



*Съвременно уеб програмиране с PHP 2012*

# УЕБ-СТРАНИЦИ С HTML5 & CSS3

# Някои истини от днешната лекция

- ◎ Уеб-програмист, който не знае HTML&CSS, се нарича  
**безработен**
- ◎ HTML5 наистина не може да прави кафе и баница
- ◎ Но ако не започнете да го ползвате сега, ще сте закъснели

# От какво се състои една страница

- От структурирано съдържание: **HTML**
- Подредено и стилизирано: **CSS**
- С което потребителят взаимодейства: **JS**

**ВАЖНО:** Поведение и интерактивност (**clientside**) не е същото като динамично съдържание (**serverside**)!

# HTML в три слайда

(от миналата година)

# Синтаксис

## ◎ Тагове

Дълъг таг: `<a>http://www.google.com</a>`

Къс таг: `<br>` или `<br/>`

## ◎ Атрибути

`<a href="www.google.com">Google</a>`

## ◎ Entities

`&ampnbsp` (няма whitespaces!) `&ampreg` `&ampgt` `&amplt` ала и `&#8217;` (въпрос за грейпфрут)

## ◎ Коментари

`<!-- I have discovered a truly marvelous proof of this, which this margin is too narrow to contain -->`

# HTML документът



```
<!DOCTYPE html>
<html>
```

```
<head>
<title>THE VENERABLE ELEPHANT</title>
<meta charset="utf-8">
... разни други неща ...
</head>
```

```
<body>
  <h1>I AM THE PHP ELEPHANT</h1>
  <div id="mainContainer">
    <p>LOREM IPSUM BLABLAB IN LATINUM</p>
  </div>
</body>
```

```
</html>
```

# Елементи

## ◎ Семантични

- Картинки
  - 

- Хиперлинкове

- <a href="http://www.google.com">
- <a name="anchor"> || <a href="#anchor">

- Форми

## ◎ Структуриращи

- DIV & SPAN
- Table
- P, H1, ...

## ◎ Форматиращи =



i know  
<html>

<!--how to meet ladies-->

# CSS в още пет слайда

*Съвременно уеб програмиране с PHP 2012*

# Правилата на CSS

## ● Селектори

- класове

- p.**highlight** { color: yellow; }

- <p **class="highlight"**>

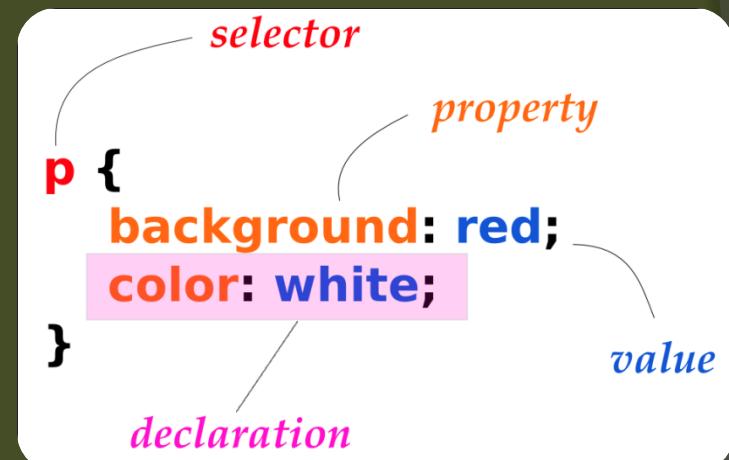
- идентификатори

- p.**#main** { color: yellow; }

- <p **id="main"**>

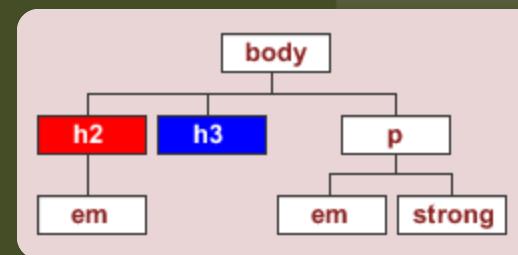
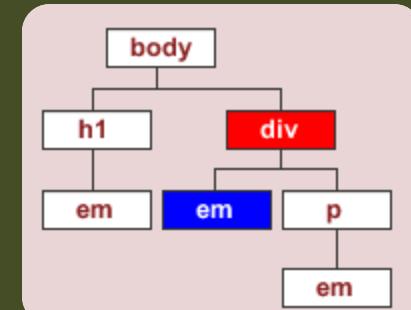
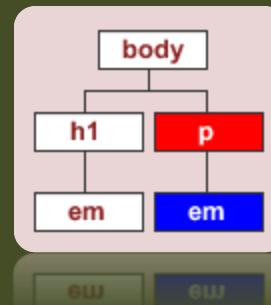
## ● Декларации

