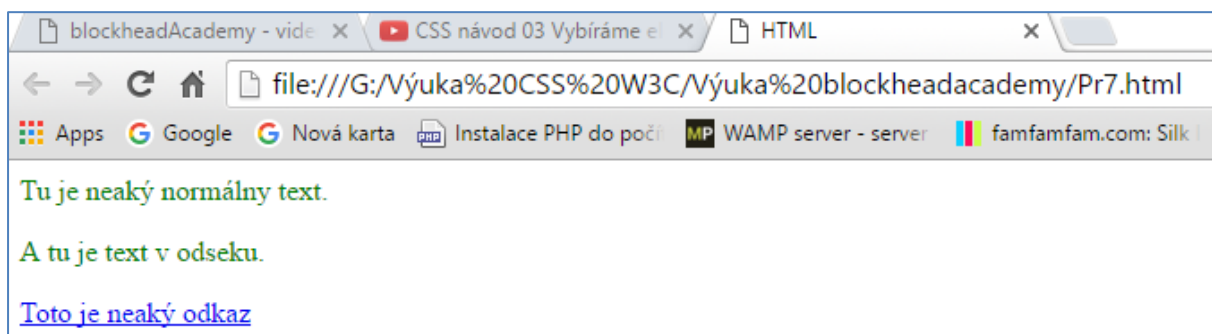
```

/*súbor main.css*/
/*Nastavenie textových prvkov v body na zelenú farbu*/

body {
  color: green;
}

```

Výstup:



Všetky texty okrem odkazu sa zobrazili zelenou farbou.

Odkazy majú v html nastavené svoje default vlastnosti. Nevzťahuje sa na ne selektor **body**

Voľba prvku **body** je vhodná ak chceme nastaviť základné vlastnosti pre celý dokument.

Preto je vhodné dať voľbu body hneď na začiatok. Môžeme nastavovať typ, veľkosť alebo farbu písma, farbu pozadia. Teda vlastnosti, ktoré budú platné pre celý dokument html.

To ale neznamená, že vlastnosti elementov nemôžu byť ďalej menené.

Ak potom chceme zmeniť v príklade PR7 farbu paragrafu napr. na modrú farbu, upravíme súbor **main.css**, tak že selektor <p> uvedieme za selektor <body> nasledovne:

Subor7.html - použijeme ten istý ako v predchádzajúcom príklade

```
<!DOCTYPE html>
```

```

<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Nastavenie vlastností viacerých elementov naraz.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    Tu je neaký normálny text.
    <p>A tu je text v paragrafu.</p>
    <a href="#">Toto je neaký odkaz</a><!--Odkaz nikam-->
  </body>
</html>

```

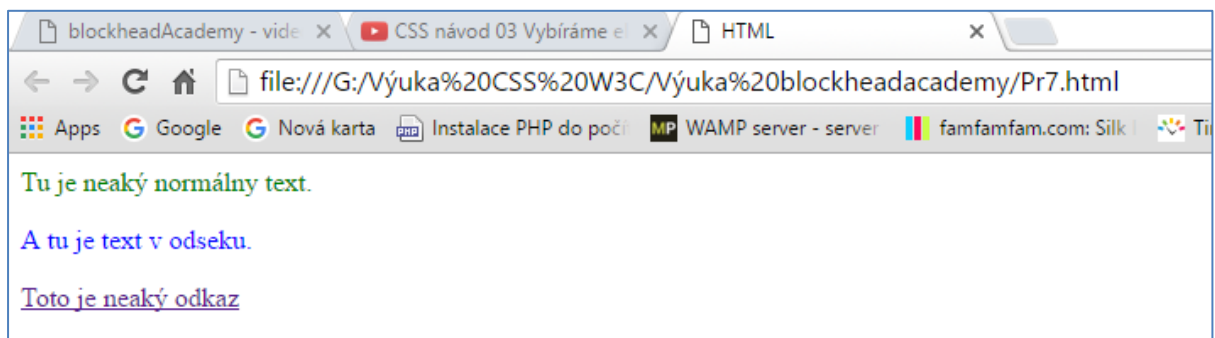
main.css

```

/*súbor main.css*/
/*Nastavenie textových prvkov v body na zelenú farbu*/
body {
  color: green;
}
/* a následne farbu textu paragrafu na modrú farbu*/
p {
  color:blue;
}

```

Výstup:



Teraz nastavíme farbu odkazu **a**

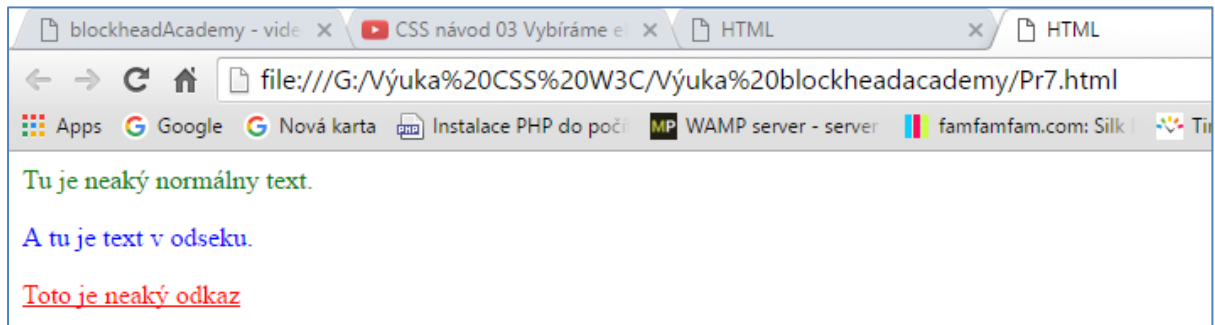
```

/*súbor main.css*/
/*Nastavenie všetkých textových prvkov v body na zelenú farbu*/
body {
  color: green;
}
/* a následne farbu textu paragrafu na modrú farbu*/
p {
  color:blue;
}

```

```
/* Teraz zmeníme aj farbu odkazu na červenú */  
a {  
    color:red;  
}
```

Výstup:



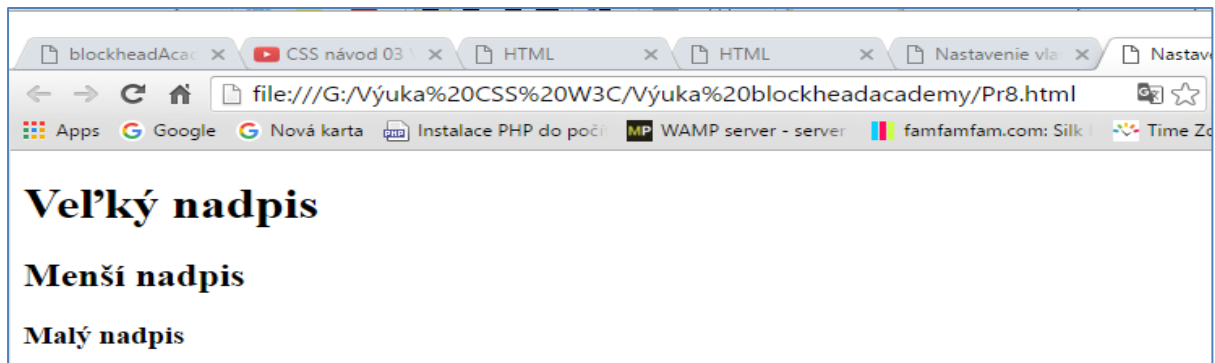
Vidíme, že napriek tomu, že sme nastavili farbu textu v celom **body** na zelenú, tak môžeme opätovným výberom prvkov zmeniť ich vlastnosti. Napr. Napriek tomu, že bol **paragraf** pomocou voľby **body** nastavený na zelenú farbu, ďalším výberom **paragrafu** môžeme zmeniť jeho farbu na modrú.

Pr8 Ako definovať rovnaký štýl pre viac elementov naraz

subor8.html použitie nadpisov bez css.

```
!DOCTYPE html>  
<html lang="sk">  
  <head>  
    <meta charset="UTF-8">  
    <title>Nastavenie vlastností viacerých elementov naraz.</title>  
  </head>  
  <body>  
    <h1>Veľký nadpis</h1>  
    <h2>Menší nadpis</h2>  
    <h3>Malý nadpis</h3>  
  </body>  
</html>
```

Výstup:



Teraz budeme chcieť nastaviť všetkým prvkom **hi** farbu naraz

Pr9.Nastavenie farieb nadpisov

subor9.html

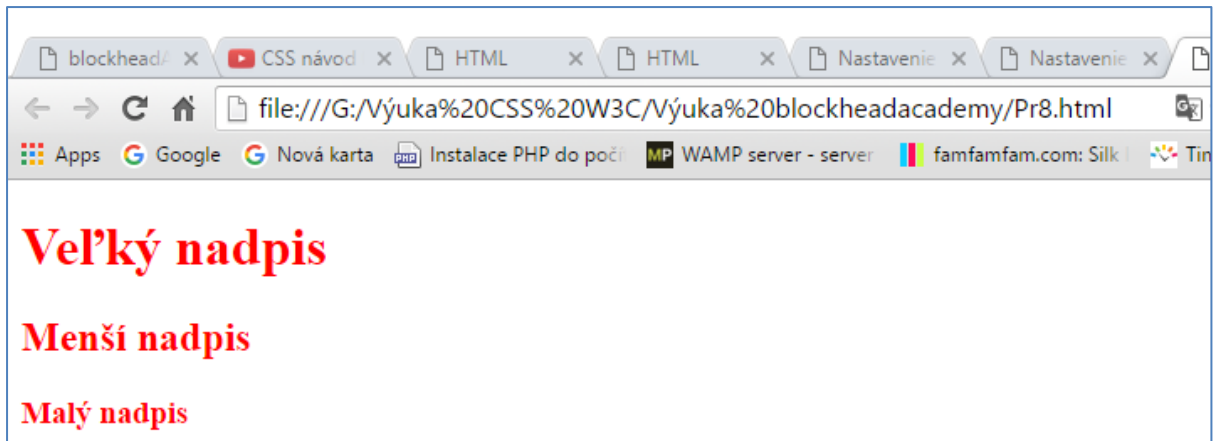
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Nastavenie vlastností viacerých elementov naraz.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <h1>Veľký nadpis</h1>
    <h2>Menší nadpis</h2>
    <h3>Malý nadpis</h3>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie všetkých nadpisov hi na červenú farbu*/
h1,h2,h3 {
  color: red;
}
```

Selektory prvkov h1,h2,h3 oddelíme čiarkou

Výstup:



Definícia štýlov pre všetky elementy.

Všetky prvky sa vyberú pomocou znaku *****

Pr9 Rozšírenie výberu na prvky hi, p, a

subor9.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Nastavenie vlastností všetkých elementov naraz.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <h1>Veľký nadpis</h1>
    <h2>Menší nadpis</h2>
    <h3>Malý nadpis</h3>
    <p>Tento paragraf bude tiež vo výbere *</p>
    <a href="#">Toto je odkaz ktorý nesmeruje nikde a je tiež vo
výbere</a>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie všetkých prvkov na červenú farbu*/
* {
  color: red;
}
```

Výstup:



Pre zakomentovanie riadkov sa používa `/* text */`. Zakomentovanie riadkov slúži aj na dočastné vypnutie príkazu alebo príkazov.

main.css

```
/*súbor main.css*/  
/*Vyradenie so spracovania nastavenae všetkých prvkov na červenú farbu  
* {  
    color: red;  
}  
*/
```

Zobrazí sa pôvodné nastavenie:

Výstup so zakomentovaným súborom main.css



4.CSS návod 04 CLASS a ID

Výber prvkov pomocou:

- tried - class
- identifikátorov- id

Pr10 Nastavenie vlastností pre všetky paragraphy

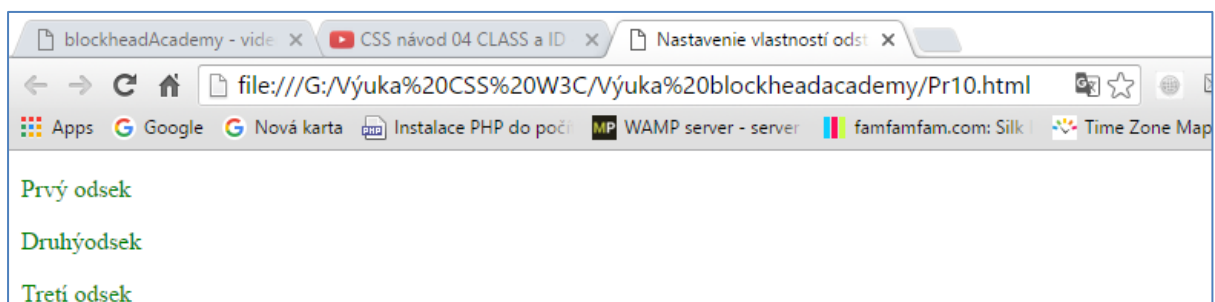
subor10.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Nastavenie vlastností pre všetky paragraphy.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Prvý paragraf</p>
    <p>Druhýparagraf</p>
    <p>Tretí paragraf</p>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie všetkých paragrafov na zelenú farbu*/
p {
  color: green;
}
```

Výstup:



Do výberu sme dali všetky paragraphy. To ale nemusí byť vždy. Môžeme chcieť dať do výberu iba určitý paragraf.

Pr11. Výber iba určitého prvku .Nie všetkých rovnakých prvkov.

Použijeme na to identifikátor **id**

```
<p id="cerveny">Druhý paragraf</p>
```

subor11.html

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Výber iba určitého prvku zo všetkých rovnakých prvkov</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Prvý paragraf</p>
    <p id="cerveny">Druhý paragraf</p>
    <p>Tretí paragraf</p>
  </body>
</html>

```

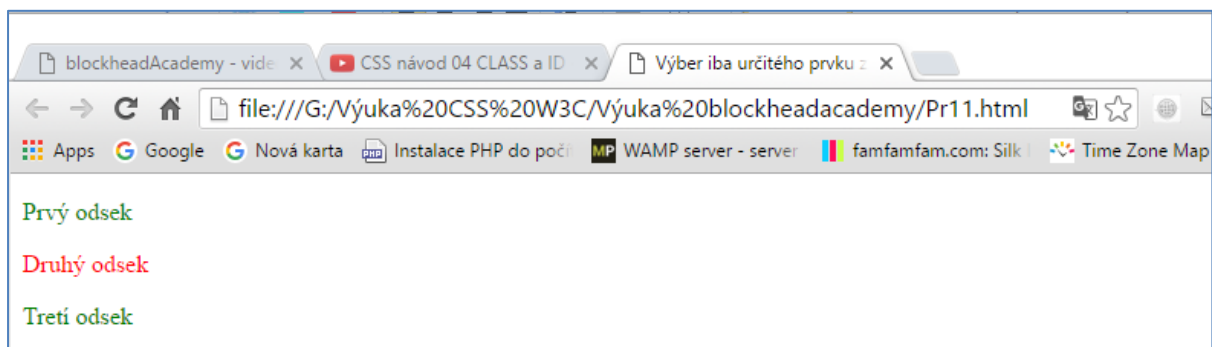
main.css

```

/*súbor main.css*/
/*Nastavenie iba určitého prvku s prechádzajúceho výberu*/
p {
  color: green;
}
#cerveny {
  color: red;
}

```

Výstup:



Prvý a tretí paragraf bude zelený a druhý vybratí identifikátorom id="cerveny" bude červený.

Identifikátor s určitým menom môžeme použiť v HTML dokumente iba jeden krát

Môžeme opäť použiť identifikátor ale s iným menom.

Identifikátor v css dokumente začíname značkou #(hash)

Pr12.Použitie id pre výber elementov

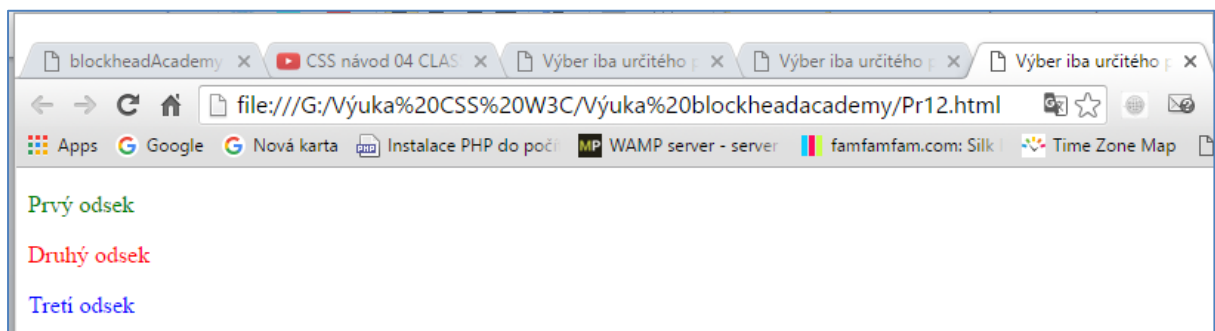
subor12.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Výber iba určitého prvku zo všetkých rovnakých prvkov</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Prvý paragraf</p>
    <p id="cerveny">Druhý paragraf</p><!--Na id sa odvolávame v súbore css-->
    <p id="modry">Tretí paragraf</p>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie iba určitého prvku s prechádzajúceho výberu pomocou
identifikatoru id*/
p {
  color: green;
}
#cerveny {
  color: red;
}
#modry {
  color: blue;
}
```

Výstup:



.....TU POKRAČOVAŤ VO ŠTVRTOK 6.Októbra 2016.....

!!!!!!!!!!!!!!!!!!!!Použiť ipsum loren!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

Čo ak budeme chcieť, aby iba text v prvom a druhom odstavci, mal inú farbu. K tomu nám môžu pomôcť triedy(class).

Triedy nám umožnia mať jeden názov triedy pre viac elementov.

```
Syntax v HTML súbore: <p class="modry">Prvý paragraf</p>
                        <p class="modry">Druhý paragraf</p>
                        <p>Tretí paragraf</p>
```

```
Syntax v CSS súbore:  .modry{
                        color:blue;
                        }
```

Pr.13 Použitie triedy pre výber prvkov

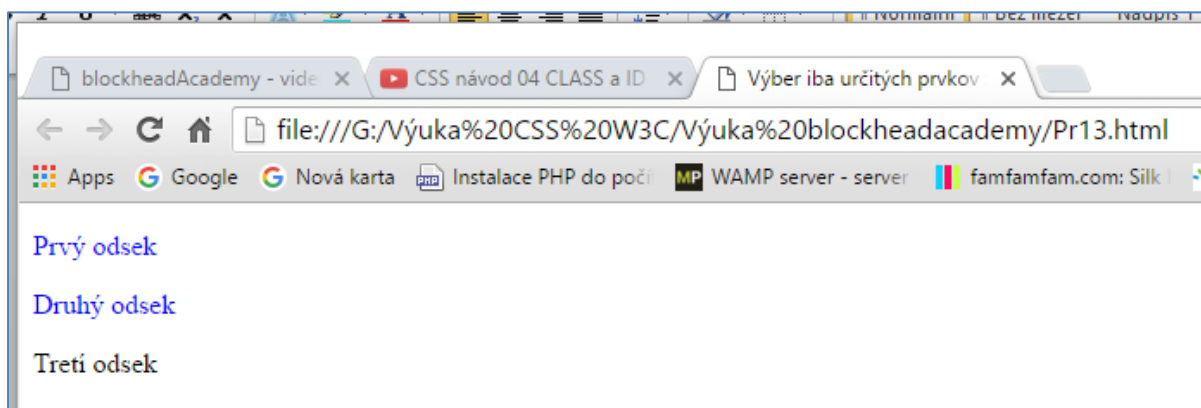
subor13.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Výber iba určitých prvkov zo všetkých rovnakých prvkov</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p class="modry">Prvý paragraf</p>
    <p class="modry">Druhý paragraf</p>
    <p>Tretí paragraf</p>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie skupiny prvkov pomocou triedy class s názvom modry */
.modry{
  color:blue;
}
```

Výstup:



Takýmto jednoducho si môžeme nadefinovať triedy s rôznymi názvami napr. pre odstavce, odkazy, kontajnery. Potom ich môžeme jednoducho priraďovať prvkom, kde budeme chcieť aby sa nám konkrétne štily prejavili.

Samozrejme i keď máme priradené triedy môžeme použiť identifikátory.

Pr14. Výber prvkov kombináciou identifikátorov a tried

subor14.html

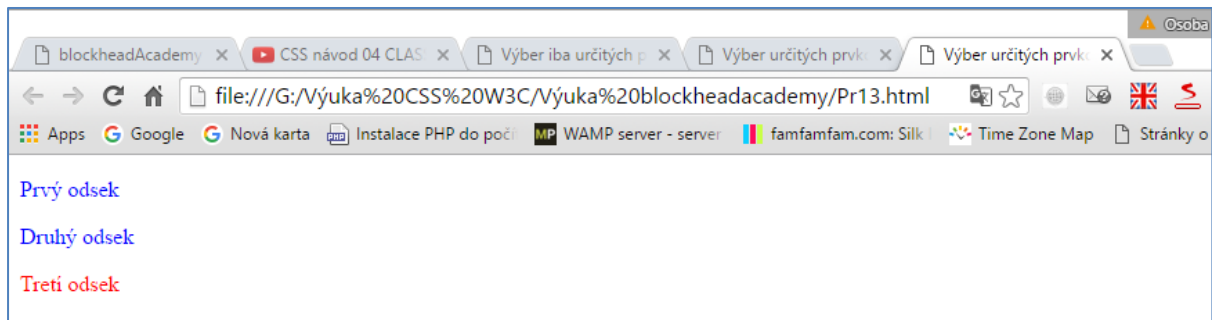
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Výber určitých prvkov pomocou id a class</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p class="modry">Prvý paragraf</p>
    <p class="modry">Druhý paragraf</p>
    <p id="cerveny">Tretí paragraf</p>
  </body>
</html>
```

main.css

```
/*súbor main.css*/
/*Nastavenie skupiny prvkov pomocou triedy a identifikátorov */
.modry{
  color:blue;
}
```

```
#cerveny{
  color:red;
}
```

Výstup



Cvičenie1-Na nastavenie farieb pomocou ID a class

5.CSS návod 05 Ďalšie možnosti výberu elementov

Komplikovanejšie možnosti výberu elementov.Použitie kontajneru pre výber elementov.

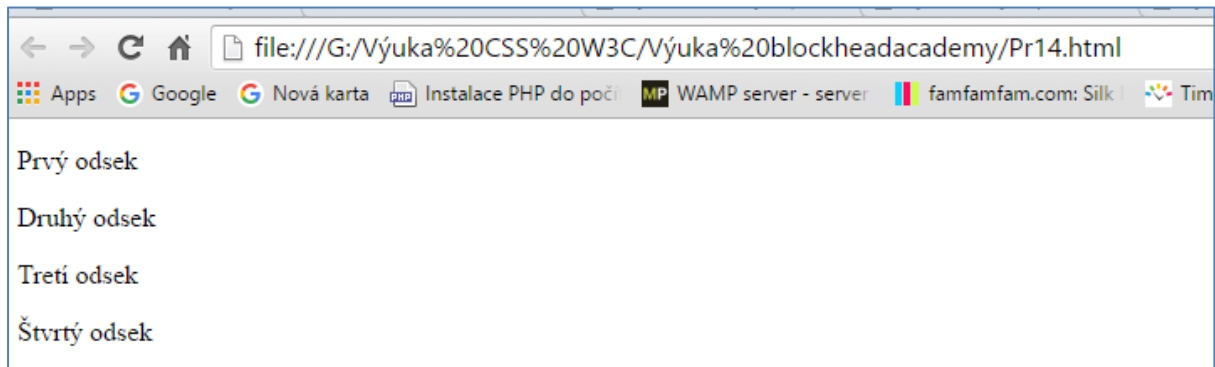
```
<div>
  <p>Prvý paragraf</p>
  <p>Druhý paragraf</p>
</div>
```

Pr15 Použitie kontajnera na výber prvkov

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia kontajnera pre výber prvkov </title>
  </head>
  <body>
    <div>
      <p>Prvý paragraf</p>
      <p>Druhý paragraf</p>
```

```
</div>
<p>Tretí paragraf</p>
<p>Štvrtý paragraf</p>
</body>
</html>
```

Výstup:



Použitím takého to divu sa nič nerieši. Výstup je podľa predefinovaného nastavenia HTML dokumentu.

Teraz budeme chcieť aplikovať neaký štýl iba na paragrafy ktoré sa nachádzajú v kontajnery **div** čiže na Prvý a Druhý paragraf.

Pr16. Aplikácia kontajnera na výber elementov **p**

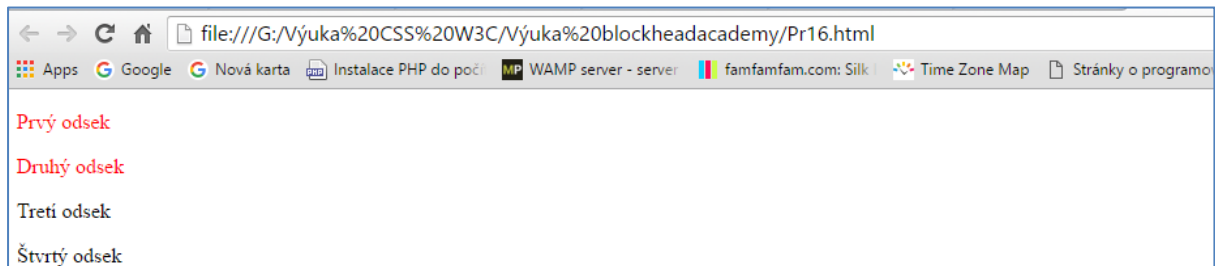
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia kontajnera pre výber prvkov </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>
      <p>Prvý paragraf</p>
      <p>Druhý paragraf</p>
    </div>
    <p>Tretí paragraf</p>
    <p>Štvrtý paragraf</p>
  </body>
</html>
```

```

/*súbor main.css*/
/*Nastavenie skupiny prvkov pomocou kontajnera */
div p {
    color:red;
}

```

Výstup:



Štýl sa nám aplikoval iba na prvé dva odstavca.Špecifikáciou cesty môžeme aplikovať štýly iba na určité vybrané skupiny elementov.V tomto príklade sme chceli aby sa štýly aplikovali iba na paragrafy ,ktoré sú umiestnené v kontainery..

Odstavce , ktoré neboli umiestnené v kontajneri zostali bez zmeny.

Pr.17.Ako elementom priradiť viacej tried

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia dvoch tried naraz </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>
      <p class="text">Prvý paragraf</p>
      <p class="text">Druhý paragraf</p>
    </div>
    <p class="text pozadie">Tretí paragraf</p>
    <p class="text pozadie">Štvrtý paragraf</p>
  </body>
</html>

```

```

/*súbor main.css*/
/*Máme triedu pre nastavenie farby textu a triedu pre pozadie*/
.text {

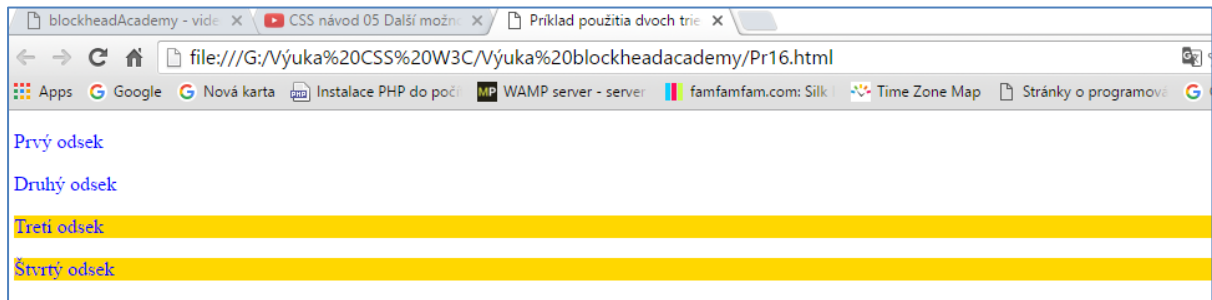
```

```

        color:blue;
    }
    .pozadie{
        background-color:gold;
    }

```

Výstup:



Vidíme ,že pre Prvý a Druhý paragraf sa zmenila iba farba textu na modrú.Na Tretí a Štvrtý paragraf sa aplikovala trieda `text` aj trieda `pozadie`.Teda zmenila sa farba textu aj pozadia.Text na modrú a súčasne pozadie na zlatú.

Takto jednoducho je možné triedy v elemente kombinovať.

Teraz priradíme nášmu kontajneru identifikátor id.Budeme chcieť zväčšiť text u paragrafov ktoré sa v tomto kontajneri nachádzajú.

Pr.18.Ako elementom v kontajneri zväčšiť písmo

```

!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia kontajnera na zmenu veľkosti textu v
paragrafoch </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div id="kontajner">
      <p class="text">Prvý paragraf</p>
      <p class="text">Druhý paragraf</p>
    </div>
    <p class="text pozadie">Tretí paragraf</p>
    <p class="text pozadie">Štvrtý paragraf</p>
  </body>
</html>

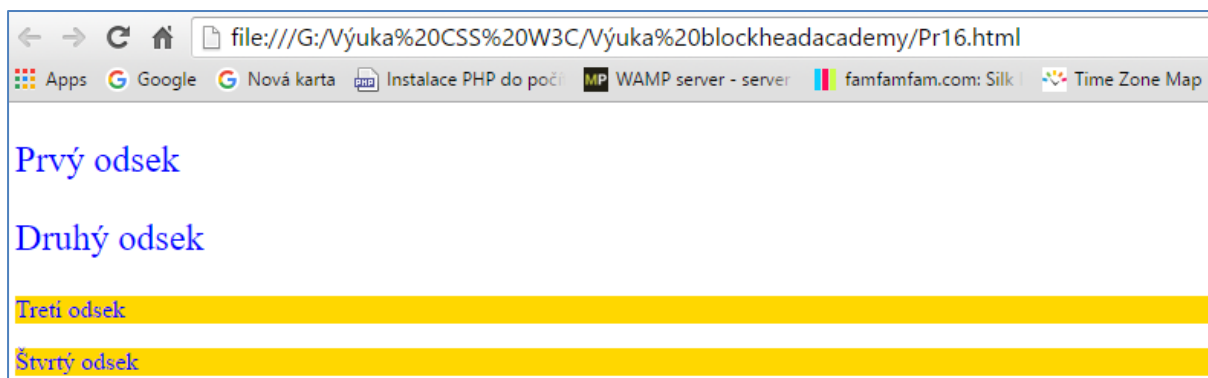
```

```

/*súbor main.css*/
/*Máme triedu pre nastavenie farby textu a triedu pre pozadie
a výberový prvok kontajner pre zväčšenie veľkosti písma v paragrafoch ktoré
sú umiestnené v kontajnery*/
.text {
    color:blue;
}
.pozadie{
    background-color:gold;
}
#kontajner p{
    font-size:150%;
}

```

Výstup:



V paragrafoch Prvý a Druhý ktoré sú v kontajnery sa zmenila veľkosť písma a súčasne sa na ne aplikuje trieda **text**, ktorá spôsobí zmenu farby textu na modrú.

V paragrafoch Tretí a Štvrtý sa zmení farba textu na modrú a farba pozadia na zlatú. Kontajner s id="kontajner" sa na Treťom a Štvrtom paragrafe neprejaví.

Takto vyberieme identifikátor(kontajner) a potom elementy na ktoré sa má identifikátor aplikovať.

Pr.19.Použitie viacerých kontajnerov

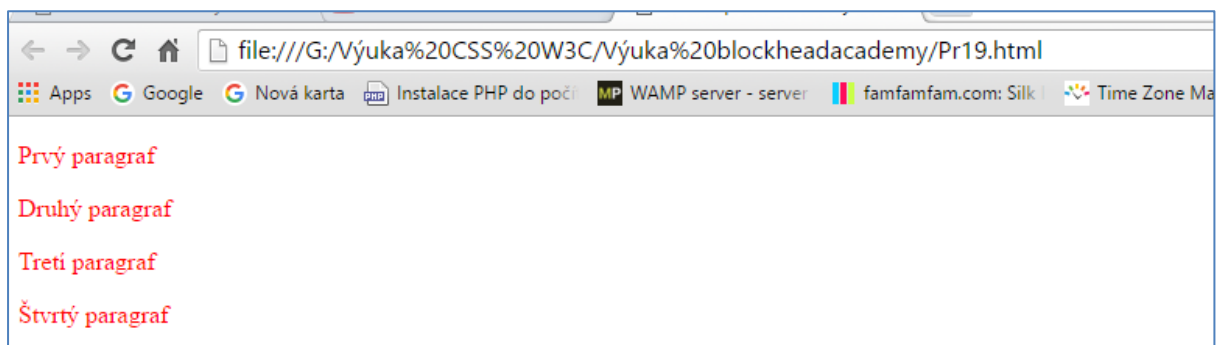
Zadefinujeme si štyri odstavce,uzatvoríme ich do kontajnera a kontajneru priradíme identifikátor id="hlavny".Tretí a Štvrtý paragraf uzatvoríme tiež do vnoreného kontajnera

Všetky štyri paragrafy sa nastavujú podľa #hlavny.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia viacerých kontajnerov v paragrafoch </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div id="hlavny">
      <p>Prvý paragraf</p>
      <p>Druhý paragraf</p>
      <div>
        <p>Tretí paragraf</p>
        <p>Štvrtý paragraf</p>
      </div>
    </div>
  </body>
</html>

/*súbor main.css*/
/* nastavenie hlavného kontajnera pre všetky paragrafy*/
#hlavny p{
  color:red;
}
```

Výstup:



Pri tomto nastavení kontajnera s id="hlavny" sú všetky paragrafy Prvý až Štvrtý nastavené na červenú farbu.

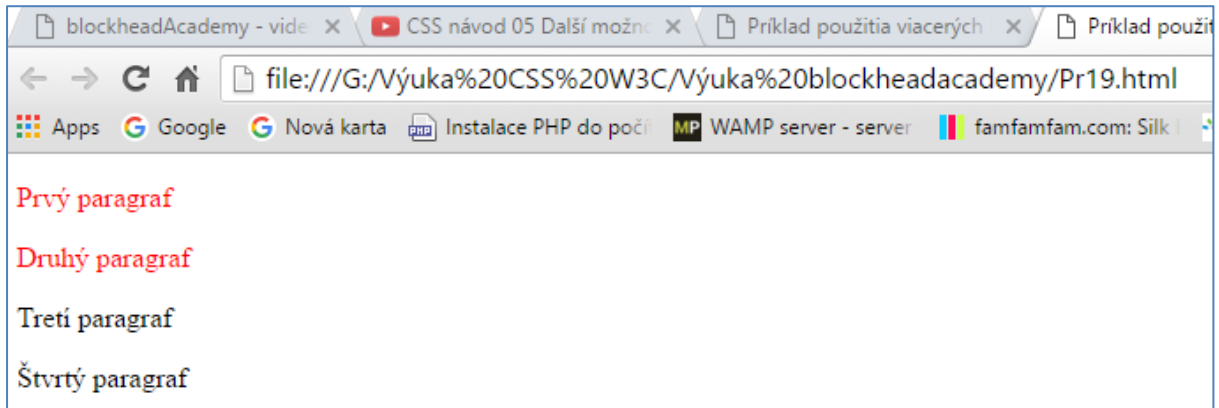
Zmeníme výber prvkov nasledovne:

```
/*súbor main.css*/
/* Nastavenie sa vzťahuje iba na priamych potomkov identifikátora "hlavny"
*/
#hlavny>p{
```

```
    color: red;
}
```

Selektor výberu **hlavny** sa vzťahuje iba na priamych potomkov teda paragraf Prvý a Druhý.

Výstup:



Iba na paragrafy, ktoré sú priamym potomkom identifikátora „hlavny“, nášho kontajnera, sa bude aplikovať nami definovaný štýl - červená farba.

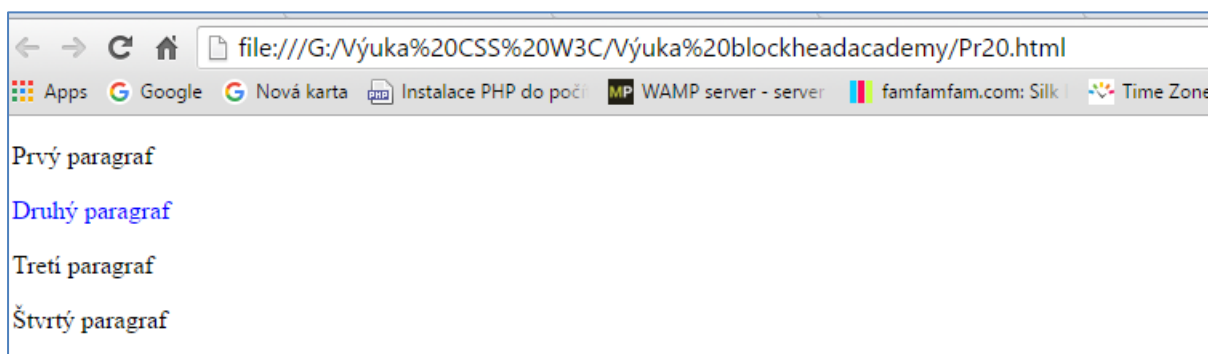
Pr.20 Použitie ďalšieho spôsobu výberu elementov

```
!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia ďalšieho výberového prvku </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Prvý paragraf</p>
    <p id="druhy">Druhý paragraf</p>
    <p>Tretí paragraf</p>
    <p>Štvrtý paragraf</p>

  </body>
</html>
```

```
/*súbor main.css*/
/*Výber identifikátorom druhy */
#druhy{
    color:blue;
}
```

Výstup:

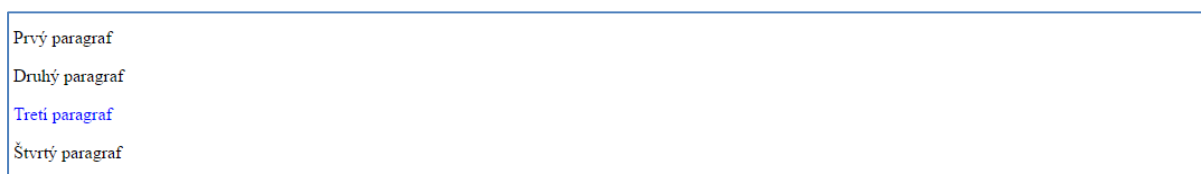


Ako sme očakávali vyberie sa iba paragraf s id="druhý" a nastaví vlastnosť color na modrú farbu.

A čo ak nastavíme main.css nasledovne:

