

9 TABUĽKY

Tabuľky väčšinou využívame na prehľadné zobrazenie údajov, ktoré nejako spolu súvisia. Istý čas sa tabuľky (pri tvorbe webových stránok) využívali aj na rozmiestňovanie objektov na stránke, napr. na vytvorenie dvojstĺpcového alebo trojstĺpcového vzhľadu stránky. Dnes sa už tento spôsob považuje za zastaralý a na rozmiestňovanie objektov sa využívajú predovšetkým kaskádové štýly (to si ukážeme v kapitole 10).

9.1 Vytváranie tabuliek

Každá tabuľka sa skladá z riadkov, v ktorých je jedna alebo viacero buniek. Tabuľky v HTML definujeme po riadkoch, nie po stĺpcoch.

Pomocou elementu `<table></table>` definujeme začiatok a koniec tabuľky. V rámci elementu `<table>` definujeme jednotlivé riadky tabuľky pomocou elementu `<tr>`. A napokon v rámci elementu `<tr>` definujeme jednotlivé bunky pomocou elementu `<td>`, resp. elementu `<th>`, ak sa jedná o bunku, ktorá je hlavičkou stĺpca alebo riadka. Od počtu buniek v riadku závisí počet stĺpcov celej tabuľky. V prípade, že nie sú žiadne bunky spojené, môžeme povedať, že počet buniek v riadku (v každom riadku rovnaký) sa rovná počtu stĺpcov tabuľky. V prípade, že sú niektoré bunky v riadku spojené, musíme si dať pozor na počet buniek v príslušnom riadku, viac pozri časť *Zlučovanie buniek*.

PRÍKLAD 9.1

V editore JSFiddle vytvoríme tabuľku o vitamínoch. Začneme prvými dvoma riadkami, ktoré budú obsahovať tieto údaje:

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch

Do HTML časti napíšme nasledujúci kód:

```
<body>
  <table>
    <tr>
      <th>názov</th>
      <th>zdroje</th>
      <th>rozpustný v</th>
    </tr>
    <tr>
      <td>vitamín A</td>
      <td>plnotučné mlieko, vajcia, pečeň</td>
      <td>tukoch</td>
    </tr>
  </table>
</body>
```



Tabuľka, ktorú sme týmto kódom vytvorili, vyzerá takto:

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch

Štandardne sa orámovanie pre tabuľky a bunky nezobrazuje, resp. žiadne orámovanie pre ne nie je nastavené. Ak chceme, aby tabuľka bola orámovaná, definujeme pomocou kaskádových štýlov (teda v časti CSS) vlastnosť `border` pre elementy `table`, `td` a `th`.

```
table, td, th {  
  border: 1px solid black;  
}
```

Poznámka: Zápisom `table, td, th` definujeme naraz štýly pre elementy `table`, `td` aj `th`. Takýto zápis s čiarkou používame vtedy, ak chceme viacerým druhom elementov definovať rovnaké vlastnosti.



ÚLOHA 9.2

Doplňte do tabuľky z príkladu 9.1 ďalšie riadky podľa obrázka 9.1.

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch
vitamín B1 - tiamín	droždie, obilné kľičky, syr	vode
vitamín C	ovocie, zelenina	vode

Obrázok 9.1 Tabuľka Vitamíny.



ÚLOHA 9.3

V JSFiddle experimentujte s orámovaním tabuľky z úlohy 9.2:

- vyskúšajte rôzne hrúbky orámovania (napr. `2px`, `3px`, `10px`),
- vyskúšajte rôzne farby orámovania,
- nastavte orámovanie:
 - len pre `table` (t.j. pre `td` a `th` nie)
 - len pre `td`
 - len pre `th`
 - len pre `tr`
 - pre `td` a `th`
 - ako uznáte za vhodné.

Otázka: Aké štýly a pre aké elementy by sme museli definovať, aby sme okolo celej tabuľky mali hrubší rámik (napr. `5px`), ale okolo jednotlivých buniek tenšie rámiky (napr. `1px`)?