## ● Коментари - /\* Спомнихте ли си С? \*/



# Контекстните селектори

- Наследници
  - **p em** {color: blue;}
- Деца
  - **p > em** {color: blue;}
- Съседи
  - **h2 + h3** {color: red;}
- Универсални
  - **\*** {color: red;}
- По атрибути
  - **img[alt]** {border: 1px solid red;}
  - **img[alt="test"]** {border: 1px solid red;}
  - Важат за: 



# Къде стоят CSS?

## ◎ link

- `<link href="styles-nons4.css" media="all" type="text/css"/>`
- Един CSS – много HTML-и

## ◎ style

- `<style>  
 p.highlight { color: red; }  
</style>`

## ◎ inline

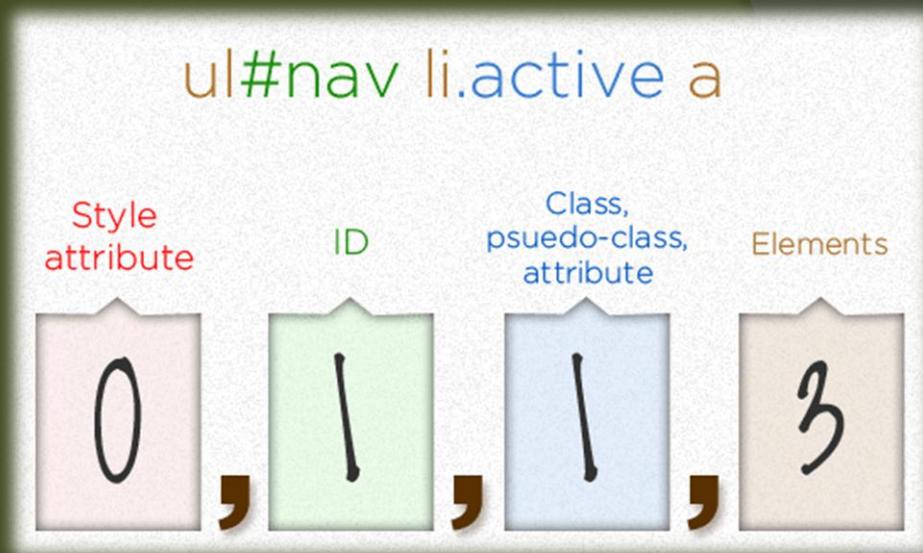
- `<p style="color: red;">`
- Единствено когато тестваме!

## ◎ @import

- `@import url('/css/layout.css');`  
`@import url('/css/color.css');`

# Прилагане на правило

- Каскаден принцип
- !important
- Ето защо ни е span
- „Наследяване“



- Ползвам два класа: `<p class="book important">`
- Или групирам правила:
  - `book, important { padding: 10px; }`
  - `important {color: red;}`
- !!!!! Мога да ползвам и SASS (<http://sass-lang.com/tutorial.html>)

# Box-model

- ◎ Една рисунка  
= хиляда думи
- ◎ Причина за  
85% от проблемите!

margin:30px

border:30px;

padding:30px;

width:80px

# И така, вече сме



## CSS НИНДЖИ

# HTML5 – що е то?

HTML

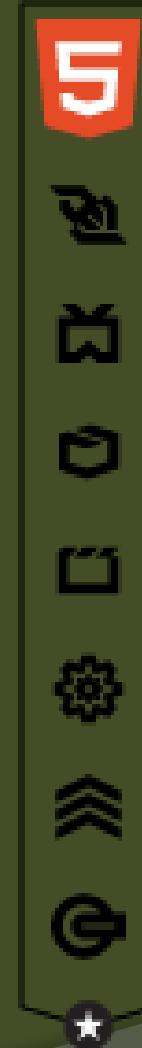


# Да изясним нещата:

- ◎ “HTML5” = HTML5 + CSS3 + JS
- ◎ Какво ни дава HTML5:
  - много нова семантика
  - мултимедия и нови типове данни
  - "умни" форми
  - наготово неща, за които ползвахме JS
  - много нови API за JS
- ◎ HTML5 е опит да се стандартизират усилията уеб-програмите да са "умни"

# Кои са тези JS API:

- Geolocation
- Web offline
- Local storage
- WebSQL / IndexedDB
- Drag&Drop
- File API
- Canvas API
- Web Workers
- Web Sockets