```
/*súbor main.css*/  
/*Výber identifikátorom + p */  
#druhý+p{  
    color:blue;  
}
```

Výstup:



Ako vidieť vyberie sa ďalší paragraf za paragrafom s identifikátorom „druhý“.

Všeobecne: Ak zadáme za identifikátorom + a značku prvku ktorého sa to má týkať ,vyberie sa nasledujúci prvok a na ňom sa aplikuje definovaný štýl.

To isté sa stane,keby selektorom nebol identifikátor ,ale napríklad trieda.

PR.22 Príklad výberu susedného prvku pomocou triedy

```
!DOCTYPE html>  
<html lang="sk">  
  <head>  
    <meta charset="UTF-8">
```

```
<title>Príklad použitia ďalšieho výberového prvku </title>
<link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
<p>Prvý paragraf</p>
<p class="druhy">Druhý paragraf</p>
<p>Tretí paragraf</p>
<p>Štvrtý paragraf</p>
</body>
</html>
```

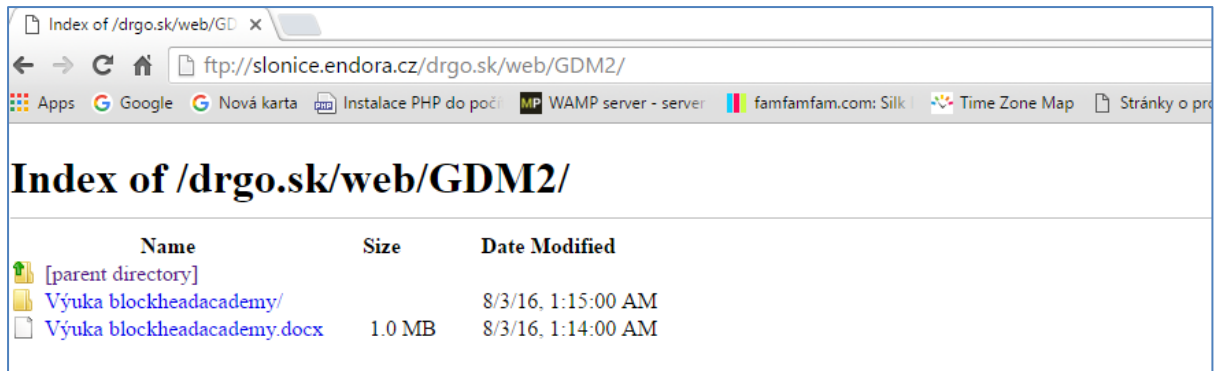
```
/*súbor main.css*/
/*Výber triedou + p */
.druhy+p{
    color:blue;
}
```

Výstup:



Vidíme , že výsledok je rovnaký ako keď sme použili na výber identifikátor, Opäť sa vyberie nasledujúci prvok ,prvku zadaného v súbore main.css.

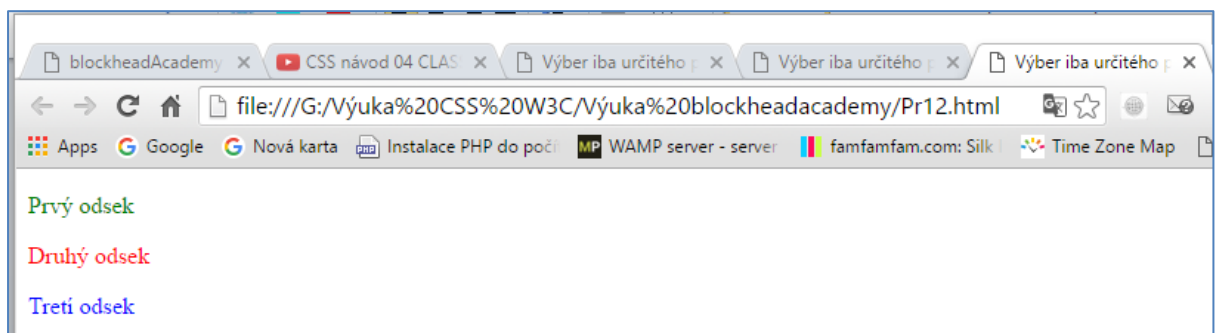
Ako stiahnuť tento materiál :[Výuka blockheadacademy.docx súbor](#) z internetu:



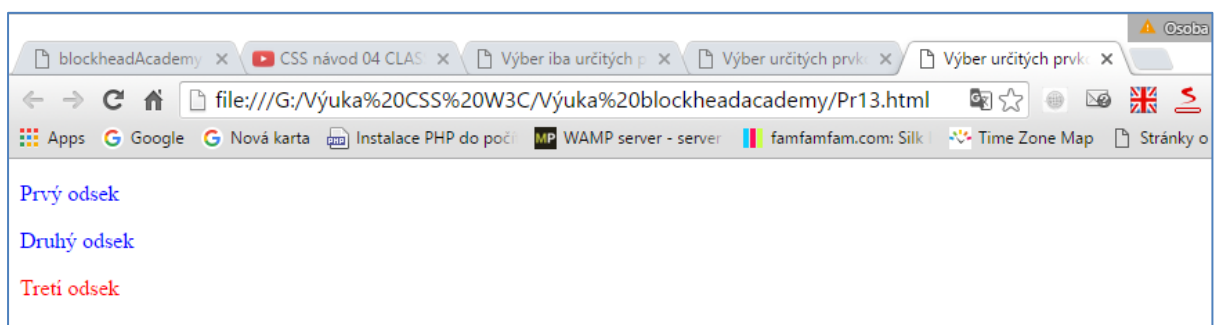
.....Tu pokračovať cez ďalšie EPU.17.októbra 2016.....

OPAKOVANIE

1.



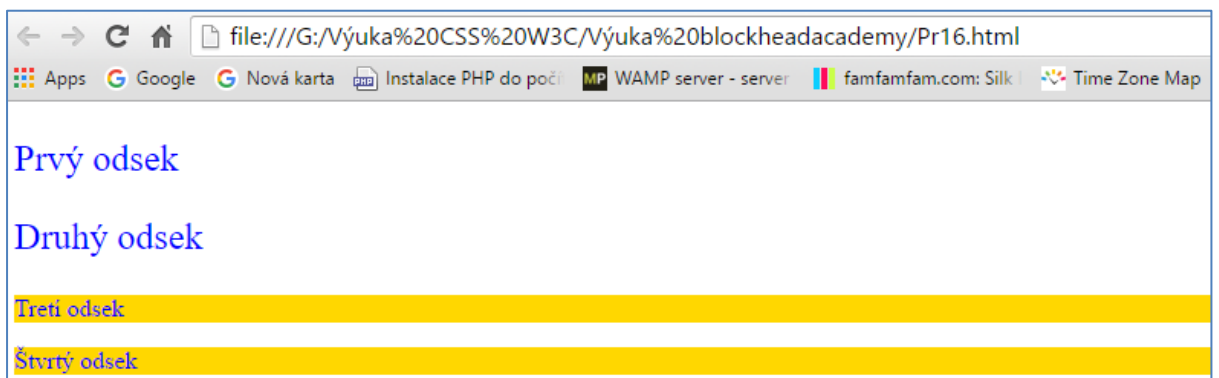
2.



3.



4.



Body	Známka
4	1
3	2
2	3
1	4
0	5

P.č.	Priezvisko	1	2	3	4	Body	Známka
1	Bazalová	1	1	1	1	4	
2	Fúseková	1	1	1	1		
3	Kapriová	1	1	1	1		
4	Krajčo	1	1	1	1		
5	Minarovič	1	1	1	1		
6	Orth	1	1	1	1		
7	Pilný	1	1	1	1		
8	Podoba	-	-	-	-		
9	Prílesan	1	1	1	1		
10	Pristaš	1	1	1	1		
11	Puček	1	1	1	1		
12	Špatný	1	1	1	1		
13	Štefková	1	1	1	1		
14	Žiaček	1	1	1	1		
15	Dašková	-	-	-	-		
16	Galovičová	-	-	-	-		
17							

.....Tu pokračovať cez ďalšie EPU.20.októbra 2016.....

6. Odkazy a pseudo triedy

Čo sú to pseudotriedy? Sú to triedy, ktorá nám pridávajú k niektorým elementom špeciálne vlastnosti. Najviac využijeme pseudotriedy pre odkazy.

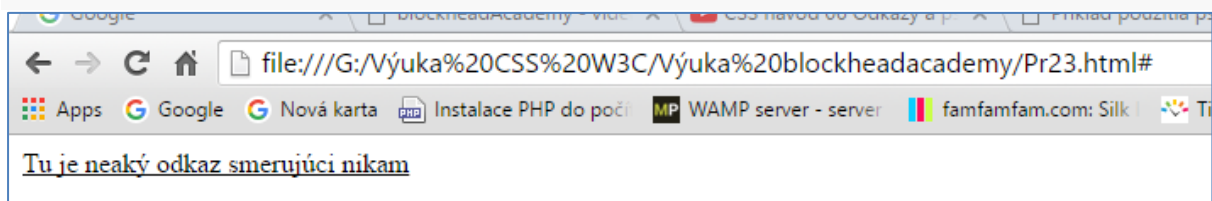
Určite ste si všimli na stránke jednoduchého efektu ,kde pri prejdutí myši sa zmení farba textu alebo pozadia pri odkaze. To môžeme dosiahnuť pomocou pseudotriedy. Toto vnesie do stránky určitú dynamiku.

Pr23. Nastavenie farby odkazu pomocou pseudotried.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia pseudotried na nastavenie odkazu</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <a href="#">Tu je neaký odkaz smerujúci nikam</a>
  </body>
</html>
```

```
/*súbor main.css*/
/*Nastavenie odkazu */
a {color:black;
}
```

Výstup:



Farba odkazu je teraz čierna a nezmení sa ani pri prejdutí myšou.

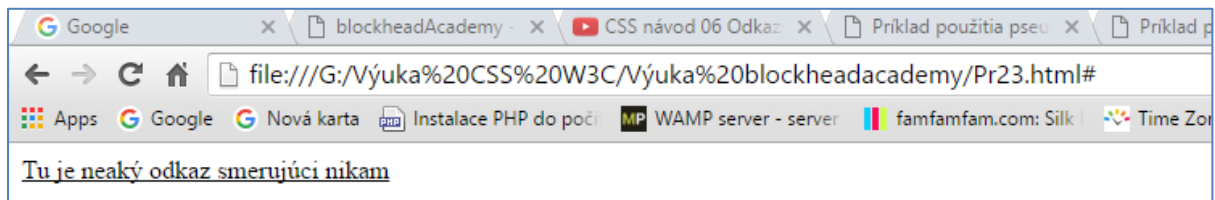
Ako dosiahnúť doho, aby sa po prejdutí myšou zmenila farba textu odkazu na inú?

Použijeme pseudotriedu : `a:hover` {

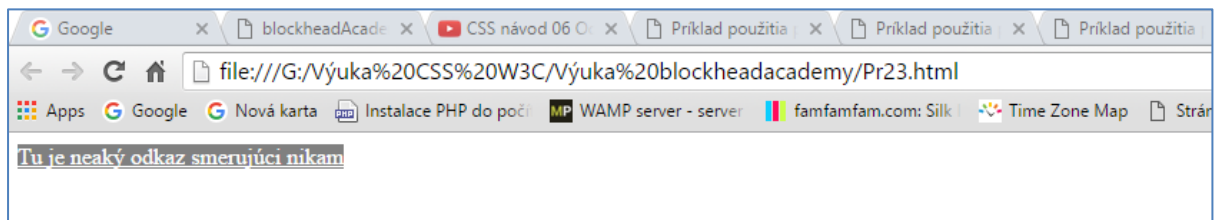
,ktorý znamená ,že po nastavení kurzora na text odkazu sa farba zmení na nastavenú farbu.

```
/*súbor main.css*/
/*Nastavenie odkazu */
a {color:black;/*Základná farba textu odkazu čierna*/
}
a:hover {
    color:white;
    background-color:gray;
}
```

Výstup: Základná farba odkazu:



A farba po nabehtutí kurzorom



Zmení sa farba textu odkazu na bielu a aj farba pozadia odkazu na zmení na šedú.

Tým môžeme vytvoriť jednoduchý dynamický efekt.

Určite ste si už všimli , že hypertextový odkaz môže mať niekoľko statusov:

- Nenvšitvený odkaz
- Navšitvený odkaz
- Reakcia na myš
- Vybraný odkaz

Každý tento status môžeme pomocou pseudotried definovať.Teraz si to ukážeme.

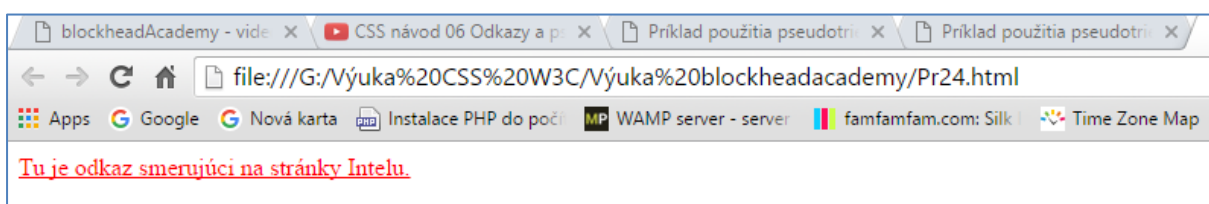
Pr24 Nastavenie statusov hyperlinkových odkazov

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia pseudotried na nastavenie odkazu</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <a href="http://intel.com">Tu je odkaz smerujúci na stránky Intelu.</a>
  </body>
</html>
```

```
/*súbor main.css*/
/*Nastavenie odkazu */
a {color:black; /*Základná farba textu odkazu čierna*/
}
a:link {
  /* nastavenie štýlu pre nenavštívený odkaz*/
  color:black;
}
/*Nastavenie farby aktívneho odkazu*/
a:active {
  color:gold ;
}
/*Nastavenie štýlu pre navštívený odkazu */
a:visited {
  color:red;
}
/*reakcia na nabehnutie kurzora na odkaz*/
a:hover {
  background-color:gray;
}
```

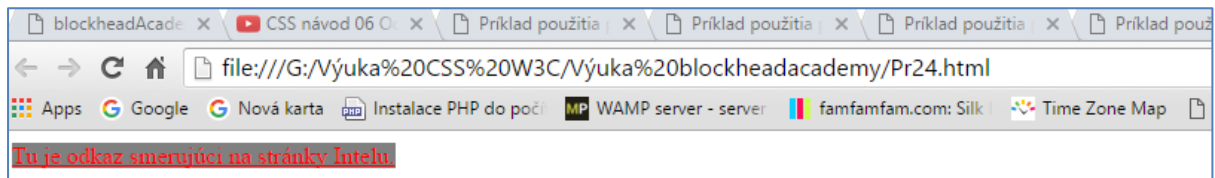
Vystup

Základné nastavenie odkazu:



Farba je červená lebo som odkaz už predtým navštívil. Teda nejde na odkaz prvý krát. Ak by som išiel na odkaz prvý krát bol by odkaz čierny.

Ak presuniem kurzor na odkaz:



Farba odkazu zostane ako u navštíveného odkazu ale sa zmení farba pozadia.

V definícii pseudotriet pre nastavenie odkazu musíme dodržať toto poradie pseudotriet:

1. `a:link`
2. `a:active`
3. `a:visited`
4. `a:hover`

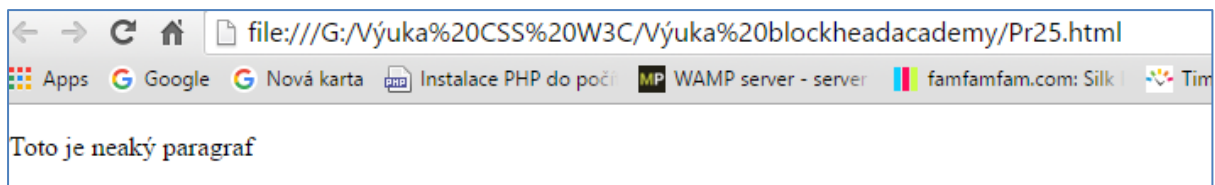
Pseudotriedu hover môžeme použiť aj u iných elementov ako je odkaz.

Pr25 Použitie pseudotriedy hover pre paragraf

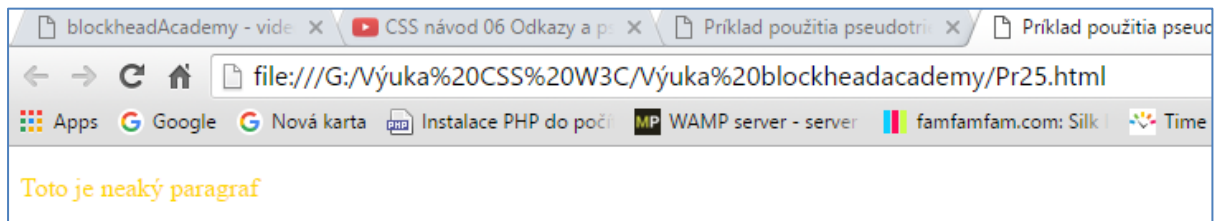
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia pseudotriet na nastavenie paragrafu</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Toto je neaký paragraf</p>
  </body>
</html>
```

```
súbor main.css*/
/*Nastavenie pseudotriedy hover pre paragraf */
p:hover {
  color:gold;
}
```

Výstup: Základná farba textu, je daná nastavením prehliadača



A toto je po nabehtutí kurzora na paragraf



Pr25 Použitie pseudotriedy focus pre výber formulárovho prvku kliknutím.

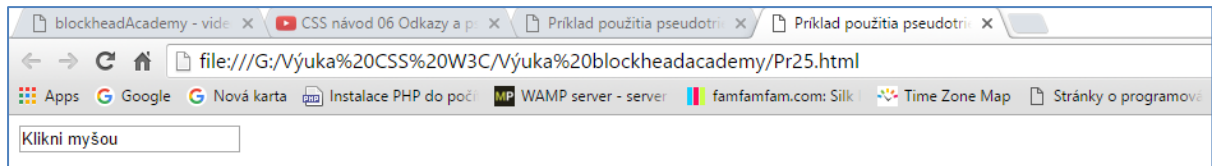
Vložíme do HTML documentu formulárový prvok **INPUT**. Bude typu **text**, ako hodnotu bude mať text „klikni myšou“. Po použití pseudotriedy focus sa kliknutím na prvok vyberie INPUT.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad použitia pseudotriedy focus na výber formulárového
prvku</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <input type="text" value="Klikni myšou"></p>
  </body>
</html>
```

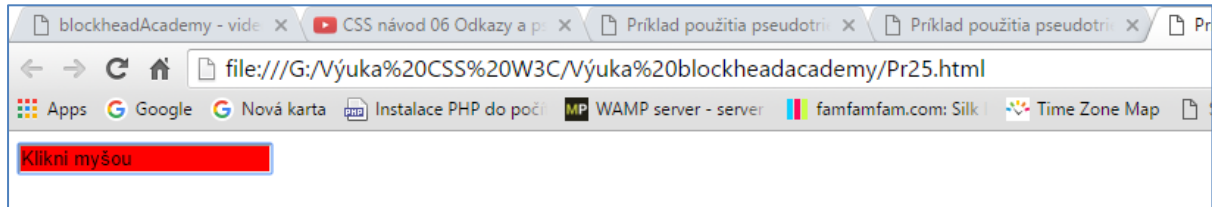
```
/*súbor main.css*/
/*Nastavenie pseudotriedy focus pre výber formulárovho prvku INPUT */
input:focus{
  background-color:red;
}
```

Výstup:

Bez použitia pseudotriedy focus



Kliknutím na prvok INPUT



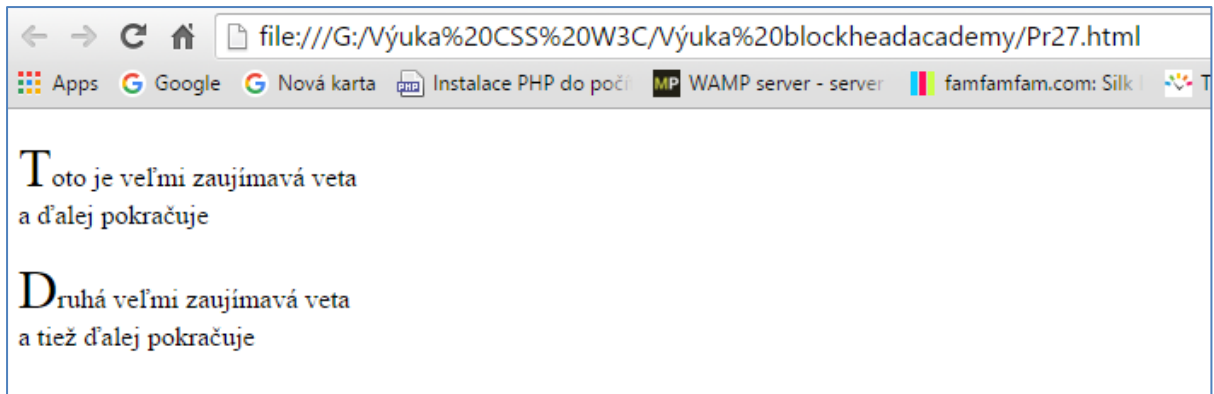
7.CSS návod 07 Pseudo triedy pokračovanie

Pr27 Ako zvýrazniť prvé písmeno z každého odstavca

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zvýraznenie prvého písmena v každom paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p> Toto je veľmi zaujímavá veta <br> a ďalej pokračuje </p>
    <p> Druhá veľmi zaujímavá veta <br> a tiež ďalej pokračuje</p>
  </body>
</html>
```

```
/*súbor main.css*/
/*Nastavenie prvého písmena veľkého v každom paragrafe */
p:first-letter{
  font-size:200%;
}
```

Výstup:

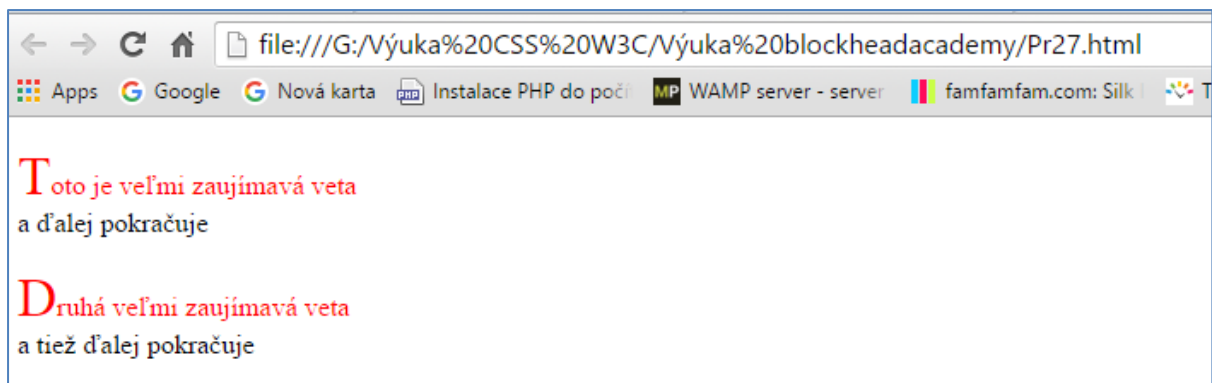


Pr28 Ako zvýrazniť prvý riadok odstavca

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
<title>Príklad na zvýraznenie prvého písmena v každom paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p> Toto je veľmi zaujímavá veta <br> a ďalej pokračuje </p>
    <p> Druhá veľmi zaujímavá veta <br> a tiež ďalej pokračuje</p>
  </body>
</html>
```

```
/*súbor main.css*/
/*Nastavenie prvého písmena veľkého v každom paragrafe */
p:first-letter{
  font-size:200%;
}
/*Nastavenie prvého riadku každého paragrafu*/
p:first-line{
  color:red;
}
```

Vystup:



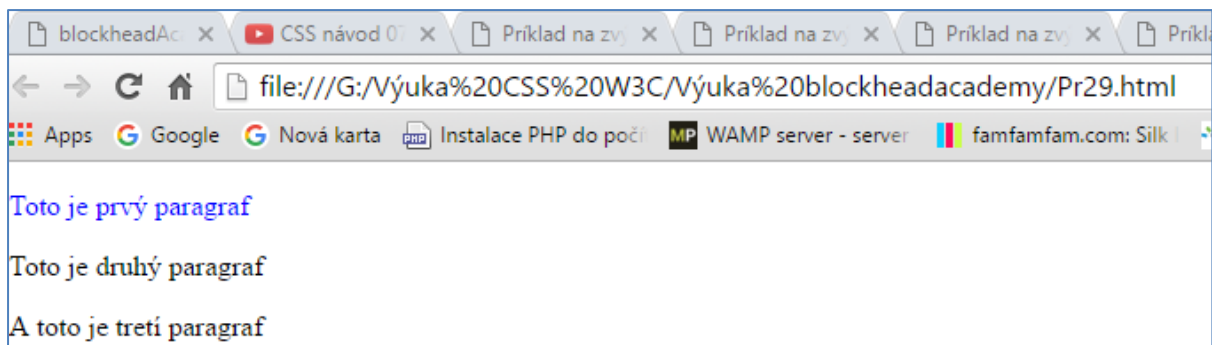
Pr29 Výber prvého potomka v kontajnery

Pre ďalší príklad si vytvoríme kontajner `<div>` a do neho vložíme tri paragrafy. Potom vyberieme prvého potomka kontajnera teda prvého odstavca z celkového počtu paragrafov.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zvýraznenie prvého odstavca v kontajnery</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>
      <p> Toto je prvý paragraf</p>
      <p> Toto je druhý paragraf</p>
      <p> A toto je tretí paragraf</p>
    </div>
  </body>
</html>
```

```
/*súbor main.css*/
/*Vybratie prvého potomka v kontajnery */
p:first-child{
  color:blue;
}
```

Výstup:

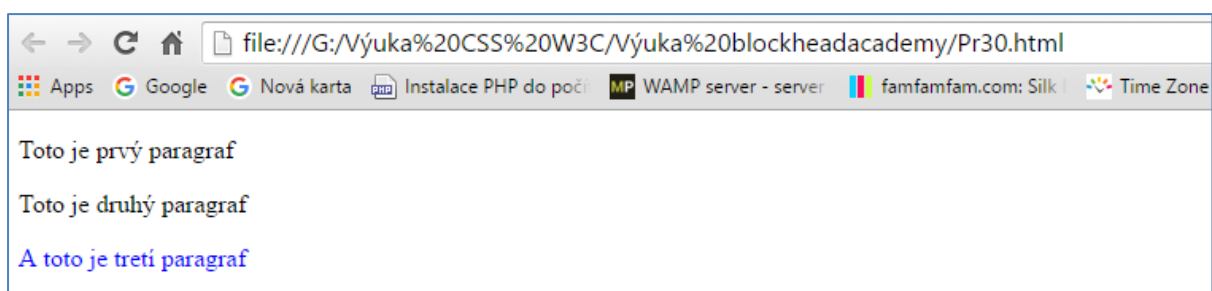


Pr30 Výber posledného potomka v kontajneri

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zvýraznenie prvého odstavca v kontajneri</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>
      <p> Toto je prvý paragraf</p>
      <p> Toto je druhý paragraf</p>
      <p> A toto je tretí paragraf</p>
    </div>
  </body>
</html>
```

```
/*súbor main.css*/
/*Vybratie posledného potomka v kontajneri */
p:last-child{
  color:blue;
}
```

Výstup:

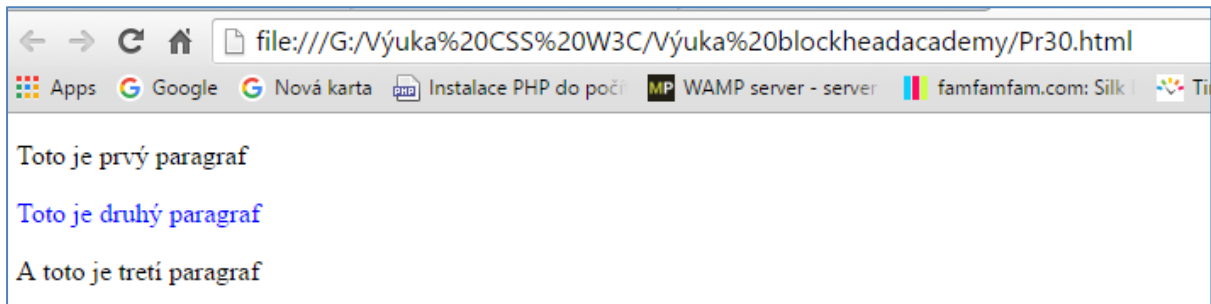


Pr31 Výber druhého potomka v kontajneri

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zvýraznenie prvého odstavca v kontajneri</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>
      <p> Toto je prvý paragraf</p>
      <p> Toto je druhý paragraf</p>
      <p> A toto je tretí paragraf</p>
    </div>
  </body>
</html>
```

```
/*súbor main.css*/
/*Vybratie určitého potomka v kontajneri */
/*My si vyberieme druhého potomka v poradí paragrafov*/
p:nth-child(2) {
  color:blue;
}
```

Výstup:



Opustíme teraz pseudo – triedy a ukážeme si ďalšie možnosti výberu prvkov.

Pr32 Voľba elementu pomocou jeho atribútu

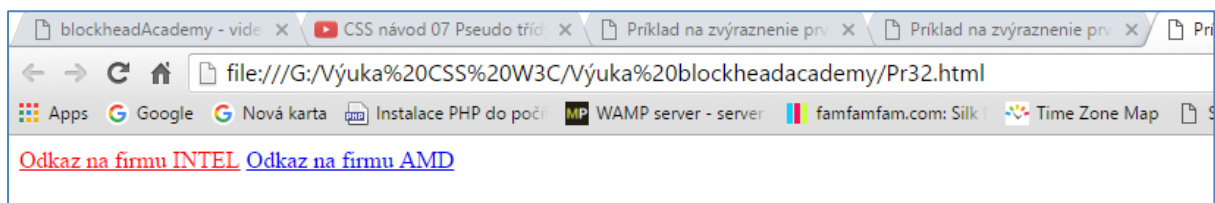
Pre ďalší príklad si vytvoríme dva odkazy. Prvý na intel.com a druhý na amd.com

Ukážeme si ako zvoliť element pomocou atribútu a jeho hodnoty.

Najskôr uvidíme značku odkazu **a** a potom do hranatej zátvorky uvidíme názov atribútu **[href="www.adresa tak ako bola uvedená v HTML dokumente"]** odkazu.

```
!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
  <title>Príklad na výber elementu a pomocou jeho atribútu href.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <a href="http://www.intel.com">Odkaz na firmu INTEL</a>
    <a href="http://www.amd.com">Odkaz na firmu AMD</a>
  </body>
</html>
```

Výstup:



Teda odkaz bol vybraný pomocou atribútu a jeho hodnoty. Odkaz je červenou farbou. Ak by sme do main.css uviedli ako hodnotu atribútu href www.amd.com,potom by červenou farbou textu bol vyznačený odkaz na amd.com.

!!!Ak tieto adresy nefungujú skúste napr. www.google.com a www.zoznam.sk!!!!

Pr33 Voľba elementu pomocou jeho atribútu title.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
  <title>Príklad na výber elementu a pomocou jeho atribútu href.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
```

```

<body>
  <a href="http://www.intel.com">Odkaz na firmu Intel</a>
  <a href="http://www.amd.com" title="Toto je odkaz na server AMD">Odkaz
na firmu AMD</a>
</body>
</html>

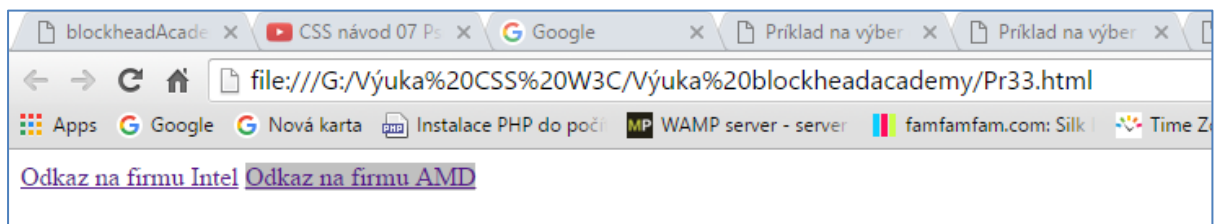
```

```

/*súbor main.css*/
/*Vybratie určitého prvku pomocou jeho atribútu a jeho hodnotyery /
/* Konkrétne sa jedná o prvok "a" a jeho atribút title v ktorom sa má
nachádzať text "odkaz" */
a[title~="odkaz"]{
  background-color:silver;
  /* Ak bude v odkaze "a" titulok "odkazy",tak sa vyberie tento odkaz
a jeho text sa zobrazí striebornou farbou*/
}

```

Výstup:



Pr34 Voľba elementu pomocou jeho atribútu title,na konci title

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na výber elementu "a" pomocou textu na konci title.</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <a href="http://www.intel.com">Odkaz na firmu Intel</a>
    <a href="http://www.amd.com" title="Toto je server AMD odkaz">Odkaz na
firmu AMD</a>
  </body>
</html>

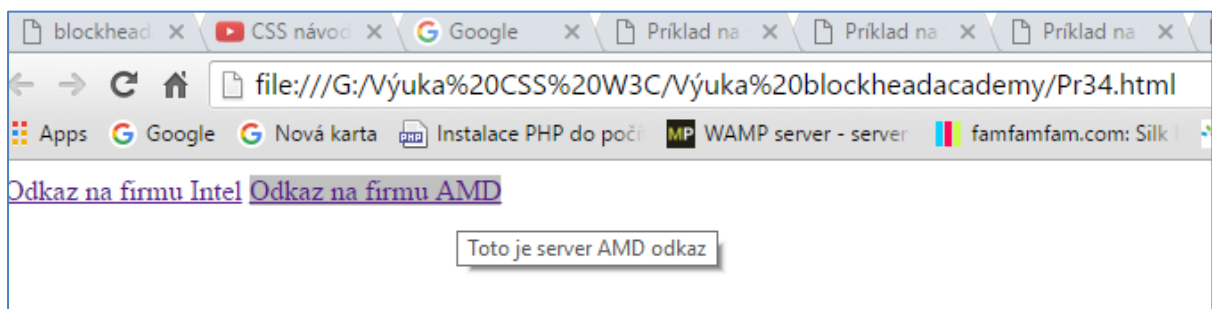
```

```

/*súbor main.css*/
/*Vybratie určitého prvku pomocou jeho atribútu a jeho hodnoty */
/* Konkrétne sa jedná o prvok "a" a jeho atribút title v ktorom sa má
nachádzať text "odkaz" na konci*/
a[title$="odkaz"]{
    background-color:silver;
    /* Ak bude v odkaze "a" titulok "odkazy",na konci title, tento
odkaz a jeho text sa zobrazí striebornou farbou*/
}

```

Výstup:



8.CSS návod 08 Základné práce s písmom

Ukážeme si ako pracovať v css s písmom.CSS nám umožní nastaviť pre písmo:

1. rodinu
2. veľkosť
3. hrúbku
4. štýl
5. Kapitálky

index.html

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Súbor pre cvičenie fontov</title>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">

```

```
<link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
<p>Dnes si ukážeme prácu s písmom</p>
</body>
</html>
```

1.Nastavenie fontu-rodiny písma

font-family

Definícia a použitie

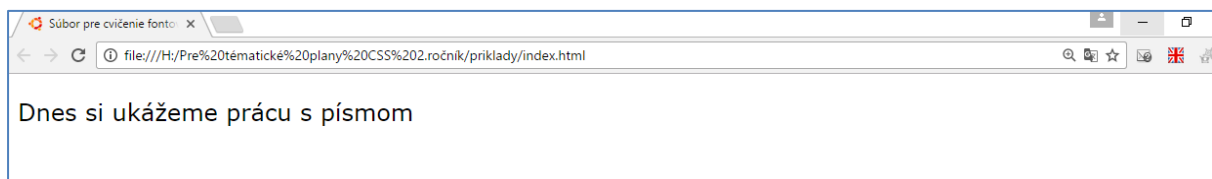
CSS vlastnosť font-family definuje použitie fontov písma v dokumente. Pokiaľ je definovaný font, ktorý u seba nemáte nainštalovaný, prehliadač použije ďalší definovaný. Keď font nie je v dokumente definovaný, prehliadač použije základné písmo daného systému, zvyčajne font Times.

Pokiaľ je v názve fontu medzera, musí sa názov fontu obaliť jednoduchými úvodzovkami

```
font-family: 'Museo Slab', Arial, Tahoma;
```

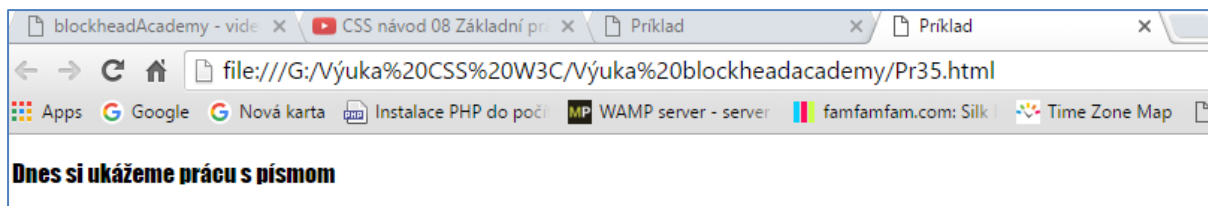
```
/*súbor main.css*/
body {
    font-family: Verdana, Arial, sans-serif; /*Nastavenie fontu pre celý
HTML dokument a aj s náhradnými fontami*/
}
```

Výstup:

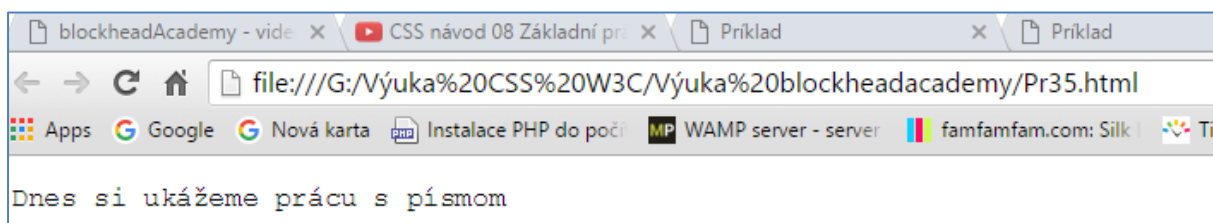


```
/*súbor main.css*/
body {
    font-family: Impact, Arial, sans-serif; /*Nastavenie fontu pre celý
HTML dokument a aj s náhradnými fontami*/
}
```

Výstup:



```
/*súbor main.css*/  
body {  
    font-family:Courier,Arial,sans-serif;/*Nastavenie fontu pre celý  
HTML dokument a aj s náhradnými fontami*/  
}
```



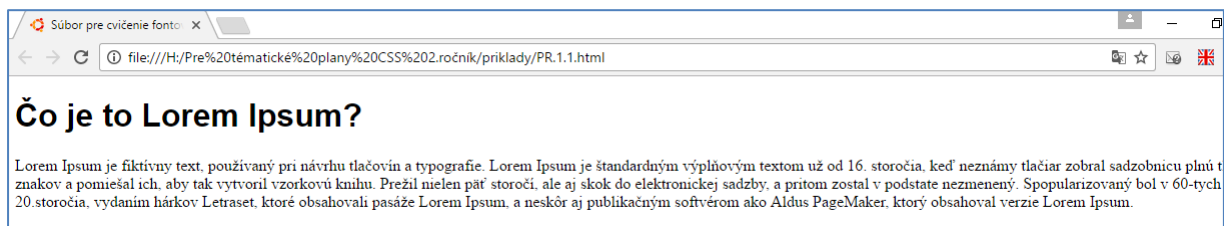
PR.1.1.html

```
<!DOCTYPE html>  
<html lang="sk">  
  <head>  
    <meta charset="UTF-8">  
    <title>Súbor pre cvičenie fontov</title>  
    <link href="images/logo.png" rel="shortcut icon" type="image/png">  
    <link rel="stylesheet" type="text/css" href="css/myStyle1.css"/>  
  </head>  
  <body>  
    <h1> Čo je to Lorem Ipsum?</h1>  
    <p>Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín a  
typografie. Lorem Ipsum je štandardným výplňovým textom už od 16. storočia,  
keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a pomiešal ich,  
aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí, ale aj skok do  
elektronickej sadzby, a pritom zostal v podstate nezmenený. Spopularizovaný  
bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset, ktoré obsahovali  
pasáže Lorem Ipsum, a neskôr aj publikačným softvérom ako Aldus PageMaker,  
ktorý obsahoval verzie Lorem Ipsum.</p>  
  </body>  
</html>
```

myStyle1.css

```
/*súbor myStyle1.css*/  
h1 {  
    font-family:"Arial";/*Nastavenie fontu pre hlavičku*/  
}  
p {  
    font-family:"Time New Roman";/*Nastavenie fontu pre odstavec*/  
}
```

Chrome:



2.Nastavenie veľkosti písma

font-size

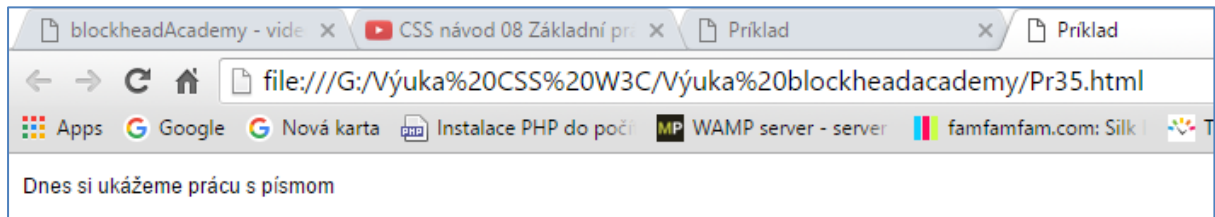
Definícia a použitie

CSS vlastnosť font-size definuje veľkosť fontu písma. Udáva sa v jednotkách, ale aj v definovaných konštantách a stupňoch.

I.Nastavenie veľkosti v pixeloch

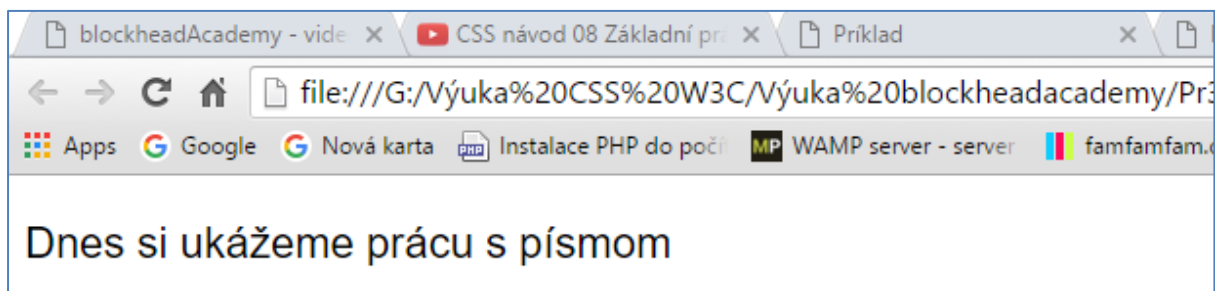
```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:12px;  
}
```

Výstup:



```
/*súbor main.css*/  
body {  
    font-family:Arial, sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:22px;  
}
```

Výstup:



II.Nastavenie veľkosti v „em“

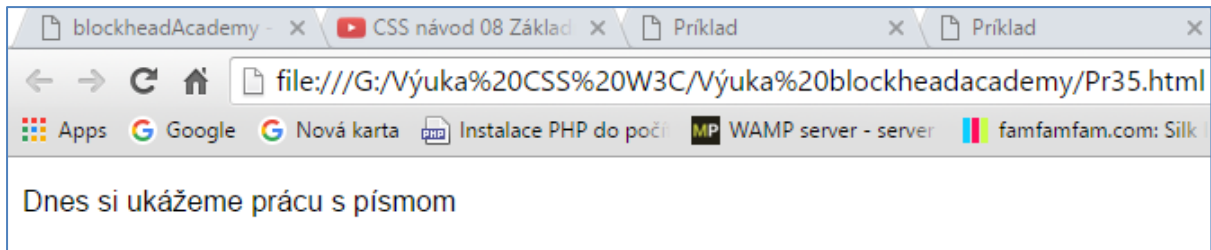
Predvolenú veľkosť textu v prehliadačoch je 16px. Takže, predvolená veľkosť 1em je 16px.

Veľkosť môže vypočítať z obrazových bodov na em pomocou

tohto vzorca: $\text{Body} / 16 = \text{em}$

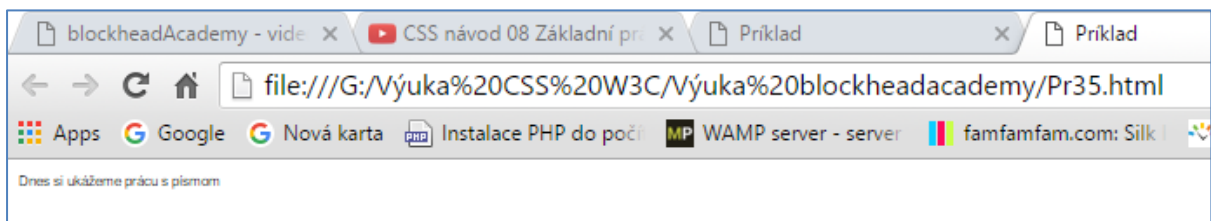
```
/*súbor main.css*/  
body {  
    font-family:Arial, sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:1 em;  
}
```

Výstup:



```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:0.5em;  
}
```

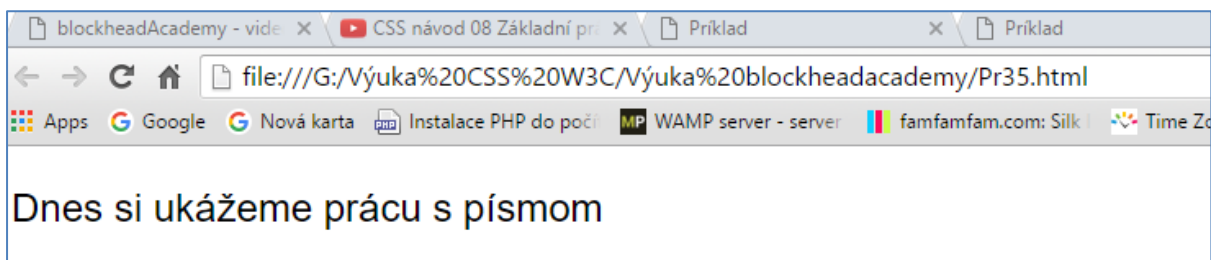
Výstup:



III.Nastavenie veľkosti v %

```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:150%;  
}
```

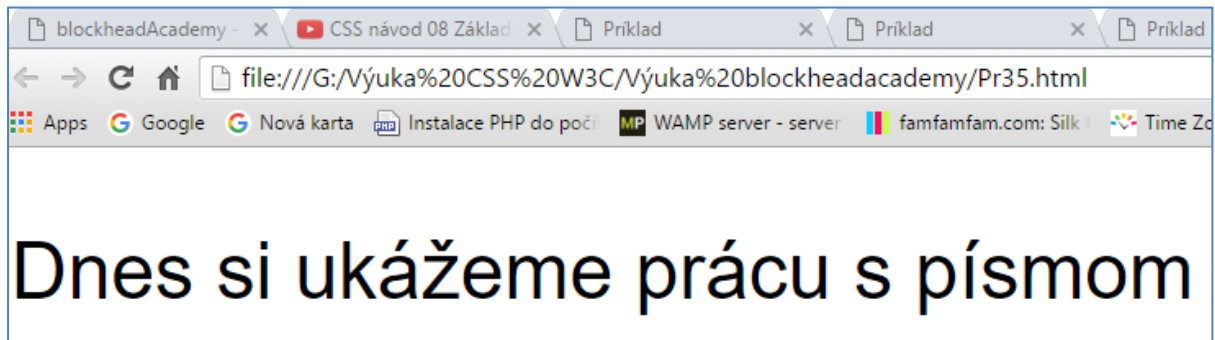
Výstup:



```
/*súbor main.css*/
```

```
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:300%;  
}
```

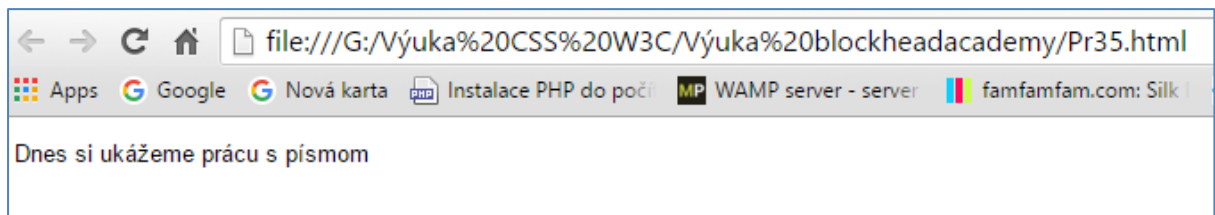
Výstup:



IV.Slovné nastavenie veľkosti

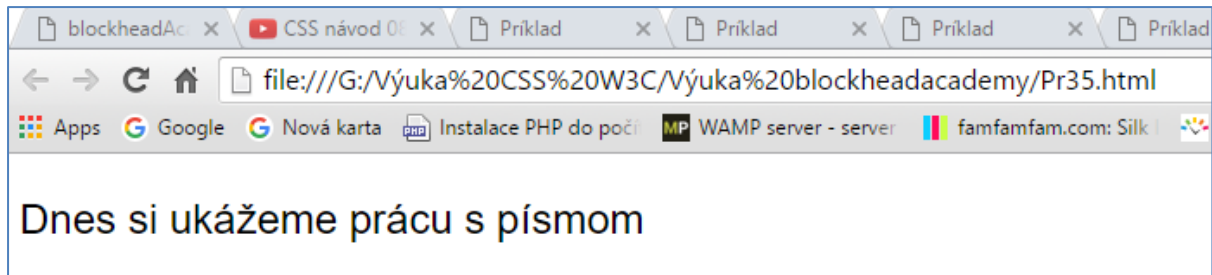
```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:small;  
}
```

Výstup:



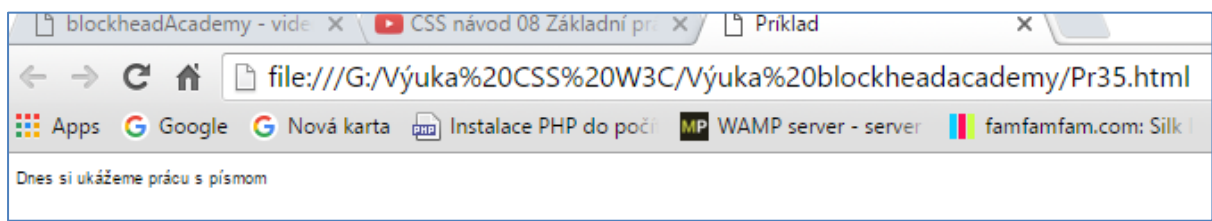
```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:x-large;  
}
```

Výstup:



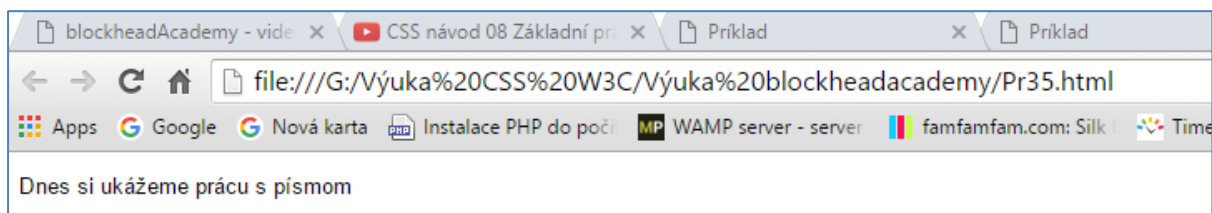
```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:xx-small;  
}
```

Výstup:



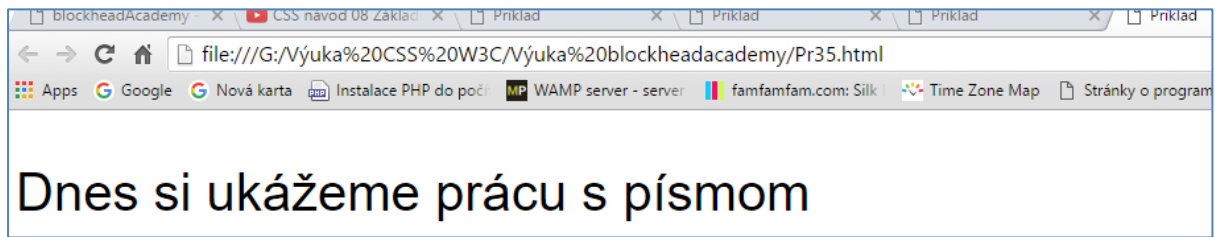
V.Nastavenie veľkosti v bodoch“pt“

```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:10pt;  
}
```



```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:30pt;  
}
```

Výstup:

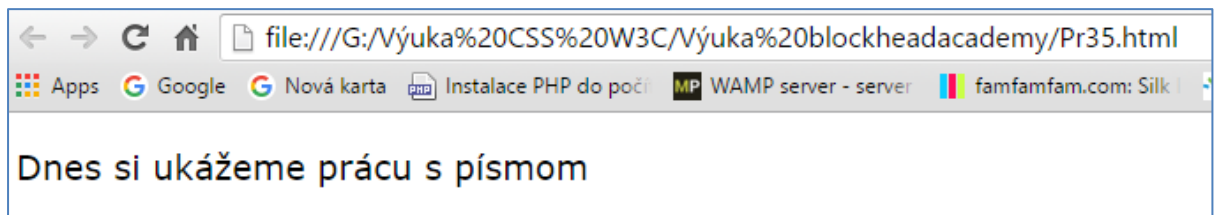


Zápis v bodoch pt je vhodný pre tlač, nie pre elektronické publikovanie.

Zápis rodiny písma a jeho veľkosti do jedného riadka

```
/*súbor main.css*/  
/* Zápis veľkosti a rodiny fontov do jedného riadka*/  
body {  
    font:1.2em Verdana,Arial,sans-serif;  
}
```

Výstup:



Pre prehľadnosť je lepšie zapisovať vlastnosti písma do samostatných riadkov. Napríklad takto:

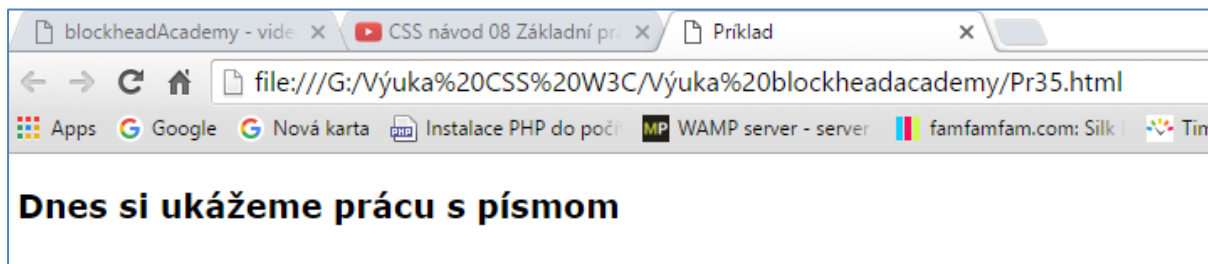
```
/*súbor main.css*/  
body {  
    font-family:Arial,sans-serif;/*Nastavenie fontu pre celý HTML  
dokument a aj s náhradnými fontami*/  
    font-size:30pt;  
}
```

3. Nastavenie hrúbky písma

```
/*súbor main.css*/  
/* Zápis veľkosti a rodiny fontov do jedného riadka*/  
body {  
    font:1.2em Verdana,Arial,sans-serif;
```

```
}
/*Nstavenie hrúbky písma */
p {
    font-weight:bold;
}
```

Výstup:



```
/*súbor main.css*/
/* Zápis veľkosti a rodiny fontov do jedného riadka*/
body {
    font:1.2em Verdana,Arial,sans-serif;
}
/*Nstavenie hrúbky písma */
p {
    font-weight:normal;
}
```

Výstup:



-----TU Pokračovať.....-----

4.Nastavenie štýlu

font-style

Definícia a použitie

CSS vlastnosť font-style definuje štýl písma. Sklonené (**italic**) a normálny (**normal**).

```
/*súbor main.css*/  
/* Definícia kurzívy */
```

```
span {  
  
    font-style: italic;  
  
}
```

PR.4.1.html

```
<span>text definovaný štýlom italic</span>
```

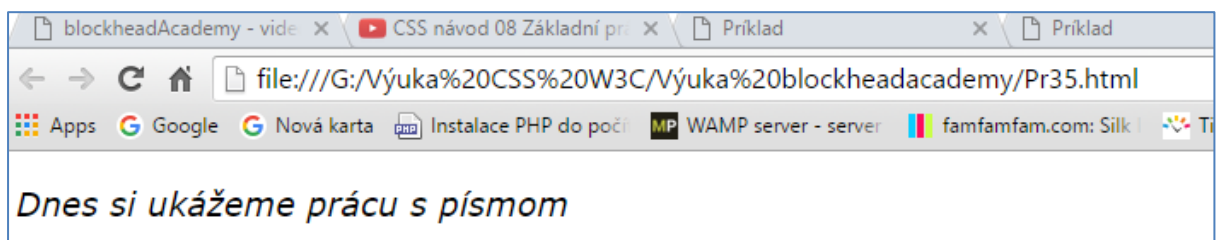
font-weight

Definícia a použitie

CSS vlastnosť font-weight definuje podporovaný rez písma. Základné sú normálne (**normal**), tučný (**bold**). Potom záleží na podpore reze písma u konkrétneho, definovaného fontu.

```
/*súbor main.css*/  
/* Zápis veľkosti ,hrúbky a štýlu */  
body {  
    font:1.2em Verdana,Arial,sans-serif;  
}  
/*Nastavenie hrúbky písma */  
p {  
    font-weight:normal;  
    font-style:italic;  
}
```

Výstup:

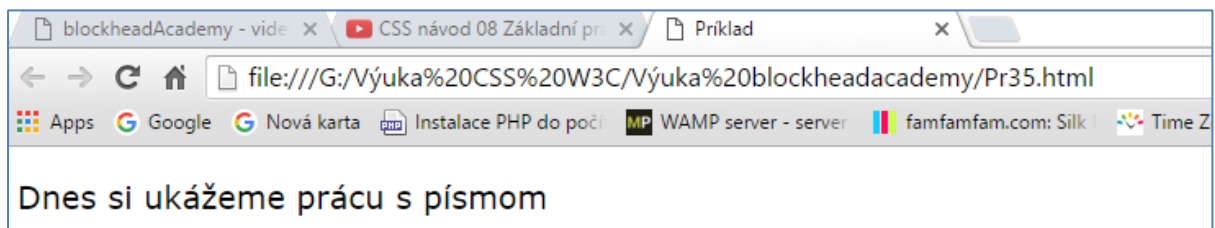


```

*súbor main.css*/
/* Zápis veľkosti ,hrúbky a štýlu */
body {
    font:1.2em Verdana,Arial,sans-serif;
}
/*Nastavenie hrúbky písma a štýlu písma */
p {
    font-weight:normal;
    font-style:normal;
}

```

Výstup:



5.Nastavenie písma kapitálky

font-variant

Definícia a použitie

CSS vlastnosť font-variant definuje transformáciu fontu,tak že veľké písmená sú zmenšená na veľkosť malého písmená a potom zobrazené všetky veľkými písmenami.Sú to tzv. kapitálky.

normal, small-caps

Text bude vypisovaný veľkými písmenami,ktoré budú mať menšiu veľkosť.Ale **prvé písmeno vo vete je väčšie.**

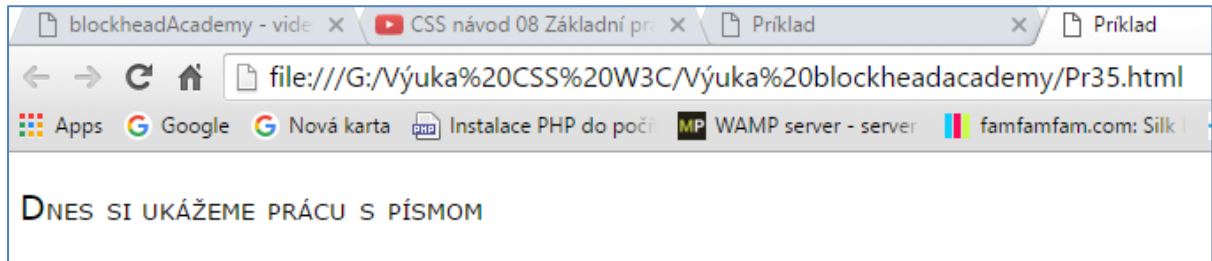
```

/*súbor main.css*/
/* Zápis veľkosti ,hrúbky a štýlu */
body {
    font:1.2em Verdana,Arial,sans-serif;
}
/*Nastavenie hrúbky písma a štýlu písma */
p {
    font-weight:normal;
    font-style:normal;
}

```

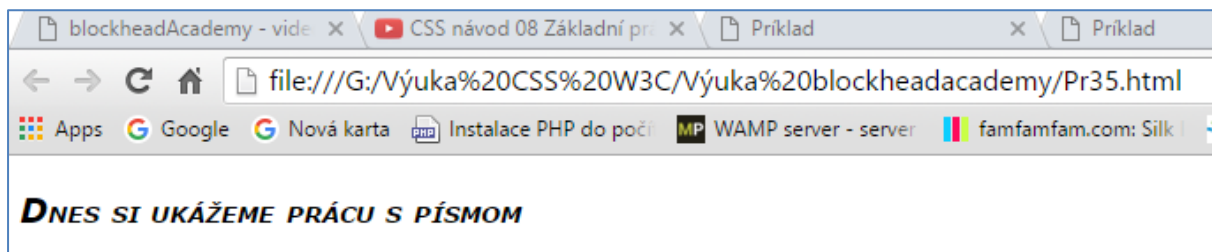
```
font-variant: small-caps;
}
```

Výstup:



```
*súbor main.css*/
/* Zápis veľkosti ,hrúbky a štýlu */
body {
    font:1.2em Verdana,Arial,sans-serif;
}
/*Nastavenie hrúbky písma a štýlu písma a varianty */
p {
    font-weight:bold;
    font-style:italic;
    font-variant: small-caps;
}
```

Výstup:



Sumárne cvičenie na font

index.html

```
<!DOCTYPE html>
<html lang="sk">
```



```

<head>
<meta charset="UTF-8">
<title>Súbor pre cvičenie fontov</title>
<link href="images/logo.png" rel="shortcut icon" type="image/png">
  <link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
<p>Toto je základná veľkosť písma 16 pixelov</p>
<p class="font1">Tento text je veľkosťou 22 pixelov</p>
<p class="font2">Tento text má veľkosť lem </p>
<p class="font3">A tento text je veľkosťou 0.5 em</p>
<p class="font4">A tento text je veľkosťou 150% základnej veľkosti</p>
<p class="font5">A tento text je veľkosťou small</p>
<p class="font6">A tento text je veľkosťou font-size:x-large</p>
<p class="font7">A tento text je veľkosťou font-size:xx-small</p>
<p class="font8">A tento text je tučný</p>
<p class="font9">A tento text je veľký a tučný</p>
<p class="font10">A tento text nie je zvýraznený</p>
<p class="font11">A tento text je zvýraznený naklonením</p>
<p class="font12">A tento text je zvýraznený hrúbkou aj naklonením </p>
<p class="font13">A tento text je zvýraznený hrúbkou, naklonením a
farbou</p>
</body>
</html>

```

main.css

```

/* main.css */
.font1{
  font-size: 22px;
}
.font2{
  font-size: 1em;
}
.font3{
  font-size: 0.5em;
}
.font4{
  font-size: 150%;
}
.font5{
  font-size: small;
}
.font6{
  font-size: x-large;
}
.font7{
  font-size: xx-small;
}

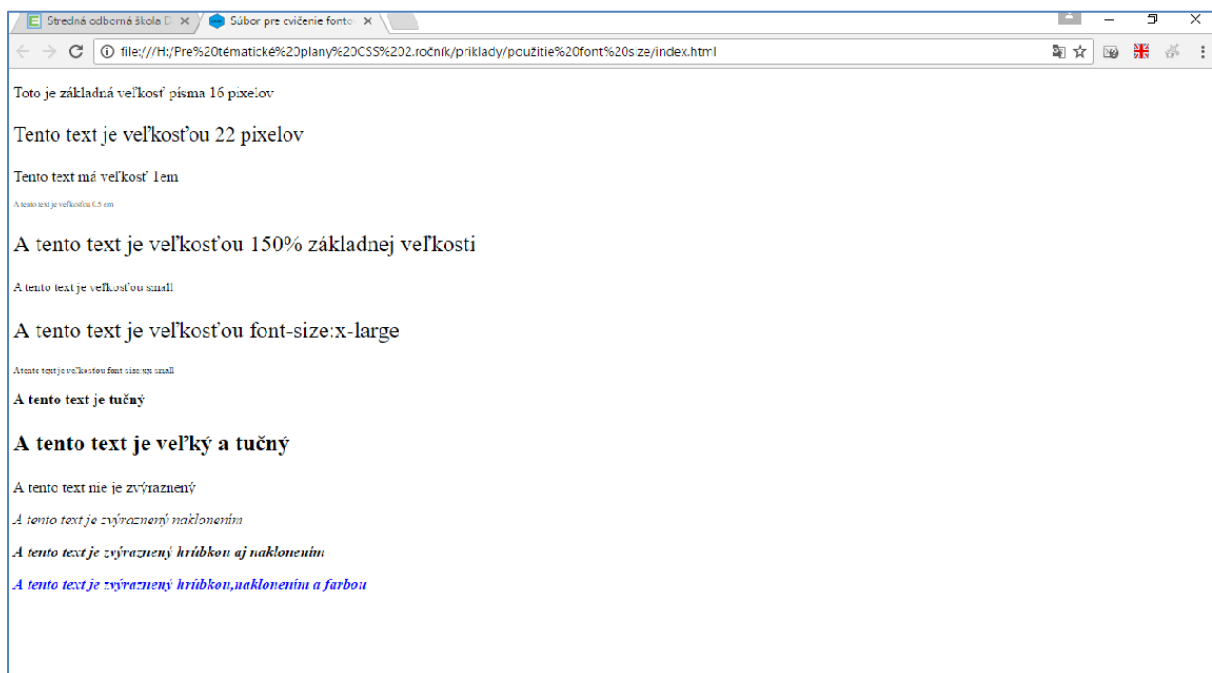
```

```

.font8 {
    font-weight: bold;
}
.font9 {
    font-size: x-large;
    font-weight: bold;
}
.font11 {
    font-style: italic;
}
.font12 {
    font-weight: bold;
    font-style: italic;
}
.font13 {
    font-weight: bold;
    font-style: italic;
    color: blue;
}

```

Výstup:



Druhé zadanie životopis

Zadanie pre samostatnú prácu:

1. Napíšte svoj životopis aspoň na 100 slov
2. Rozdelte ho aspoň do troch kapitol, každá kapitola bude mať vlastný nadpis (hlavičku) prvej úrovne (napríklad: Detstvo, Základná škola, Stredná škola)
3. Text rozumne triedte do odstavcov
4. Všetky mena osôb napíšte tučným písmom Názvy miest, obcí, riek, alebo iných neživých predmetov podčiarknite
5. Čísla napíšte kurzívou
6. Svoje meno píšete kurzívou i tučným písmom zároveň

Index.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Súbor pre cvičenie fontov</title>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p class="hlavnyNadpis">Životopis</p>
    <p class="obdobie">Detstvo</p>
    <p class="detstvo">Volám sa <span class="mojeMeno" >Pavel Drgo</span>.
    Narodil som sa 18.12.1958 v <u>Púchove</u>. Moji rodičia sú <span>Pavol
    Drgo</span> a <span>
    Anna Drgová</span> rodená <span>Matúšová</span>. Otec bol šoférom
    nákladného automobilu v podniku Jednota so sídlom v <u>Púchove</u>. Mama
    pracovala ako chemička v Slovenských mliekarňach so sídlom tiež v
    <u>Púchove</u>. Mám 5 súrodencov ,všetko sú to sestry, ktoré sú mladšie ako
    ja. Ich mená sú <span>Milka,Viera,Zuzana,Anna a Janka</span>. Svoje detstvo
    som prežil v obci <u>Lednica</u>. Tu som navševola aj materskú škôlku.
    Hlavnú časť svojho detstva som prežil v spoločnosti mojej babky
    <span>Magdalény</span> pod majestátnym hradom.</p>
    <p class="obdobie">Základná škola</p>
    <p class="zakladnaSkola">Základnú deväťročnú dochádzku do školy som
    zahájil v školskom roku <span class="kurziva">1966-1967</span> v <span
    class="mesto">Lednici</span>. V tretej triede som sa stal iskričkou a v
    piatej triede pionierom.Aj napriek tomu som chodil do kostola a v <span
    class="mesto">Lednickom</span>kostole som prijal sviatosť prvého svätého
    prijímania. Školskú dochádzku som často vynechával lebo som musel pomáhať
    mame s poľnohospodárskymi prácami. V lete som okrem iného trávil tak, že
    som pásol kravy a husi.Najradšej spomienkam na letné prázdniny, ktoré boli
    plné hier a chleba s rajčinou. Školskú dochádzku som skončil v roku <span
    class="kurziva">1974.</span></p>
```

```
<p class="obdobie">Stredná škola</p>
<p class="strednaSkola"> V školskom roku <span
class="kurziva">1974/75</span> som nastúpil na štúdium do Stredného
odborného učilišťa pri Strojárskych a metalurgických závodoch v <span
class="mesto">Dubnici nad Váhom</span>. Prijali ma na trojročné štúdium
odboru montér elektrotechnických zariadení.Štúdium som úspešne skončil
záverečnou skúškou a získal som výučný list a bol som zaradený do <span
class="kurziva">5</span>.platobnej triedy.Po vyučení som pracoval dva
mesiace v odbore elektrotechnika.V <span class="kurziva">1977 </span> som
urobil rozdielové skúšky a nastúpil som do druhého ročníka odboru
elektrické stroje a prístroje na Strednej priemyselnej škole v <span
class="mesto">Dubnici nad Váhom</span>.Štúdium som ukončil maturitnou
skúškou v roku <span class="kurziva">1980.</p>

</body>
</html>
```

Main.css

```
/* main.css*/
.hlavnyNadpis{
    font-size:60px;
    font-weight:bold;
    text-align: center;
}
.obdobie{
    font-size:40px;
}

.detstvo{
    font-family:Arial, sans-serif;
    font-size: 20px;
    text-indent: 40px;
    line-height:3;
}
.mojeMeno{
    font-style: italic;
    font-weight: bolder;
}
p.detstvo:first-letter{
    font-size:150%;
}
span{
    font-weight: bolder;
}

.zakladnaSkola{
    font-family:Verdana, sans-serif;
    font-size: 20px;
    text-indent: 40px;
```

```

        line-height:2;
    }
    p.zakladnaSkola:first-letter{
        font-size:150%;
    }
    .mesto{
        text-decoration:underline;
        font-weight:normal;
    }
    .kurziva{
        font-weight:normal;
        font-style: italic;
    }
    .strednaSkola{
        font-family:sans-serif;
        font-size: 20px;
        text-indent: 40px;
        line-height:1;
    }
    p.strednaSkola:first-letter{
        font-size:150%;
    }

```

Životopis

Detstvo

Volám sa **Pavel Drgo**. Narodil som sa 18.12.1958 v Púchove. Moji rodičia sú **Pavol Drgo** a **Anna Drgová** rodená **Matúšová**. Otec bol šoférom nákladného automobilu v podniku Jednota so sídlom v Púchove. Mama pracovala ako chemička v Slovenských mliekárňach so sídlom tiež v Púchove. Mám 5 súrodencov ,všetko sú to sestry, ktoré sú mladšie ako ja. Ich mená sú **Milka,Viera,Zuzana,Anna a Janka**. Svoje detstvo som prežil v obci Lednica. Tu som navštevola aj materskú škôlku. Hlavnú časť svojho detstva som prežil v spoločnosti mojej babky **Magdalény** pod majestátnym hradom.

Základná škola

Základnú deväťročnú dochádzku do školy som zahájil v školskom roku 1966-1967 v Lednici. V tretej tiede som sa stal iskričkou a v piatej triede pionierom.Aj napriek tomu som chodil do kostola a v Lednickomkostole som prijal sviatosť prvého svätého prijímania. Školskú dochádzku som často vynechával lebo som musel pomáhať mame s poľnohospodárskymi prácami. V

Tu pokračovať!!!!

9.CSS návod 09 Nastavujeme vlastnosti textu

Dnes si ukážeme ako nastavovať vlastnosti textu.Môžeme tak ovplyvniť,či bude text:

1. Text bude podtrhnutý
2. Prečiarknutý
3. Zarovnaný vpravo,vľavo,nastred
4. Nastaviť veľkosť medzery medzi slovami
5. Nastaviť vzdialenosť medzi písmenami

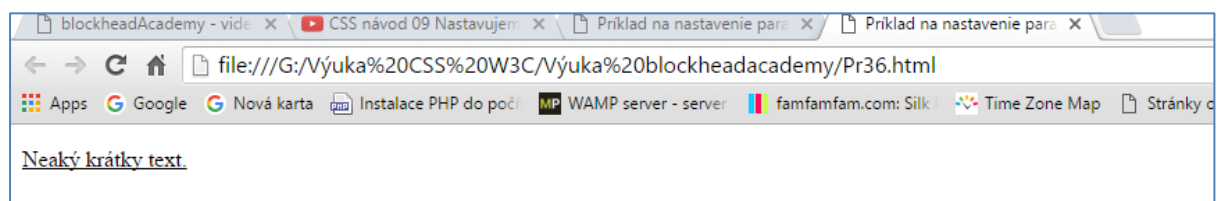
1.Text podčiarknutý

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie parametrov textu</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">

  </head>
  <body>
    <p>Neaký krátky text.</p>
  </body>
</html>
```

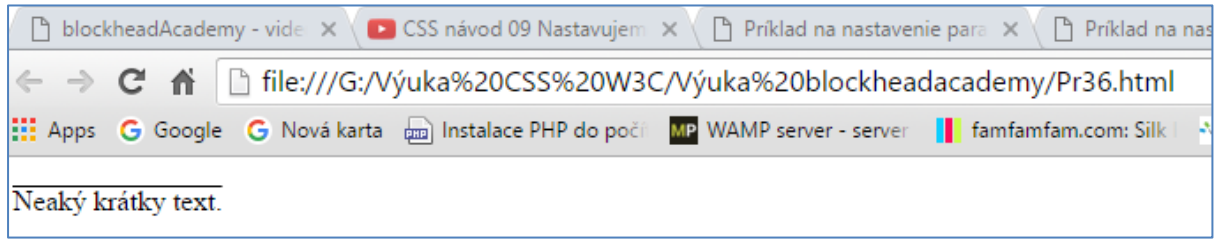
```
/*súbor main.css*/
/* Zápis dekorácie textu */
p {
    text-decoration:underline;
}
```

Výstup:



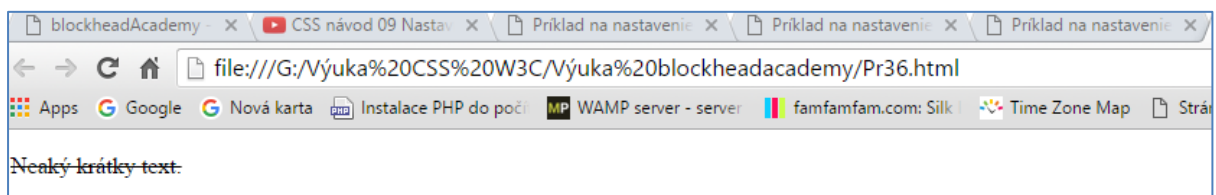
```
/*súbor main.css*/
```

```
/* Zápis dekorácie textu s čiarou nad textom*/  
p {  
    text-decoration: overline;  
}
```



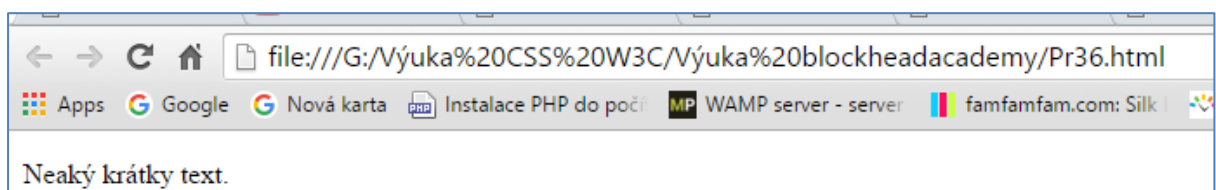
```
/*súbor main.css*/  
/* Zápis dekorácie textu prečiarknutím */  
p {  
    text-decoration: line-through;  
}
```

Výstup

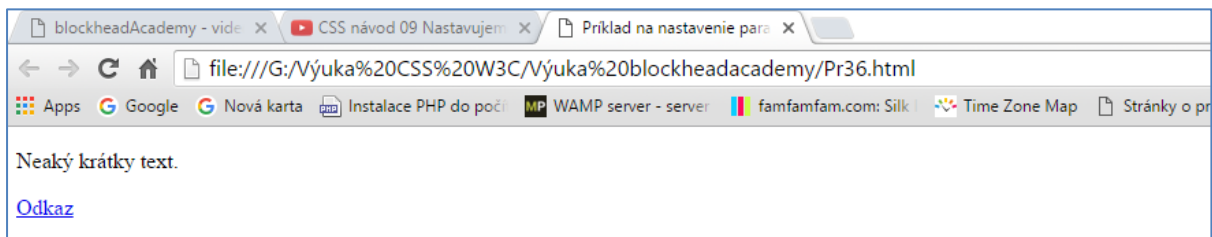


```
/*súbor main.css*/  
/* Zápis dekorácie zrušenia dekorácie textu */  
p {  
    text-decoration: none; /*Zruší default nastavenie*/  
}
```

Výstup:



To sa dá použiť ak chceme zrušiť defaultné nastavenie odkazu:

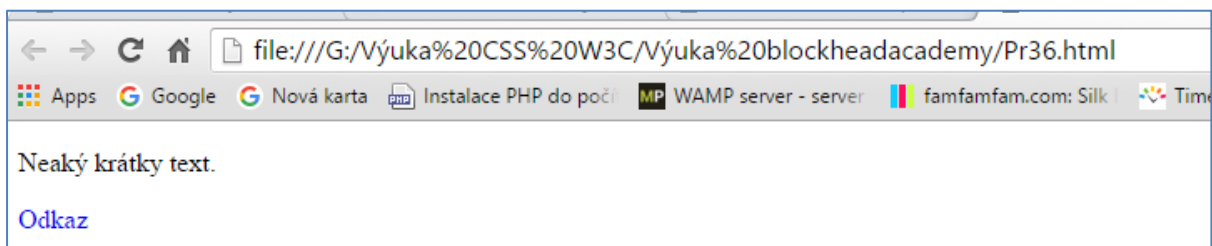


Defaultne nastavenie odkazu je modrý text odkazu a podčiarknutý.

My toto počiaroknutie pomocou text-decoration:none, zrušíme.Odkaz však zostane odkazom.