PRÍKLAD 9.4

Do tabuľky z predchádzajúcich úloh pridáme popis (legendu). Využijeme element `<caption>`.

```

<body>
  <table>
    <caption>Prehľad vitamínov</caption>
    <tr>
      <th>názov</th>
      ...

```

Popis tabuľky sa štandardne zobrazuje nad tabuľkou zarovnaný na stred.



ÚLOHA 9.5

V kóde z predchádzajúceho príkladu experimentujte s nastavením šírky, resp. výšky elementu v %.

- Meňte šírku okna prehliadača a pozorujte, ako sa mení šírka elementu.
- Postupne meňte šírku elementu na 100%, 80%, 20%, 5% a pozorujte. Čo sa stane, keď zvolíte príliš malú hodnotu?
- Zopakujte predchádzajúce dva body pre výšku. Je správanie stránky rovnaké ako pri nastavení šírky?

Tabuľka 9.1 Základné elementy pre tabuľku.

element	popis
<code><table></table></code>	začiatok a koniec celej tabuľky (vnútri musí byť všetko, čo sa týka tabuľky)
<code><tr></tr></code>	definuje celý riadok (vnútri sú bunky)
<code><th></th></code>	definuje bunku v rámci riadka, ktorá je nadpisom (zvyčajne zvýraznená) – používame len pre nadpisy riadkov/stĺpcov, nie kvôli zvýrazneniu
<code><td></td></code>	definuje bunku v rámci riadka
<code><caption></caption></code>	legenda celej tabuľky, zobrazuje sa nad okrajom tabuľky, musí byť definovaná v rámci elementu <code><table></table></code>

ÚLOHA 9.6

Vytvorte tabuľku informatických súťaží podľa obrázku 9.2. Údaje do tabuľky nájdete v súbore 09/sutaze.html.



INFORMATICKÉ SÚŤAŽE		
Názov	Cieľová skupina	Zameranie
iBobor	ZŠ a SŠ	informatika všeobecne
Olympiáda v informatike	stredoškóoláci	algoritmizácia a programovanie
Korešpondenčný seminár z programovania	stredoškóoláci	programovanie
PALMA	stredoškóoláci	programovanie
ISTROBOT		robotika
First Lego League	9-16 rokov	robotika

Obrázok 9.2 Tabuľka Informatické súťaže.

Obsah buniek (teda obsah elementu `<td>`) môžeme ľubovoľne štruktúrovať a formátovať. Takisto môžeme do buniek vložiť obrázky, či ďalšie tabuľky (tzv. vnáranie tabuliek).



ÚLOHA 9.7

Prerobte ponuku píz v súbore 09/ponuka.html pomocou tabuľky (pozri obrázok 9.3).

- Tabuľka bude mať stĺpce názov, zloženie, obrázok, alergény a cena.
- Názvy píz budú formátované ako nadpis úrovne 2.
- Zloženie bude zoznam jednotlivých položiek.
- Názov stĺpca Alergény bude odkazom na súbor `alergeny.html`.
- Bunky budú mať zobrazené len spodné orámovanie, tabuľka nebude mať orámovanie.
- Hlavičky stĺpcov budú zarovnané vľavo a väčším písmom (napr. 120%).

Názov	Zloženie	Obrázok	Alergény	Cena
Cardinale	<ul style="list-style-type: none"> • paradajková omáčka • syr • šunka 		1, 7	malá 4,00 € veľká 5,50 €
Funghi	<ul style="list-style-type: none"> • paradajková omáčka • syr • šampiňony 		1, 7	malá 5,00 € veľká 5,50 €

Obrázok 9.3 Ponuka píz ako tabuľka.

9.2 Vlastnosti tabuliek

V predchádzajúcich dvoch kapitolách sme si ukázali ako pomocou kaskádových štýlov nastaviť rôzne vlastnosti písma a textu, orámovanie a okraje blokov. Väčšinu týchto vlastností môžeme využiť aj pre elementy tvoriace tabuľky, nie všetky však majú zmysel pre každý z tabuľkových elementov. V tejto časti budeme veľa experimentovať s rôznymi vlastnosťami pre elementy `table`, `tr`, `td` a `th` a ukážeme si, resp. sami pridáte na to, ktoré nastavenia pre ktoré elementy sú vhodné. Tiež si ukážeme nové vlastnosti špecifické len pre tabuľky.