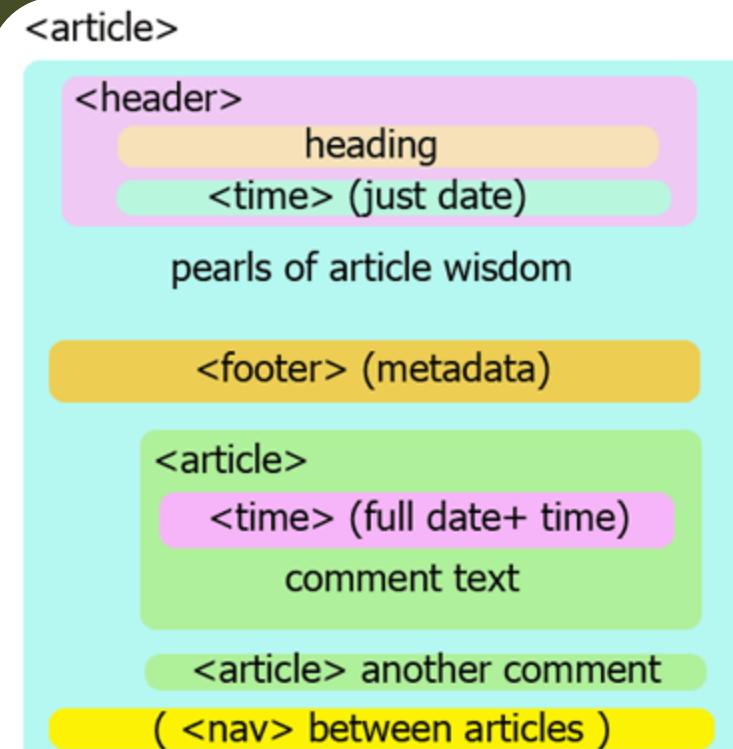
# НОВ СИНТАКСИС

# Малко, но важни промени:

- ◎ **<!DOCTYPE html>**
  - VS <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
- ◎ **<meta charset="utf-8">**
  - VS <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
- ◎ Таговете може да не се затварят
- ◎ Атрибутите официално и без кавички
- ◎ Валидатор: <http://validator.nu>

# НОВИ СЕМАНТИКИ

# Структурни елементи



!!! Препоръката е **section** да е логически обособена част от съдържанието, която е неразбираема без контекст, т.е. стои под **article**. За случая на картинката си ползваме **div**.

# По-надълбоко в семантиката

## ◎ Защо семантиката е важна?

- Програмна обработка на сайта ни
  - Например търсачките (+15 to SEO)
- Навигация за незрящи хора
- SemanticWeb – бъдещето на Интернет

## ◎ Microdata в HTML5

```
<div itemscope itemtype = "http://schema.org/Movie">
  <h1 itemprop="name">Avatar</h1>
  Director: <span itemprop="director">James Cameron</span>
  <a href =".. /movies/avatar-trailer.html" itemprop="trailer">Trailer</a>
</div>
```

# МУЛТИМЕДИЯ И СПЕЦИАЛНО СЪДЪРЖАНИЕ

# Видео & Аудио

```
<video width="320" height="240"  
       controls autoplay loop>  
<source src="pr6.mp4">  
<source src="pr6.webm">  
<source src="pr6.ogv">  
<track kind="subtitles" src="brave.en.vtt"  
srclang="en" label="English">  
</video>
```

# Видео & Аудио - 2

- Аудиото работи аналогично

```
<audio controls>
```

```
  <source src="song.ogg" type="audio/ogg" />
  <source src="song.mp3" type="audio/mpeg" />
```

Your browser does not support the audio element.

```
</audio>
```

- Различните браузъри поддържат различни кодеки!
- На youtube.com/html5 можете да пробвате HTML5 видео
- Можете да ембеднете Flash (object/embed) видео вътре в новия таг, така че стари браузъри пак да показват видеото ви

# MathML

```
<math display="block">
  <mrow>
    <msubsup>
      <mo>&Integral;</mo>
      <mn>0</mn>
      <mn>1</mn>
    </msubsup>
    <msup>
      <mi>x</mi>
      <mi>x</mi>
    </msup>
    <mo rspace="mediummathspace">&InvisibleTimes;</mo>
    <mo rspace="o">&DifferentialD;</mo>
    <mi>x</mi>
  </mrow>
</math>
```

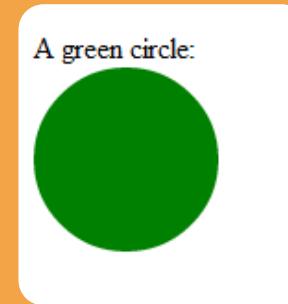
A 3D surface plot of the mathematical function y = x^x. The vertical axis (y) ranges from 0 to 1, with a peak at (1, 1). The horizontal axes (x and z) range from 0 to 1. The surface is smooth and curves upwards towards the point (1, 1, 1).

# SVG

```
<p>  
A green circle:
```

```
<svg>  
  <circle r="50" cx="50"  
        cy="50" fill="green"/>  
</svg>
```

```
</p>
```



!!! SVG (vector, markup) **VS** Canvas (bitmap, code)

