



НИЕ ВЯРВАМЕ ВЪВ ВАШЕТО БЪДЕЩЕ

# HTML5 Style Guide and Coding Conventions

# Конвенция?

- Когато дадено нещо може да се направи по няколко начина, но ангажираната общност приеме само един от тях за правилен, се създава конвенция
- Програмистите са общност, която обича да създава конвенции за правилно и пригледно писане на код
- [https://developer.mozilla.org/en-US/docs/Mozilla/Developer\\_guide/Coding\\_Style](https://developer.mozilla.org/en-US/docs/Mozilla/Developer_guide/Coding_Style)
- [http://www.w3schools.com/html/html5\\_syntax.asp](http://www.w3schools.com/html/html5_syntax.asp)

# Как да валидирате кода?

- Онлайн валидатори:  
<https://validator.w3.org/>
- Extension-и за Brackets:  
<https://www.raymondcamden.com/2013/11/01/HTMLHint-extension-for-Brackets/>

- Linters

Това е наименование за програма или плъгин, който проверява синтаксиса на кода и маркира местата, където той не е валиден

# Индентация

```
16
17 ▼  <ul>
18      <li>asd
19          </li>
20      <li>dsa</li>
21          <li>
22              aed</li>
23      </ul>
24
```

```
17
18 ▼  <ul>
19      <li>asd
20          </li>
21      <li>dsa</li>
22 ▼      <li>
23          aed
24          </li>
25      </ul>
```

```
17
18 ▼  <ul>
19      <li>asd</li>
20      <li>dsa</li>
21          <li>aed</li>
22      </ul>
23
24
```

**Very bad!**

**Bad!**

**Good!**

Лошо индентирианият код, демонстрира неуважение, към този, който после ще трябва да го чете!

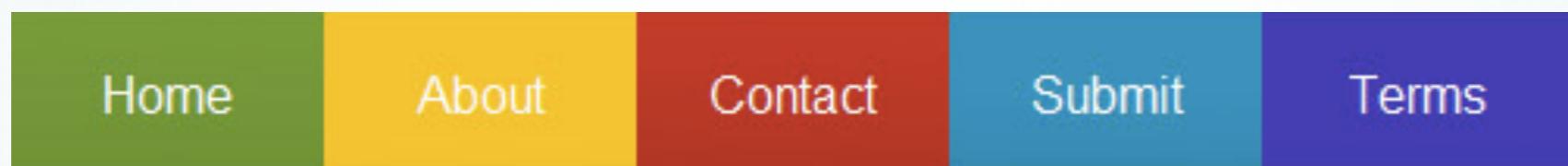
# Още за Brackets

- <https://www.youtube.com/watch?v=pC0oi-4dAZI>
- <https://www.youtube.com/watch?v=SBYb98IWo44>
- Полезни екстеншъни:
  - Beautify (задължително)
  - Strip trailing spaces
  - Autosave Files on Window Blur
  - Nu Markup Checker for Brackets

I will not write any more bad code  
I will not write any more bad code



# HTML навигация



# Що е то?

- Всеки сайт трябва да има навигация, ако съдържа повече от една логически обособени секции
- Целта на навигацията е бързо преминаване от секция в секция
- Навигацията е много важна за UX, затова трябва да се поставя на удобно място
- Част от навигацията са и препратките от дъното на страницата - "нагоре"

# Хипервръзки и котви

- Вече разгледахме хипервръзките (<a>), които могат да сочат към външен ресурс (към страница или файл на друг домейн)
- Когато искаме линка да сочи към елемент от текущата страница, използваме т. нар. "котви"
- Синтаксис:  
`<a href="#id_of_an_element">Go there</a>`
- Пример:  
`<a href="#contact">Контакти</a>`  
`<section id="contact"> .... </section>`

# Още малко за хипервръзките

- Прието е, когато линка сочи към външен ресурс, да отваряме този линк в друг прозорец/таб а не в същия
- Това се постига чрез атрибута `taret="_blank"`
- Прието е също, ако изписваме email адрес, той да бъде линк, който отваря прозорец за изпращане в мейл клиента ни (напр. Outlook или Thunderbird)
- Това става по следния начин:  
`<a href="mailto:emanolova@gmail.com">emanolova@gmail.com</a>`  
или даже: `<a href="mailto:emanolova@gmail.com">mail me!</a>`

# Упражнение

- ▶ Направете нов HTML документ. Можете да го създадете в папка hw04 във вашата папка от проекта с домашните
- ▶ Създайте топ навигация към следните секции:  
За мен | История | Проекти | Контакти
- ▶ Използвайте елемента nav за навигацията
- ▶ В самите секции сложете по едно заглавие h2 и по един параграф със текст
- ▶ Накрая, във всяка секция трябва да има линк за връщане нагоре (<a href="#top">нагоре</a>)

