

# 9. CSS

1

## Umiestňovanie objektov (1)

# Obsah

2

- Normálne (+ relatívne) umiestňovanie
- Plávajúce umiestňovanie
- Zmena spôsobu zobrazenia elementov - vlastnosť **display**

# Čo chýba našim stránkam?

3

TWD tour

TWD TOUR

úvod | Grécko | Bulharsko | kontakt

*Rezervujte si miesto u nás ...*

Vitajte na stránkach cestovnej kancelárie TWD tour.

V súčasnosti poskytujeme letecké zájazdy do Grécka a Bulharska. Našu ponuku však v blízkej budúcnosti rozšírime napr. o Egypt, Turecko a exotický Dubai.

Pozrite si našu ponuku v pohodni Vášho domova alebo kancelárie a objednajte si pomocou nášho on-line formulára.

V prípade, že máte návrhy/pripomienky ku našej stránke alebo ponuke, napíšte nám.

## Akcie na tento týždeň

- ♦ Grécko - ostrov Kos: Hotel Cosmopolitan \*\*, dĺžka zájazdu: 11 dní, strava: raňajky, cena: **464.38 EUR**
- ♦ Pre rodiny (2+2) zľava **20%**

vytvoril: *TWD team*, informácie sú prevzaté zo stránky [zajazdy.sk](http://zajazdy.sk)

# Spôsoby umiestňovania objektov

4

- Existujú 3 spôsoby, ako môžeme objekty na stránke umiestniť:
  - Normálne
  - Plávajúco (často je dôležité poradie elementov v zdrojovom kóde)
  - Absolútne (v ďalšej prednáške)

# Normálne umiestňovanie

5

- Každý objekt v normálnom radení môžeme posunúť relatívne k jeho základnej pozícii, tzv. **relatívne umiestňovanie**.
- Posun **nemá vplyv** na umiestnenie nasledujúcich objektov (môžu sa teda prekrývať).
- Využijeme vlastnosť **position** s hodnotou **relative**.
- Posunutie objektov určujeme pomocou vlastností **left**, **right**, **top**, **bottom** (od ktorého okraja chceme posun, nemusíme uviesť všetky).
- **Nepoužíva sa na umiestňovanie základných objektov (štruktúry) na stránke**, ale používa sa skôr na „dokreslenie“ objektov.

# Normálne umiestňovanie – ukážka

6

- Dočasne posunieme odkazy v navigácii o 1/2 znaku dolu.

```
nav a {
```

```
...
```

```
position: relative;
```

```
top: 0.5em;
```

```
}
```

# Plávajúce umiestňovanie

7

- Plávajúce objekty sú **posunuté úplne vľavo alebo vpravo v rámci bloku**, v ktorom sa nachádzajú.
- Plávajúci objekt sa vždy správa blokovo.
- Každý plávajúci objekt má definovanú šírku (buď explicitne – vlastnosť **width** alebo prirodzene – podľa obsahu).
- Plávajúce objekty môžu spolu susediť (aj na 1 riadku).
- Plávajúci objekt definujeme pomocou vlastnosti **float** s hodnotami **left** | **right** | **none** (ku ktorému okraju sa má objekt posunúť, resp. zrušenie plávania objektu).
- Zvyčajne nekombinujeme plávajúce a relatívne umiestňovanie.

# Zmena dizajnu (krok 1)

8

- Plávajúce objekty môžeme využiť pri hlavnom obsahu stránky a časti s akciami.
- Nestačí, ak obom častiam nastavíme vlastnosť **float: left** (prípadne **float: right**). Problémom je, že objekty nemajú definovanú šírku. Preto by sa skoro nič neudialo (až na päť – nebola by správne umiestnená).
- Musíme nastaviť aj šírky: section – **73%**, aside – **23%**
- Prečo len 73%, resp. 23%?
  - Elementy **section** a **aside** majú nastavené **padding: 1%**.
  - Pozor na tzv. box model – prehliadače ho rôzne interpretujú



# Zmena dizajnu (krok 2)

9

- Ako donútiť pätičku, aby bola vždy pod časťami s obsahom a akciami?
- Použijeme vlastnosť **clear**, ktorá môže nadobúdať hodnoty **left** | **right** | **none** | **both**
- Vlastnosť clear **zakazuje obtekanie** ďalších objektov na danej strane (napr. hodnota left znamená, že na ľavej strane objektu nemôže byť plávajúci objekt).
- Pri pätičke využijeme hodnotu **both** (stačilo aj by left).
- Upozornenie: Zvyčajne vlastnosť **clear** nedefinujeme plávajúcemu objektu, ale objektom pred/za plávajúcim objektom.