# "УМНИ" ФОРМИ

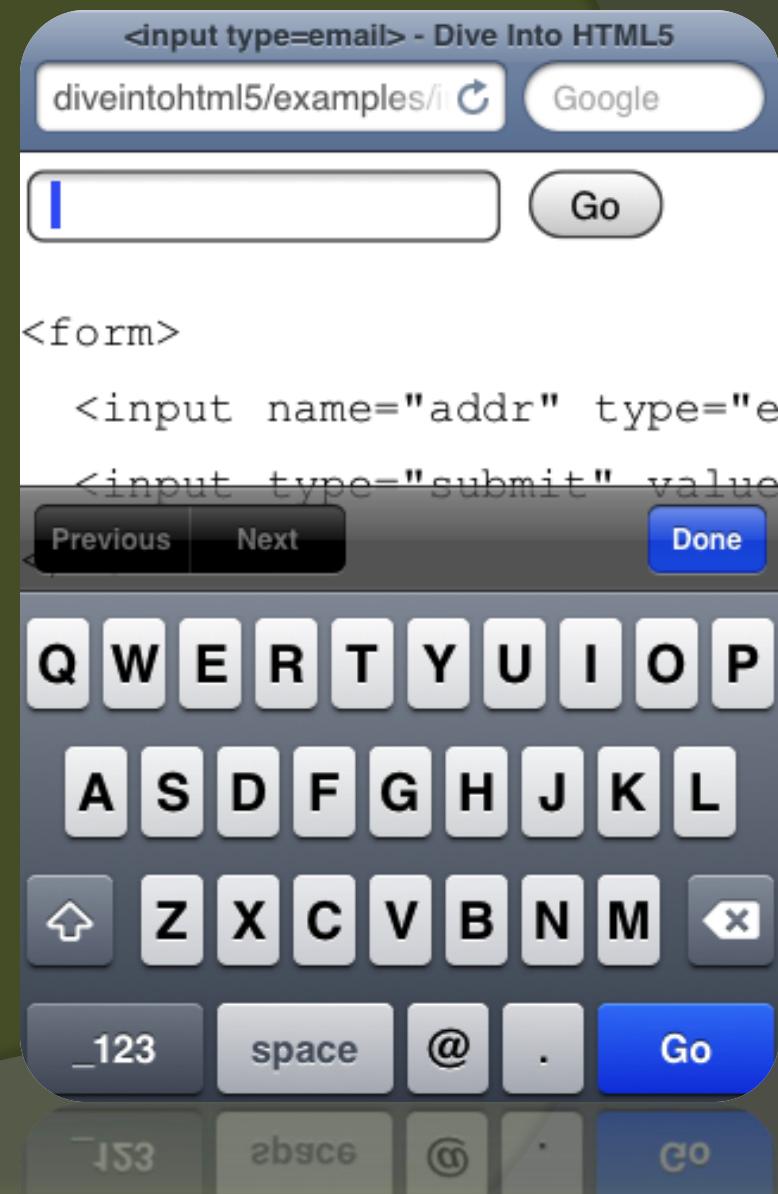
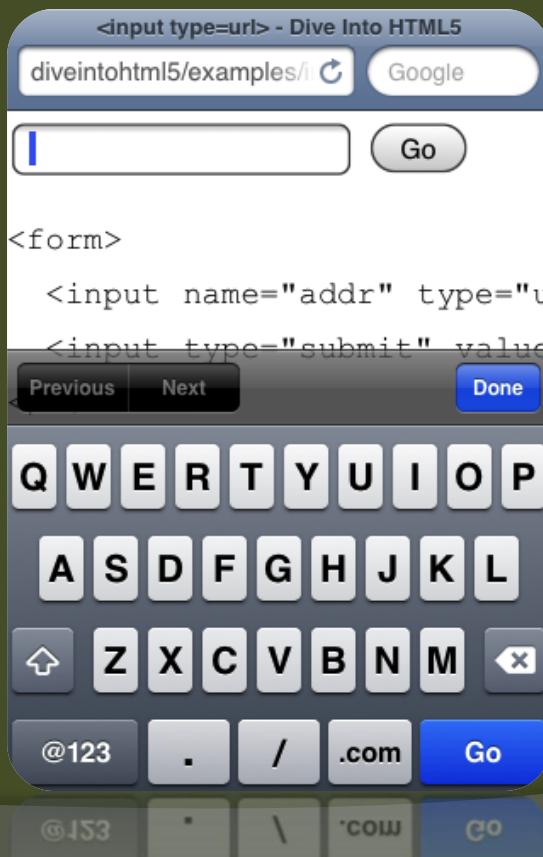
# Нови типове полета

- ◎ <input type="XXX">
- ◎ Визуализация наготово
- ◎ Валидация наготово
- ◎ Когато не се поддържат, се показва просто текстово поле (HTML5 = graceful degradation)