```
/*súbor main.css*/  
/* Zápis dekorácie zrušenia dekorácie textu odkazu */  
a {  
    text-decoration:none; /*Zruší default nastavenie*/  
}
```

Výstup



Toto môžeme použiť a na stránkach budeme chcieť vytvoriť menu s odkazov a podčiarknutie by nás esteticky rušilo.

Pre ďalší príklad budeme potrebovať dlhší text v paragrafe ,ktorý si stiahneme zo začiatku dohto tútorialu.

Pr.37 Nastavovanie šírky odstavca

```
<!DOCTYPE html>  
<html lang="sk">  
  <head>  
    <meta charset="UTF-8">  
  <title>Príklad na nastavenie parametrov textu</title>
```



```

<link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
<p>Ako napríklad farbu textu,, veľkosť písma, farbu pozadia. Umožňujú nám
nastaviť vlastnosti , ktoré sa v HTML dokumente nastaviť nedajú. Práca s
kakádovými štýlmi je omnoho efektívnejšia než nastavovanie vlastností
elementu priamo v HTML dokumente,</p>
</body>
</html>

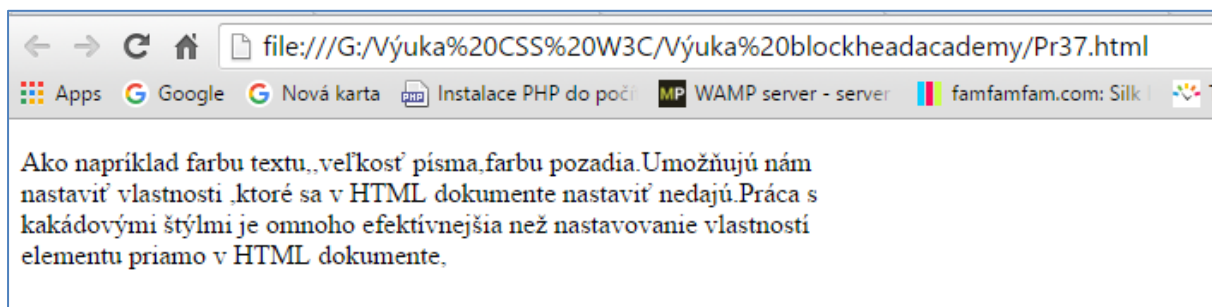
```

```

*súbor main.css*/
/* Nastavenie šírky paragrafu */
p {
    width: 500px;
}

```

Výstup:



Vidíme ,že text je defaultne zarovnaný doľava a pravá strana vyzerá tak ako nám ktoré slovo výjde.

Zarovnanie môže byť:

1. Left
2. Right
3. Center
4. Justify

Pr.38 Zarovnanie textu do bloku

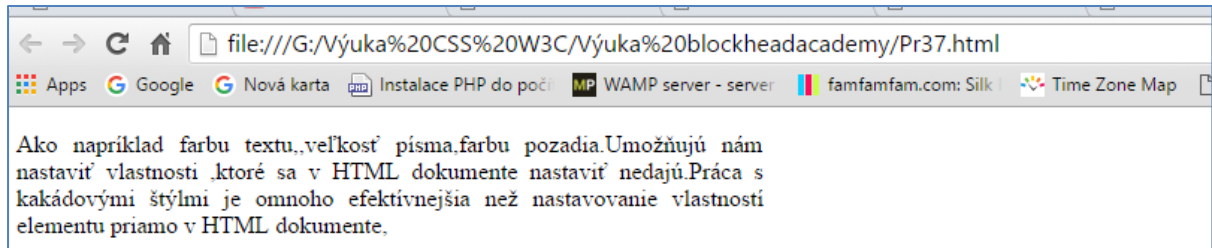
```

/*súbor main.css*/
/* Nastavenie zarovnania textu */
p {

```

```
width: 500px;  
text-align: justify;  
}
```

Výstup:



10.CSS návod10 Nastavujeme vlastnosti textu pokračovanie

Ukážeme si ako :

- I. Nastavenie medzier medzi slovami
- II. Nastavenie medzier medzi písmenami
- III. Veľkosť medzeri vo vnútri odstavca
- IV. Výšku riadku v elemente
- V. Text transform. Umožní nastavovať pre vybraný text veľké a malé písmená

I. Nastavenie medzier medzi slovami

Pr.38 Nastavovanie medzier medzi slovami

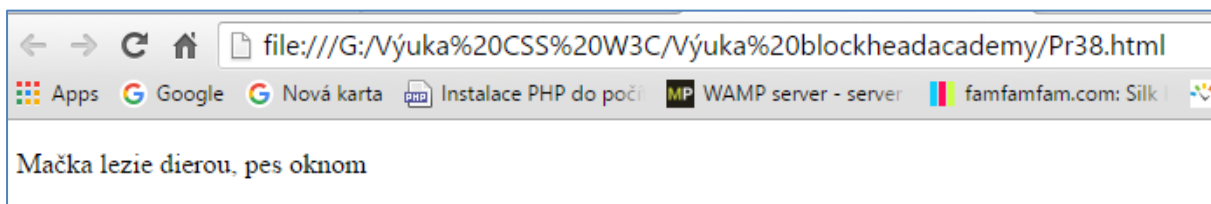
```
<!DOCTYPE html>  
<html lang="sk">  
  <head>
```

```

<meta charset="UTF-8">
<title>Príklad na nastavenie medzier medzi slovami v paragrafe</title>
<link rel="stylesheet" type="text/css" href="css/main.css"/>
<link href="images/logo.png" rel="shortcut icon" type="image/png">
</head>
<body>
<p>Mačka lezie dierou, pes oknom</p>
</body>
</html>

```

Defaultne je nastavená takáto vzdialenosť medzi slovami.



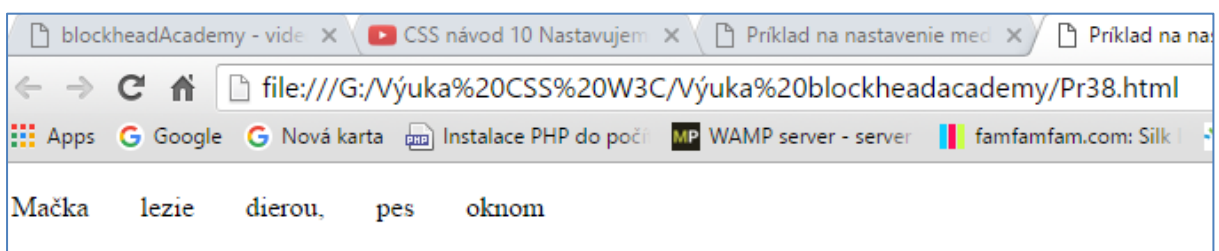
Na zmenu vzdialenosti použijeme nasledovný tvar súboru main.css:

```

/*súbor main.css*/
/* Nastavenie medzery medzi slovami */
p {
    word-spacing: 25px;
}

```

Výstup:



Veľkosť medzier medzi slovami sa výrazne zväčšila.

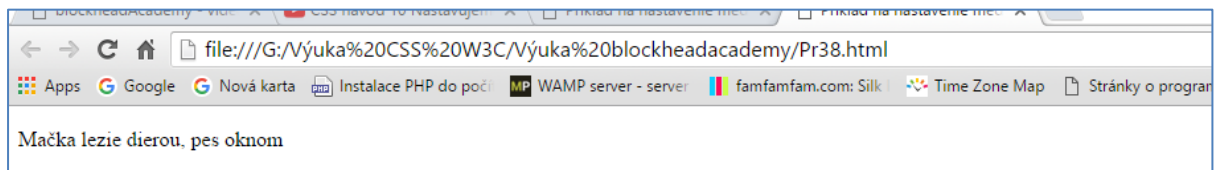
Ak chceme vrátiť medzery do pôvodného stavu, použijeme nasledovný súbor main.css

```

/*súbor main.css*/
/* Nastavenie medzery medzi slovami na normálnu veľkosť */
p {
    word-spacing: normal;
}

```

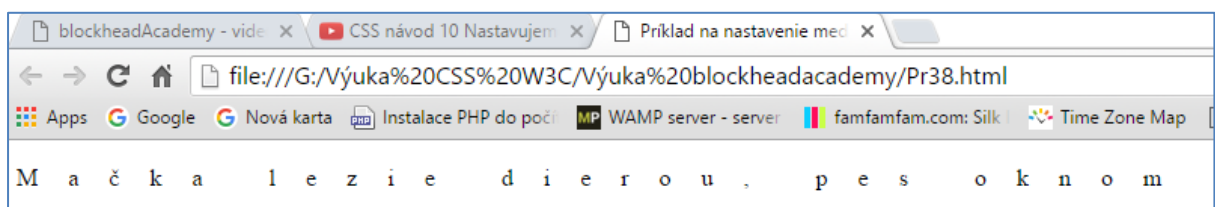
Výstup:



II. Nastavenie medzier medzi písmenami

```
*súbor main.css*/  
/* Nastavenie medzery medzi slovami na normálnu veľkosť */  
p {  
    word-spacing: normal;  
}  
/*Nastavenie medzier medzi písmenami*/  
p {  
    letter-spacing: 20px;  
}
```

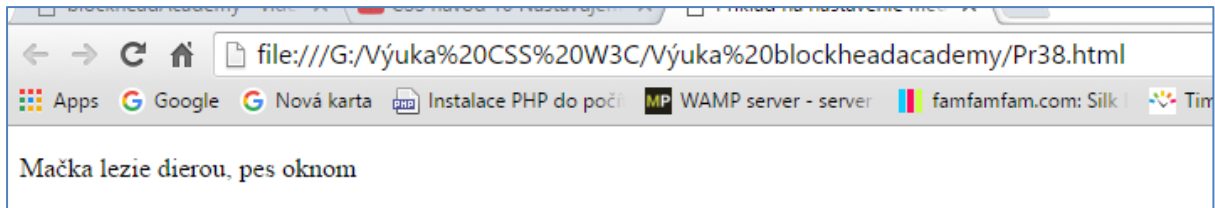
Výstup:



Návrat na pôvodnú hodnotu veľkosti medzier medzi slovami a písmenami:

```
/*súbor main.css*/  
/* Nastavenie medzery medzi slovami na normálnu veľkosť */  
p {  
    word-spacing: normal;  
}  
/*Nastavenie medzier medzi písmenami na pôvodnú hodnotu*/  
p {  
    letter-spacing: normal;  
}
```

Výstup: je zobrazenie nastavenia HTML prehliadačom



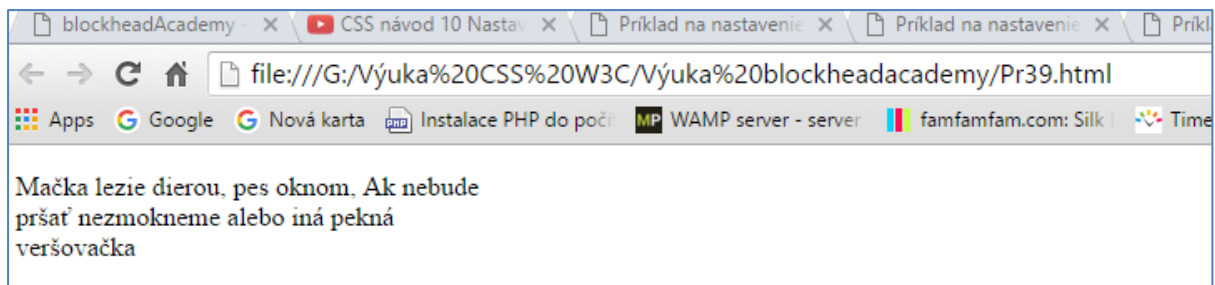
III. Veľkosť medzeri vo vnútri odstavca

Pr39

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie medzier riadkami v paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Mačka lezie dierou, pes oknom, Ak nebude pršať nezmokneme alebo iná
    pekná veršovačka</p>
  </body>
</html>
```

```
*súbor main.css*/
/* */
p {
  width:300px;/*Nastavenie dĺžky paragrafu*/
}
```

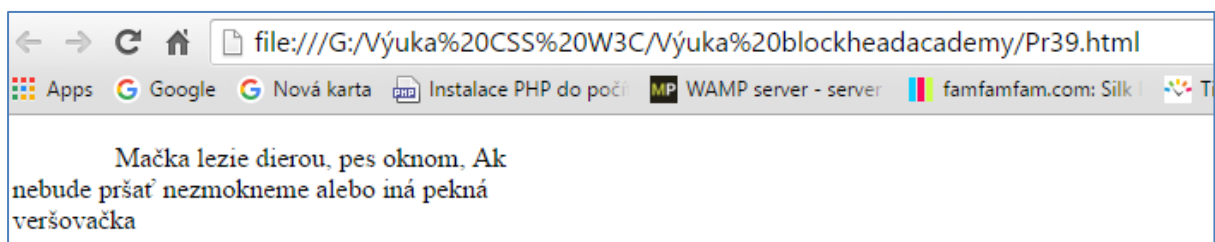
Výstup



V predu uvedenom príklade je defaultná vzdialenosť medzi riadkami paragrafu.

```
*súbor main.css*/  
p {  
    width:300px;/*Nastavenie dĺžky paragrafu*/  
    text-indent: 60px;/*Nastavenie vzdialenosti pre odsadenie v prvom  
riadku v paragrafe*/  
}
```

Výstup:



Hodnota odsadenia môže byť a j percentách

```
*súbor main.css*/  
p {  
    width:300px;/*Nastavenie dĺžky paragrafu*/  
    text-indent: 20%;/*Toto je bežné odsadenie v paragrafe*/  
}
```

IV. Výšku riadku v elemente

Pr.40 Nastavenie vzdialenosti medzi riadkami elementu

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="utf-8">
  <link rel="stylesheet" type="text/css" href="css/main.css"/>
  <link href="images/logo.png" rel="shortcut icon" type="image/png">
</head>
<body>

<p>
Toto je paragraf so štandardnou výškou riadu v odstavci.<br>
Východisková výška riadku vo väčšine prehliadačov je asi 110% až
120%, štandardnej výšky riadku.<br>
</p>

<p class="small">
Tento paragraf je s menšou výškou riadku .<br>
Tento paragraf je s menšou výškou riadku .line-height.<br>
</p>

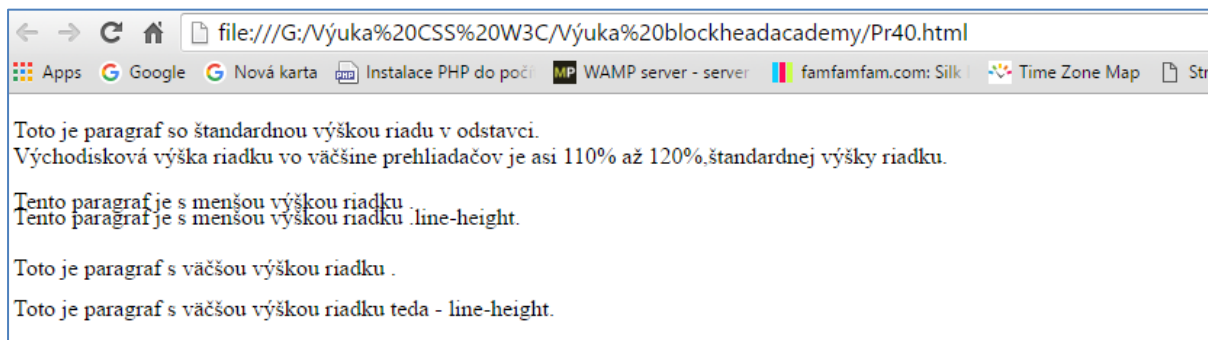
<p class="big">
Toto je paragraf s väčšou výškou riadku .<br>
Toto je paragraf s väčšou výškou riadku teda - line-height.<br>
</p>

</body>
</html>
```

```
* CSS subor s menom main.css*/
p.small {
  line-height: 0.7;
}

p.big {
  line-height: 1.8;
}
```

Výstup:

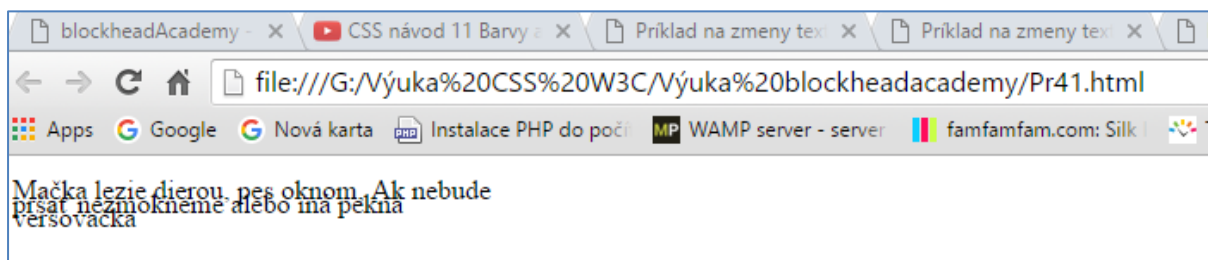


Vzdialenosť medzi riadkami môžeme nastavovať aj v percentách.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zmeny textu v paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Mačka lezie dierou, pes oknom, Ak nebude pršať nezmokneme alebo iná
    pekná veršovačka</p>
  </body>
</html>
```

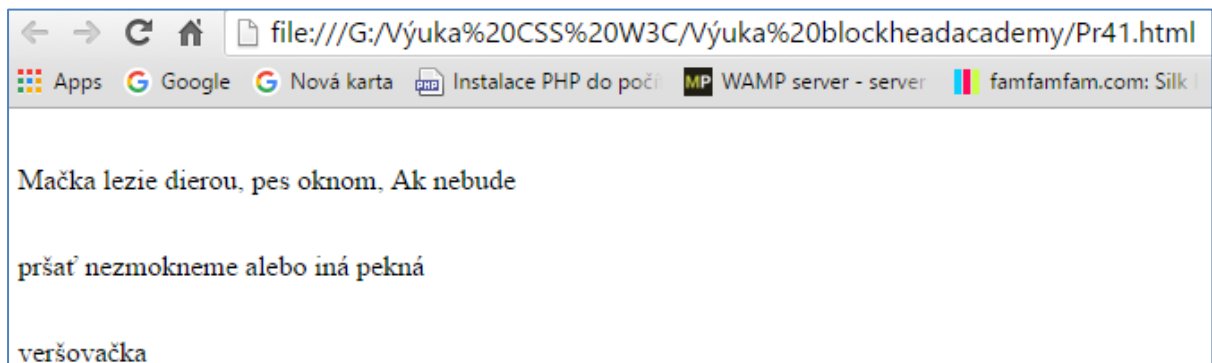
```
/* CSS subor s menom main.css*/
p{
  line-height:50%;/* Nastaví sa polovičná veľkosť štandardnej výšky
riadku.*/
}
```

Výstup:



Výšku riadku môžeme nastaviť aj číselnou hodnotou

```
* CSS subor s menom main.css*/
p{
    width: 300px;
    line-height:3;/* Nastaví sa trojnásobná veľkosť štandardnej výšky
riadku.*/
}
```



V.Text transform.

Umožní nastavovať pre vybraný prvok veľké a malé písmená , pomocou vlastnosti text-transform.

Pr.41.Nastavenie textu pomocou text-transform.

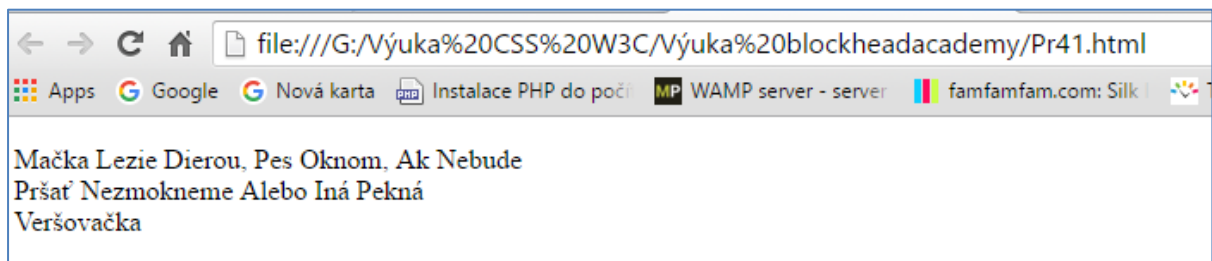
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zmeny textu v paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">
  </head>
  <body>
    <p>Mačka lezie dierou, pes oknom, Ak nebude pršať nezmokneme alebo iná
pekná veršovačka</p>
  </body>
</html>
```

```

/* CSS subor s menom main.css*/
p{
    width:300px;
    text-indent: initial;/*Nastavenie nulového odsadenia prvého riadku
v vybranom elemente*/
}
p{
    text-transform: capitalize;/* Nastavenie prvých písmen slov na
veľké.*/
}

```

Výstup:

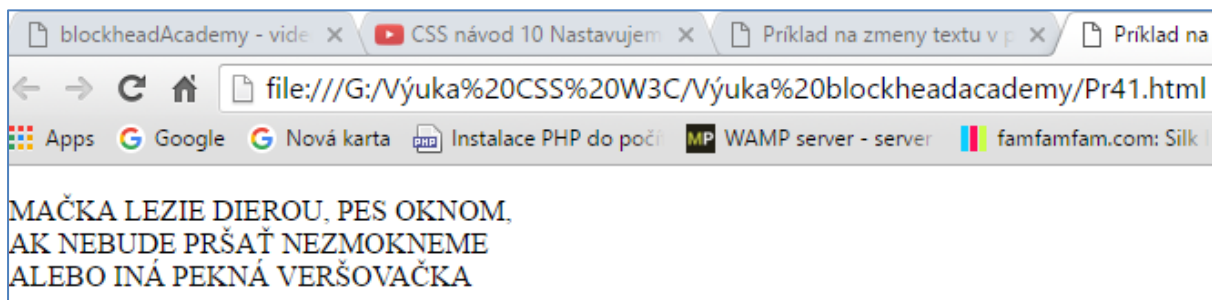


```

* CSS subor s menom main.css*/
p{
    width:300px;
    text-indent: initial;/*Nastavenie nulového odsadenia prvého riadku
v vybranom elemente*/
}
p{
    text-transform: uppercase;/* Nastavenie všetkých písmen vybraného
prvku na veľké.*/
}

```

Výstup

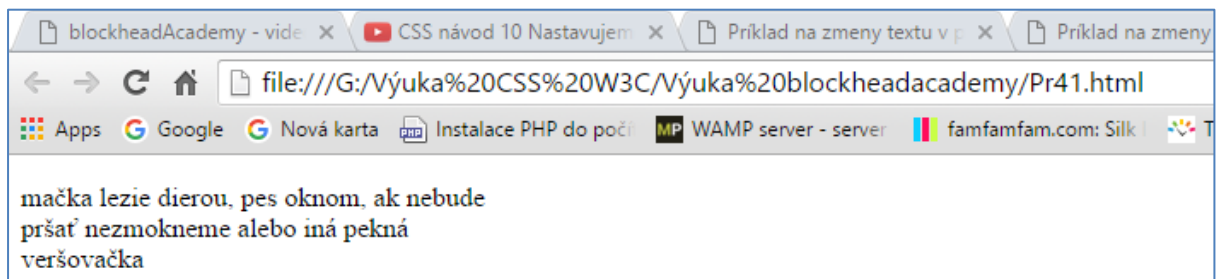


```

/* CSS subor s menom main.css*/
p{
    width:300px;
    text-indent: initial;/*Nastavenie nulového odsadenia prvého riadku
v vybranom elemente*/
}
p{
    text-transform: lowercase;/* Nastavenie všetkých písmen vybraného
prvku na malé.*/*
}

```

Výstup:

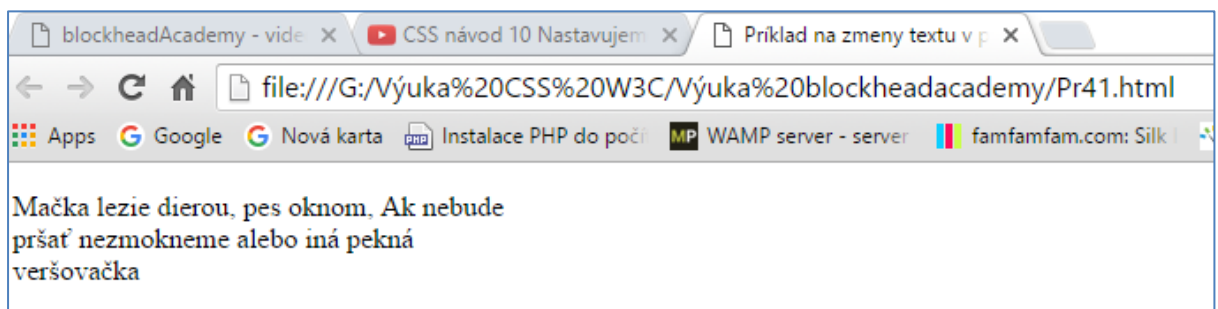


```

* CSS subor s menom main.css*/
p{
    width:300px;
    text-indent: initial;/*Nastavenie nulového odsadenia prvého riadku
v vybranom elemente*/
}
p{
    text-transform: none;/* Text sa chová štandardne, teda podľa
velkostí v paragrafe.*/*
}

```

Výstup:



11.CSS návod 11 farby a obrázkov na pozadí

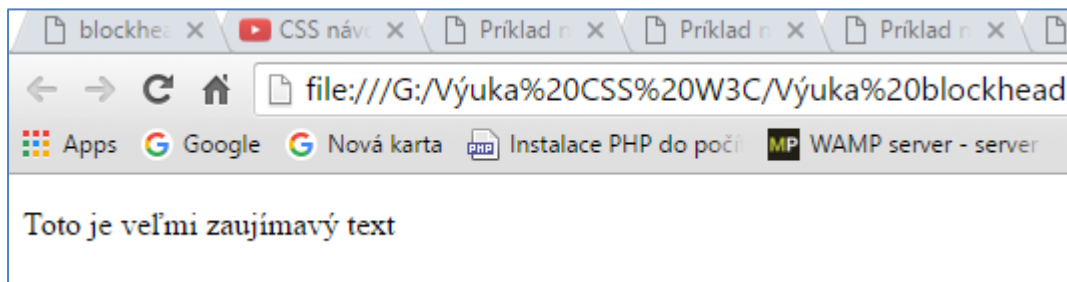
Ukážeme si prácu s farbami v texte a pozadí, a tiež ako na pozadí nastaviť obrázok.

Práca s farbami v texte vybraného elementu

```
<body>
<p>Toto je veľmi zaujímavý text</p>
</body>
```

V tomto prípade sa nastaví farba textu paragrafu na čiernu a pozadie na bielu farbu. Čiže štandardne.

Výstup:



Nastavenie farby textu vybraného elementu:

[Tu pokračovať 1. decembra 2016](#)

Cvičenie

Index.html

```
<!DOCTYPE html>
<html lang='sk'>
  <head>
    <title>Počítačový servis ALL Computer s.r.o.</title>
    <meta charset='utf-8'>
      <link href='images/logo.png' rel='shortcut icon'
type='image/png'>
```

```

        <link href='css/style.css' rel='stylesheet'
type='text/css'>
    </head>
<body>
    <header>
        <p class="Nadpis">Počítačový servis</p>
        <p class="ALL">ALL Computer s.r.o.</p>
    </header>
    <article>
        <p class="textFirma">Firma All Computer s.r.o. ponúka široký
sortiment produktov a tovarov v oblasti výpočtovej techniky.Vyskúšajte naše
služby a sami sa budete môcť presvedčiť o vysokej kvalite a úrovni.To čo
Vám dokážeme sľubiť nikto nedokáže splniť.</p>
        <p class="ponuka">PONÚKAME:</p>
        <nav>
            <ul>
                <li>Predaj nových počítačov</li>
                <li>Predaj počítačových zostáv</li>
                <li>Predaj komponentov od procesora po zvukové
karty</li>
                <li>Oprava nových počítačov</li>
                <li>Repasovanie starých počítačov</li>
                <li>Odvírovanie počítačov a záchrana údajov</li>
                <li>...A to s rýchlymi dodacími lehotami</li>
            </ul>
        </nav>
        <p class ="pata">Na nás sa môžete spoľahnúť!!!</p>
    </article>
</body>
</html>

```

style.css

```

body {
    background-image: url("../images/wall.jpg");
    background-color: #eee;
}
p.Nadpis{
    font-family: Arial;
    font-size: 30px;
    line-height:3;
    border-bottom: 3px solid blue;
}
p.ALL{
    font-family: Calibri,Arial;
    font-size: 16px;
    text-align: right;
}

```

```

p.textFirma{
    font-family: "Times New Roman";
    font-size: 20px;
    text-align: justify;
    line-height: 2;
}
p.ponuka{
    font-family: Arial;
    font-size: 16px;
    font-weight: bold;
    text-decoration: underline;
    text-indent: 40px;
}
li{
    list-style-position: inside; /*má za následok ďalšie odsadenie v
zozname*/
    line-height: 2;
    text-indent: 40px;
}
p.pata{
    font-family: Arial;
    font-size: 16px;
    text-align: center;
    line-height: 3;
    border-style: solid;
    border-width: 2px;
    border-color: blue;
}

```

Výstup:

Počítačový servis

ALL Computer s.r.o.

Firma All Computer s.r.o. ponúka široký sortiment produktov a tovarov v oblasti výpočtovej techniky. Vyskúšajte naše služby a sami sa budete môcť presvedčiť o vysokej kvalite a úrovni. To čo Vám dokážeme sľubiť nikto nedokáže splniť.

PONÚKAME:

- Predaj nových počítačov
- Predaj počítačových zostáv
- Predaj komponentov od procesora po zvukové karty
- Oprava nových počítačov
- Repasovanie starých počítačov
- Odvírovanie počítačov a záchrana údajov
- ...A to s rýchlymi dodacími lehotami

Na nás sa môžete spoľahnúť!!!

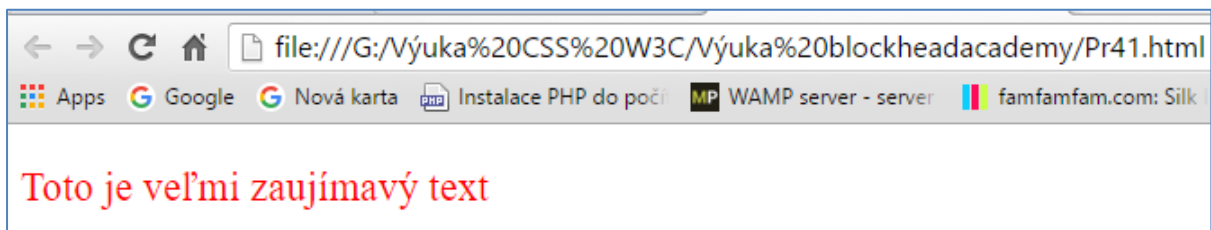
I. Anglické meno farby (red, blue, green, silver, gold, gray, aqua)

PR42 Nastavenie farby textu vybraného elementu

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zmeny textu v paragrafe</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <p>Toto je veľmi zaujímavý text</p>
  </body>
</html>
```

```
/* CSS subor s menom main.css */
p {
  font-size: 1.4em;
  color: red;
}
```

Výstup:

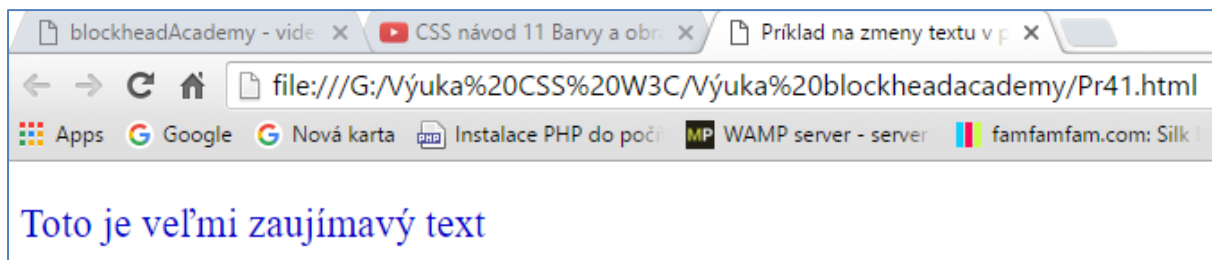


II. Zadanie kódu farby v hexadecimálnom tvare

Umožňuje rozlíšiť viacej farieb. Farba začína znakom hash #.