Vlastnosti elementu table

V časti 9.1 sme tabuľke nastavovali vlastnosť `border`. S orámovaním súvisia dve vlastnosti špecifické len pre element `table`: `border-collapse` a `border-spacing`. Ich význam si ukážeme na príkladoch.

PRÍKLAD 9.8

Vráťme sa k tabuľke vitamínov v editore JSFiddle z úlohy 9.4 (kód je v súbore 09/vitaminy.html). Chceme, aby tabuľka nemala separátne orámovanie pre bunky a celú tabuľku, ale aby tieto dve orámovania splynuli do jedného (porovnajme tabuľky na obrázku 9.1 a 9.4). Do CSS kódu doplníme nastavenie `border-collapse` pre element `table`.

```
table, td, th {
  border: 1px solid black;
}
table {
  border-collapse: collapse;
}
```

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvori sa v koži pôsobením slnečného žiarenia	tukoch
vitamín B1 - tiamín	droždie, obilné kľičky, syr	vode
vitamín C	ovocie, zelenina	vode

Obrázok 9.4 Splynutie orámovania buniek a tabuľky.

ÚLOHA 9.9

Meňte hrúbku, štýl a farbu orámovania. Prejaví sa zmena?

ODPOVEDZTE

Je možné pri splynutých orámovaniach buniek a celej tabuľky nastaviť inú farbu „vonkajšieho“ rámika a inú farbu pre „vnútorné čiary“?

Hodnota vlastnosti `border-collapse` rozhoduje o tom, či budú orámovania buniek a tabuľky oddelené (hodnota `separate`, prednastavená) alebo splynú (hodnota `collapse`).



PRÍKLAD 9.10

V editore JSFiddle nahradíme vlastnosť `border-collapse` vlastnosťou `border-spacing` s hodnotu `10px`. Tým definujeme, aká má byť vzdialenosť (priestor) medzi jednotlivými bunkami (pozri obrázok 9.5).

```
table, td, th {
  border: 1px solid black;
}
table {
  border-spacing: 10px;
}
```

názov	zdroje	rozpustný v
vitamin A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamin D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch
vitamin B1 - tiamín	droždie, obilné kľičky, syr	vode
vitamin C	ovocie, zelenina	vode

Obrázok 9.5 Tabuľka s nastavenou vzdialenosťou susedných buniek.



ÚLOHA 9.11

V tabuľke z príkladu 9.10:

- experimentujte s hodnotou `border-spacing`,
- zmeňte hodnotu `border-spacing` na dvojicu `5px 10px`,
- nastavte hodnotu `border-spacing` na dvojicu `20px 5px` a zrušte vlastnosť `border` pre `th` a `td`, pre `table` ju zachovajte!,
- vráťte vlastnosť `border` pre `td` a `th`, pridajte `border-collapse` s hodnotou `collapse` a skúšajte rôzne hodnoty pre `border-spacing`. Prejaví sa nejaká zmena hodnôt `border-spacing`? Prečo asi?



ÚLOHA 9.12

V tabuľke z predchádzajúcej úlohy experimentujte s nastaveniami LEN pre element `table`:

- zrušte `border-spacing`, ponechajte `border-collapse: collapse`,
- definujte šírku (`width`), vyskúšajte napr. hodnoty `70%`, `100%`, `50%`, `300px`, `600px`, ..,
- definujte výšku (`height`), vyskúšajte napr. hodnoty `200px`, `100px`, `130px`, ...,
- nastavte vonkajší okraj (`margin`) na `20px`,
- nastavte farbu pozadia (`background-color`) na `#ECD9C6`,
- nastavte `font-family` na `Arial`,
- vnútorný okraj (`padding`) na `20px`.