# Нови типове полета

- <input type="email">
- <input type="url">



# Нови типове полета

- ◎ <input type="number" min="0" max="10" step="2" value="6">

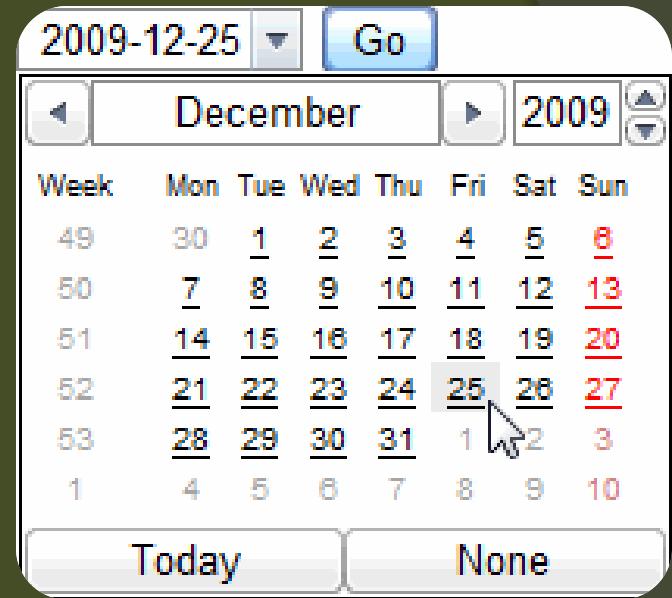


- ◎ <input type="range" ... >



# Нови типове полета

- <input type="date">
- <input type="color">



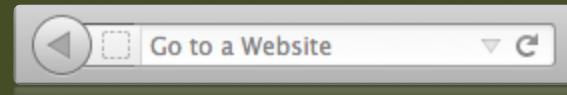
# Нови атрибути

◎ **autofocus**

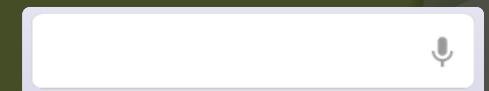
◎ **required**

◎ **placeholder**("Go to a Website")

◎ **novalidate** (на form!)



◎ **form** (вече не е нужно полетата да са вложени във формата)



◎ **x-webkit-speech** – за speech input

# Clientside autocomplete

- ◎ 1 картийка = 1000 думи

```
<input list="players">  
<datalist id="players">  
    <option value="Miroslav Klose">  
    <option value="Bastian Schweinsteiger">  
    <option value="Philipp Lahm">  
    <option value="Mesut Oezil">  
</datalist>
```



- ◎ Не се скалира много добре
- ◎ Може да зарежда от файл:

```
<input list="players">  
<datalist id="players" data="data.xml">
```

# ДРУГИ ЕКСТРИ

# Редактиране!!!!

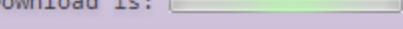
- Атрибутът **contenteditable**
- Можете да го сложите на **style** тага (заедно с **display:block**) и да позволите на потребителя да редактира CSS (**!!!OMGHACKS!!!**)
- Съхраняване на промяна (ако трябва)
  - AJAX
  - localStorage
- Тествайте тук: <http://jsbin.com/owavu3>

# И още

- Ruby (транскрипции)
- Soft line breaks
- Block-level links
- Интерактивни елементи: **details**
- Семантични inline елементи: **time** & **meter** (отново тази семантика!)
- Индикатор за прогреса: **progress**

こうかくきどうたい  
攻殻 機動隊

```
<meter min="0" max="100" low="40" high="90" optimum="100" value="91">A+</meter>
Your score is:  91/100

<progress>working...</progress>
Download is:  100/100

<progress value="75" max="100">3/4 complete</progress>
Goal is:  75/100
```

# CSS3 – МЕЧТАТА НА УЕБ-ПРОГРАМИСТА



# Прозрачности и цветове

- Вече не е нужно да ползваме Photoshop за прозрачност

**opacity: 0.5**



- Или ползваме новите модели за цветове
  - background: **rgba(255,0,0,0.5)**
  - background: **hsla(200,30%,70%,0.5)**

# Сенки и отразяване

- ◎ Сенки на текст

**text-shadow: 2px 4px 3px 0px black;**

- ◎ Сенки на div

**box-shadow: 2px 4px 3px 4px #ccc;**

- ◎ Отразяване

**-webkit-box-reflect: below 10px;**

Shadows



Web 2.0

Мер 5.0

# Градиенти

`background: linear-gradient(top, red 0%, orange 15%, yellow 30%, green 45%, blue 60%, indigo 75%, violet 100%);`

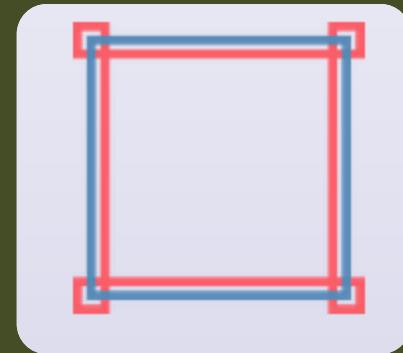


`background-image: radial-gradient(50% 50%, circle contain, rgba(171, 171, 171, 0.1), rgba(0, 0, 255, 0.4) 100%)`

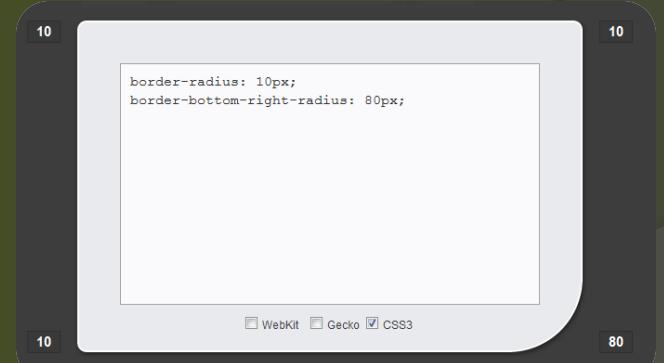


# Заоблените ъгли и други благинки

- ◎ **border-image:**  
`url(border_image_1.png) 20%`  
`stretch stretch;`



- ◎ **border-radius: 10px;**
- ◎ **border-bottom-right-radius: 85px;**



- ◎ **resize: horizontal;**

Try to resize the test `<p>` in both dimensions.