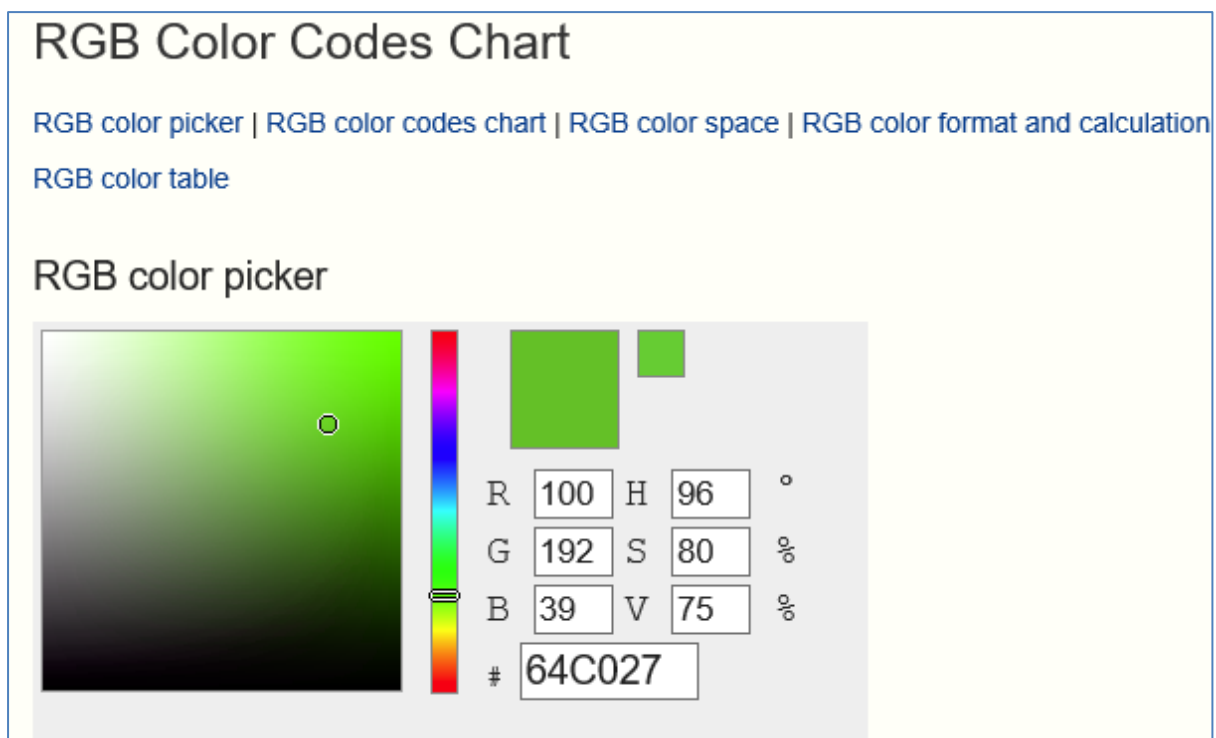
```
/* CSS subor s menom main.css */
p {
  font-size: 1.4em;
  color: #0000cc; /*Modrá farba textu vybraného elementu*/
}
```

Výstup



Kódy farieb si môžeme nájsť na internete pod kľúčový slovom: **color mixer online**

III.Zadanie farieb v kódovaní RGB



Každá farba v RGB môže nadobúdať hodnoty od 0 do 255

PR.42.1.Zadávanie farieb pomocou metódy RGB

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
```



```

<title>Príklad na nastavenie farieb textu a pozadia elementu pomocou
farieb RGB</title>
  <link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
<div>
<h1>London je hlavné mesto Anglicka.</h1>
<p>To je najviac zaľudnené mesto v Spojenom kráľovstve ,
s metropolitnej oblasti viac ako 13 miliónov obyvateľov.</p>
</div>
</body>
</html>

```

```

/* CSS subor s menom main.css*/
div {
  width: 500px; /* Nastavenie šírky kontajneru*/
  background-color: rgb(0, 191, 255); /*Farba pozadia kontajnera*/
  color: rgb(255, 255, 255); /*Farba textu v kontajneri.V kontajneri je
prvok "p" a "h1"*/
  padding: 20px; /* Nastavenie výstelky kontajnera*/
}

```

Výstup:



Nastavenie farby pozadia vybraného elementu:

Farby môžu byť zadané slovné alebo v hexa kóde

PR43.Nastavenie farby pozadia vybraného elementu

```

<!DOCTYPE html>

<html lang="sk">

```

```

<head>
  <meta charset="UTF-8">
<title>Príklad na zmeny textu v paragrafe</title>
  <link rel="stylesheet" type="text/css" href="css/main.css"/>
</head>
<body>
  <p>Toto je veľmi zaujímavý text</p>
</body>
</html>

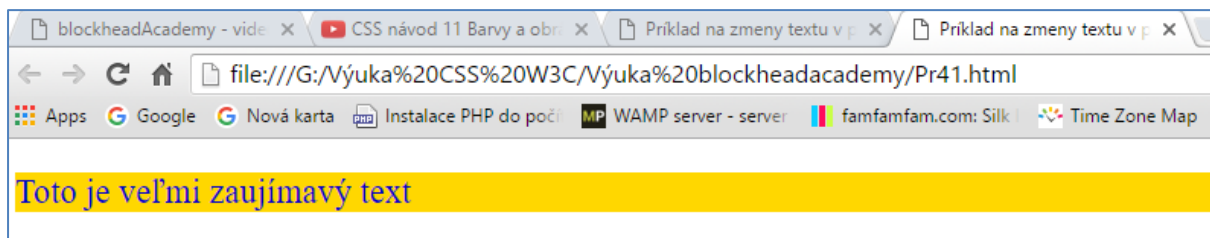
```

```

/* CSS subor s menom main.css*/
p{
  font-size: 1.4em;
  color:#0000cc; /*Modrá farba textu vybraného elementu*/
  background-color:gold; /*Nastavenie farby pozadiavybraného elementu*/
}

```

Výstup:



Ako nastaviť obrázok na pozadie:

Je treba správne nastaviť cestu k obrázku. Obrázok umiestnime do priečinku images v rovnakom priečinku ako je priečinok css. Cesta je:

```
url("../images/pozadie.jpg");
```

Obrázok pozadie.jpg je v priečinku images. A bude nastavený pre celý HTML dokument. Ak je obrázok menší ako šírka elementu, obrázok sa zobrazí viac krát po celej šírke html dokumentu.

PR. Nastavenia obrázku na pozadí

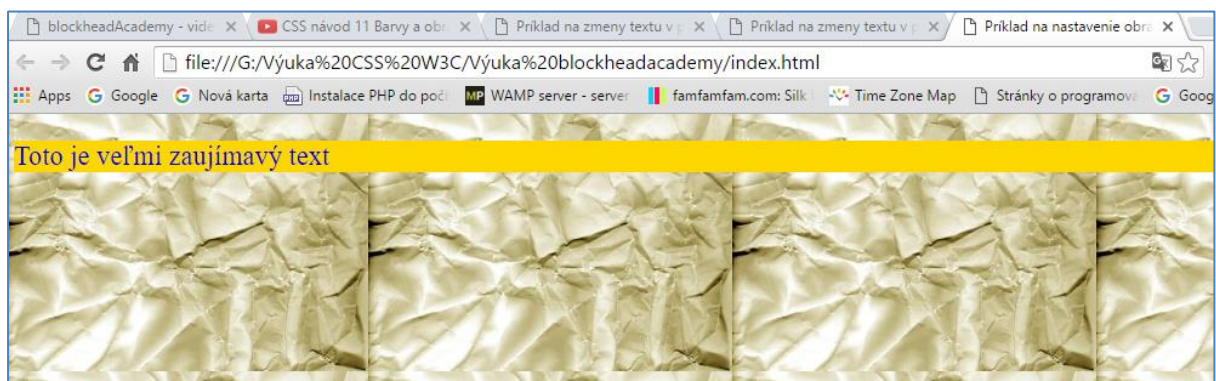
Súbor:index.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie obrázku na pozadí html dokumentu</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">
  </head>
  <body>
    <p>Toto je veľmi zaujímavý text</p>
  </body>
</html>
```

```
/* CSS subor s menom main.css*/

body{
  background-image:url("../images/pozadie.jpg");
  /*Nastavenie obrázku na celý dokument html,Například na index.html*/
}
p{
  font-size: 1.4em;
  color:#0000cc; /*Modrá farba textu vybraného elementu*/
  background-color:gold; /*Nastavenie farby pozadia vybraného elementu*/
}
```

Výstup:



Prípadne môže byť zapísaná adresa url aj bez úvodzoviek

```
background-image:url(../images/pozadie.jpg);
```

12.CSS návod 12 Obrázok na pozadí pokračovanie

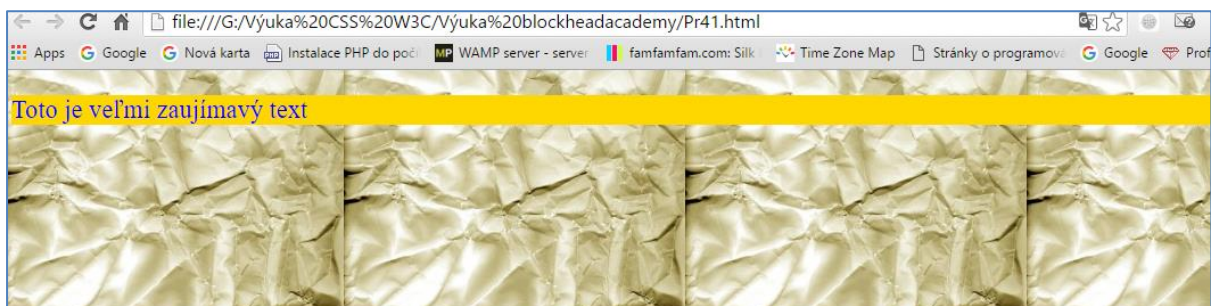
V prechádzajúcich príkladoch bol obrázok menší a preto sa zobrazoval ako dlaždice po celej ploche dokumentu. Ak chceme na pozadí zobraziť jeden obrázok musíme zmeniť spôsob zápisu. Defaultne je nastavené opakovanie doprava a dolu.

Ak chceme aby sa obrázok opakova iba doprava nastavíme opakovanie nasledovne:

Pr.12.1. Opakovanie obrázku v smere osi X

```
/* CSS subor s menom main.css */  
body {  
    background-image:url(../images/pozadie.jpg);  
    background-repeat: repeat-x;  
}  
  
p {  
    font-size: 1.4em;  
    color: #0000ff;  
    background-color: gold;  
}
```

Výstup:



Ak chceme aby sa obrázok opakoval iba dolu použijeme v repeat-y.

Najčastejšie budeme používať nastavenie aby sa obrázok neopakoval:

```

/* CSS subor s menom main.css*/
body {
    background-image:url(../images/pozadie.jpg);
    background-repeat:no-repeat;
}

p{
    font-size:1.4em;
    color:#0000ff;
    background-color:gold;
}

```

Výstup:



Ďalšie nastavenie je či sa bude obrázok pohybovať pri rolovaní stránky.

```

/* CSS subor s menom main.css*/
body {
    background-image:url(../images/pozadie.jpg);
    background-repeat:no-repeat;
    background-attachment: scroll; /* Defaultne je nastavenie aby
obrázok roloval s textom stránky, čiže akoby sme použili scroll*/
}

p{
    font-size:1.4em;
    color:#0000ff;
    background-color:gold;
}

```

.Aby sme to mohli vyskúšať potrebovali by sme dostatočne veľkú stránku.,čo my nemáme.

Ďalšia možnosť je nastaviť hodnotu fixed

```
/* CSS subor s menom main.css*/  
body {  
    background-image:url(../images/pozadie.jpg);  
    background-repeat:no-repeat;  
    background-attachment: fixed;/* Pri rolovaní je obrázok stále na to  
istom mieste a obsah stránky sa bude pohybovať*/  
}  
  
p{  
    font-size:1.4em;  
    color:#0000ff;  
    background-color:gold;  
}
```

Ako ostatné nastavenie je určenie presnej polohy na ktorej sa má obrázok na stránke objavovať.

```
/* CSS subor s menom main.css*/  
body {  
    background-image:url(../images/pozadie.jpg);  
    background-repeat:no-repeat;  
    background-attachment: scroll;/* Pri rolovaní sa obrázok posúva s  
textom*/  
    background-position: 50% 50%;/*Nastavenie pozície obrázku na  
stránke*/  
}  
  
p{  
    font-size:1.4em;  
    color:#0000ff;  
    background-color:gold;  
}
```

Výstup:



Pozíciu môžeme zadávať aj v pixeloch.

```
/* CSS subor s menom main.css */  
body {  
    background-image: url(../images/pozadie.jpg);  
    background-repeat: no-repeat;  
    background-attachment: scroll; /* Pri rolovaní sa obrázok posúva s  
    textom */  
    background-position: 30px 30px; /* Nastavenie stálej pozície obrázku  
    na stránke */  
}  
  
p {  
    font-size: 1.4em;  
    color: #0000ff;  
    background-color: gold;  
}
```



Ďalšie pozície obrázku:

- I. top left -obrázok hore vľavo
- II. top center-hore uprostred
- III. right top-hore vpravo
- IV. right center-vpravo v strede
- V. left center-vľavo v strede

VI. A ďalšie

Všetky nastavenia pre obrázok na pozadí môžeme napísať do jedného riadka nasledovne:

```
/* CSS subor s menom main.css*/
body {
    background:#eee url(..images/pozadie.jpg) no-repeat fixed 30px
30px;
}

p{
    font-size:1.4em;
    color:#0000ff;
    background-color:gold;
}
```

Obrázok na pozadí bol nastavený nasledovne:

- Farba pozadia: svetlo šedivá #eeeeee
- Obrázok na pozadí:pozadie.jpg
- Obrázok neopakovať_: no-repeat
- Pri rolované obrázok fixovať na pozícií 30 30

13.CSS návod 13 Rozmery a okraje elementu

Ukážeme si ako môžeme s css ovplyvniť rozmery a okraje elementov.Najskôr si nastavíme rozmery elementov.

PR.13.1

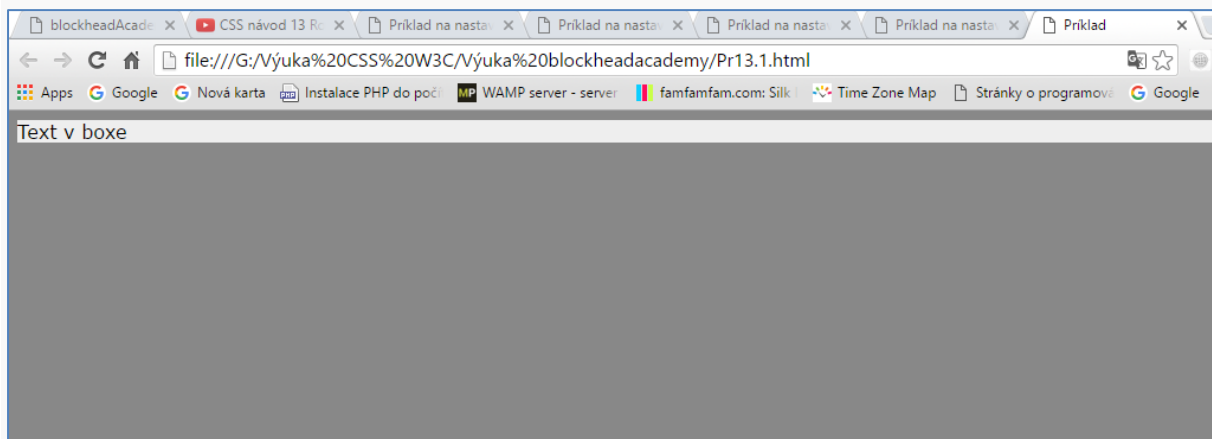
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad</title>
    <link href='images/logo.png' rel='shortcut icon' type='image/png'>
    <link href="css/main.css" rel="stylesheet" type="text/css" />
  </head>
</body>
<div>Text v boxe</div>
</body>
```



```
</html>

/* CSS subor s menom main.css*/
/* Nastavenie vzhľadu celého dokumentu*/
body {
background-color: #888;
font-family: Verdana;
font-size: 1em;
}
/* Nastavenie kontajneru*/
div {
    background-color: #eee;
}
```

výstup



Máme nastavenú farbu pozadia stránky na šedú a farbu kontajneru na bledú farbu. Kontajner je však pretiahnutý do konca stránky a okolo elementu div je veľmi málo miesta a preto pôsobí stiesnene.

Najskôr niečo urobíme s dĺžkou riadku width, potom nastavíme výšku boxu height a potom nastavíme okraje padding. Potom pomocou elementu span vyberieme časť riadku elementu na nastavenie hodnôt elementu.

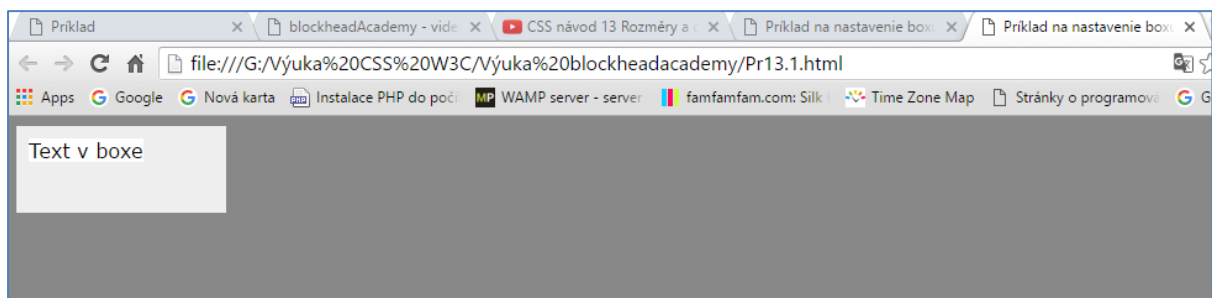
Pr13.2.Nastavenie boxu div

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie boxu</title>
    <link href='images/logo.png' rel='shortcut icon' type='image/png'>
    <link href="css/main.css" rel="stylesheet" type="text/css" />
  </head>
  <body>
    <div><span>Text v boxe</span></div>
  </body>
</html>
```

```
/* CSS subor s menom main.css*/
/* Nastavenie vzhľadu celého dokumentu*/
body {
background-color: #888;
font-family: Verdana;
font-size: 1em;
}

/* Nastavenie kontajneru*/
div {
    background-color: #eee;/*Nastavenie farby boxu div na
odtieň sivej*/
    width: 150px;/*Nastavenie šírky boxu*/
    height: 50px;/* Nastavenie výšky boxu*/
    padding: 10px;/* Nastavenie okraja celého boxu */
}
span {
    background-color: #fff;/* Nastavenie pozadia in-line výberového
elementu)*/
}
```

Vystup:

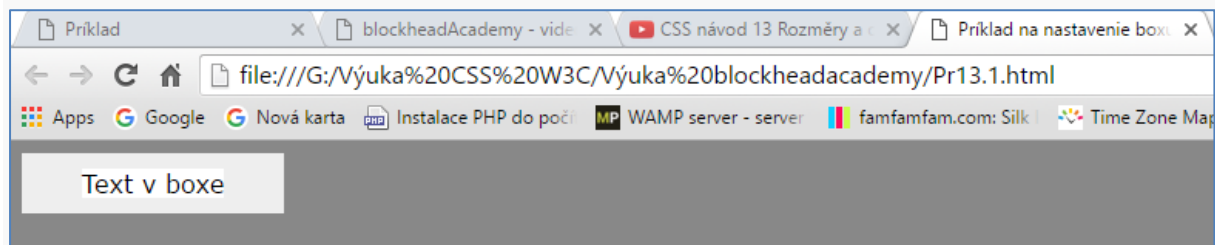


Máme nastavenú šírku a výšku boxu. Takže musíme počítať s tým, že sa nám okraj vpravo pripočíta k dĺžke a dole k výške.

Pr.13.3 Výšku teraz zrušíme a text boxu vycentrujeme:

```
/* CSS subor s menom main.css*/
/* Nastavenie vzhľadu celého dokumentu*/
body {
background-color: #888;
font-family: Verdana;
font-size: 1em;
}
/* Nastavenie kontajneru*/
div {
background-color: #eee; /*Nastavenie farby boxu div na
odtieň sivej*/
width: 150px; /*Nastavenie šírky boxu*/
padding: 10px; /* Nastavenie okraja celého boxu */
text-align: center; /* Text v boxe vycentrujeme*/
}
span {
background-color: #fff; /* Nastavenie in-line výberového elementu)*/
}
```

Výstup:



Vnútorňý okraj je po všetkých stranách rovnaký. Ale čo ak budeme chcieť rôzne hodnoty po stranách:

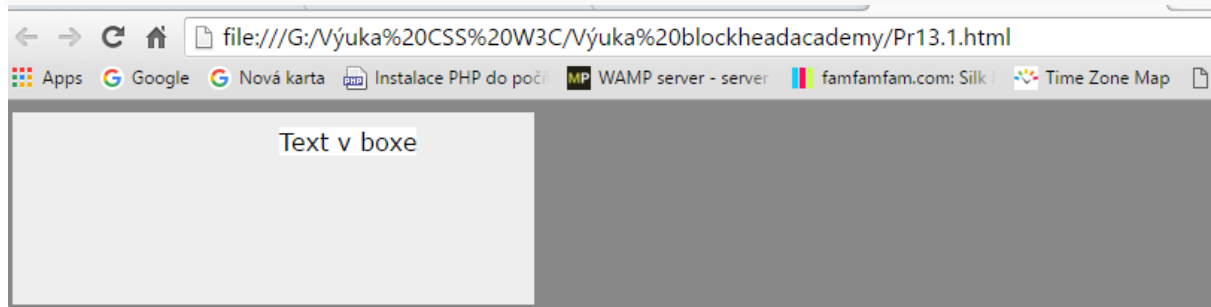
Pr.13.4

```
/* CSS subor s menom main.css*/
/* Nastavenie vzhľadu celého dokumentu*/
body {
background-color: #888;
font-family: Verdana;
font-size: 1em;
}
/* Nastavenie kontajneru*/
div {
background-color: #eee; /*Nastavenie farby boxu div na
odtieň sivej*/
```

```

width: 150px;/*Nastavenie šírky boxu*/
padding: 10px 50px 100px 150px;/* Nastavenie okraja celého boxu */
text-align: center;/* Text v boxe vycentrujeme*/
}
span {
background-color: #fff;/* Nastavenie in-line výberového elementu)*/
}

```



Okraje padding sa zadávajú v smere hodinových ručičiek:

1. Horný okraj
2. Právý okraj
3. Dolný okraj
4. Ľavý okraj

Ak budú rovnaké hodnoty hore a dolu a súčasne rovnaké hodnoty vľavo a vpravo môžeme to zapísať:

Pr.13.5. Rovnaké hodnoty v protíhlých stranách

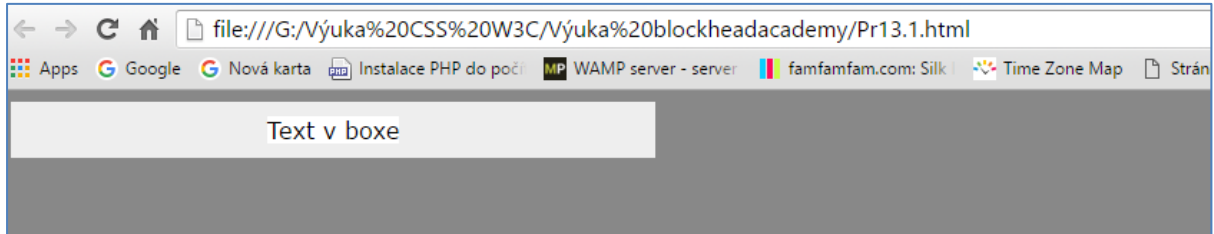
```

/* CSS subor s menom main.css*/
/* Nastavenie vzhľadu celého dokumentu*/
body {
background-color: #888;
font-family: Verdana;
font-size: 1em;
}
/* Nastavenie kontajneru*/
div {
background-color: #eee;/*Nastavenie farby boxu div na
odtieň sivej*/
width: 150px;/*Nastavenie šírky boxu*/
padding: 10px 150px ;/* Rovnaký horný a dolný a rovnaký ľavý a
pravý */
text-align: center;/* Text v boxe vycentrujeme*/
}
span {

```

```
background-color: #fff;/* Nastavenie in-line výberového elementu)*/  
}
```

Výstup

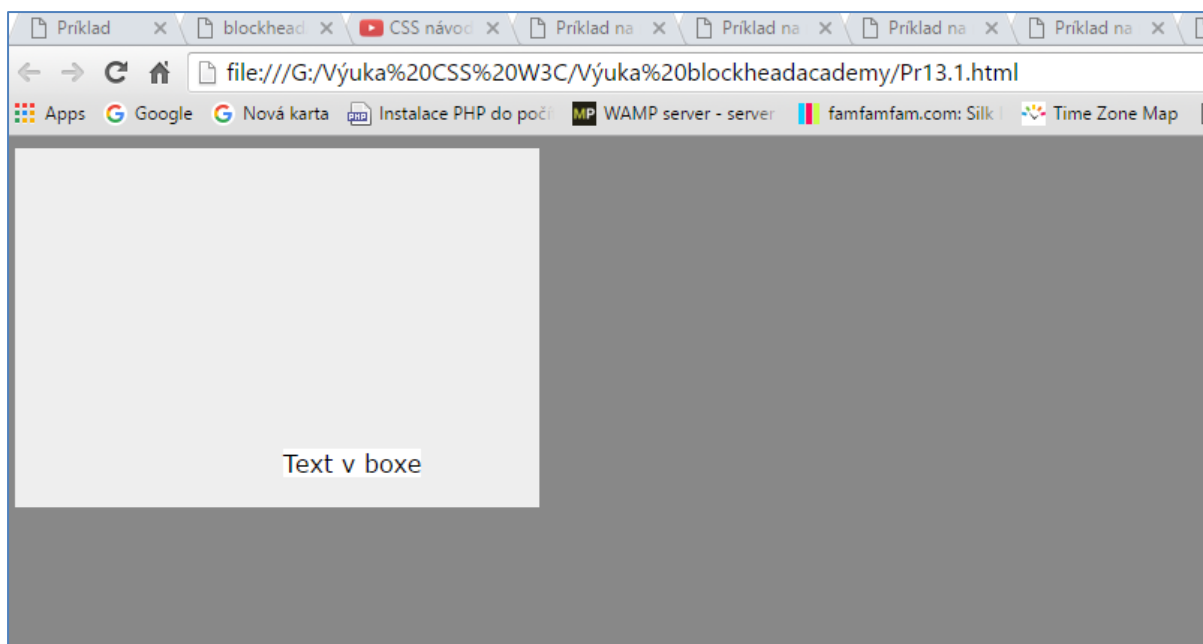


Text v boxe vyzerá akoby bol vycentrovaný. Ak by sa vám to zdalo zložité je možné zadať jednotlivé okraje do samostatných riadkov.

Pr.13.6 Nastavenie okrajov boxu samostatne.

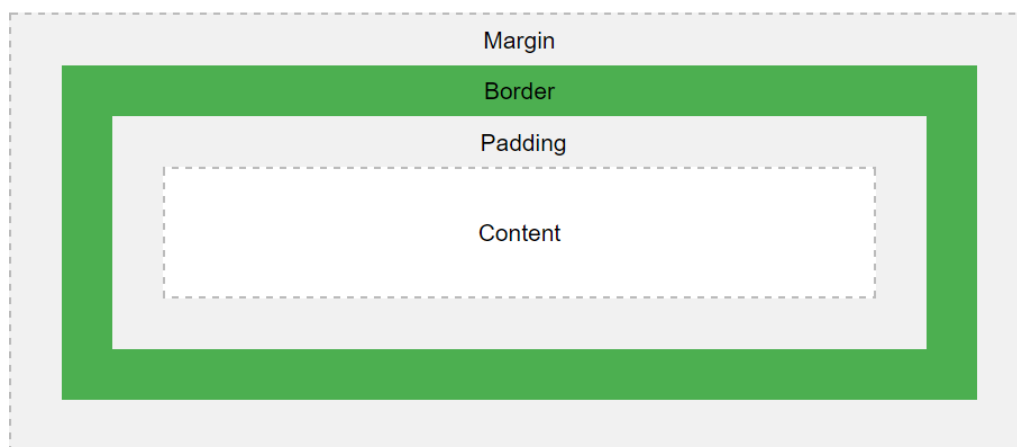
```
/* CSS subor s menom main.css*/  
/* Nastavenie vzhľadu celého dokumentu*/  
body {  
background-color: #888;  
font-family: Verdana;  
font-size: 1em;  
}  
/* Nastavenie kontajneru*/  
div {  
background-color: #eee;/*Nastavenie farby boxu div na  
odtieň sivej*/  
width: 150px;/*Nastavenie šírky boxu*/  
padding-left: 150px;/* Nastavenie ľavého okraja */  
padding-right: 50px;/* Nastavenie pravého okraja*/  
padding-top: 200px;/* Nastavenie horného okraja */  
padding-bottom: 20px;/* Nastavenie dolného okraja */  
text-align: center;/* Text v boxe vycentrujeme*/  
}  
span {  
background-color: #fff;/* Nastavenie in-line výberového elementu)*/  
}
```

Výstup:



V prechádzajúcom návode sii ukázali ako nastaviť vnútorné okraje elementu.pomocou vlastnosti PADDING.

14.CSS Box Model



Margin-okraj,Border-hranica,Padding-vypchávka,Content-obsah

Vysvetlenie rôznych častí:

- **content** - obsah krabice, kde sa objaví text a obrázky
- **padding** - Vyčistí oblasť okolo obsahu.
- **Border** - ohraničenie, ktorá vedie okolo vypchávky a obsahu
- **Margin** - Vymaže oblasti mimo hraníc.

Aby bolo možné nastaviť šírku a výšku prvku správne vo všetkých prehliadačoch, je potrebné vedieť, ako tento model box funguje.

Příklad 14.1. Nastavenia parametrov prvku

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  width: 320px; /*šírka prvku div*/
  padding: 10px; /*vypchávká */
  border: 5px solid gray; /*hranica*/
  margin: 0; /*Okraj*/
}
</style>
</head>
<body>
<h2>Výpočet celkovej šírky prvku:</h2>

<div>Obrázok hore je 350px široký. Celková šírka tohto prvku je tiež
350px.</div>
</div>
</body>
</html>
```

Výpočet celkovej šírky prvku:



Obrázok hore je 350px široký. Celková šírka tohto prvku je tiež 350px.

Matematická kalkulácia vychádzajúca s parametrov prvku div:

320 (šírka)
+ 20px (ľavá + pravá padding)
+ 10px (ľavý + pravý border)
+ 0px (ľavý + pravý margin)
= **350px**

PREKLAD:

- **WIDTH-šírka**
- **PADDING-výplň**
- **BORDER-hranica**
- **MARGIN-okraj**

Celková šírka prvku by sa mala vypočítať takto:

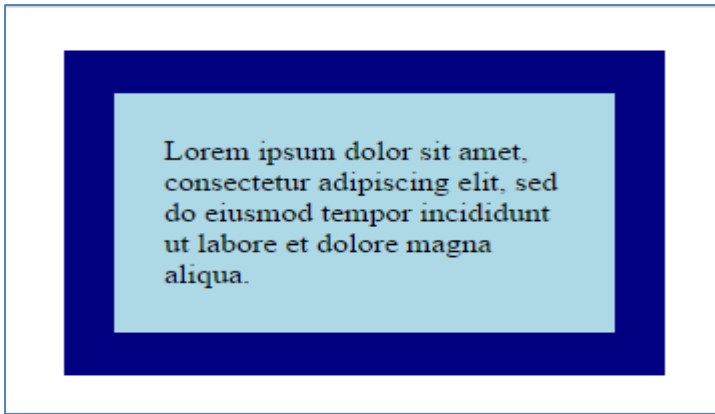
Celková šírka prvok = šírka + ľavý padding + vpravo padding + vľavo border + pravý border + ľavý margin + pravý margin

Celková výška prvku by sa mala vypočítať takto:

Celková výška prvoku = výška + horné padding + dolný padding + horný border + spodný border + horný margin + spodný margin

Cvičenie1

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  background-color: lightblue;
  width: 200px;
  padding: 25px;
  border: 25px solid navy;
  margin: 25px;
}
</style>
</head>
<body>
<div>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do
eiusmod tempor incididunt ut labore et dolore magna aliqua.</div>
</body>
</html>
```

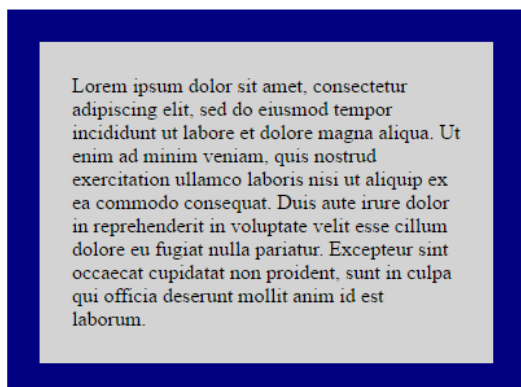



Cvičenie2

```
<!DOCTYPE html>
<html>
<head>
<style>
div {
  background-color: lightgrey;
  width: 300px;
  padding: 25px;
  border: 25px solid navy;
  margin: 25px;
}
</style>
</head>
<body>
<h2>Demonštrácia Box Modelu</h2>
<p>Tento model CSS boxu je v podstate box, ktorý obaľuje každý prvok HTML. Skladá sa z: okrajov(margin), hranice(border), výstielky(padding) a skutočného obsahu(content).</p>
<div>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</div>
</body>
</html>
```

Demonštrácia Box Modelu

Tento model CSS boxu je v podstate box, ktorý obaľuje každý prvok HTML. Skladá sa z: okrajov, hranice, výstielky a skutočného obsahu.



Príklad trocha odveci - prvok inline

Pr14.1. Použitie prvku span na výber časti textu

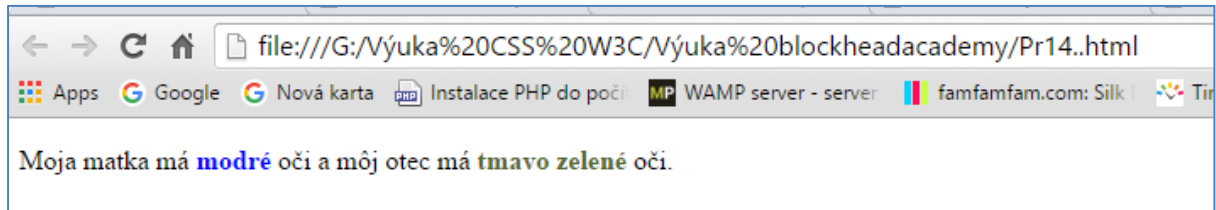
```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na zvýraznenie vybranej časti textu v elemente "p".
  </title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>

  <p>Moja matka má <span class="matka" >modré</span> oči a môj otec má <span
class="otec">tmavo zelené </span> oči.</p>

</body>
</html>
```

```
/* CSS subor s menom main.css*/
.matka{
  color:blue;
  font-weight:bold;
}
.otec{
  color:darkolivegreen;
  font-weight:bold;
}
```

Výstup:



V prechádzajúcom návode si ukázali ako nastaviť výplň elementu pomocou vlastnosti PADDING.

Dnes budeme pokračovať vonkajšími okrajmi(margin) a rámčekom(border).

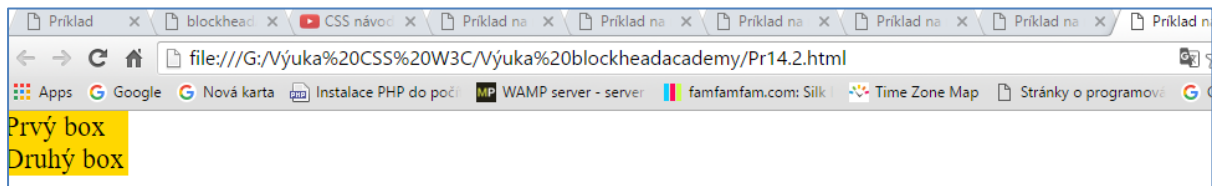
Vonkajšie okraje elementu sa nastavujú voľbou vlastnosti **margin**.

Pr14.2.Nastavovanie vonkajších okrajov pre element div.

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie vonkajších okrajov vybraného elementu</title>
    <link href='images/logo.png' rel='shortcut icon' type='image/png'>
    <link href="css/main.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <div>Prvý box</div>
    <div>Druhý box</div>
  </body>
</html>
```

```
/* CSS subor s menom main.css*/
/*Východzie nastavenie pre všetky prvky "body"*/
body {
  padding:0;
  margin:0;
  font-size: 1.4em;
}
/* Nastavovanie vlastností iba pre obidva boxy*/
div {
  width:100px;/* Nastavenie šírky boxu*/
  background-color: gold;
}
```

Výstup:

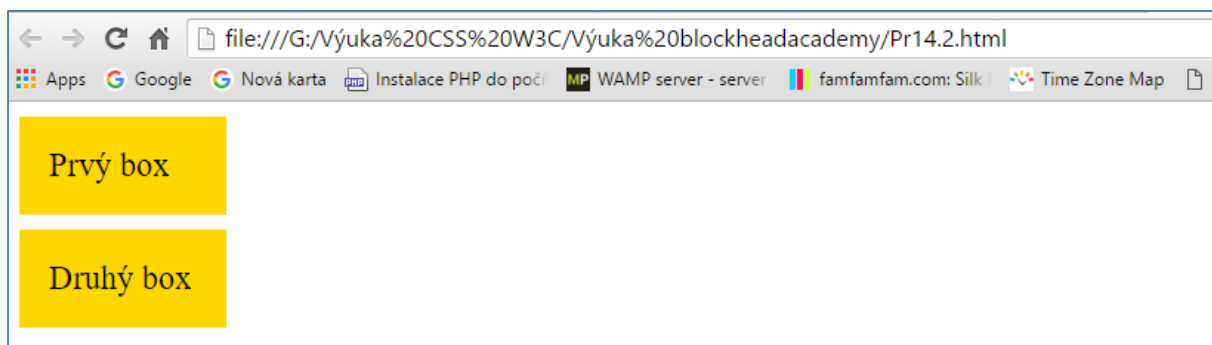


Toto je východzie nastavenie padding a margin. Oba boxy sú umiestnené do ľavého horného rohu a nevidieť kde končí prvý box a kde začína druhý box.

Pr14.2. Nastavenie vlastností boxov.

```
/* CSS subor s menom main.css */
/*Východzie nastavenie pre všetky prvky "body"*/
body {
    padding: 0;
    margin: 0;
    font-size: 1.4em;
}
/* Nastavovanie vlastností iba pre obidva kontajnéry */
div {
    width: 100px; /* Nastavenie šírky kontajneru */
    background-color: gold;
    padding: 20px; /* Nastavenie vnútorných okrajov obom divom */
    margin: 10px; /* Nastavenie vonkajších okrajov */
}
```

Výstup:



Teraz už nie sú boxy úplne vľavo hore a tiež vidieť kde ktorý box začína a končí.

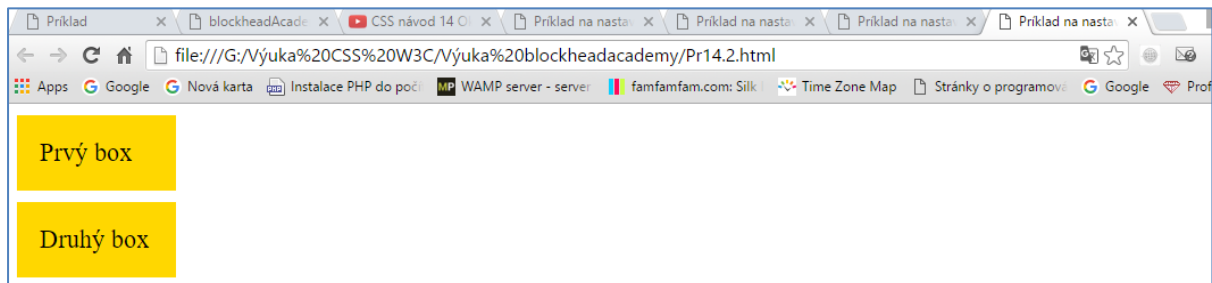
Pre zadávanie hodnôt MARGIN platí to isté ako pre PADDING.