# HTML таблици

- Когато искаме да изобразим таблица, HTML ни дава специална група от елементи, предназначена за таблично представяне на данни
- Тези елементи са:

<code>&lt;table&gt;&lt;/table&gt;</code>	таблица
<code>&lt;tr&gt;&lt;/tr&gt;</code>	ред
<code>&lt;th&gt;&lt;/th&gt;</code>	клетка (table header)
<code>&lt;td&gt;&lt;/td&gt;</code>	клетка (table data)

- *Забележка: използването на таблици за изграждане на layout се счита за много лош стил на програмиране!*

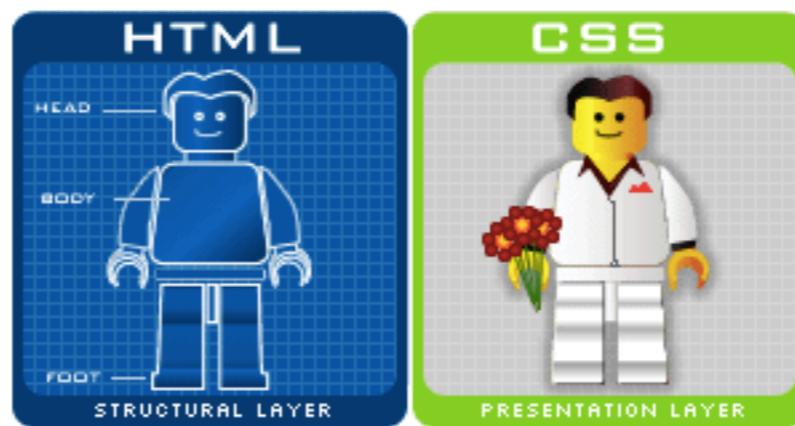
# Упражнение

- ▶ Направете таблица в секцията за контакти от предното упражнение
- ▶ Таблицата ще се състои от 3 реда и 2 колони
- ▶ Нека първата колона в тази таблица да е header
- ▶ На първи ред трябва да има съответно email, на втори: телефон, а на трети - линк към linkedIn

# Решение

```
29 <table>
30   <tr>
31     <th>Телефон:</th>
32     <td>+359 888 373 103</td>
33   </tr>
34   <tr>
35     <th>email:</th>
36     <td><a href="mailto:emanolova@gmail.com">emanolova@gmail.com</a></td>
37   </tr>
38   <tr>
39     <th>LinkedIn:</th>
40     <td><a href="https://bg.linkedin.com/in/evgenia-manolova-16696a39"
        target="_blank">Евгения Манолова</a></td>
41   </tr>
42 </table>
```

# CSS



# Как се ползва?

За да включим css стилизиране в нашата html страница, използваме един от следните подходи:

- с линк (препоръчително!):

```
<link rel="stylesheet" href="css/style.css">
```

- директно в html-а чрез **style** тагове (трябва да са в *head-a*):

```
<style>
  body {
    font-family: Helvetica;
    font-size: 12pt;
  }
</style>
```

- или просто *inline* като **style** атрибут на елемента (избягвайте!):

```
<body style="font-size: 25px">
```

# Синтаксис

```
selector1 {  
    property1: value1;  
    property2: value2;  
    ...  
}
```

```
selector2 {  
    property1: value1;  
    property2: value2;  
    ...  
}
```

*/\* това е коментар и не се чете от браузъра \*/*

# Правила

Конструкцията:

```
selector {  
    property: value;  
    ...  
}
```

се нарича правило.

- CSS файла представлява набор от правила
- Селекторите указват за кои елементи важи правилото
- Отделните свойства (*properties*) - задават стиловете, които ще се приложат на съответните елементи

# Селектори

- Чрез селектора указваме за кои елементи важат стиловете от правилото
- Например, имаме следната структура:

```
<body>
  <section id="main-section">
    <h1>My main section</h1>
  </section>
</body>
```
- Както вече знаем, за да зададем стил на body елемента, използваме body{...} записа, за стил на хединга - h1{...} записа.
- Т.е. **body**, **section** и **h1** са селектори

- Други селектори за същия код са
  - body section { ... }
  - body h1 { ... }
  - body section h1 { ... }
  - section h1 { ... }
  - #main-section { ... }
  - и дори: #main-section h1 { ... }
- Т.е. използването на id атрибута, ни дава селектор за този елемент
- [http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp)