# Свойства за фона

## ◎ Multiple backgrounds

background: url(image1.png) right bottom,  
url(image2.png) center, url(image3.png)

## ◎ Позициониране

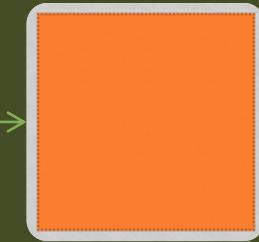
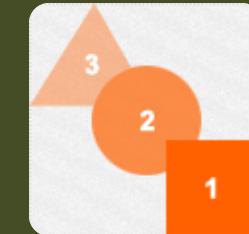
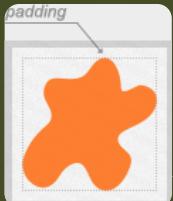
- **background-clip:** border-box

padding-box

- **background-origin:** border-box

content-box

- **background-size:** 50% 100%, 33% 33%



# Решения за layout

## ◉ Най-големият проблем на CSS2.1

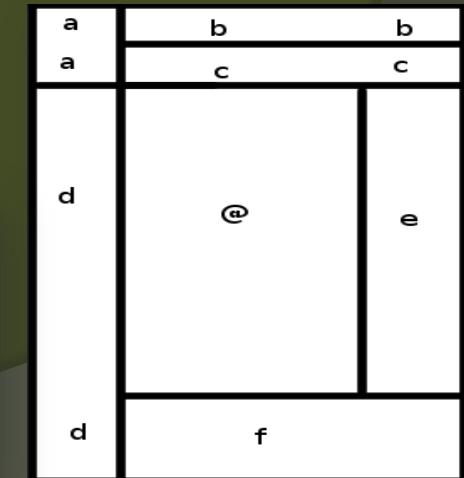
- Таблиците не се поддържат лесно
- **float** е тромаво и изисква много хакове
- Абсолютното позициониране е абсолютно

## ◉ Няма универсално решение за CSS3 ☺

- Flexbox – засега прието, но ограничено
- GridLayout – има го в IE10!
- AdvancedLayout (Templates)



```
body {  
    display: "abb"  
    "acc"  
    "d@e"  
    "dff";  
}  
  
#logo { position: a; }  
#header { position: b; }
```



(jQuery имплементация: <http://code.google.com/p/css-template-layout/>)

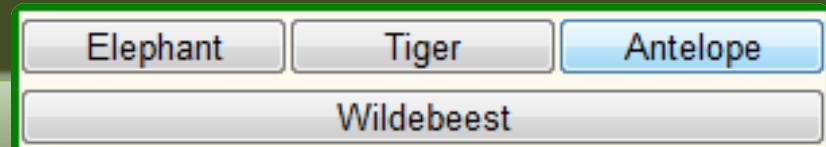
# Flexible Box Layout

- ul { display: **flexbox**; flex-pack: **distribute**; }



- #main { display: flexbox; flex-flow: row wrap; width: 300px; }  
button { flex: 80px 1; }

Първоначално всеки бутон заема по **80px**, а останалото място се разпределя по-равно, всеки бутон иска **1** част от свободното място.



- #main { display: flexbox; }  
#main > article { flex: 1; flex-order: 2; }  
#main > nav { width: 200px; flex-order: 1; }  
#main > aside { width: 200px; flex-order: 3; }



# Grid Layout

## ◎ Описваме мрежата в родителя

```
div#grid { display: grid;
```

**grid-columns:** 2fr 150px 1fr; /\* средната колона = 150px, останалите са разпределени 2:1 лява към дясна \*/

```
grid-rows: 150px 150px 150px; }
```

Задаваме по колко редове и колони заема всяко дете на родителя

<pre>grid-column: 1; grid-row: 1; grid-column-span: 2;</pre>	<pre>grid-column: 3; grid-row: 1; grid-row-span: 2;</pre>
<pre>grid-column: 1; grid-row: 2; grid-row-span: 2;</pre>	<pre>grid-column: 2; grid-row: 2;</pre>
<pre>grid-column: 2; grid-row: 3; grid-column-span: 2;</pre>	