```
margin:10px 20px 30px 40px;/*Nastavenie vonkajších okrajov v poradí  
horný,pravý,dolný,ľavý*/
```

alebo

```
/* CSS subor s menom main.css*/  
/*Východzie nastavenie pre všetky prvky "body"*/  
body {  
    padding:0;  
    margin:0;  
    font-size: 1.4em;  
}  
/* Nastavovanie vlastností iba pre obidva kontajnéry*/  
div {  
    width:100px;/* Nastavenie šírky kontajneru*/  
    padding: 20px;/*Nastavenie vnútorných okrajov*/  
    background-color: gold;  
    margin-top:10px;/* Nastavenie vonkajších okrajov*/  
    margin-right:10px;  
    margin-bottom:10px;  
    margin-left:10px;  
}
```

Výstup:



Vonkajší okraj už nemá farbu, ktorú sma nastavili pri „body“ elementu „DIV“.Vonkajší okraj slúži na nastavenie vzdialenosti od ostatných elementov.

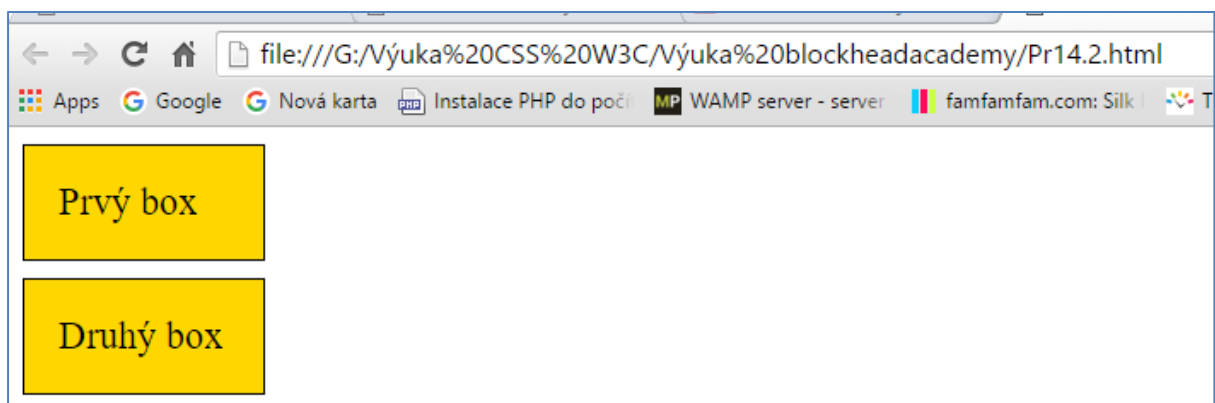
PR14.3 .Ako nastaviť rámček elementu DIV

```
<!DOCTYPE html>  
<html lang="sk">  
    <head>  
        <meta charset="UTF-8">  
<title>Príklad na nastavenie vonkajších okrajov vybraného elementu</title>  
        <link rel="stylesheet" type="text/css" href="css/main.css"/>  
    </head>
```

```
<body>
<div>Prvý box</div>
<div>Druhý box</div>
</body>
</html>
```

```
/* CSS subor s menom main.css*/
/*Východzie nastavenie pre všetky prvky "body"*/
body {
    padding:0;
    margin:0;
    font-size: 1.4em;
}
/* Nastavovanie vlastností iba pre obidva kontajnéry*/
div {
    width:100px;/* Nastavenie šírky kontajneru*/
    padding: 20px;/*Nastavenie vnútorných okrajov*/
    background-color: gold;
    margin-top:10px;/* Nastavenie vonkajších okrajov*/
    margin-right:10px;
    margin-bottom:10px;
    margin-left:10px;
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu"DIV"
}
}
```

Výstup:



Okolo prvkov „DIV“ je teraz rámček, hrúbkou 1 pixel čiernou a súvislou čiarou. SOLID určuje aký štýl bude mať rámček. Rámček môže byť aj bodkovanou čiarou, prípadne iným štýlom.

CVIČENIE

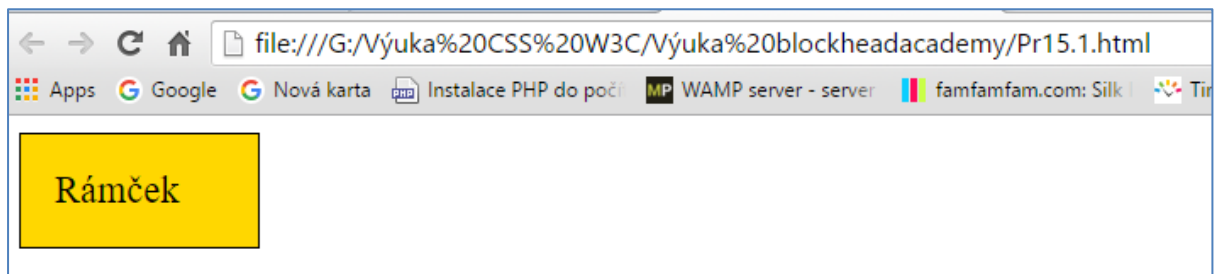
15. CSS návod 15 Rámček BORDER detailnejšie

Pr15.1 Príklad nastavenia vlastnosti a ramčeka kontajnera DIV

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
  <title>Príklad na nastavenie rámčeka</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div>Rámček</div>
  </body>
</html>
```

```
/* CSS subor s menom main.css*/
div{
  background-color:#CCC; /*Odtieň sivej*/
  width:200px; /*Šírka kontajnere 200 pixelov*/
  padding:10px; /*Nastavenie vnútorných okrajov na rovnakú hodnotu*/
  font-size: 1.4em; /* 1em = 16px, teda je to 22.4px*/
  border:1px solid #888; /*Rámček hrubý 1px ,plná čiara, Tmavo šedá farba*/
}
```

Výstup:



Zmeníme nastavenie rámčeka:

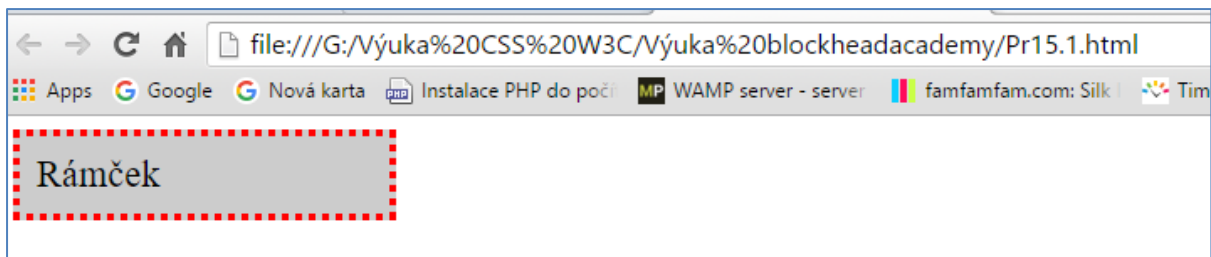
```
* CSS subor s menom main.css*/
div{
```

```

    background-color:#CCC;/*Odtieň sivej*/
    width:200px;/*Šírka kontajnere 200 pixelov*/
    padding:10px;/*Nastavenie vnútorných okrajov na rovnakú hodnotu*/
    font-size: 1.4em;/* 1em = 16px,teda je to 22.4px*/
    border:4px dotted #ff0000;/*Rámček hrubý 4px ,bodkovaná čiara,červená
farba rámčeka */
}

```

Výstup:



Štýl čiary môže byť:

1. solid-plná čiara
2. dotted-bodkovaná čiara
3. dashed- čiarkovaná čiara
4. double-dvojitý plný rámček

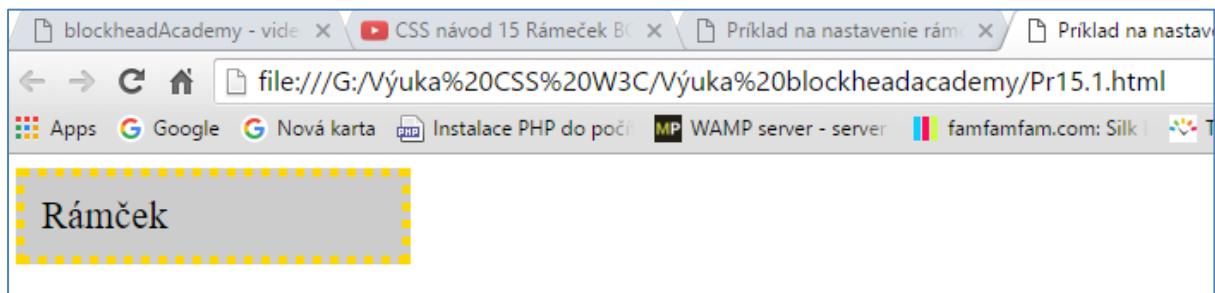
Vlastnosti rámčeka môžeme nastavovať individuálne vo viacerých riakoch.Ak sme na začiatku použili neaké nastavenie a potom sme ho zmenili , tak platí nastavenie uvedené ako posledné.

```

/* CSS subor s menom main.css*/
div{
    background-color:#CCC;/*Odtieň sivej*/
    width:200px;/*Šírka kontajnere 200 pixelov*/
    padding:10px;/*Nastavenie vnútorných okrajov na rovnakú hodnotu*/
    font-size: 1.4em;/* 1em = 16px,teda je to 22.4px*/
    border:4px dotted #ff0000;/*Rámček hrubý 4px ,bodkovaná čiara,červená
farba rámčeka */
    /*Toto nastavenie bolo urobené neskôr a preto platí toto nastavenie*/
    border-width: 5px;
    border-style: dotted;
    border-color: gold;
}

```


Výstup:



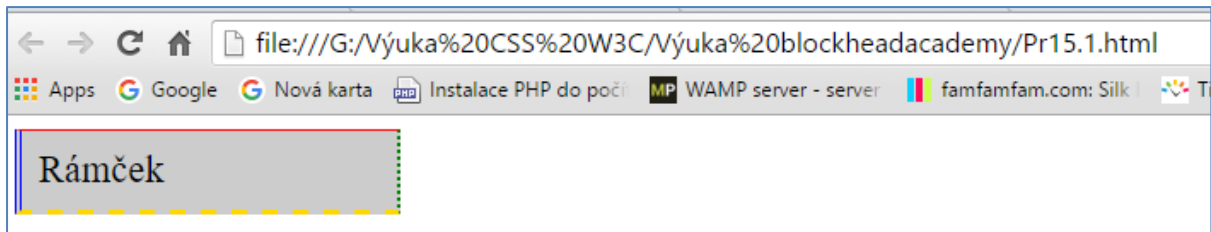
Zatiaľ sme nastavovali rámček pre všetky strany rovnký. Môžeme však nastaviť jednotlivé strany rámčeka individuálne:

Pr.15.2. Nastavenie strán ramčeka individuálne

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
<title>Príklad na nastavenie rámčeka</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
<body>
<div>Rámček</div>
</body>
</html>
```

```
/* CSS subor s menom main.css*/
div{
  background-color:#CCC; /*Odtieň sivej*/
  width:200px; /*Šírka kontajnera 200 pixelov*/
  padding:10px; /*Nastavenie vnútorných okrajov na rovnakú hodnotu*/
  font-size: 1.4em; /* 1em = 16px, teda je to 22.4px*/
  border-top:1px solid red; /*Plná*/
  border-right:2px dotted green; /*bodkovaná*/
  border-bottom:3px dashed gold; /*ciarkovaná*/
  border-left:4px double blue; /*dvojitá*/
}
```

Výstup

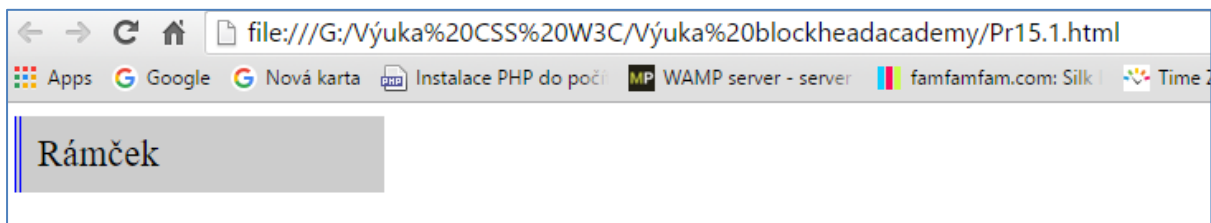


Pri pohľade do prehliadača je táketo nastavenie málo používané.Ak je vôbec používané.

Možné je však takéto použitie.

```
/* CSS subor s menom main.css */
div{
    background-color:#CCC; /*Odtieň sivej*/
    width:200px; /*Šírka kontajnera 200 pixelov*/
    padding:10px; /*Nastavenie vnútorných okrajov na rovnakú hodnotu*/
    font-size: 1.4em; /* 1em = 16px, teda je to 22.4px*/
    border-left:4px double blue; /*dvojitá*/
}
```

Výstup



16.CSS návod 16 Zaguľatený rámeček BORDER RADIUS

Pr.16.1.Úvodné nastavenie kontajnera

```
<!DOCTYPE html>
```

```
<html lang="sk">
  <head>
    <meta charset="UTF-8">
  <title>Príklad na nastavenie kontajnera</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div></div>
  </body>
</html>
```

```
/* CSS subor s menom main.css*/
```

```
div{
  width: 300px; /*Šírka kontajnera*/
  height: 150px; /*výška kontajnera DIV*/
  background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
  margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
}
```

výstup



Úprava pre vytvorenie zaguľatenia rohov kontajnera

```
/* CSS subor s menom main.css*/
div{
  width: 300px; /*Šírka kontajnera*/
```

```
height: 150px;/*výška kontajnera DIV*/
background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri
rohy*/
}
```

Výstup:



S zagulatením môže robiť nasledovné úpravy:

1.U každého rohu iné zagulatenie s zagulatením pomocou polomeru kružnice

To urobíme tak že do zápisu rádiusov zadáme aj ostatné rohy.

```
/* CSS subor s menom main.css*/
div{
    width: 300px;/*Šírka kontajnera*/
    height: 150px;/*výška kontajnera DIV*/
    background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
    margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
    /*Zadávanie rádiusov začína v ľavom hornom rohu a pokračuje ďalej
    v smere hodinových ručičiek*/
    border-radius: 10px 20px 30px 40px;/*Nastavenie rádiusu pre všetky
každý roh samostatný rádius*/
}
```

Výstup:



1. Prvá hodnota nastaví ľavý horný roh
2. druhá pravý horný roh
3. pravý dolný roh
4. ľavý dolný roh

Nastavenie rádiusov môžeme umiestniť do samostatných riadkov:

```
/* CSS subor s menom main.css */
div{
    width: 300px; /*Šírka kontajnera*/
    height: 150px; /*výška kontajnera DIV*/
    background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
    margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
    border-top-left-radius: 10px; /* ľavý horný*/
    border-top-right-radius: 20px; /*pravý horný*/
    border-bottom-right-radius: 30px; /*pravý dolný*/
    border-bottom-left-radius: 40px; /*dolný ľavý*/
}
```

Výstup:



Výsledok je rovnaký ako v prechádzajúcom príklade. Zapisovanie môže byť výhodné z hľadiska prehľadnosti.

Najčastejšie sa používa v prípade, keď chceme nastaviť jeden roh inak ako ostatné rohy.

```
/* CSS subor s menom main.css */
div{
    width: 300px; /*Šírka kontajnera*/
    height: 150px; /*výška kontajnera DIV*/
    background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
    margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
    border-radius: 20px; /* Všetky rohy okrem ľavého dolného 20 pixelov*/
    border-bottom-left-radius: 0px;
}
```

Výstup:

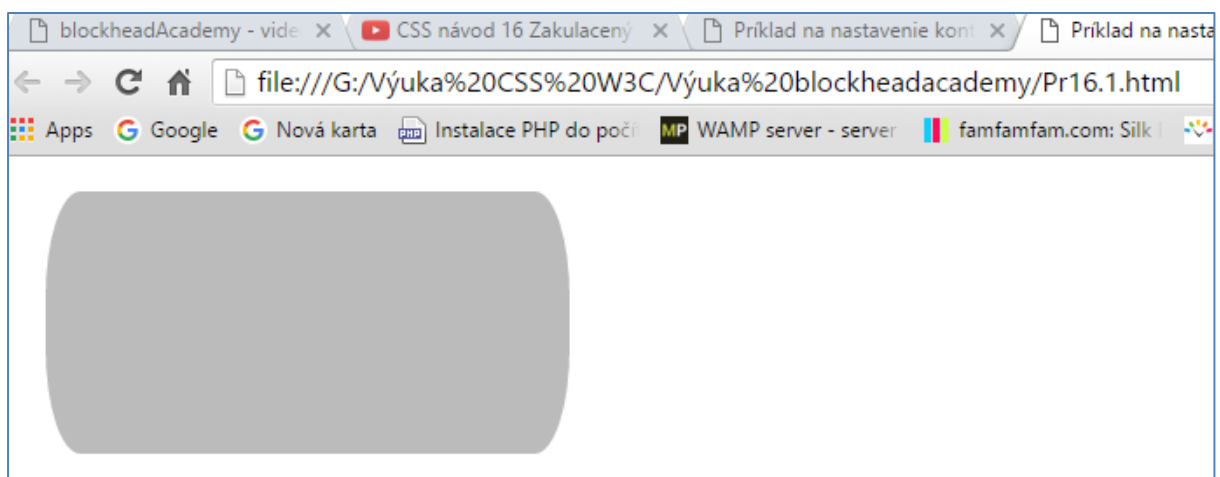


2. U každého rohu zaogúľatenie pomocou polomerov dvoch kružníc

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie kontajnera</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div></div>
  </body>
</html>
```

```
/* CSS subor s menom main.css */
div{
  width: 300px; /*Šírka kontajnera*/
  height: 150px; /*výška kontajnera DIV*/
  background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
  margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
  border-radius: 20px/60px; /* Všetky rohy zaguľatené pomocou dvoch
polomerov */
}
```

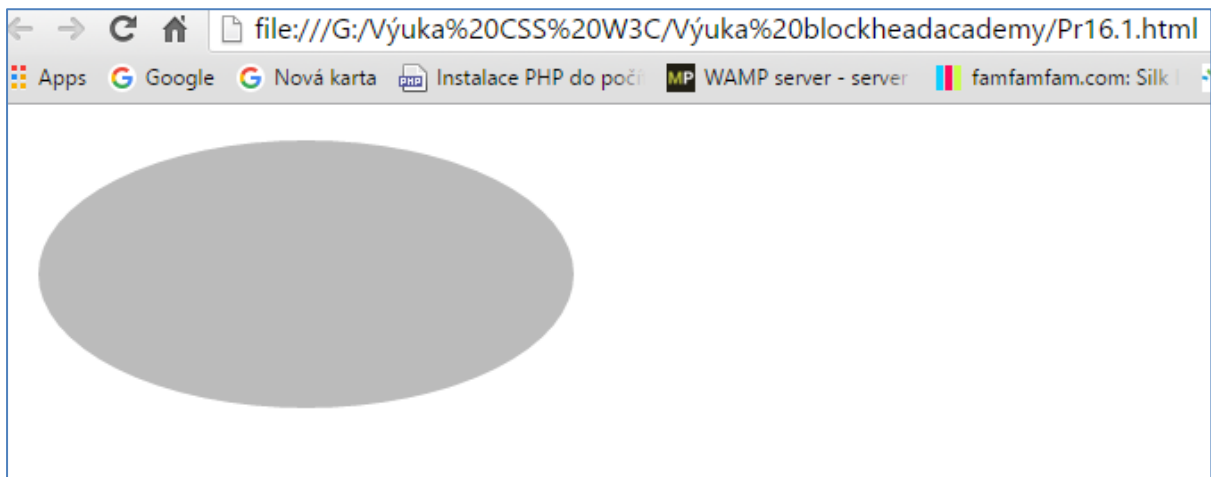
Výstup:



Vrátíme sa k zadávanie zaguľatenia pomocou jedného polomeru. Pomocou zaguľatenia elementu môžeme vytvoriť elipsu:

```
/* CSS subor s menom main.css*/  
  
div{  
    width: 300px; /*Šírka kontajnera*/  
    height: 150px; /*výška kontajnera DIV*/  
    background-color: #bbb; /* Šedivá farba pozadia kontajneru*/  
    margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/  
    border-radius: 50%; /* Všetky rohy zaguľatené pomocou percent */  
}
```

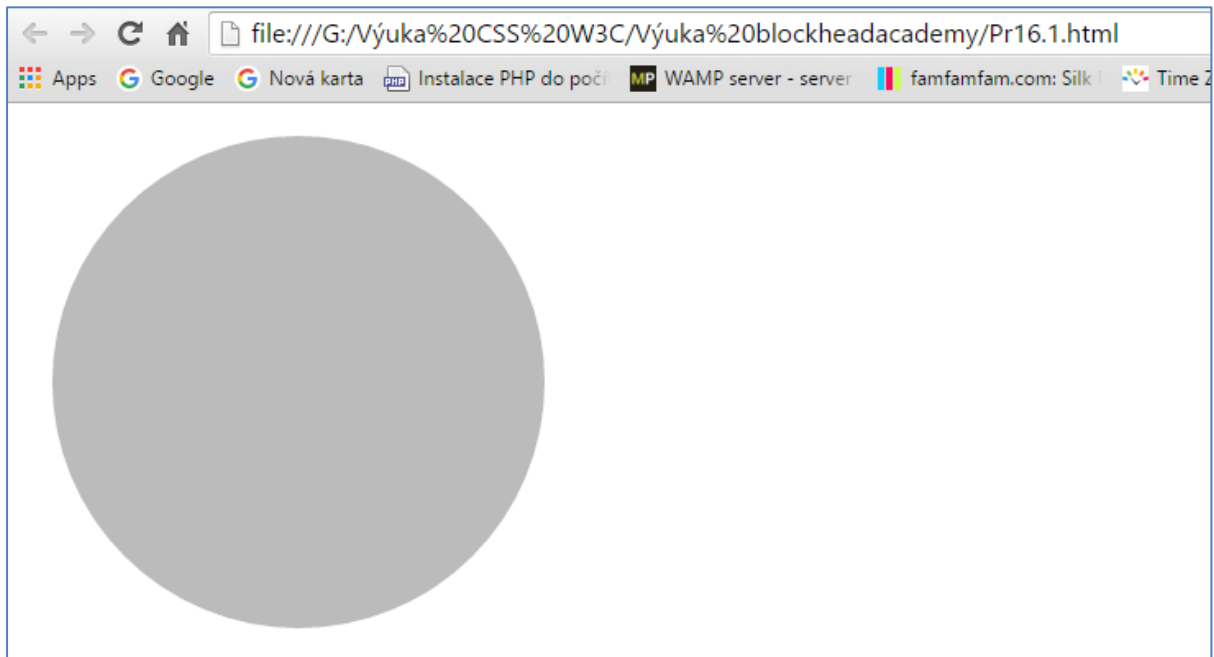
Výstup



Pomocou rádiusov môžeme vytvoriť aj kruh - width a height musia byť rovnaké

```
/* CSS subor s menom main.css*/  
  
div{  
    width: 300px; /*Šírka kontajnera*/  
    height: 300px; /*výška kontajnera DIV*/  
    background-color: #bbb; /* Šedivá farba pozadia kontajneru*/  
    margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/  
    border-radius: 50%; /* Všetky rohy zaguľatené pomocou percent */  
}
```


Výstup



Pr.snehuliak.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na vykreslenie kružníc</title>
    <link rel="stylesheet" type="text/css" href="css/gula.css"/>
  </head>
  <body>
    <div id="horna"></div>
    <div id="stredna"></div>
    <div id="dolna"></div>
  </div>
</body>
</html>
```

gula.css

```
/* CSS subor s menom gula.css */
#horna{
  width: 200px; /*Šírka kontajnera*/
  height: 200px; /*výška kontajnera DIV*/
  background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
  margin-left: 100px;
  border-radius: 50%; /* Všetky rohy zaguľatené pomocou percent */
```

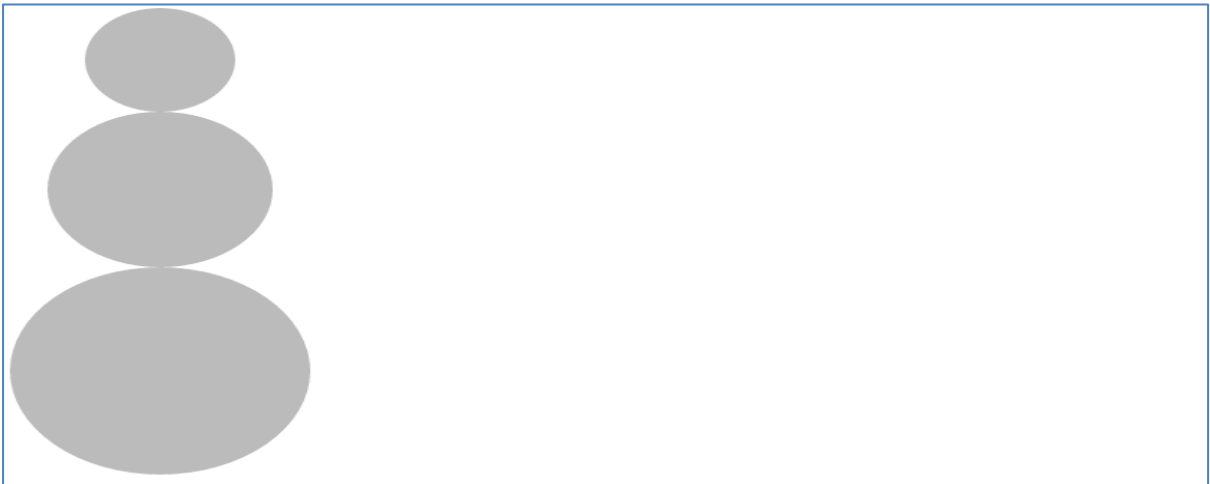
```

    }
#stredna{
    width: 300px;/*Šírka kontajnera*/
    height: 300px;/*výška kontajnera DIV*/
    background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
    /*margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
    border-radius: 50%;/* Všetky rohy zaguľatené pomocou percent */
    margin-left: 50px;

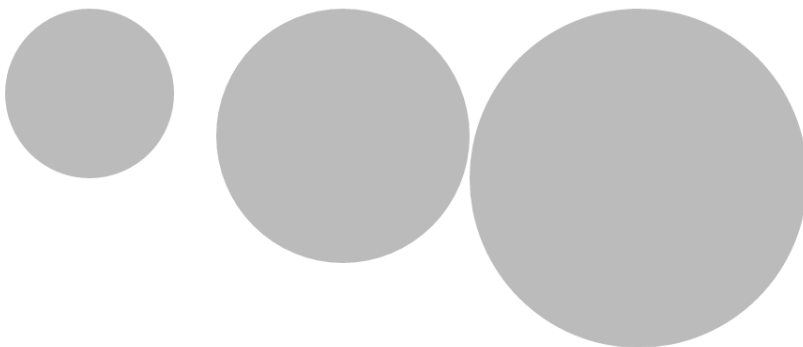
    }
#dolna{
    width: 400px;/*Šírka kontajnera*/
    height: 400px;/*výška kontajnera DIV*/
    background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
    /*margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
    border-radius: 50%;/* Všetky rohy zaguľatené pomocou percent */

```

CHROME:

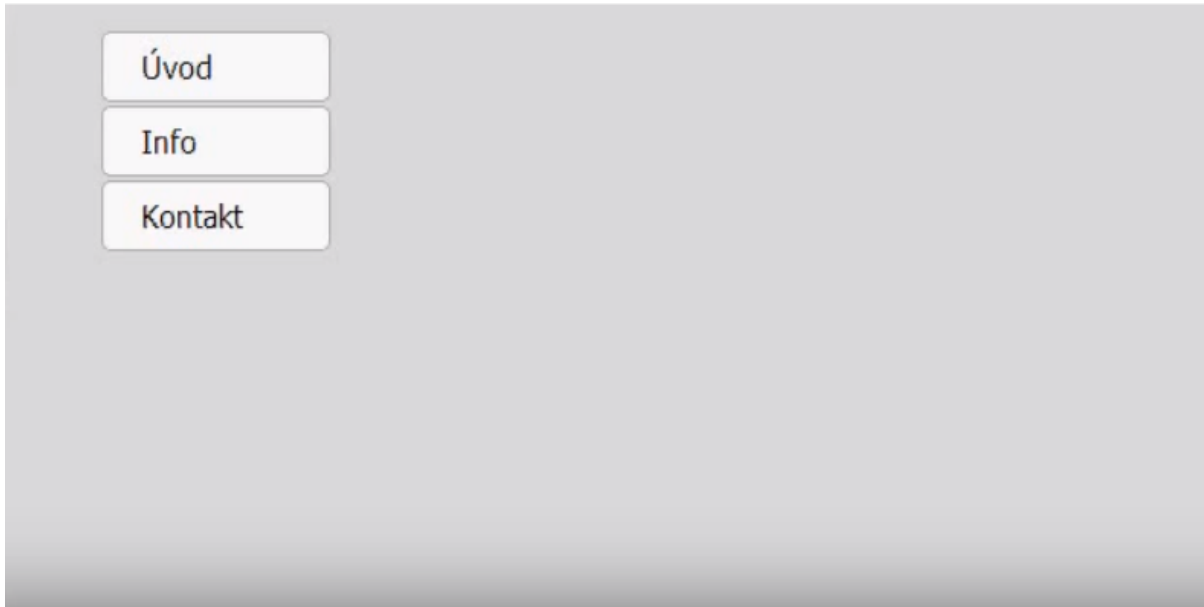


Toto dostaneme ak použijeme na snehuliaka **float**



17.CSS návod 17 Jednoduché menu pomocou CSS

Použitím vedomostí získaných v prechádzajúcich návodov sa pokúsime vytvoriť jednoduché menu.Chceme vytvoriť nasledovnú stránku:



Ide o jednoduché menu vytvorené zo zoznamu a odkazov.Ako odkaz je použitý znak # hash,takže odkazy nevedú nikam.Nie je však problém uviesť namiesto # skutočný odkaz.

Pokiaľ prejdeme na položku zoznamu myšou ,zmení sa farba pozadia položky.

Pr.17.1.Vytvorenie menu

Subor

- 1.index.html- html dokument
- 2.main.css –css dokument

index.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na vytvorenie menu podľa predlohy</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
  <body>
    <div id="menu">
      <ul>
        <li><a href="#">Úvod</a></li>
```

```
<li><a href="#">Info</a></li>
<li><a href="#">Kontakt</a></li>
</ul>
</div>
</body>
</html>
```

main.css

```
/* CSS subor s menom main.css*/
body{
    font-family: Tahoma,Arial,sans-serif;
    font-size: 1em;/*16 pixelov*/
    background-color: #ddd;/*svetlo šedé pozadie html dokumentu*/
}
/* Toto nastavenie platí pre zoznam v kontajneri "menu"*/
#menu ul {
    list-style: none;/* týmto nastavením odstránime odrážky so
zoznamu*/
}
/*Zobrazenie kružnice na pozadí*/
div{
    width: 300px;/*Šírka kontajnera*/
    height: 300px;/*výška kontajnera DIV*/
    background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
    margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
    border-radius: 50%;/* Všetky rohy zaguľatené pomocou percent */
}
```

Výstup čiastočného návrhu

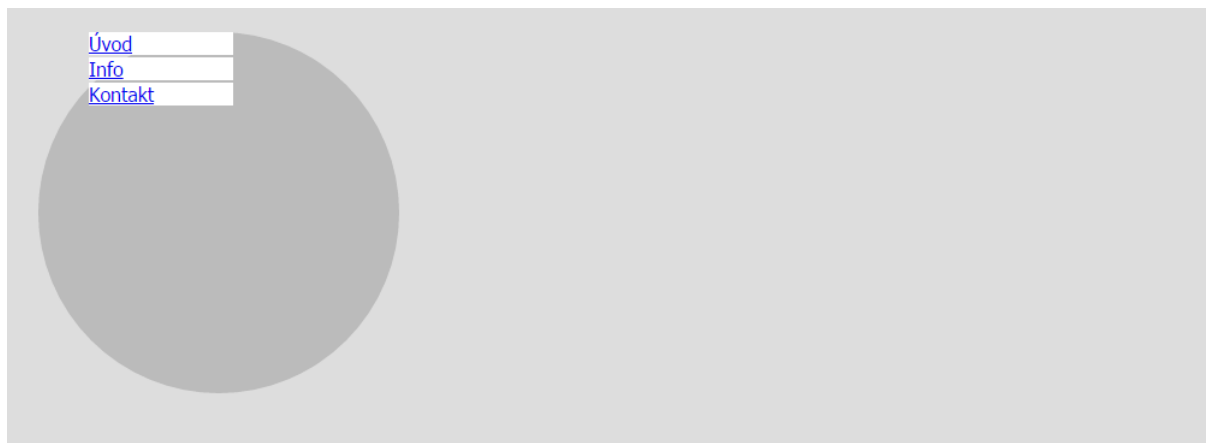


```

/* CSS subor s menom main.css*/
body{
    font-family: Tahoma,Arial,sans-serif;
    font-size: 1em;
    background-color: #ddd;/*svetlo šedé pozadie html dokumentu*/
}
/* Toto nastavenie platí pre zoznam v kontajnery "menu"*/
#menu ul {
    list-style: none;/* týmto nastavením odstránime odrážky so
zoznamu*/
}
/* Týmto nastavíme položky zoznamu*/
#menu li{
    width:120px;
    margin: 2px;
}
/* Nastavenie odkazu v položke zoznamu*/
#menu li a{
    display:block;/*Toto nastavenie pre odkaz preto,aby sme s odkazom
mohli pracovať ako s blokom,a mohli sme lepšie nastaviť vlastnosti odkazu*/
    background-color: #fff;/* Pozadie odkazu je biele*/
}
/*Zobrazenie kružnice na pozadí*/
div{
    width: 300px;/*Šírka kontajnera*/
    height: 300px;/*výška kontajnera DIV*/
    background-color: #bbb;/* Šedivá farba pozadia kontajneru*/
    margin: 20px;/* vonkajší okraj,aby nebol kontajner úplne vľavo hore*/
    border-radius: 50%;/* Všetky rohy zaguľatené pomocou percent */
}

```

Výstup čiastkový



Nastavenie odkazov v položkách zoznamu „li“

```
/* Toto nastavenie platí pre zoznam v kontajneri "menu"*/
#menu ul {
    list-style: none; /* týmto nastavením odstránime odrážky so
zoznamu*/
}
/* Týmto nastavíme položky zoznamu*/
#menu li{
    width:120px;
    margin: 2px;
}
/* Nastavenie odkazu v položke zoznamu*/
#menu li a{
    display:block; /*Toto nastavenie pre odkaz preto, aby sme s odkazom
mohli pracovať ako s blokom, a mohli sme lepšie nastaviť vlastnosti odkazu*/
    background-color: #fff; /* Pozadie odkazu je biele*/
    padding: 8px 20px; /* Hore a dolu 8px, Vľavo a vpravo 20px*/
    border:1px solid #aaa; /*Nastavenie rámčeka okolo odkazu*/
    border-radius: 5px; /*Nastavenie rádiusu pre zaoblenie všetkých
rohov odkazu*/
}
/*Zobrazenie kruhu na pozadí*/
div{
    width: 300px; /*Šírka kontajnera*/
    height: 300px; /*výška kontajnera DIV*/
    background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
    margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
    border-radius: 50%; /* Všetky rohy zaguľatené pomocou percent */
}
```

Výstup čiastkový



Pr.17.2.Hotové riešenie menu skoro podľa predlohy

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
<title>Príklad na vytvorenie menu podľa predlohy</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
<body>
<div id="menu">
<ul>
  <li><a href="#">Úvod</a></li>
  <li><a href="#">Info</a></li>
  <li><a href="#">Kontakt</a></li>
</ul>
</div>
</body>
</html>
```

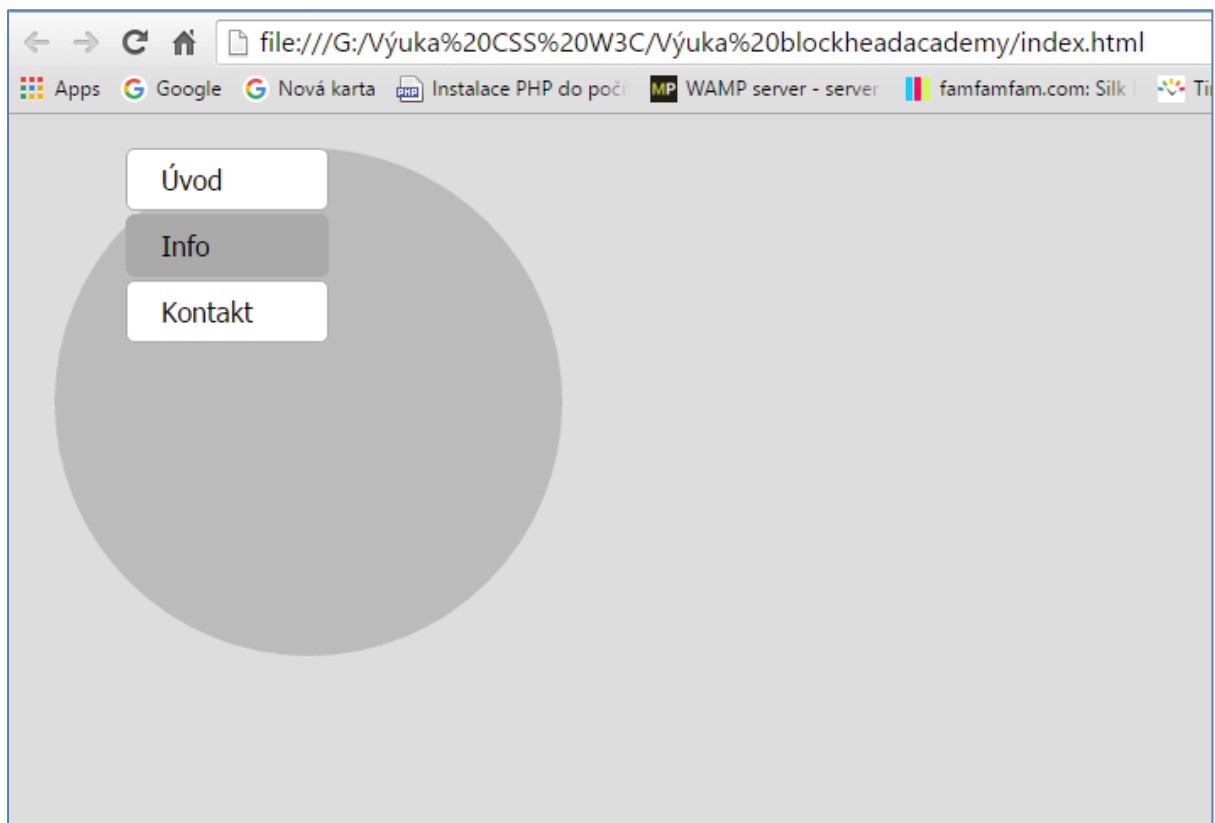
```
/* CSS subor s menom main.css*/
body{
  font-family: Tahoma,Arial,sans-serif;
  font-size: 1em;
  background-color: #ddd;/*svetlo šedé pozadie html dokumentu*/
}
/* Toto nastavenie platí pre zoznam v kontajnery "menu"*/
#menu ul {
  list-style: none;/* týmto nastavením odstránime odrážky so
zoznamu*/
}
/* Týmto nastavíme položky zoznamu*/
#menu li{
  width:120px;
  margin: 2px;
}
/* Nastavenie odkazu v položke zoznamu*/
#menu li a{
  display:block;/*Toto nastavenie pre odkaz preto,aby sme s odkazom
mohli pracovať ako s blokom,a mohli sme lepšie nastaviť vlastnosti odkazu*/
  background-color: #fff;/* Pozadie odkazu je biele*/
  padding: 8px 20px;/* Hore a dolu 8px,Vľavo a vpravo 20px*/
  border:1px solid #aaa;/*Nastavenie rámčeka okolo odkazu*/
```