Tabuľka 9.2 Vlastnosti tabuľky.

element		popis
<code>border-collapse</code>	collapse separate	<p>Definuje správanie orámovania buniek tabuľky.</p> <p>collapse – Orámovania tabuľky a jednotlivých buniek budú splývať. Čiary orámovania budú vycentrované v priestore medzi bunkami.</p> <p>separate – Celá tabuľka a každá bunka tabuľky má svoje vlastné orámovanie a vzdialenosť medzi bunkami je riadená vlastnosťou <code>border-spacing</code>.</p>
<code>border-spacing</code>	<p><i>dĺžka1 dĺžka2 dĺžka</i></p> <p>napr. 10px 2px</p> <p>alebo len 5px</p>	<p>Určuje vzdialenosť medzi orámovaním susedných buniek. Priestor medzi orámovaním je vyplnený pozadím tabuľky.</p> <p>Ak zadáme dve hodnoty, prvá určuje vzdialenosť v horizontálnom a druhá vo vertikálnom smere.</p> <p>Ak zadáme len jednu hodnotu, určuje vzdialenosť pre oba smery.</p>
<code>caption-side</code>	top bottom	Určuje polohu legendy/hlavičky vzhľadom k tabuľke.

Okrem uvedených špecifických vlastností môžeme elementu `table` definovať takmer všetky nám už známe vlastnosti, napr. rôzne vlastnosti textu (`font-family`, `color`, `font-weight`, ...), farbu pozadia, vonkajšie či vnútorné okraje, šírku, výšku.

Vlastnosti elementu tr

PRÍKLAD 9.13

V tabuľke z úlohy 9.12 nastavme prvému riadku (riadku s hlavičkami) tmavé pozadie (napr. farby `#130D06`), svetlú farbu textu (napr. `#ECD9C6`) a zarovnanie vľavo (obrázok 9.6).

Riešenie

```
tr {
  background-color: #130D06;
  color: #ECD9C6;
  text-align: left;
}
```



zrejme nie je úplne správne, pretože takto sme zmenili všetky riadky tabuľky. Už vieme, že ak chceme definovať štýl práve jednému výskytu elementu, musíme tento element nejako identifikovať. Definujme prvému riadku tabuľky v HTML kóde identifikátor `id="prvy"`.

```
<table>
  <caption>Prehľad vitamínov</caption>
  <tr id="prvy">
    <th>názov</th>
    ...
```

a v CSS kóde definujme štýl pre `#prvy` (namiesto pre element `tr`).

```
<table>
  <caption>Prehľad vitamínov</caption>
  <tr id="prvy">
    <th>názov</th>
    ...
```



ODPOVEDZTE

Vedeli by ste (pre našu konkrétnu tabuľku) vyriešiť *príklad 9.13* aj bez použitia atribútu `id` (resp. bez zásahu do HTML časti, len použitím CSS)? Bolo by vami navrhnuté riešenie použiteľné aj pre iné tabuľky? Pripomíname že chceme zvýrazniť len prvý riadok.

Prehľad vitamínov

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch
vitamín B1 - tiamín	droždie, obilné klíčky, syr	vode
vitamín C	ovocie, zelenina	vode

Obrázok 9.6 Zvýraznenie prvého riadka tabuľky.



ÚLOHA 9.14

V CSS kóde z *príkladu 9.13* pridajte do štýlu pre `#prvy`:

- nastavenie šikmého písma,
- nastavenie výšky, napr. na `30px`,
- nastavenie šírky – skúšajte rôzne hodnoty,
- nastavenia `margin`, `padding`, `border`.

Riadkom tabuľky (elementu `tr`) môžeme nastavovať rôzne vlastnosti textu, farbu pozadia a výšku. Nemá však zmysel nastavovať im akékoľvek okraje či orámovanie ani šírku.

Otázka: Zamyslite sa (resp. vyskúšajte), čo sa stane, ak definujeme štýl pre pseudotriedu `tr:hover` (napr. `tr:hover {background-color: white}`).