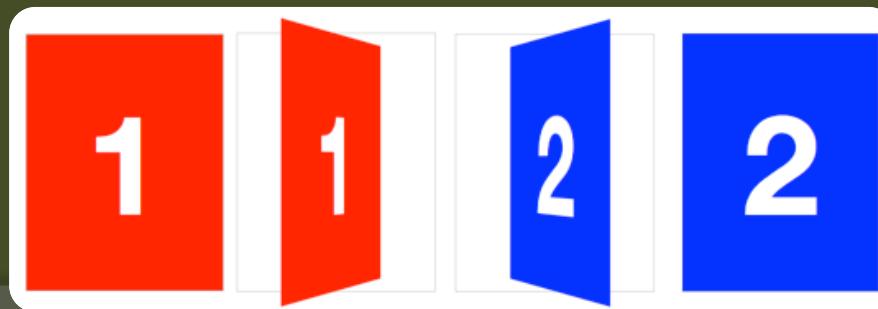
# Трансформации

- **transform: rotate(4deg) scale(0.772) skew(-6deg) translate(12px);**

- !rotate важи и за текст!

- Същото за 3D:

```
#transformed {  
    transform: rotateY(-40deg);  
    perspective: 800px; разстояние на гледане  
}
```



# Transitions

- Transition = анимиране на промяната на CSS свойствата

**transition: transform 2s ease-in-out 2s;**

- Можем да сложим transition на всяко свойство (например тук е transform)
- **Всяка** промяна на указаните свойства става анимация между предната и новата стойност
- Анимацията е някакъв вид интерполяция, но можем да я зададем ръчно:  
<http://matthewlein.com/ceaser/>

# АНИМАЦИИ

```
@keyframes resize {  
    0% { padding: 0;  
    }  
    50% { padding: 0 20px;  
           background-color:rgba(255,0,0,0.2);  
    }  
    100% { padding: 0 100px;  
            background-color:rgba(255,0,0,0.9);  
    }  
}  
  
#box:hover {  
    animation-name: resize;  
    animation-duration: 1s;  
    animation-iteration-count: 4;  
    animation-direction: alternate;  
    animation-timing-function: ease-in-out;  
}
```

Задаваме кадрите на анимацията. За разлика от transitions тук можем да имаме повече кадри с различни стойности за свойствата.

Асоцираме анимацията с елемента, тя ще се задейства при посочване и ще се превърти 4 пъти – т.е. по два пъти в двете посоки (заради alternate)

# Нови селектори

## ◎ Контекстно мачване на атрибути

a[href<sup>^</sup>="ftp"] a[href<sup>\*</sup>="zamunda"] a[href\$="php"]

## ◎ Нови псевдо класове

- :checked, :enabled, :disabled
- :focus
- :nth-child(an+b), :nth-last-child(an+b)
- :nth-of-type, :nth-last-of-type
- :only-child, :first-child, :last-child
- :only-of-type, :empty
- ::selection (селекцията на потребителя)

## ◎ Отрицание

a:not([href<sup>^</sup>="mailto"]) { opacity: 80%; }

```
.row:nth-child(even) {  
    background: #dde;  
}  
.row:nth-child(odd) {  
    background: white;  
}
```

Row 1

Row 2

Row 3

Row 4

Синтаксисът за псевдо-  
елементи тип :before  
вече е ::before, но се  
поддържа и старият

# Броячи

```
<dl>                                HTML  
<dt>term</dt>  
<dd>definition</dd>  
<dt>term</dt>  
<dd>definition</dd>  
</dl>
```

```
dl { counter-reset: term; }  
dt { counter-reset: definition; }  
dt:before { counter-increment: term;  
content: counter(term) ". " ; }  
dd:before { counter-increment: definition;  
content: counter(term) "." counter(definition) " " ; }
```

```
dl {  
    counter-reset: term;  
}  
dt:before {  
    counter-increment: term;  
    content: counter(term);  
    padding: 1px 2px;  
    margin-right: 0.2em;  
    background: #ffc;  
    border: 1px solid #999;  
}
```

1 term  
definition  
2 term  
definition

1. term  
1.1 definition  
2. term  
2.1 definition

# Шрифтове и текст

- ◎ Стандартизирано вмъкване на шрифт

```
@font-face {  
font-family: 'SecretTypewriterRegular';  
src: url('fonts/veteran_typewriter-webfont.eot');  
src: url('fonts/veteran_typewriter-webfont.woff') format('woff'),  
url('fonts/veteran_typewriter-webfont.ttf') format('truetype'); }  
h1 { font-family: "SecretTypewriterRegular", sans-serif; }
```

- ◎ Съкращаване  
**text-overflow: ellipsis;**



- ◎ Колони  
**column-count: 3;**  
**column-gap: 40px;**  
**column-rule: 4px outset #f0f;**

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure

dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet

doming id quod mazim placerat facer possim assum. Typi non habent claritatem insitam; est usus legentis in iis qui facit eorum claritatem. Investigations demonstraverunt lectores legere me lius quod ii legunt saepius.