# Rôzne potenciálne problémy

10

- Pozor na bočné orámovanie obsahu a/alebo akcií. Ak by sme ho niekde pridali, časti by neboli vedľa seba – nezmestili by sa (100% + orámovanie).
- Ak sa dá, treba mať šírku a rôzne okraje v rovnakých jednotkách (zvyčajne %). Ak to nie je možné, niekedy treba použiť pomocný element, napr. `<div>`.
- Čo, ak by obsah mal nejaké pozadie? Keďže obsah je kratší na výšku ako akcie, pozadie obsahu by nebolo až po pätičku. Prípadne naopak.

# Na zamyslenie

11

- Čo ak by sme mali v zdrojovom kóde skôr definované akcie a až potom obsah?

```
<body> ...
```

```
<aside>...</aside>
```

```
<section>...</ section >
```

```
...
```

```
</body>
```

- Stačilo by, aby sme **float: right** nastavili len časti s akciami => text hlavnej časti by obtekal okolo časti s akciami.

# Na zamyslenie – iné riešenie

12

- Stačí časti s hlavným obsahom nastaviť pravý okraj aspoň na šírku akcií (nemusíme definovať šírku).
- Rozdiel je v tesnosti textu obsahu a akcií a v obtekaní textu. Pri tomto riešení môžeme nastaviť bočné orámovanie obsahu.

```
section { margin-right: 25%; }
```

- V čom bol „problém“ – neumiestňované objekty pred a za plávajúcim objektom (v zdrojovom kóde) sú umiestnené tak, ako keby plávajúci objekt neexistoval.

# 3-stĺpcový dizajn (verzia 1)

13

- Pokúsme sa zmeniť dizajn stránky tak, že navigácia bude prvým stĺpcom ešte pred hlavným obsahom.
- Navigácii nastavíme šírku **8%** a **float: left** (už má nastavené **padding: 0 1%**).
- Hlavnému obsahu zmenšíme priestor – pridáme **margin-left: 10%** (urobíme tak priestor pre navigáciu).
- Poznámka: navigácia je vedľa reklamy

# 3-stĺpcový dizajn (verzia 2)

14

- Reklamu presunieme pred navigáciu (v zdrojovom kóde)
- Oproti verzii 1:
  - Hlavnému obsahu **zrušíme** margin-left aj margin-right.
  - Hlavnému obsahu **nastavíme width: 63%** a **float: left**.

# Zmena navigácie pri 3-stĺpcovom dizajne

15

- Pri 3-stĺpcovom dizajne nemáme peknú zvislú navigáciu. Skúsime ju vylepšiť.
- Zrušíme znak | zo zdrojového kódu a aj medzery medzi odkazmi navigácie.
- Čo spraviť, ak chceme mať odkazy v navigácii v samostatných riadkoch?
  - Môžeme ich oddeliť pomocou `<br>`, alebo
  - ich dať do samostatných blokov, napr. cez `<div>`, `<p>` alebo
  - použiť zoznam s odrážkami alebo
  - **použiť vlastnosť `display`** (zmeníme správanie odkazov).

# Vlastnosť display

16

- Môžeme zmeniť spôsob zobrazenia elementov.
- Môžeme použiť/nastaviť **ľubovoľnému** elementu – **treba zvážiť dôsledky**.
- Môže mať hodnoty: **none** | **inline** | **block** | **list-item** | run-in | compact | marker | table | inline-table | table-row-group | table-header-group | table-footer-group | table-row | table-column-group | table-column | table-cell | table-caption
- [www.w3schools.com/cssref/pr\\_class\\_display.asp](http://www.w3schools.com/cssref/pr_class_display.asp)



# Odkazy ako bloky

17

- Ak nastavíme odkazu vlastnosť **display: block**, stane sa z neho blokový element (zaberá celú šírku v obsahovom elemente, vizuálne sa zmení aj :hover).
- Pomocou **display: none** skryjeme element.
- Námet na doma:
  - Vyskúšajte si spraviť navigáciu ako zoznam s odrážkami.
  - Potom sa pokúste ju zobrazíť pomocou CSS do jedného riadka (ako pôvodné riešenie v ukážke 05).

# Užitečné zdroje

- <http://interval.cz/clanky/trisloupcovy-layout-s-hlavickou-a-patickou/>
- <http://interval.cz/clanky/trisloupcovy-layout-webu-pomoci-css/>
- <http://www.alistapart.com/articles/flexiblelayouts/>
- <http://www.csszengarden.com/>
- Meyer E.: *Eric Meyer o CSS – ovládněte kaskádové styly* (projekt 9)
- Cederholm D.: *Webdesign s webovými standardy*
- ...
- Další dobré zdroje můžete napísať do systému Moodle

# Ďakujem za pozornosť 😊

19

Priestor na vaše otázky