PRÍKLAD 9.15

V editore JSFiddle v CSS časti ponechajme v štýle pre element table len nasledujúce vlastnosti:

```
...
table {
  border-collapse: collapse;
  width: 350px;
  background-color: #ECD9C6;
  font-family: Arial;
}
#prvy {
  ...
```

Tabuľku sme zámerne zúžili na 350px, aby nám vznikli bunky s viacerými riadkami textu. Všimnime si teraz napr. bunku s textom *vitamín D*. Text v nej je vodorovne zarovnaný vľavo a zvislo sa v rámci bunky nachádza v strede. Aj všetky ostatné texty sú zvislo umiestnené v strede buniek.

Zmeňme zvislé zarovnanie textu vo všetkých bunkách tabuľky tak, aby text bol umiestnený v rámci bunky hore – definujme pre `td` a `th` vlastnosť `vertical-align`.

```
...
td, th {
  vertical-align: top;
}
```

názov	zdroje	rozpustný v	názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch	vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch	vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch
vitamín B1 - tiamín	droždie, obilné klíčky, syr	vode	vitamín B1 - tiamín	droždie, obilné klíčky, syr	vode
vitamín C	ovocie, zelenina	vode	vitamín C	ovocie, zelenina	vode

Obrázok 9.7 Zvislé zarovnanie textu: na stred a hore.

Pomocou vlastnosti `vertical-align` definujeme zvislé zarovnanie textu v rámci bunky. Môže nadobúdať hodnoty `middle` (teda akési zvislé centrovanie), `top` a `bottom`. Vlastnosť môžeme použiť aj pre element `tr`, nie však pre `table`.

ÚLOHA 9.16



V CSS kóde z príkladu 9.15 experimentujte s vlastnosťami pre elementy `td` a `th`:

- zmeňte nastavenie `vertical-align` na `middle`, resp. `bottom`,
- nastavte hodnotu `padding` na `3px`, `10px`, resp. štvoricu `5px 10px 5px 10px`,
- zrušte vlastnosť `width` pre element `table` a definujte `width` pre `td` a `th` s hodnotou

- 150px; experimentujte s inými hodnotami, skúste príliš malé aj príliš veľké hodnoty, pridajte nastavenie `height` s hodnotou napr. 30px, 50px, 80px.



ODPOVEDZTE

Pre aký typ údajov by bolo vhodné použiť vodorovné zarovnanie s hodnotou `right`?

Ako by ste nastavili šírku len prvého stĺpca tabuľky na 100px?

Bunkám tabuľky môžeme nastavovať vlastnosti textu (vrátane horizontálneho aj vertikálneho zarovnanie), farbu pozadia, šírku, výšku, vnútorné okraje a orámovanie. Nemá zmysel nastavovať im vlastnosť `margin`, lebo okolie buniek určujú vlastnosti elementu `table`.

CSS – triedy



PRÍKLAD 9.17

V JSFiddle upravíme štýly pre tabuľku tak, aby sa striedali farby riadkov ako na obrázku 9.8. Najskôr si do CSS časti nakopírujeme štýly zo súboru 09/styl09pr17.txt.

Prehľad vitamínov

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	tukoch
vitamín B1 - tiamín	droždie, obilné klíčky, syr	vode
vitamín C	ovocie, zelenina	vode

Obrázok 9.8 Tabuľka so striedajúcimi sa farbami riadkov.

S doterajšími znalosťami by sme to vedeli vyriešiť napr. takto: v HTML kóde by sme tretiemu a piatemu riadku definovali atribút `k` (pre každý z týchto riadkov rôzne, lebo `id` musí byť unikátne) a v CSS kóde by sme pre tieto dva identifikátory vytvorili štýl, v ktorom by sme nastavili farbu pozadia. Toto riešenie by bolo funkčné, ale nie veľmi efektívne. Ak by tabuľka mala viac riadkov, museli by sme pre polovicu riadkov definovať rôzne identifikátory a pridať všetky tieto identifikátory do CSS časti, čím vlastne vytvárame toľko štýlov, koľko je identifikátorov, hoci všetky tieto štýly sú rovnaké. Potrebujeme jeden spoločný štýl pre viacero výskytov (ale nie všetky) istého elementu. Na to môžeme použiť tzv. triedy.