# Data URI

- DataURI е стандартен начин за пакетиране на ресурс в URL



```
data:image/png;base64,iVB  
ORwoKGgoAAAANSUhEU  
gAAAXAAAAAE8CAYAAAD  
Zg+ooAAAAAXNSRoIArs4  
c6QAAAARnQU1BAACxjw  
v8YQUAAAA....
```

- ... полезно например за

```
<link rel="icon" src="data:image/png;base64,iv...">
```

- или пък

```
a:[href$="pdf"]::before { content: url(data:image/png;base64,...); }
```

- а между другото

```
<link rel="pingback" src="mycounter.php">
```

ни спасява от javascript броячи на посещения



# Media queries

- Прозрец между 600 и 900 пиксела

```
@media screen and (min-width: 600px) and (max-width: 900px) {  
    .class { background: #333; }  
}
```

- Зареждаме изцяло нов stylesheet

```
<link rel="stylesheet" media="screen and (max-width: 600px)"  
      href="small.css" />
```

- Проверка на ориентацията на iPad

```
<link rel="stylesheet" media="all and (orientation:portrait)"  
      href="portrait.css">  
  
<link rel="stylesheet" media="all and (orientation:landscape)"  
      href="landscape.css">
```

ЕКСПЕРИМЕНТИРАЙТЕ!

# А БРАУЗЪРИТЕ?

# Съвместимост

- ◎ Браузърите постоянно имплементират неща от HTML5&CSS3
- ◎ Някои CSS свойства все още са с префикси:
  - **-webkit-transform: rotate(45deg)**
  - **-o-transform: rotate(45deg)**
  - **-moz-transform: rotate(45deg)**
  - **-ms-transform: rotate(45deg)**
  - Можем да ползваме <http://leaverou.github.com/prefixfree/>
- ◎ Поне десет уебсайта постоянно следят какво се случва и са добър reference
- ◎ **Modernizr**

# Какво са polyfills

- ◎ Javascript скриптове, които осигуряват CSS3/HTML5/JS функционалност, където тя липсва
- ◎ ...или имплементират неща, които още не са твърдо в стандарта
- ◎ Можете да намерите добра селекция тук:  
<https://github.com/Modernizr/Modernizr/wiki/HTML5-Cross-browser-Polyfills>

# ЗА ФИНАЛ – МАЛКО ПОЛЕЗНИ ЧЕТИВА

## ◎ Tutorials

<http://www.net-kit.com/html5-css3-layout-tutorials-and-templates/>

<http://www.noupe.com/css3/css3-exciting-functions-and-features-30-useful-tutorials.html>

<http://www.netmagazine.com/features/top-10-css3-techniques>

<http://onwebdev.blogspot.com/2012/02/css-counters-tutorial.html>

<http://css-tricks.com/data-uris/>

## ◎ Transforms, Animations, Transitions:

<http://css3 bradshawenterprises.com/all/>



<http://www.joelambert.co.uk/flux/>

<http://www.the-art-of-web.com/css/css-animation/>

<http://24ways.org/2010/intro-to-css-3d-transforms>

## ◎ CSS3 общо

<http://www.css3files.com/> 

<http://www.css3explained.com/>

<http://www.css3.com/>

<http://www.css3.info/> - старо, но има ценни неща

<http://www.csstothe-point.com/>

<http://www.standardista.com/>

<http://htmldog.com/>

<http://css-tricks.com/snippets/css/>

<http://www.impressivewebs.com/>

<http://lea.verou.me/css3-secrets/#intro> - натиснете надясно 

<http://net.tutsplus.com/tutorials/html-css-techniques/the-30-css-selectors-you-must-memorize/>

## ◎ Места за пробване и генератори

<http://westciv.com/tools/gradients/> & the rest

<http://www.css3maker.com/index.html> 

<http://css3please.com/>

<http://border-radius.com/>

<http://brucelawson.co.uk/tests/html5-forms-demo.html>

## ◎ Проверка за поддръжка

<http://www.findmebyip.com/litmus> 

<http://caniuse.com/> 

<http://html5please.com/>

<http://html5test.com/> 

<http://www.quirksmode.org/css/contents.html>

[http://en.wikipedia.org/wiki/Comparison\\_of\\_layout\\_engines\\_\(HTML5\)](http://en.wikipedia.org/wiki/Comparison_of_layout_engines_(HTML5))

<http://www.modernizr.com/>

## ◎ HTML5 общо



<http://slides.html5rocks.com>



<http://www.html5rocks.com>

<http://html5doctor.com/>

<http://introducinghtml5.com/>



<http://diveintohtml5.info/>

<http://dev.w3.org/html5/html4-differences/>

<http://coding.smashingmagazine.com/2011/11/18/html5-semantics/>

<http://www.alistapart.com/articles/semanticsinhtml5/>

## ◎ Polyfills

<http://css3pie.com/>

<http://addyosmani.com/blog/writing-polyfills/>

<https://github.com/Modernizr/Modernizr/wiki/HTML5-Cross-browser-Polyfills>

<http://remysharp.com/2009/01/07/html5-enabling-script/>

## ◎ Галерия от примери

<http://www.html5laboratory.com/>

<http://html5demos.com/>

<http://html5gallery.com/>