```

border-radius: 5px; /*Nastavenie rádiusu pre zaoblenie všetkých
rohov odkazu*/
color:#000; /* Nastavenie farby textu v odkaze na čiernu*/
text-decoration:none; /* Zrušenie podčiarknutia textu v odkaze*/
}
/* zmena pozadia položky zoznamu odkazu po prejdutí myšou*/
#menu li a:hover{
background-color: #aaa; /* Po nastavení kurzora na položku zoznamu
sa zmení farba položky na sivú*/
}
/*Zobrazenie kruhu na pozadí*/
div{
width: 300px; /*Šírka kontajnera*/
height: 300px; /*výška kontajnera DIV*/
background-color: #bbb; /* Šedivá farba pozadia kontajneru*/
margin: 20px; /* vonkajší okraj, aby nebol kontajner úplne vľavo hore*/
border-radius: 50%; /* Všetky rohy zaguľatené pomocou percent */

```

Konečný výstup



18.Obtekanie prvkov(blokových elementov)

Blokové elementy sa vyznačujú tým, že:

- je možné nastaviť výšku a šírku (height a width),
- dva blokové elementy umiestnené v kóde za sebou sa na stránke objavia pod sebou.

Blokové elementy sú elementy za, ktorými sa zalamuje riadok.Príkladom môže byť odsek p, nadpis h1, oddiel div.Preto ak umiestnime vo vnútri v bloku div ďalší div, umiestní sa druhý div na nový riadok.

Elementy DIV sú blokové (ako napríklad odseky). Ich predvolené správanie je, že sa rozťahnu cez celú šírku elementu, v ktorom sú vložené (teda cez celý DIV) a poskladajú sa pod seba.

Používanie **inline-blocku** je tak spôsob, ako dostať elementy s nastavenými rozmermi vedľa seba bez použitia obtekanie (float).

Prvky s touto vlastnosťou:

- Sa zobrazujú vedľa seba (ako inline).
- Môžu mať nastavenú šírku a výšku.

Iným prípadom sú **inline**(vložené) elementy.Ak umiestníme za sebou viac inline elementov,tieto sa umiestňujú v riadku vedľa seba.Na ďalší riadok sa prejde až keď je zaplnený riadok.

Príklady inline prvkov:

-
- <a>
-

Vlastnosti prvku float

Float-Určuje, či by mal prvok plávať alebo nie

Určuje ako sa má obrázok alebo text správať vo vnútri iného prvku.Napr.obrázok vo vnútri odseku.

-môže byť :- left

-right

Clear-Určuje, po ktorých stranách plávajúceho elementu, nie je dovolené plávať

-Vlastnosť sa používa na riadenie správania sa plávajúcich prvkov.

-ruší všetky predchádzajúce obtekania prvkov

- môže byť: - left

- right

-both

Overflow-Určuje, čo sa stane, keď obsah pretečie box prvku

Float najjednoduchšie použitie .

Vlastnosť môže byť použitá pre obtekanie textu okolo obrázkov.

Pr.1. Obrázok bude plávať po pravej strane textu:

```
<!DOCTYPE html>
<!--vzťah paragrafu a obrázka-->
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Súbor pre cvičenie obtekania odstavca vpravo</title>
    <link href="images/logo.png" rel="shortcut icon" type="image/png">
    <link href="css/main.css" rel="stylesheet" type="text/css">
  </head>
  <body>
    <h2> V zápise kódu je za odsek umiestnený obrázok .V tomto príklade
bude obrázok plávať po pravej strane odseku a text do odseku sa zalomí
okolo obrázka.</h2>

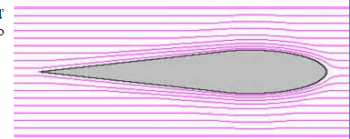
    <p>
Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín a typografie.
Lorem Ipsum je štandardným výplňovým textom už od 16. storočia, keď neznámy
tlačiar zobral sadzobnicu plnú tlačových znakov a pomiešal ich, aby tak
vytvoril vzorkovú knihu. Prežil nielen päť storočí, ale aj skok do
elektronickej sadzby, a pritom zostal v podstate nezmenený. Spopularizovaný
bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset, ktoré obsahovali
pasáže Lorem Ipsum, a neskôr aj publikačným softvérom ako Aldus PageMaker,
ktorý obsahoval verzie Lorem Ipsum.</p>
  </body>
</html>
```

main.css

```
img {  
    float: right;  
    margin: 0 0 10px 10px;  
}
```

V zápise kódu je za odsek umiestnený obrázok .V tomto príklade bude obrázok plávať po pravej strane odseku a text do odseku sa zalomí okolo obrázka.

Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín a typografie. Lorem Ipsum je štandardným výplňovým textom už od 16. storočia, keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a pomiešal ich, aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí, ale aj skok do elektronickej sadzby, a pritom zostal v podstate nezmenený. Spopularizovaný bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset, ktoré obsahovali pásáže Lorem Ipsum, a neskôr aj publikačným softvérom ako Aldus PageMaker, ktorý obsahoval verzie Lorem Ipsum.



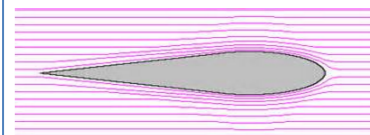
Ak zmeníme spôsob obtekania nasledovne:

main.css

```
img {  
    float: left;  
    margin: 0px 10px 0px 10px;  
}
```

Dostaneme :

V zápise kódu je za odsek umiestnený obrázok .V tomto príklade bude obrázok plávať po ľavej strane odseku a text do odseku sa zalomí okolo obrázka.



Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín a typografie. Lorem Ipsum je štandardným výplňovým textom už od 16. storočia, keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a pomiešal ich, aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí, ale aj skok do elektronickej sadzby, a pritom zostal v podstate nezmenený. Spopularizovaný bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset, ktoré obsahovali pásáže Lorem Ipsum, a neskôr aj publikačným softvérom ako Aldus PageMaker, ktorý obsahoval verzie Lorem Ipsum.

Pr.2 Obtekanie span prvku

```
<!DOCTYPE html>  
<html lang="sk">  
    <head>  
        <meta charset="UTF-8">  
        <title>Súbor pre cvičenie obtekania inline prvku vľavo</title>  
        <link href="images/logo.png" rel="shortcut icon" type="image/png">  
        <link href="css/main.css" rel="stylesheet" type="text/css">  
    </head>  
    <body>  
        <p>  
<span>A</span>
```

k ešte nepoznáte animované rozprávky Máša a medveď, je čas sa s nimi zoznámiť. Máša je neposedné ruské dievčatko, ktoré musí všade byť a o všetkom vedieť. Medveď má naopak rád pokoj a preto to má s Mášou niekedy skutočne ťažké. Pozrite si aj vy rozprávky Máša a medveď a sami sa presvedčte, prečo sú medzi deťmi, ale aj dospelými také populárne.

</p>

<p>

V predchádzajúcom odseku, je prvé písmeno textu v rozpätí span prvku.Span element má šírku, ktorá je 0,7 krát väčšia ako aktuálny font.Font-size span prvku je 400% (pomerne veľká) a výška riadka je 80%.Písmo span prvku bude v "Algerian".

</p>

</body>

</html>

main.css

```
span {  
    float: left;  
    width: 0.7em;  
    font-size: 400%;  
    font-family: algerian, courier;  
    line-height: 80%;  
}
```

Ak ešte nepoznáte animované rozprávky Máša a medveď, je čas sa s nimi zoznámiť. Máša je neposedné ruské dievčatko, ktoré musí všade byť a o všetkom vedieť. Medveď má naopak rád pokoj a preto má s Mášou niekedy skutočne ťažké. Pozrite si aj vy rozprávky Máša a medveď a sami sa presvedčte, prečo sú medzi deťmi, ale aj dospelými také populárne

V predchádzajúcom odseku, je prvé písmeno textu v rozpätí span prvku.Span element má šírku, ktorá je 0,7 krát väčšia ako aktuálny font.Font-size span prvku je 400% (pomerne veľká) a výška riadka je 80%.Písmo span prvku bude v "Algerian".

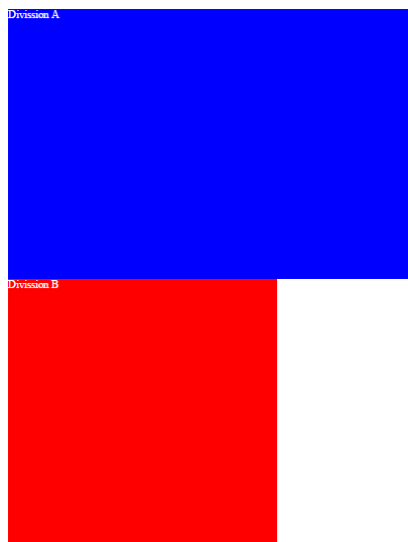
Index.html

```
<!DOCTYPE html>  
<html lang='sk'>  
    <head>  
        <title>Toto je príklad na obtekanie elementu div</title>  
        <meta charset='utf-8'>  
            <link href='images/logo.png' rel='shortcut icon'  
type='image/png'>  
            <link href='css/style.css' rel='stylesheet'  
type='text/css'>  
        </head>  
<body>  
    <div id = "logo">Division A</div>  
    <div id = "menu">Division B</div>  
</body>  
</html>
```

style.css

```
#logo{
  width:600px;
  height:400px;
  color:white;
  background:blue;
  /*float:left;*/
}
#menu{
  width:400px;
  height:400px;
  color:white;
  background:red;
  /*float:left;*/
}
```

chrome



DIV logo a DIV menu budeme chcieť vedľa seba, nie pod sebou. Docielime toho tým, že ich označíme ako plávajúce. Plávajúce elementy sa radí vedľa seba za predpokladu, že im nastavíme rozmery.

Plávajúce elementy sa radia vedľa seba a sú neplávajúcim obsahom obtekané. Ak máme okolo plávajúcich elementov nejaký neplávajúci obsah, napríklad text okolo plávajúceho obrázku, bude text obrázok obtekať.

Ak vedľa seba skladáme plávajúce elementy, musíme za posledným elementom v rade ukončiť plávajúce obsah. To sa robí CSS vlastnosťou clear (ako vyčistiť), do CSS si vložíme nový triedny selektor:

```
.cistic {
  clear: both;
}
```

A do html doplníme:

```
<div class="cistic"></div>
```

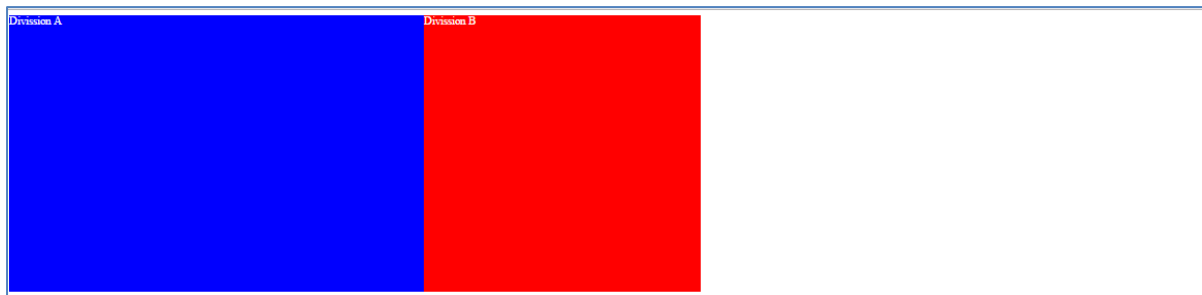
PR.3.Vyriešenie radenia blokov div za sebou

```
<!DOCTYPE html>
<html lang='sk'>
  <head>
    <title>Toto je príklad na obtekanie elementu div</title>
    <meta charset='utf-8'>
      <link href='images/logo.png' rel='shortcut icon'
type='image/png'>
      <link href='css/style.css' rel='stylesheet'
type='text/css'>
    </head>
  <body>
    <div id = "logo">Divission A</div>
    <div id = "menu">Divission B</div>
    <div class="cistic"></div>
  </body>
</html>
```

style.css

```
#logo{
  width:600px;
  height:400px;
  color:white;
  background:blue;
  float:left;
}
#menu{
  width:400px;
  height:400px;
  color:white;
  background:red;
  float:left;
}
.cistic{
  clear:both;
}
```

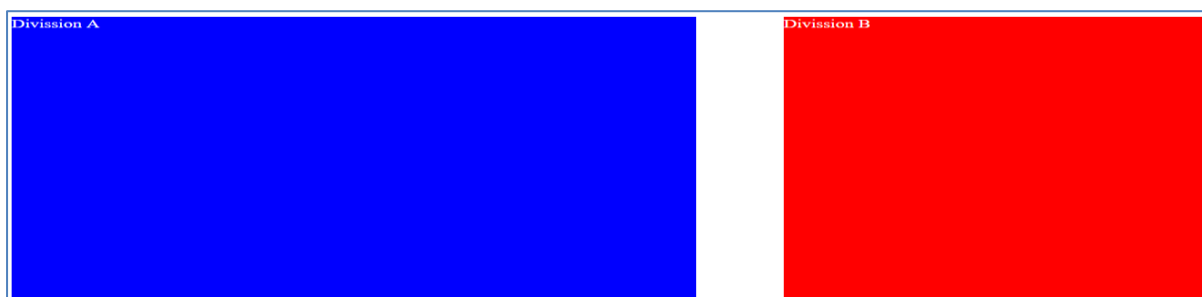
Chrome:



Ak div #menu bude mať obtekanie :

```
float:right;
```

bude usporiadanie nasledovné:



Příklad radenia obrázkov

```
<!DOCTYPE html>
<html lang='sk'>
  <head>
    <title>Toto je príklad na obtekanie elementu img</title>
    <meta charset='utf-8'>
      <link href='images/logo.png' rel='shortcut icon'
type='image/png'>
      <link href='css/style.css' rel='stylesheet'
type='text/css'>
    </head>
  <body>

  <h1 align=center> Modely LAMBORGHINI</h1>
    <div id = "auto1">
      
    </div>
    <div id = "auto2">
      
    </div>
    <div class="cistic">
      <span>A</span>utomobili Lamborghini S.p.A (akciová spoločnosť).,
známa predovšetkým ako Lamborghini, je výrobca traktorov[3], motorov do
pretekárskych člnov[4], najviac ale preslávená ako výrobca vysokovýkonných
a superšportových automobilov. Sídli v talianskom mestečku Sant'Agata
```

Bolognese. Spoločnosť založil v roku 1963 Ferruccio Lamborghini (1916-1993).

Pre Lamborghini navrhujú alebo navrhovali autá dizajnéri ako Franco Scaglione, Touring of Milan, Zagato, Mario Marazzi, Bertone, ItalDesign, Marcello Gandini a Luc Donckerwolke.

<p>

Automobili Lamborghini S.p.A (akciová spoločnosť) ., známa predovšetkým ako Lamborghini, je výrobca traktorov[3], motorov do pretekárskych člnov[4], najviac ale preslávená ako výrobca vysokovýkonných a superšportových automobilov. Sídlí v talianskom mestečku Sant'Agata Bolognese. Spoločnosť založil v roku 1963 Ferruccio Lamborghini (1916-1993).

Pre Lamborghini navrhujú alebo navrhovali autá dizajnéri ako Franco Scaglione, Touring of Milan, Zagato, Mario Marazzi, Bertone, ItalDesign, Marcello Gandini a Luc Donckerwolke.</p>

</div>

</body>

</html>

Style.css

```
#auto1{
  float:left;
  margin:10px 10px 10px 10px;
}

#auto2{
  float: right;
  margin:10px 10px 10px 10px;
}

.cistic{
  /*clear: both;*/
  line-height: 2;
}

span {
  float: left;
  width: 0.7em;
  font-size: 400%;
  font-family: algerian, courier;
  line-height: 80%;
}
```


Modely LAMBORGHINI



Automobilí Lamborghini S.p.A (akciová spoločnosť), známa predovšetkým ako Lamborghini, je výrobca traktorov[3], motorov do pretekárskych člnov[4], najviac ale preslávená ako výrobca vysokovýkonných a superšportových automobilov. Sídlí v talianskom mestečku San'Agata Bolognese. Spoločnosť založil v roku 1963 Ferruccio Lamborghini (1916–1993). Pre Lamborghini navrhujú alebo navrhovali autá dizajnéri ako Franco Scaglione, Touring of Milan, Zagato, Mario Marazzi, Bertone, ItalDesign, Marcello Gandini a Luc Donckerwolke.



Automobilí Lamborghini S.p.A (akciová spoločnosť), známa predovšetkým ako Lamborghini, je výrobca traktorov[3], motorov do pretekárskych člnov[4], najviac ale preslávená ako výrobca vysokovýkonných a superšportových automobilov. Sídlí v talianskom mestečku San'Agata Bolognese. Spoločnosť založil v roku 1963 Ferruccio Lamborghini (1916–1993). Pre Lamborghini navrhujú alebo navrhovali autá dizajnéri ako Franco Scaglione, Touring of Milan, Zagato, Mario Marazzi, Bertone, ItalDesign, Marcello Gandini a Luc Donckerwolke.

Ako sa správa img keď ho použijeme opakované?

```
<!DOCTYPE html>
<html lang='sk'>
  <head>
    <title>Toto je príklad na radenie elementu img</title>
    <meta charset='utf-8'>
      <link href='images/logo.png' rel='shortcut icon'
type='image/png'>
    </head>
  <body>

  <h1 align=center>Hrad Lednica</h1>
    <div>
      
      
      
      
    </div>
  </body>
</html>
```

Hrad Lednica



Pri opakovanom použití elementu img sa obrázky radia vedľa seba. Prechod na ďalší riadok bude po dosiahnutí konca riadku.

Hrad Lednica



Pr.boxy bez plávania

index.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
<title>Príklad na nastavenie parametrov boxov</title>
    <link rel="stylesheet" type="text/css" href="css/main.css"/>
  </head>
<body>
<div id="blok1">Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín
a typografie. Lorem Ipsum je štandardným výplňovým textom už od 16.
storočia, keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a
pomiešal ich, aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí,
ale aj skok do elektronickej sadzby, a pritom zostal v podstate nezmenený.
Spopularizovaný bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset,
ktoré obsahovali pasáže Lorem Ipsum, a neskôr aj publikačným softvérom ako
Aldus PageMaker, ktorý obsahoval verzie Lorem Ipsum.</div>
<div id="blok2">Napriek všeobecnému presvedčeniu nie je Lorem Ipsum len
náhodný text. Jeho korene sú v časti klasickej latinskej literatúry z roku
45 pred n.l., takže má viac ako 2000 rokov. Richard McClintock, profesor
latinčiny na Hampden-Sydney College vo Virgínii, hľadal jedno z menej
určitých latinských slov, consectetur, z pasáže Lorem Ipsum, a ako
vyhľadával výskyt tohto slova v klasickej literatúre, objavil jeho
nepochybný zdroj. Lorem Ipsum pochádza z odsekov 1.10.32 a 1.10.33
Cicerovho diela "De finibus bonorum et malorum" (O najvyššom dobre a zle),
napísaného v roku 45 pred n.l. Táto kniha je pojednaním o teórii etiky, a
bola veľmi populárna v renesancii. Prvý riadok Lorem Ipsum, "Lorem ipsum
dolor sit amet..", je z riadku v odseku 1.10.32.</div>
<div id="blok3">Existuje mnoho podôb pasáží Lorem Ipsum, ale väčšina trpela
rôznymi zmenami, vložením humoru, alebo náhodných slov, ktoré nevyzerajú
ani trocha dôveryhodne. Ak sa chystáte použiť pasáž z Lorem Ipsum, mali by
ste sa presvedčiť, že uprostred textu nie je skrytá žiadna časť, ktorá by
vás mohla priviesť do nepríjemnej situácie. Všetky generátory Lorem Ipsum
na internete opakujú vopred definované časti textu, takže náš generátor je
prvým skutočným generátorom na internete. Používa slovník viac ako 200
latinských slov, a kombinuje ich niekoľkými modelovými vetnými štruktúrami,
takže generuje Lorem Ipsum, ktoré vyzerá hodnoverne. Vygenerované Lorem
```

```
Ipsum je týmto spôsobom vždy bez opakujúcich sa častí, bez vtipov a
nenáležitých výrazov, atď.</div>
</body>
</html>
```

Main.css

```
/* CSS subor s menom main.css*/
body {
    font-size: 1.4em;
}
/* Nastavovanie vlastností pre bloky*/
#blok1{
    background-color: gold;
    width:960px;/* Nastavenie šírky kontajneru*/
    padding: 20px;/*Nastavenie vnútorných okrajov*/
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri
rohy*/
    margin:10px;/* Nastavenie vonkajších okrajov*/
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu "DIV"*/
}
#blok2{
    background-color: lightblue;
    width:960px;/* Nastavenie šírky kontajneru*/
    padding: 20px;/*Nastavenie vnútorných okrajov*/
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri
rohy*/
    margin:10px;/* Nastavenie vonkajších okrajov*/
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu "DIV"*/
}
#blok3{
    background-color: lightgreen;
    width:960px;/* Nastavenie šírky kontajneru*/
    padding: 20px;/*Nastavenie vnútorných okrajov*/
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri
rohy*/
    margin:10px;/* Nastavenie vonkajších okrajov*/
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu "DIV"*/
}
```

CHROME:

Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín a typografie. Lorem Ipsum je štandardným výplňovým textom už od 16. storočia, keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a pomiešal ich, aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí, ale aj skok do elektronickej sadzby, a pritom zostal v podstate nezmenený. Spopularizovaný bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset, ktoré obsahovali pasáže Lorem Ipsum, a neskôr aj publikačným softvérom ako Aldus PageMaker, ktorý obsahoval verzie Lorem Ipsum.

Napriek všeobecnému presvedčeniu nie je Lorem Ipsum len náhodný text. Jeho korene sú v časti klasickej latinskej literatúry z roku 45 pred n.l., takže má viac ako 2000 rokov. Richard McClintock, profesor latinčiny na Hampden-Sydney College vo Virginii, hľadal jedno z menej určitých latinských slov, consectetur, z pasáže Lorem Ipsum, a ako vyhľadával výskyt tohto slova v klasickej literatúre, objavil jeho nepochybný zdroj. Lorem Ipsum pochádza z odsekov 1.10.32 a 1.10.33 Cicerovho diela "De finibus bonorum et malorum" (O najvyššom dobre a zle), napísaného v roku 45 pred n.l. Táto kniha je pojednaním o teórii etiky, a bola veľmi populárna v renesancii. Prvý riadok Lorem Ipsum, "Lorem ipsum dolor sit amet..", je z riadku v odseku 1.10.32.

Existuje mnoho podob pasáží Lorem Ipsum, ale väčšina trpela rôznymi zmenami, vložením humoru, alebo náhodných slov, ktoré nevyzerajú ani trocha dôveryhodne. Ak sa chystáte použiť pasáž z Lorem Ipsum, mali by ste sa presvedčiť, že uprostred textu nie je skrytá žiadna časť, ktorá by vás mohla priviesť do nepríjemnej situácie. Všetky generátory Lorem Ipsum na internete opakujú vopred definované časti textu, takže náš generátor je prvým skutočným generátorom na internete. Používa slovník viac ako 200 latinských slov, a kombinuje ich niekoľkými modelovými vetnými štruktúrami, takže generuje Lorem Ipsum, ktoré vyzerá hodnoverne. Vygenerované Lorem Ipsum je týmto spôsobom vždy bez opakujúcich sa častí, bez vtipov a nenáležitých výrazov, atď.

Pr.boxy s plávaním

Index1.html

```
<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Príklad na nastavenie parametrov boxov</title>
    <link rel="stylesheet" type="text/css" href="css/style.css"/>
  </head>
  <body>
    <div id="blok1">Lorem Ipsum je fiktívny text, používaný pri návrhu tlačovín
    a typografie. Lorem Ipsum je štandardným výplňovým textom už od 16.
    storočia, keď neznámy tlačiar zobral sadzobnicu plnú tlačových znakov a
    pomiešal ich, aby tak vytvoril vzorkovú knihu. Prežil nielen päť storočí,
    ale aj skok do elektronickej sadzby, a pritom zostal v podstate nezmenený.
    Spopularizovaný bol v 60-tych rokoch 20.storočia, vydaním hárkov Letraset,
    ktoré obsahovali pasáže Lorem Ipsum, a neskôr aj publikačným softvérom ako
    Aldus PageMaker, ktorý obsahoval verzie Lorem Ipsum.</div>
    <div id="blok2">Napriek všeobecnému presvedčeniu nie je Lorem Ipsum len
    náhodný text. Jeho korene sú v časti klasickej latinskej literatúry z roku
    45 pred n.l., takže má viac ako 2000 rokov. Richard McClintock, profesor
    latinčiny na Hampden-Sydney College vo Virginii, hľadal jedno z menej
    určitých latinských slov, consectetur, z pasáže Lorem Ipsum, a ako
    vyhľadával výskyt tohto slova v klasickej literatúre, objavil jeho
    nepochybný zdroj. Lorem Ipsum pochádza z odsekov 1.10.32 a 1.10.33
    Cicerovho diela "De finibus bonorum et malorum" (O najvyššom dobre a zle),
    napísaného v roku 45 pred n.l. Táto kniha je pojednaním o teórii etiky, a
    bola veľmi populárna v renesancii. Prvý riadok Lorem Ipsum, "Lorem ipsum
    dolor sit amet..", je z riadku v odseku 1.10.32.</div>
```

```
<div id="blok3">Existuje mnoho podôb pasáží Lorem Ipsum, ale väčšina trpela rôznymi zmenami, vložením humoru, alebo náhodných slov, ktoré nevyzerajú ani trocha dôveryhodne. Ak sa chystáte použiť pasáž z Lorem Ipsum, mali by ste sa presvedčiť, že uprostred textu nie je skrytá žiadna časť, ktorá by vás mohla priviesť do nepríjemnej situácie. Všetky generátory Lorem Ipsum na internete opakujú vopred definované časti textu, takže náš generátor je prvým skutočným generátorom na internete. Používa slovník viac ako 200 latinských slov, a kombinuje ich niekoľkými modelovými vetnými štruktúrami, takže generuje Lorem Ipsum, ktoré vyzerá hodnoverne. Vygenerované Lorem Ipsum je týmto spôsobom vždy bez opakujúcich sa častí, bez vtipov a nenáležitých výrazov, atď.</div>
```

```
</body>
```

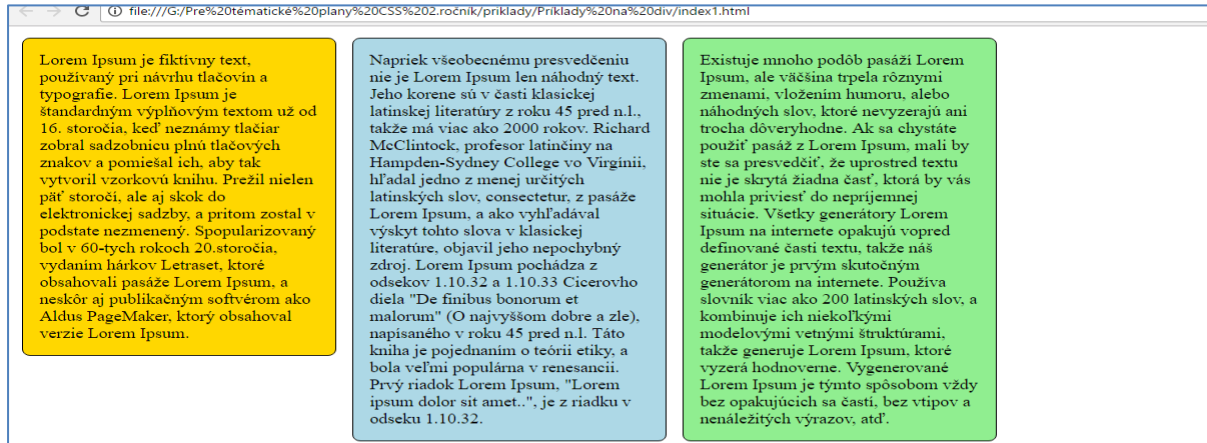
```
</html>
```

style.css

```
/* CSS subor s menom style.css*/  
body {  
    font-size: 1.4em;  
}  
/* Nastavovanie vlastností pre bloky*/  
#blok1{  
    background-color: gold;  
    width:350px;/* Nastavenie šírky kontajneru*/  
    padding: 20px;/*Nastavenie vnútorných okrajov*/  
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri  
rohy*/  
    margin:10px;/* Nastavenie vonkajších okrajov*/  
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu "DIV"*/  
    float: left;  
}  
#blok2{  
    background-color: lightblue;  
    width:350px;/* Nastavenie šírky kontajneru*/  
    padding: 20px;/*Nastavenie vnútorných okrajov*/  
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri  
rohy*/  
    margin:10px;/* Nastavenie vonkajších okrajov*/  
    border:1px solid #000;/* Nastavenie rámčeku okolo elementu "DIV"*/  
    float:left;  
}  
#blok3{  
    background-color: lightgreen;  
    width:350px;/* Nastavenie šírky kontajneru*/  
    padding: 20px;/*Nastavenie vnútorných okrajov*/  
    border-radius: 10px;/*Nastavenie rádiusu rovnako pre všetky štyri  
rohy*/
```

```
margin:10px;/* Nastavenie vonkajších okrajov*/
border:1px solid #000;/* Nastavenie rámčeka okolo elementu "DIV"*/
float:left;
}
```

Chrome:



LAYOUT-rozvrhnutie stránky

Webová stránka obsahuje mnoho html prvkov, ako napríklad nadpisy, podnadpisy, odseky, obrázky, hypertextové odkazy na ďalšie stránky a pod. Tieto prvky sú zoskupené a umiestnené v takzvaných blokoch, ktoré si môžeme predstaviť ako obdĺžniky vymedzujúce určitý priestor. Ich veľkosť, prípadne farby, pozadia, rámčeky a podobne (bloky môžu byť ale aj priehľadné) sú nastavené pomocou css štýlov.

Jednotlivé bloky môžu byť pod sebou, vedľa seba i vnorené v iných blokoch.

Rozmiestnenie týchto blokov vytvára rozvrhnutie webovej stránky, jej návrh, grafický dizajn.

Trendy moderného webu

- jednoduchosť
- rezpozivita

Pre vytvorenie zložitejšieho rozvrhnutie webovej stránky sú potrebné znalosti radu rôznych vlastností css štýlov.

Pre začiatok si vytvoríme úplne jednoduchý návrh zložený zo 4 blokov (hlavička, navigačné menu, článok a pätička) umiestnených v piatom bloku (obaľujúca, zapouzdrujúci blok).

Tento návrh je jednoduchý (k jeho vytvoreniu je potreba minimum vlastnosťou css štýlov), využijeme ho však aj neskôr, pri vytváraní responzívneho webu (web prispôsobujúce sa veľkosti zariadenia) ako rozvrhnutie pre mobily.

Layout stránky-teória

Obsah webovej strany

- **hlavné problémy**
 - polohovanie textov, popisov, navigácia, grafiky a obrázkov
 - zachovanie polohy pri zmene šírky okna
 - jednotné rozloženie a vzhľad všetkých stránok webu
 - prispôsobenie obsahu veľkosti okna a typu zariadenia
- Riešenie pomocou šablóny webu - layoutu

Layout - šablóna rozloženia

- spôsob polohovanie oblastí stránky s rôznym typom obsahu
 - záhlavia, päty, riadkové menu, stĺpcové menu, obsah ...

Typy layoutu

1.Podľa počtu stĺpcov

• **jedno stĺpcový, dvojtĺpcový, trojtĺpcový**

2.Podľa nastavenia šírky stránky

- **fixné layout** - pevná šírka stránky s ohľadom na rozlíšenie monitora - cca 960 - 980 px
- **fluidné layout** - šírka je premenná podľa aktuálnej šírky okna - obvykle 90 - 95%
- **responzívne layout** - rozvrhnutie sa prispôsobuje šírke okna a typu zobrazovacieho zariadenia - monitor, tablet, mobil ...

Technológia layoutu

1.tabuľkový layout

- využitie html tabuliek pre rozloženie obsahu
- ukážka výberu tabuľky rozvrhnutia
 - **1** - Hlavička, Ľavý stĺpec, Pravý horný a Základný text
 - **2** - Hlavička, Ľavý stĺpec, Základný text a Päta

Bez rozložení			
1		2	

- **výhody**
 - prehľadný a jednoduchý kód
 - využitie tabuliek rozloženie v SPD
- **nevýhody**
 - pomalé načítanie - načítavanie celá tabuľka naraz
 - obmedzené možnosti formátovania buniek tabuľky

PR5.Tabuľkové rozvrhnutie stránky

Pr.5.1

```

<!DOCTYPE html>
<html lang="sk">
  <head>
    <meta charset="UTF-8">
    <title>Návrh stránky pomocou tabuľky</title>
  </head>
  <body>
    <section>
      <table border="3" width="600" height="400" cellpadding="5"
cellspacing="5">
        <tr height="70" align="center" bgcolor="yellow">
          <td colspan="3">Hlavička</td>
        </tr>

        <tr valign="top">
          <td width="100" bgcolor="red">Ľavý blok</td>
          <td bgcolor="lightblue">Hlavná časť stránky</td>
          <td width="100" bgcolor="red">Pravý blok</td>
        </tr>

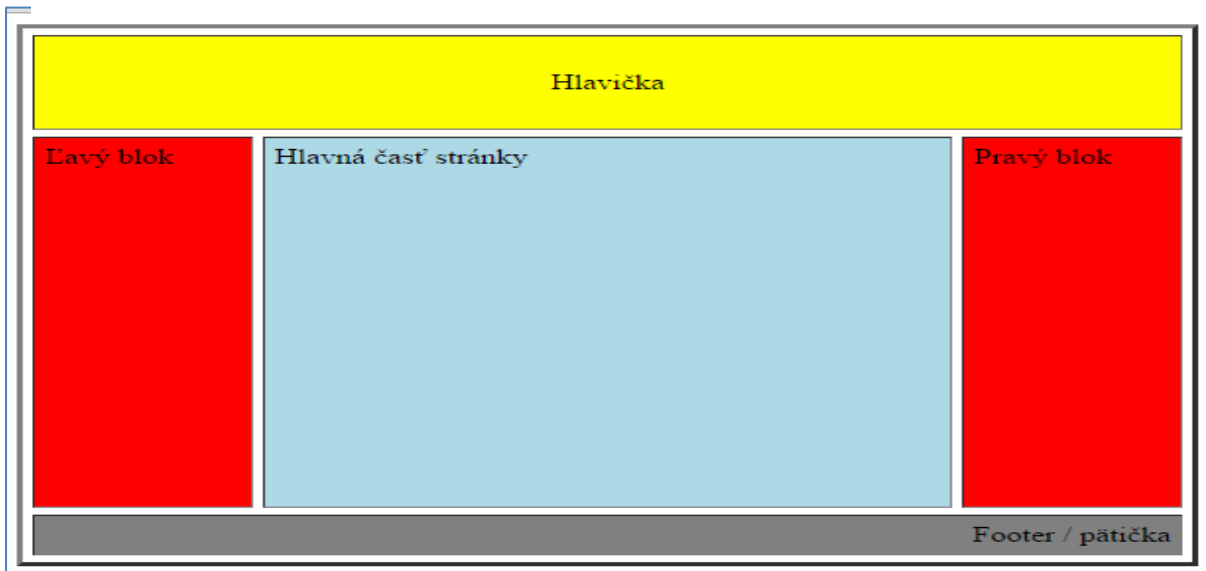
        <tr height="30" align="right" bgcolor="gray">
          <td colspan="3">Footer / pätička</td>
        </tr>
      </table>
    </section>

```



```
</body>
</html>
```

chrome:



2.rámcový layout

- zskupenie obsahu niekoľkých html stránok pomocou špeciálnej rámcovej stránky
- rámcová strana definuje rozloženie obsahu stránok do rámcových oblastí
- z jedného rámcu možné volať stránky do iného rámcu

- **výhody**
 - jednoduché a efektívne riešenie rozloženie obsahu
 - spoločné menu poro všetky strany
- **nevýhody**
 - pomalé načítanie
 - nevhodné pre vyhľadávače

index.html

- **<HTML>**
- **<HEAD>**
- **<TITLE>** Príklad použitia rámcov **</TITLE>**
- **</HEAD>**
- **<FRAMESET cols="200,*">***<!--200 PÍXELOV LAVY ramec zvysoK pravy ramec-->*
- **<FRAME SRC="lavy.html" NAME="Lavy">**
- **<FRAME SRC="pravy.html" NAME="Pravy">**
- **</FRAMESET>**
- **</HTML>**

Lavy.html

- `<HTML>`
- `<HEAD>`
- `<TITLE> Ponuka </TITLE>`
- `</HEAD>`
- `<body bgcolor="#008080" text="#000000" link="#FF8000" vlink="#FF8000" alink="#FF0000">`
- `Menu<p>`
- `<p>`
- `<p>`
- `<p>`
- `<p>`
- `<p>`
- `<!--target preto aby sa stranka otvarala v pravom ramci-->`
- `

`
- `<p>`
- `<i>by Drgo Pavel</i>`
- `<i>18.marec 2013</i>
`
- `</body>`
- `</HTML>`



3. Blokový layout riadený kaskádovými štýlmi

- vyžitie html tagov `<div>` alebo `` pre členenie obsahu do blokov
- vyžitie štýlových pravidiel **CSS** pre nastavenie veľkosti, polohy a vzhľadu oblastí

- **výhody**
 - rýchle načítanie - úsporný kód, riadenie poradie načítaných prvkov
 - možnosti formátovania - väčšie možnosti nastavenia vzhľadu
 - úprava vzhľadu - centrálny šablóna štýlov využívaná viac stránkami
- **nevýhody**
 - nutnosť znalosti pravidiel CSS
 - rozdiely medzi prehliadačmi v interpretácii CSS

základné bloky

Pre jednoduché rozloženie stránky použijeme bloky:

- `div` - všeobecný blok - obalujúci ďalšie bloky
- `header` - hlavička - názov webu, logo, obrázok
- `nav` - navigačné menu - odkazy na ďalšie stránky webu
- `article` - článok - vlastný obsah stránky
- `footer` - pätička - copyright, rýchle odkazy, kontakty a pod.

Do hlavičky *header* zapíšeme názov webu.

Do bloku *nav* neskôr vložíme odkazy na ostatné stránky webu a vytvoríme tak navigačné menu.

Do bloku *article* podľa ďalších kapitol vložíme článok - vlastný obsah webu - nadpisy, odseky, obrázky, tabuľky atď.

Do pätičky *footer* môžeme zapísať napr. Kontaktné údaje či meno autora alebo copyright a pod.

umiestnenie menu

Menu odkazov na stránky webu (blok `nav`) môže byť umiestnené pod hlavičkou webu, nad hlavičkou alebo vnútri hlavičky.