V HTML kóde tabuľky si „označíme“ všetky tie elementy `tr`, ktoré majú mať svetlejšie pozadie – v našom prípade 2. a 4. riadok. Označíme ich pridaním atribútu `class` s hodnotou napr. "svetly".

```
<table>
<caption>Prehľad vitamínov</caption>
<tr class="prvy">
...
</tr>
<tr class="svetly">
  <td>vitamín A</td>
...

```

```

</tr>
<tr>
  <td>vitamín D</td>
  ...
</tr>
<tr class="svetly">
  <td>vitamín B1 - tiamín</td>
  ...
</tr>
...

```

V CSS kóde definujeme triedu (štýl) s názvom `.svetly` a jedinou vlasnosťou – farbou pozadia. V definícii kaskádového štýlu musíme meno triedy uvádzať s bodkou na začiatku (bez medzery), napr. `.svetly`.

```

.svetly {
  background-color: #F9F2EC;
}

```

Takto definovanú triedu môžeme použiť nielen pre element `tr`, ale pre ľubovoľný element. Môžeme napríklad doplniť pred tabuľku nadpis `h1` a naň tiež aplikovať štýl definovaný triedou `.svetly`.

```

<body>
  <h1 class="svetly">Vitamíny</h1>
  <table>
  ...

```

ZAPAMÁTAJTE SI

V kaskádových štýloch zadefinujeme triedu spôsobom

```
.nazov_triedy { vlastnosť: hodnota; ... }
```

a v HTML kóde označíme elementy, pre ktoré chceme túto triedu použiť pomocou atribútu `class="nazov_triedy"`.

ÚLOHA 9.18

V súbore `09/body.html` máme definovanú tabuľku s bodmi žiakov na seminári z informatiky. Využite triedu (`class`) na to, aby stĺpec tabuľky, v ktorom je súčet bodov, mal šírku `100px` a text v ňom bol zarovnaný vpravo a tučný. Zarovnajte vpravo tiež obsah všetkých ostatných buniek, v ktorých sú čísla (nie je potrebná trieda).

Zlučovanie buniek

Pomocou HTML vieme definovať aj tabuľky so zlúčenými bunkami. Môžeme zlučovať niekoľko buniek vedľa seba v rámci riadka alebo niekoľko buniek pod sebou v rámci jedného stĺpca.





PRÍKLAD 9.19

V súbore 09/vitaminy2.html máme podobnú tabuľku s vitamínmi, s akou sme pracovali doteraz, len s väčším množstvom údajov a takmer bez štýlov. Zlúčime v 3. stĺpci všetky po sebe idúce bunky, v ktorých je rovnaká hodnota (pozri obrázok 9.9). Tabuľku môžeme upravovať priamo v súbore alebo ju nakopírujeme do JSFiddle.

názov	zdroje	rozpustný v
vitamín A	plnotučné mlieko, vajcia, pečeň	tukoch
vitamín D	tvorí sa v koži pôsobením slnečného žiarenia	
vitamín B1 - tiamín	droždie, obilné klíčky, syr	vode
vitamín B2 - riboflavín	kvasnice, mlieko, mäso, ryby	
vitamín C	ovocie, zelenina	

Obrázok 9.9 Zlúčenie buniek v rámci stĺpca.

Ak chceme zlúčiť bunky pod sebou v rámci jedného stĺpca, použijeme atribút `rowspan`. Jeho hodnotou je číslo, vyjadrujúce koľko buniek pod sebou chceme zlúčiť do jednej. Atribút definujeme pre tú bunku tabuľky, ktorá je prvá (najvyššia) zo zlučovaných. Vo všetkých nasledujúcich riadkoch, ktoré sa majú zlúčiť, musíme túto jednu bunku vynechať!

```
<table>
  <tr>
    <th>názov</th>
    <th>zdroje</th>
    <th>rozpustný v</th>
  </tr>
  <tr>
    <td>vitamín A</td>
    <td>plnotučné mlieko, vajcia, pečeň</td>
    <td rowspan="2">tukoch</td>
  </tr>
  <tr>
    <td>vitamín D</td>
    <td>tvorí sa v koži pôsobením slnečného žiarenia</td>
    <td>tukoch</td>
  </tr>
  <tr>
    <td>vitamín B1 - tiamín</td>
    <td>droždie, obilné klíčky, syr</td>
    <td rowspan="3">vode</td>
  </tr>
  <tr>
    <td>vitamín B2 - riboflavín</td>
    <td>kvasnice, mlieko, mäso, ryby</td>
    <td>vo vode</td>
  </tr>
  <tr>
    <td>vitamín C</td>
    <td>ovocie, zelenina</td>
    <td>vo vode</td>
  </tr>
</table>
```

Podobne funguje zlučovanie buniek vedľa seba v rámci jedného riadka. Do najľavejšej bunky zo zlučovaných pridáme atribút `colspan`, ktorého hodnotou bude počet zlučovaných buniek.

Definície zvyšných zlučovanych buniek v danom riadku potom vynecháme (t.j. vynecháme príslušné `td` elementy).

PRÍKLAD 9.20

V súbore `09/adresy.html` zlúčime bunky, kde fakturačná adresa a doručovacia adresa sú rovnaké (pozri obrázok 9.10).

meno	doručovacia adresa	fakturačná adresa
Jožko Mrkvička	Hronská 5, 974 01 Banská Bystrica	Nevädzová 54, 821 01 Bratislava
Aneta Múdra	Alejová 7, 040 01 Košice	
Boris Malina	Chalupkova 17, 010 09 Žilina	Hečkova 4, 010 01 Žilina
Milena Veselá	L. Novomeského 22, 911 08 Trenčín	

Obrázok 9.10 Zlúčenie buniek v rámci stĺpca.

```
<table>
  <tr>
    <th>meno</th>
    <th>doručovacia adresa</th>
    <th>fakturačná adresa</th>
  </tr>
  <tr>
    <td>Jožko Mrkvička</td>
    <td>Hronská 5, 974 01 Banská Bystrica</td>
    <td>Nevädzová 54, 821 01 Bratislava</td>
  </tr>
  <tr>
    <td>Aneta Múdra</td>
    <td colspan="2">Alejová 7, 040 01 Košice</td>
    <td>Alejová 7, 040 01 Košice</td>
  </tr>
  <tr>
    <td>Boris Malina</td>
    <td>Chalupkova 17, 010 09 Žilina</td>
    <td>Hečkova 4, 010 01 Žilina</td>
  </tr>
  <tr>
    <td>Milena Veselá</td>
    <td colspan="2">L. Novomeského 22, 911 08 Trenčín</td>
    <td>L. Novomeského 22, 911 08 Trenčín</td>
  </tr>
</table>
```

ÚLOHA 9.21

V tabuľke so súťažami z úlohy 9.6 zlúčte tie po sebe idúce bunky v stĺpci Zameranie, v ktorých je rovnaká hodnota (pozri obrázok 9.11).

INFORMATICKÉ SÚŤAŽE

Názov	Cieľová skupina	Zameranie
iBobor	ZŠ a SŠ	informatika všeobecne
Olympiáda v informatike	stredoškóoláci	programovanie
Korešpondenčný seminár z programovania	stredoškóoláci	
PALMA	stredoškóoláci	
ISTROBOT		robotika
First Lego League	9-16 rokov	

Obrázok 9.11 Tabuľka súťaží so zlúčenými bunkami.



ÚLOHA 9.22

Pomocou tabuľky vytvorte svoj rozvrh hodín. Môže byť vymyslený, s menším počtom hodín, ako ten reálny, ale nech sú v ňom aspoň štyri rôzne predmety a aspoň dve dvojhodinovky. Tabuľka by mala spĺňať tieto vlastnosti:

- bude sa zobrazovať orámovanie tabuľky aj orámovanie buniek, pričom tieto čiary splynú,
- tabuľka bude zaberáť 80% šírky stránky,
- obsah buniek bude zarovnaný na stred horizontálne aj vertikálne,
- obsah buniek bude od orámovania mierne odsadený,
- každý predmet bude mať inú farbu, rovnaké predmety tú istú farbu (ak máte veľa predmetov, rozdeľte ich do troch skupín, prírodovedné, humanitné a ostatné a každá skupina nech má svoju farbu),
- dvojhodinovky (prípadne viachodinovky) budú zlúčené bunky,
- hlavičky stĺpcov a riadkov budú tučným písmom.