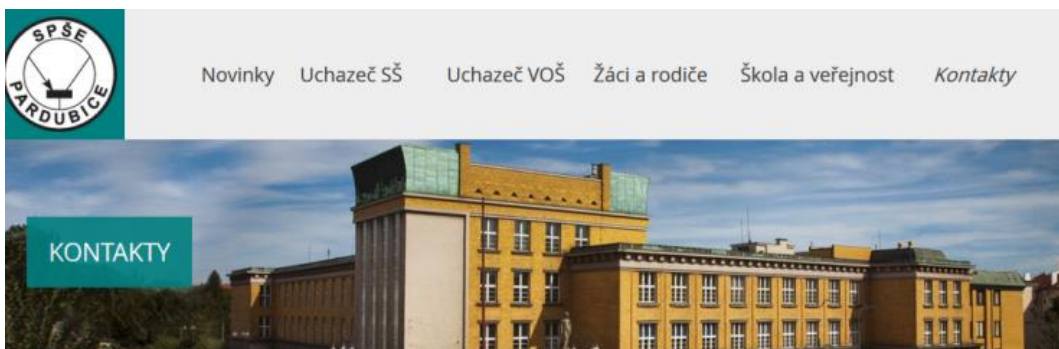
Ukážky rôznych umiestnenie menu

1) Menu pod hlavičkou:



```
<div>
  <header></header>
  <nav>
    <a href="index.html">Svět</a>
    <a href="postavy.html">Postavy</a>
    <a href="kouzla.html">Kouzla</a>
  </nav>
  <article>
    <h1>Melio Unar</h1>
    <p>Melio Unar je svět ...</p>
  </article>
</div>
```

2. Menu úplně hore nad hlavičkou - může tu být pevně ukotvené a být dostupné aj při rolování:



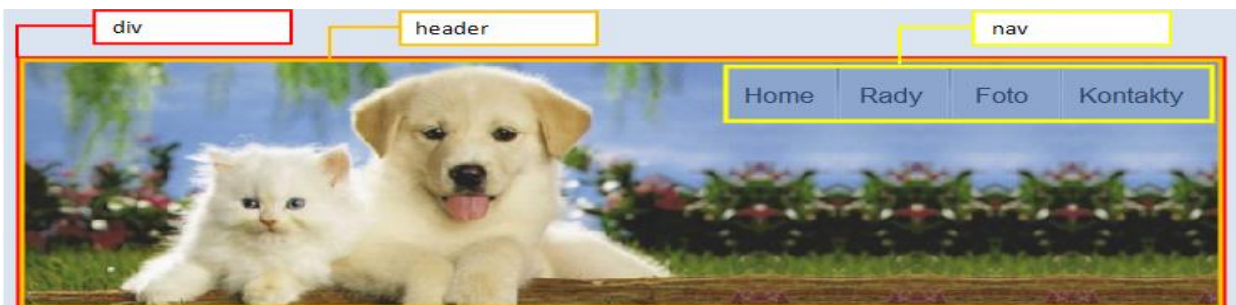
```
<div>
```

```

<nav>
  <a href="index.html">Novinky</a>
  <a href="uchazec-ss.html">Uchazeč SŠ</a>
  <a href="uchazec-vos.html">Uchazeč VOŠ</a>
  <a href="zaci-rodice.html">Žáci a rodiče</a>
  <a href="skola-verejnost.html">Škola a veřejnost</a>
  <a href="kontakty.html">Kontakty</a>
</nav>
<header></header>
<article>
  <!-- hlavní obsah stránky -->
</article>
</div>

```

Menu vnútri hlavičky:



```

<div>
  <header>
    <nav>
      <a href="index.html">Home</a>
      <a href="rady.html">Rady</a>
      <a href="foto.html">Foto</a>
      <a href="kontakty.html">Kontakty</a>
    </nav>
  </header>
  <article>
    <!-- hlavní obsah stránky -->
  </article>
  <footer></footer>
</div>

```

2.HTML5 Sémantické prvky

Sémantika je štúdium významov slov a fráz v jazyku.

Sémantické prvky sú prvky s významom.

Sémantický prvok – element, jasne opisuje svoj význam ako pre prehliadače tak aj vývojárov.

Príklad na **non-sémantické** prvky: <div> a - nič nevytvorí o jeho obsahu.

Príklady **sémantických-významových** prvkov: <form>, <table> a <Article><header ><footer>- Jasne definujú jeho obsah.

HTML5 sémantické prvky sú podporované vo všetkých moderných prehliadačoch.

New sémantické prvky v HTML5

Poznámka:

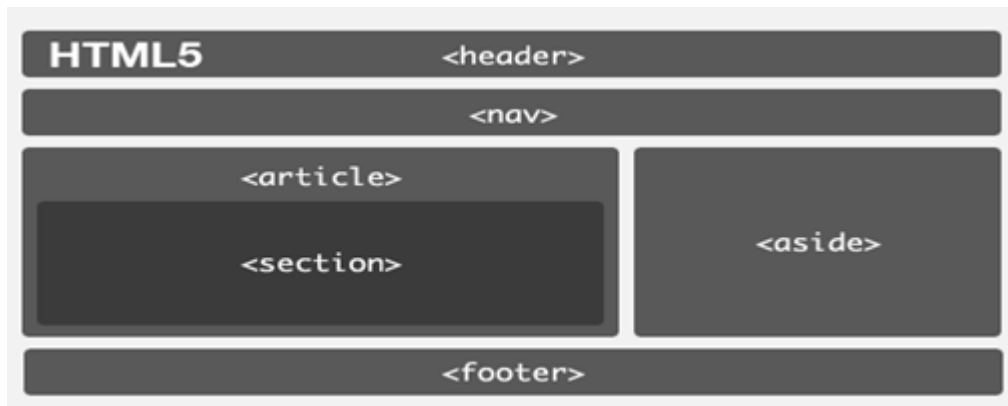
Mnohé webové stránky obsahujú kód HTML ako: <div id = "nav"> <div class = "header"> <div id = "päta"> na označenie navigácie , hlavičky a päty.



HTML5 ponúka nové sémantické prvky, ktoré definujú rôzne časti webovej stránky:

- <Article>
- <aside>
- <Details>
- <Figcaption>
- <figure>
- <Footer>
- <Header>
- <main>
- <Mark>
- <Nav>
- <Section>
- <Summary>
- <Time>

Príklad použitia sémantických prvkov na rozloženie obsahu stránky:



Popis sémantických elementov

HTML5 prvok <section> oddiel

Sekcia je tématické zoskupenie obsahu, typicky s nadpisom. Domovská stránka by za normálnych okolností bude rozdelená do sekcií pre zavedenie, obsah a kontaktné informácie.

Nosná časť kódu

```
<section>
  <h1>WWF</h1>
  <p>Svetový fond na ochranu prírody (WWF) je.....</p>
</section>
```

Celý príklad-section.html

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok section</title>

  <!--<link rel="stylesheet" href="css/style.css" type="text/css"-->
</head>
<body>
<section>
  <h1>WWF</h1>
  <p>Svetový fond na ochranu prírody (WWF) je medzinárodná organizácia pracujúca na otázkach týkajúcich sa ochrany, výskum a obnovu životného prostredia, predtým s názvom World Wildlife Fund. WWF bola založená v roku 1961.</p>
</section>
```

```

<section>
  <h1>Panda symbol WWF organizácie</h1>
  <p>Panda sa stala symbolom WWF organizácie. Známe WWF logo pandy
pochádza od pandy s názvom Chi Chi, ktorá bola prevedená zo Zoo v Pekingu
do Londýnskej Zoo v tom istom roku ako bola organizácia WWF založená</p>
</section>

</body>
</html>

```

Chrome:

WWF

Svetový fond na ochranu prírody (WWF) je medzinárodná organizácia pracujúca na otázkach týkajúcich sa ochrany, výskum a obnovu životného prostredia, predtým s názvom World Wildlife Fund. WWF bola založená v roku 1961.

Panda symbol WWF organizácie

Panda sa stala symbolom WWF organizácie. Známe WWF logo pandy pochádza od pandy s názvom Chi Chi, ktorá bola prevedená zo Zoo v Pekingu do Londýnskej Zoo v tom istom roku ako bola organizácia WWF založená

HTML5 prvok <article> článok

Prvok <article>určuje nezávislý, sebestačný obsah.

Článok by mal mať zmysel sam o sebe, a malo by byť možné ho čítať nezávisle od zvyšku webových stránok.

Ako príklad, kde môže byť použitý <article> element:

- príspevok:
- príspevok na blogu
- Novinový článok

Nosná časť článku

```

<article>
  <h1>Čo Svetový fond na ochranu prírody (WWF) robí</h1>
  <p>Poslaním WWF je zastaviť degradáciu prírodného prostredia našej
planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s
prírodou.</p>
</article>

```

Celý príklad – **article.html**

```
<!DOCTYPE html>
```



```

<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok article</title>

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

</head>
<body>
<article>
  <h1>Čo Svetový fond na ochranu prírody (WWF) robí</h1>
  <p>Poslaním WWF je zastaviť degradáciu prírodného prostredia našej
planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s
prírodou.</p>
</article>

</body>
</html>

```

Chrome:

Čo Svetový fond na ochranu prírody (WWF) robí ?

Poslaním WWF je zastaviť degradáciu prírodného prostredia našej planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s prírodou.

HTML5 prvok <header>

prvok <header> určuje hlavičku dokumentu alebo oddielu.

Prvok by mal byť použitý ako obal(kontajner) pre úvodný obsah.

Môžete mať niekoľko prvkov <záhlavie> v jednom dokumente.

Nasledujúci príklad definuje hlavičku <article>článku:

Nosná časť definovania hlavičky článku

```

<article>
  <header>
    <h1>Čo Svetový fond na ochranu prírody (WWF) robí ?</h1>
    <p>Poslanie WWF:</p>
  </header>
  <p>Poslaním WWF je zastaviť degradáciu prírodného prostredia našej
planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s
prírodou.</p>
</article>

```

Celý príklad **header.html**

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok header</title>

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

</head>
<body>
<article>
  <header>
    <h1>Čo Svetový fond na ochranu prírody (WWF) robí ?</h1>
    <p>Poslanie WWF:</p>
  </header>
  <p>Poslaním WWF je zastaviť degradáciu prírodného prostredia našej
planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s
prírodou.</p>
</article>
</body>
</html>
```

Chrome:

Čo Svetový fond na ochranu prírody (WWF) robí ?

Poslanie WWF:

Poslaním WWF je zastaviť degradáciu prírodného prostredia našej planéty a vytvárať budúcnosť, v ktorej ľudia žijú v harmónii s prírodou.

HTML5 prvok <footer>

Prvok `<footer>` je zápätie dokumentu alebo oddielu.(section)

Mal by obsahovať informácie o svojom obsahu.

Päta zvyčajne obsahuje autora, informácie o dokumente, autorských práv, spojenie s podmienkami používania, kontaktné informácie atď

Môžete mať niekoľko `<footer>` prvkov v jednom dokumente.

príklad

```
<footer>
  <p>Posted by: Hege Refsnes</p>
  <p>Contact information: <a href="mailto:someone@example.com">
    someone@example.com</a>.</p>
</footer>
```

HTML5 prvok <nav>

Prvok <nav> definuje sadu navigačných odkazov.

Všimnite si, že nie všetky väzby(linky) dokumentu by mal byť vo vnútri prvku <nav> . Prvok je určený len pre hlavný blok navigačných odkazov.

Nosná časť príkladu

```
<nav>
  <a href="#html">HTML</a> |
  <a href="#css">CSS</a> |
  <a href="#js">JavaScript</a> |
  <a href="#jquery">jQuery</a>
</nav>
```

Celý príklad `nav.html`

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

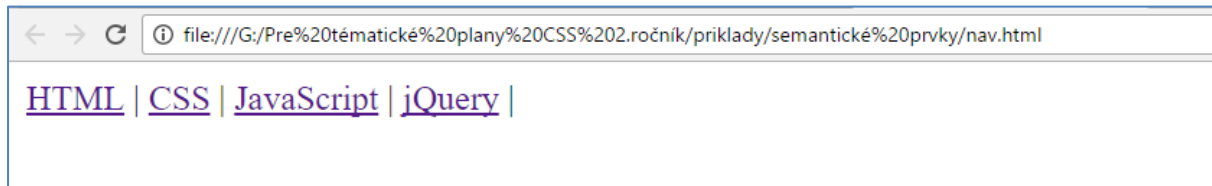
  <title>Semantický prvok nav</title>

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

</head>
<body>
<nav>
  <a href="#html">HTML</a> |
  <a href="#css">CSS</a> |
  <a href="#js">JavaScript</a> |
  <a href="#jquery">jQuery</a> |
</nav>
</body>
```

```
</html>
```

Chrome:



HTML5 prvok <aside-stranou>

Prvok definuje obsah je umiestnený v bočnom paneli.

Obsah by mal byť v súvislosti s okolitým obsahom.

Nosná časť príkladu

```
<body>
<p>Ja a moja rodina sme navštívili Epcot centrum tento rok v lete.</p>

<aside>
  <h4>Epcot Center</h4>
  <p>Epcot Center je zábavný park v Disney World na Floride.</p>
</aside>
</body>
```

Celý program- [aside.html](#)

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok aside</title>

  <!--<link rel="stylesheet" href="css/style.css" type="text/css"-->
</head>
<body>
<p>Ja a moja rodina sme navštívili Epcot centrum tento rok v lete.</p>

<aside>
  <h4>Epcot Center</h4>
  <p>Epcot Center je zábavný park v Disney World na Floride.</p>
</aside>
</body>
```

```
</html>
```

Chrome:

Ja a moja rodina sme navštívili Epcot centrum tento rok v lete.

Epcot Center

Epcot Center je zábavný park v Disney World na Floride.

HTML5 <figure-postava> a <figcaption >

Účelom titulku <figure> je pridať vizuálne vysvetlenie obrazu.

Nosná časť príkladu

```
<figure>
  
  <figcaption>Obrazok č.1 - The Pulpit Rock, Norway.</figcaption>
</figure>
```

V HTML5, obraz a titulok môže byť zoskupené do prvku <Figure> :

Prvok ** definuje obrazok, prvok **<figcaption> definuje titulok obrázku.

Celý príklad-figure.html

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok figure</title>

  <!--<link rel="stylesheet" href="css/style.css" type="text/css"-->
</head>
<body>
<p>The Pulpit rock je masívny útes 604 metrov (1982 stôp) nad Lysefjorden,
naproti Kjerag plošine, v Forsand, Ryfylke, Nórsko. V hornej časti útesu je
```

približne 25 x 25 metrov ploch, a je známe turistické atrakcie v Nórsku.</p>

<figure>

<figcaption>Obrazok č.1 - The Pulpit Rock, Norway.</figcaption>

</figure>

</body>

</html>

Chrome:

The Pulpit rock je masivny útes 604 metrov (1982 stôp) nad Lysefjorden, naproti Kjerag plošine, v Forsand, Ryfylke, Nórsko. V hornej časti útesu je približne 25 x 25 metrov ploch, a je známe turistické atrakcie v Nórsku.



Obrazok č.1 - The Pulpit Rock, Norway.

Prečo používať Sémantické prvky?

Vývojári ktorí používali štandard HTML4, používali vlastné mená id / class do štýlových prvkov ako: záhlavie, hore, dole, päty, menu, navigácia, hlavné, kontajner, obsah, článok, bočnom paneli, topnav, atď.

Z tohto dôvodu bolo nemožné pre vyhľadávače identifikovať správny obsah webových stránok.

S novými HTML5 prvkov (<header> <footer> <nav> <sekcia> <article>), to bude jednoduchšie.

Podrobný popis zvyšných sématických prvkov HTML5

<article> Tag uvádza nezávislý, sebestačné obsah.

Článok by mal mať zmysel sám o sebe a malo by byť možné ho distribuovať nezávisle na zvyšku stránky.

Potenciálne zdroje pre <Article> prvk:

- príspevok:

- príspevok na blogu
- príspevok do novín
- komentár

Východiskové nastavenie prvku

```
article {
    display: block;
}
```

priklad **article2.html**

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">

  <title>Semantický prvok article</title>

  <!--<link rel="stylesheet" href="css/style.css" type="text/css">-->
</head>
<body>
<article>
  <h1>Google Chrome</h1>
  <p>Google Chrome je bezplatný open-source webový prehliadač vyvinutý spoločnosťou Google, vyšlo v roku 2008.</p>
</article>

<p><strong>Poznámka:</strong> Tento prvok, article, nie je podporovaný v programe Internet Explorer 8 a predchádzajúcich verziách.</p>
</body>
</html>
```

Chrome:

Google Chrome

Google Chrome je bezplatný open-source webový prehliadač vyvinutý spoločnosťou Google, vyšlo v roku 2008.

Poznámka: Tento prvok, article, nie je podporovaný v programe Internet Explorer 8 a predchádzajúcich verziách.

<aside> Obsah by mal byť v súvislosti s okolitým obsahom

<main> Prvok <hlavný> určuje hlavný obsah dokumentu.

Obsah vnútri <main> prvku by mal byť jedinečný v dokumente. Nemal by obsahovať žiadny obsah, ktorý sa opakuje v celom dokumente ako sú postranné prvky, navigačné prepojenia, informácie o autorských právach, logá webu a vyhľadávacie formuláre

Poznámka: Nesmie existovať viac ako jeden <main-hlavný> prvok v dokumente. Prvok <hlavný> nesmie byť potomok <article>, <aside>, <footer>, <header> alebo <nav> prvku.

Príklad použitia prvku <main>

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Semantický prvok main</title>
  <!--<link rel="stylesheet" href="css/style.css" type="text/css">-->
</head>
<body>
<main>
  <h1>WEB prehliadače</h1>
  <p>Google Chrome, Firefox, Internet Explorer sú najpoužívanejšie
prehliadače dnes.</p>
  <article>
    <h1>Google Chrome</h1>
    <p>Google Chrome je bezplatný open-source webový prehliadač vyvinutý
spoločnosťou Google, vyšiel v roku 2008.</p>
  </article>

  <article>
    <h1>Internet Explorer</h1>
    <p>Internet Explorer je bezplatný webový prehliadač od Microsoftu,
ktorý vyšiel v roku 1995.</p>
  </article>

  <article>
    <h1>Mozilla Firefox</h1>
    <p>Firefox je slobodný, open-source webový prehliadač Mozilla, ktoré
vyšlo v roku 2004.</p>
  </article>
</main>
</body>
</html>
```

Chrome:

WEB prehliadače

Google Chrome, Firefox, Internet Explorer sú najpoužívanejšie prehliadače dnes.

Google Chrome

Google Chrome je bezplatný open-source webový prehliadač vyvinutý spoločnosťou Google, vyšiel v roku 2008.

Internet Explorer

Internet Explorer je bezplatný webový prehliadač od Microsoftu, ktorý vyšiel v roku 1995.

Mozilla Firefox

Firefox je slobodný, open-source webový prehliadač Mozilla, ktoré vyšlo v roku 2004.

<mark> značka

Prvok **<mark>** definuje označený text. Používa sa ak chceme zvýrazniť časti textu.

Príklad-mark.html

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Semantický prvok main</title>
  <!--<link rel="stylesheet" href="css/style.css" type="text/css">-->
</head>
<body>
<p>Dnes si nezabudnite kúpiť <mark> mlieko </ mark></p>
<p>
  <strong> Poznámka: </strong> Prvok mark nie je podporovaný v programe
  Internet Explorer 8 a predchádzajúcich verziách </ p>.
</body>
</html>
```

Chrome:

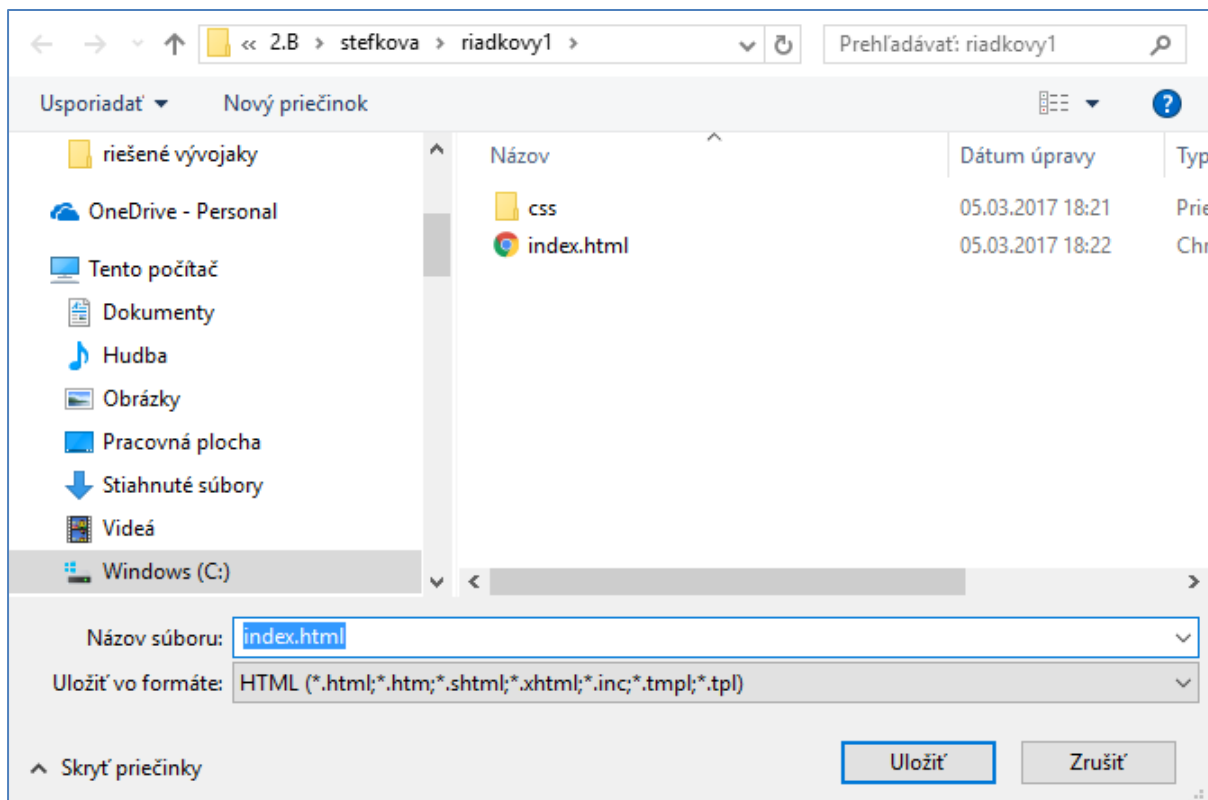
Dnes si nezapodíte kúpiť **mlieko**

Poznámka: Prvok mark nie je podporovaný v programe Internet Explorer 8 a predchádzajúcich verziách .

Príklady rozvrhnutia stránky

1. Riadkový layout-responzivne rozloženie

Index.html(c:\2.B\stefkova\riadkovy1\index.html)



```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Riadkový layout1</title>
  <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
<header>Hlavička</header>
<nav>Navigácia</nav>
<article>Článok</article>
<footer>Patička</footer>
```

```
</body>  
</html>
```

Style.css (c:\2.B\stefkova\riadkovy1\css\style.css)

```
/*css/style.css*/  
body{  
  margin: 0px;  
}  
header{  
  width:100%;  
  background-color:#993333;  
  line-height: 3;  
  color:white;  
  padding-left: 20px;  
  }  
nav{  
  width:100%;  
  background-color:#cc3300;  
  line-height: 3;  
  color:white;  
  padding-left: 20px;  
  }  
article{  
  width:100%;  
  background-color:#ffcc66;  
  line-height: 26;  
  color:white;  
  padding-left: 20px;  
  }  
footer{  
  width:100%;  
  background-color:#660033;  
  line-height: 3;  
  color:white;  
  padding-left: 20px;  
  }
```

chrome:



POZNÁMKA: Výpočet šírky boxov

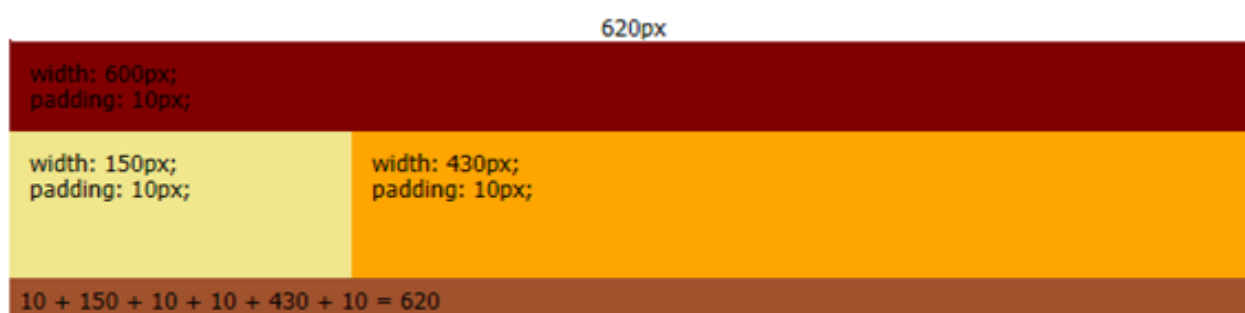
Stĺpcový layout - šírka vnútorných blokov

Vnútrošná šírka obalovacího bloku (width) určuje celkovú šírku iných blokov (tejto šírka sa musí rovnať súčet width, ľavého a pravého padding, prípadne rámčeku). O koľko zväčšíme rámček vnútorného divu, o toľko musíme zmenšiť jeho width. Ak zväčšíme padding o 1px (ľavý aj pravý), musíme zmenšiť width o 2px a naopak (to platí pri nezadaných vlastnosti box-sizing,

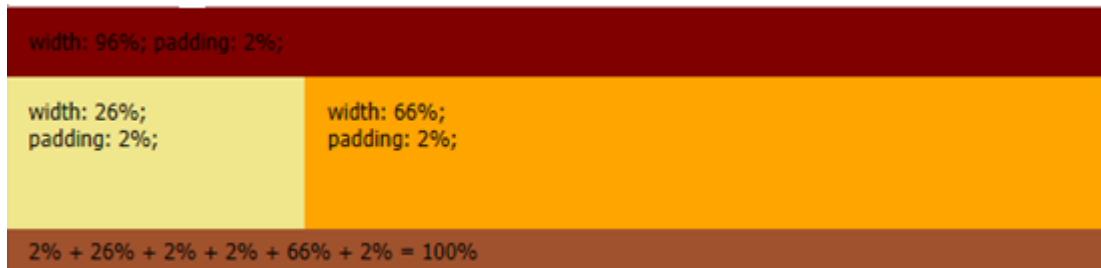
Pri hodnote `box-sizing: content-box` (alebo nezadaných vlastnosti box-sizing) platí spôsob počítania celkovej šírky popísaný vyššie (celková šírka bloku = width + padding + border + margin).

Pri hodnote `box-sizing: border-box` určuje width celkovú šírku bloku až po rámček.

Tento obrázok ilustruje box-sizing: `content-box`, alebo box-sizing nie je použitý



Percentuálne rozvrhnutie stránkv



Druhá varianta jednoriadkového layoutu - fixný rozdelenie

Index.html (c:\2.B\stefkova\riadkovy2\index.html)

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Riadkový layout s rozložením fixným</title>
  <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
<header>Hlavička</header>
<nav>Navigácia</nav>
<article>Článok</article>
<footer>Patička</footer>
</body>
</html>
```

Style.css (c:\2.B\stefkova\riadkovy2\css\style.css)

```
/*css/style.css*/
body {
  margin: 0;
}
header{
  width:960px;
  height:100px;
  background-color:#993333;
  color:white;
  line-height: 100px;
  padding-left: 20px;
}
nav{
  width:960px;
  height:100px;
  background-color:#993;
```

```
    color:white;
    line-height: 100px;
    padding-left: 20px;
  }
article{
    width:960px;
    height:300px;
    padding-left: 20px;
    background-color:#ffcc66;
    color:white;
    line-height: 300px;
  }
footer{

    width:960px;
    height:100px;
    line-height: 100px;
    padding-left: 20px;
    background-color:#660033;
    color:white;
  }
```

Chrome:



Riadkový layout – s vlastnosťami obalujúceho bloku main

Poznámka: Ak bloky header, nav, article a footer vložíme do obalovacieho bloku bloku main, môžeme nastaviť štýly tomuto obalujúcimú bloku.

Index.html c:\2.B\stefkova\riadkovy3\index.html)

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Riadkový layout s rozložením fixným a kontajnerom</title>
  <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
<main>
<header>Hlavička</header>
<nav>Navigácia</nav>
<article>Článok</article>
<footer>©copy Ing.Drgo Pavel,26.januar2016,17:33</footer>
</main>
</body>
</html>
```

Style.css c:\2.B\stefkova\riadkovy3\css\style.css)

```
/*css/style.css*/
body{
  margin:0px; /*umiestnenie polohy všetkých prvkov do lavého horného rohu*/
}
main {
  width:980px; /*width+padding-left*/
  border:3px solid #74514b;
  border-radius: 13px;
  box-shadow: 0px 0px 25px 0px #6e4941;
  margin-left: 10px;
  margin-top: 10px;
}
header{
  width:960px;
  height:100px;
  background-color:#993333;
```

```
    color:white;
    line-height: 100px;
    padding-left: 20px;
    }
nav{
    width:960px;
    height:100px;
    background-color:#993;
    color:white;
    line-height: 100px;
    padding-left: 20px;
    }
article{
    width:960px;
    height:300px;
    padding-left: 20px;
    background-color:#ffcc66;
    color:white;
    line-height: 300px;
    }
footer{

    width:960px;
    height:100px;
    line-height: 100px;
    padding-left: 20px;
    background-color:#660033;
    color:white;
    }
}
```

chrome



Dvojstĺpcový layout

Index.html (c:\2.B\stefkova\dvojstĺpcovy\index.html)

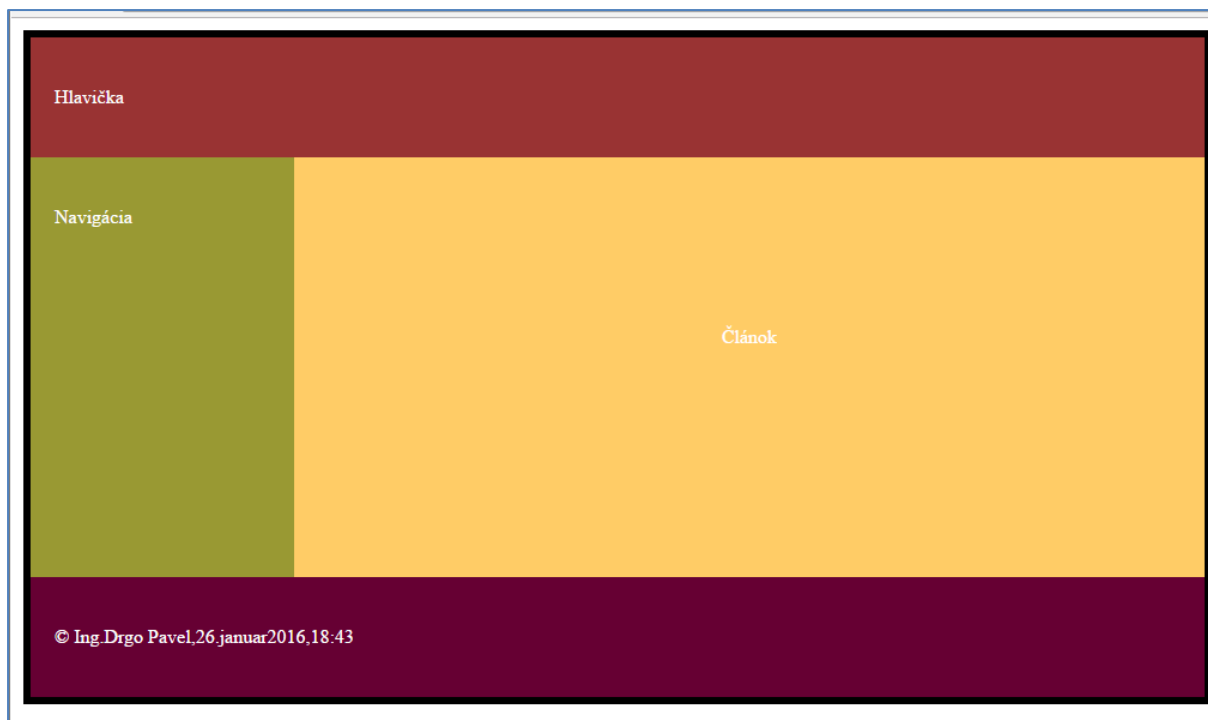
```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Dvojstĺpcový layout</title>
  <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
<main>
<header>Hlavička</header>
<nav>Navigácia</nav>
<article>Článok</article>
<footer>© Ing. Drgo Pavel, 26. januar 2016, 18:43</footer>
</main>
</body>
</html>
```

Style.css (c:\2.B\stefkova\dvojstĺpcovy\css\style.css)

```
/*css/style.css*/
body{
  margin:0px; /*umiestnenie polohy všetkých prvkov do ľavého horného rohu*/
}
```

```
main {
  width:980px;/*width+padding-left*/
  border:6px solid black;
  margin-left: 10px;
  margin-top: 10px;
}
header{
  width:960px;
  height:100px;
  background-color:#993333;
  color:white;
  line-height: 100px;
  padding-left: 20px;
}
nav{
  width:200px;
  height:350px;
  background-color:#993;
  color:white;
  line-height: 100px;
  padding-left: 20px;
  float: left;
}
article{
  width:960px;
  height:350px;
  padding-left: 20px;
  background-color:#ffcc66;
  color:white;
  line-height: 300px;
  text-align: center;
}
footer{
  width:960px;
  height:100px;
  line-height: 100px;
  padding-left: 20px;
  background-color:#660033;
  color:white;
}
```

Chrome:



Trojstĺpcový layout

Index.html(c:\2.B\stefkova\trojstĺpcovy\index.html)

```
<!DOCTYPE html>
<html lang="sk">
<head>
  <meta charset="UTF-8">
  <title>Trojstĺpcový layout</title>
  <link rel="stylesheet" href="css/style.css" type="text/css">
</head>
<body>
<main>
<header>Hlavička</header>
<nav>Navigácia</nav>
<article>Článok</article>
<aside>Bočný stĺpec</aside>
<footer>&copy; Ing.Drġo Pavel,26.januar2016,19:26</footer>
</main>
</body>
</html>
```

Style.css(c:\2.B\stefkova\trojstĺpcovy\css\style.css)

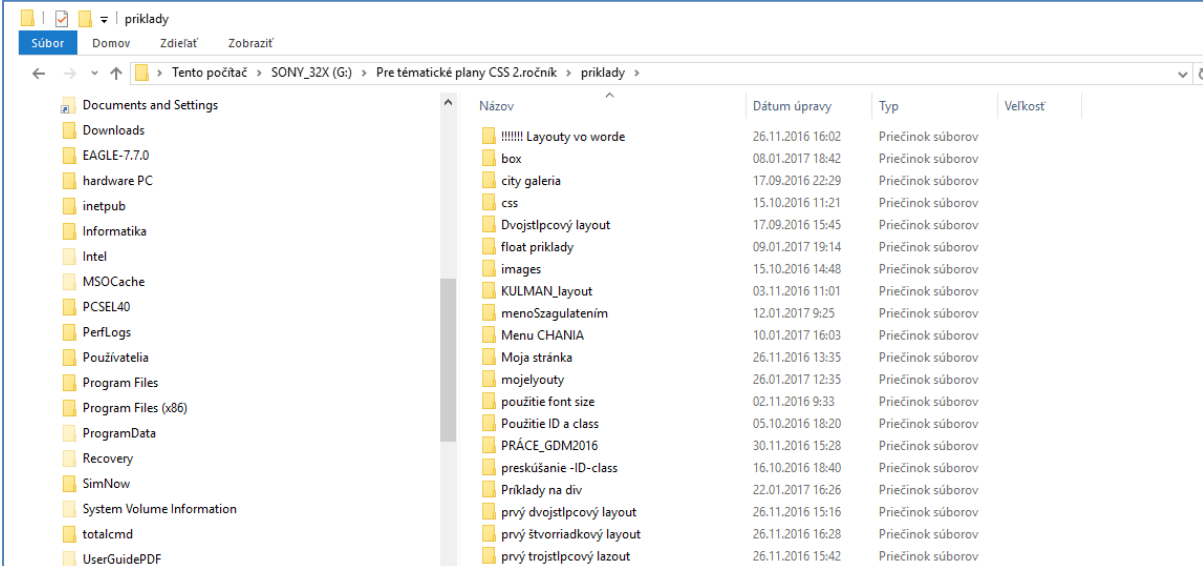
```
/*css/style.css*/
body{
  margin:0px; /*umiestnenie polohy všetkých prvkov do ľavého horného rohu*/
}
main {
  width:980px;/*width+padding-left*/
  border:6px solid black;
  margin-left: 10px;
  margin-top: 10px;
}
header{
  width:960px;
  height:100px;
  background-color:#993333;
  color:white;
  line-height: 100px;
  padding-left: 20px;
  border-bottom: 2px dashed white;
}
nav{
  width:200px;
  height:350px;
  background-color:#993;
  color:white;
  line-height: 100px;
  padding-left: 20px;
  float: left;
  border-right: 2px dashed white;
}
article{
  width:550px;
  height:350px;
  padding-left: 20px;
  background-color:#ffcc66;
  color:white;
  line-height: 300px;
  text-align: center;
  float :left;
  border-right: 2px dashed white;
}
aside{
  width:186px;
  height: 350px;
  background-color:#ff6;
  float: right;
  line-height: 300px;
}
```

```
text-align: center;
color: white;
}
footer{
    width:960px;
    height:100px;
    line-height: 100px;
    padding-left: 20px;
    background-color:#660033;
    color:white;
    clear: both;
    border-top: 2px dashed white;
}
```

Chrome:



PRÍKLADY RIEŠENIA STRÁNOK



The screenshot shows a Windows File Explorer window with the following details:

- Address bar: Tento počítač > SONY_32X (G:) > Pre tématické plány CSS 2.ročník > priklady
- Left sidebar: Shows the navigation pane with 'priklady' selected.
- Main pane: Displays a list of files and folders in a table view.

Názov	Dátum úpravy	Typ	Veľkosť
!!!!!!! Layouty vo worde	26.11.2016 16:02	Priečink súborov	
box	08.01.2017 18:42	Priečink súborov	
city galeria	17.09.2016 22:29	Priečink súborov	
css	15.10.2016 11:21	Priečink súborov	
Dvojstĺpcový layout	17.09.2016 15:45	Priečink súborov	
float priklady	09.01.2017 19:14	Priečink súborov	
images	15.10.2016 14:48	Priečink súborov	
KULMAN_layout	03.11.2016 11:01	Priečink súborov	
menoSzagulatenim	12.01.2017 9:25	Priečink súborov	
Menu CHANIA	10.01.2017 16:03	Priečink súborov	
Moja stránka	26.11.2016 13:35	Priečink súborov	
mojelyouty	26.01.2017 12:35	Priečink súborov	
použitie font size	02.11.2016 9:33	Priečink súborov	
Použitie ID a class	05.10.2016 18:20	Priečink súborov	
PRÁCE_GDM2016	30.11.2016 15:28	Priečink súborov	
preskúšanie -ID-class	16.10.2016 18:40	Priečink súborov	
Priklady na div	22.01.2017 16:26	Priečink súborov	
prvý dvojtĺpcový layout	26.11.2016 15:16	Priečink súborov	
prvý štvorriadkový layout	26.11.2016 16:28	Priečink súborov	
prvý trojtĺpcový lazout	26.11.2016 15:42	Priečink súborov	