Ďalšie vlastnosti doplňte podľa vlastnej fantázie.

9.3 Metodické pokyny pre učiteľa

CIEĽ

Cieľom je naučiť sa vytvárať tabuľku pomocou HTML jazyka, experimentovať s vlastnosťami tabuliek, preskúmať, ktoré vlastnosti majú zmysel pre celú tabuľku, ktoré pre riadky a ktoré len pre bunky.

VÝKLAD

K poznámke za príkladom 9.1: Ak chceme rovnaké vlastnosti zdefinovať viacerým druhom elementov, stačí ich oddeliť čiarkou, napr.

- `h1, h2 {...}` - tento štýl bude patriť všetkým výskytom elementov `h1` a `h2` na stránke,
- `#id1, #id2: {...}` - tento štýl sa aplikuje na element s identifikátorom `id1` a element s identifikátorom `id2`,
- `img, #galeria: {...}` - tento štýl ovplyvní všetky výskyty elementu `img` a element s identifikátorom `galeria`,
- `h1, footer p: {...}` - týmto štýlom nastavíme vzhľad elementov `h1` a tých elementov `p`, ktoré sú vnorené v elemente `footer`.

Úloha 9.6: Nie je podstatné, aby tabuľka vyzerala presne ako na obrázku, ale mali by sa aspoň trochu pohrať aj s orámovaním tabuľky a buniek. Na obrázku boli použité nasledujúce štýly:

- celá tabuľka je orámovaná hrubším súvislým čiernym rámikom
- bunky sú orámované 1px bodkovaným sivým rámikom
- `th` sú zarovnané doľava
- `caption` je tučným písmom, kapitálkami a má pridaný spodný vonkajší okraj

K tabuľke 9.2: Vlastnosť `caption-side` nefunguje v starších verziách IE, Opera.

Príklad 9.13: Treba vyskúšať, aj toto nesprávne riešenie - aspoň pekne farebne vynikne `padding`, lebo celá tabuľka má svetlohnedú farbu pozadia a riadky budú mať tmavú, tie prekryjú tú svetlohnedú, ale nie vo vnútornom okraji tabuľky.

K otázke za príkladom 9.13: Príslušný štýl môžeme nastaviť elementu `th`, ale len pre tabuľky, ktoré majú `th` len v prvom riadku (hlavičku majú len stĺpce). Pre ľubovoľnú tabuľku nie, lebo `th` môže byť súčasne použité aj ako hlavičky riadkov, a teda by sme zvýraznili okrem 1. riadku aj 1. stĺpec.

K vertical align

Vlastnosť `vertical-align` môžeme používať nielen pre tabuľkové elementy. Vlastnosť môže nadobúdať hodnoty `baseline` | `sub` | `super` | `top` | `text-top` | `middle` | `bottom` | `text-bottom` | `length` | `%`. Pre elementy `td`, `th` a `tr` však majú zmysel len hodnoty `middle`, `top` a `bottom`.

Dá sa to skúšať na na:

https://www.w3schools.com/cssref/playit.asp?filename=playcss_vertical-align&preval=baseline

K triedam (class)

V učebnici sme definovali tzv. všeobecné triedy, také, ktoré môžeme aplikovať na ľubovoľný element. Triedu môžeme priradiť len istému elementu, napr. `p.kapitalky {font-variant: small-cass}`. V takom prípade sa trieda aplikuje len na tie elementy `p`, ktoré majú definovaný atribút `class="kapitalky"`. Ak by sme atribút `class="kapitalky"` definovali napr. pre element `h1`, nadpis by to nijako nezmenilo, lebo trieda `.kapitalky` je určená výhrane pre elementy `p`.

Jednému elementu môžeme priradiť aj viacero tried, napr. `<p class="kapitalky" class="svetly">`.

Úloha 9.22 je opakovacia.

Nadväznosť na predchádzajúce kapitoly: Kapitola vyžaduje všetky predchádzajúce kapitoly.