# HTML **id** и **class** на елемента

- *id* и *class* са атрибути, които могат да ползват от всеки html element
- *id* атрибута задава уникален идентификатор на съответния елемент. Той трябва да е винаги различен за различните елементи

```

```

- *class* атрибута задава наименование на клас, към който html елемента принадлежи. Използва се за групиране на сходни елементи и идентифицира уникално група от елементи

```

```

# id и class в CSS-а

- HTML:

```
<p id="unique">First</p>
<p class="grouped">Second</p>
<p class="grouped">Third</p>
<p>Just a regular p without id or class</p>
```

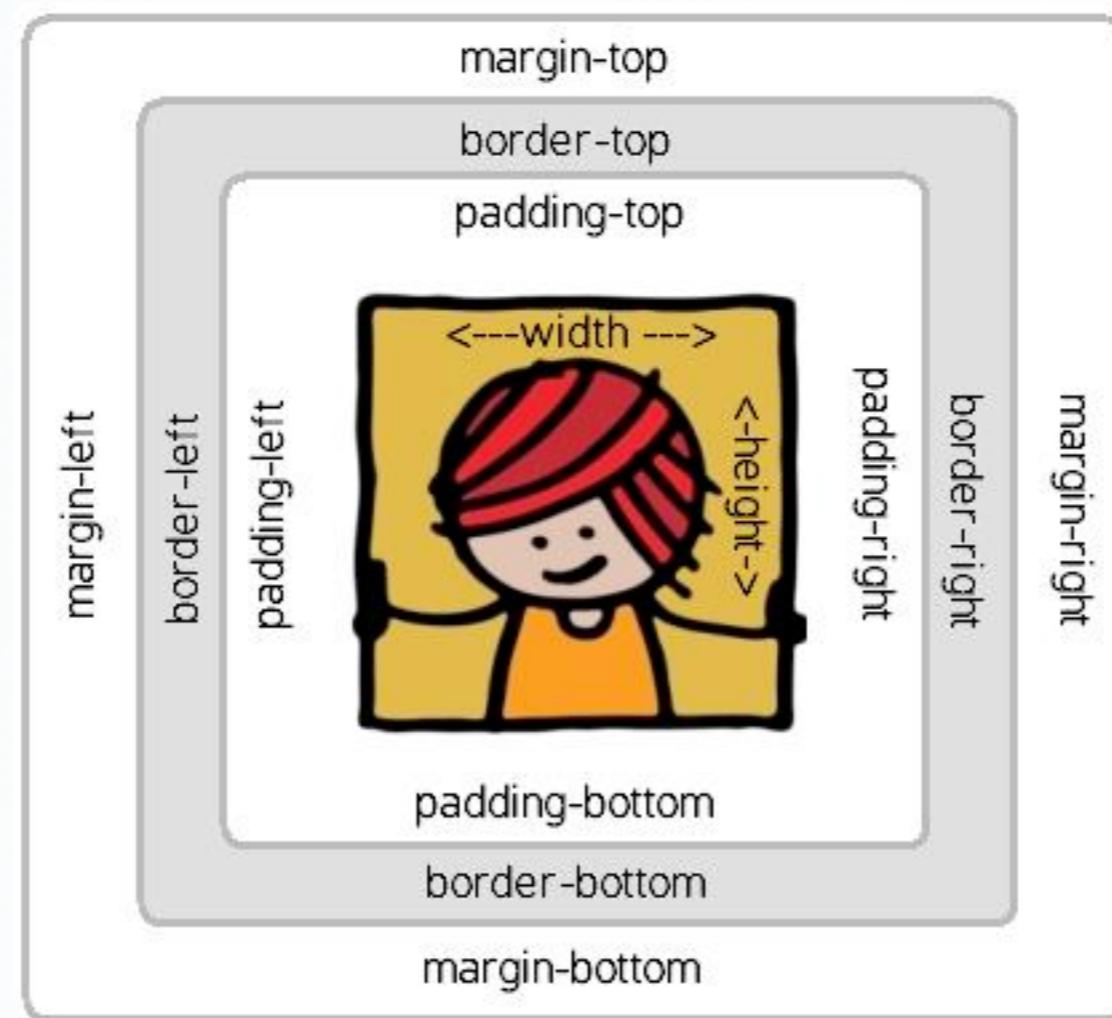
- CSS:

```
p { color: green; }
p.grouped { color: blue; }
p#unique { color: red; }
```

- резултат:

First  
Second  
Third  
Just a regular p without id or class

# The box model



# margin

- Това са отстъпите, които поставяме от *външната* страна на елемента.
- Т.е. разстоянието на което ще отместим настоящия елемент, от всички обграждащи го
- Може да се задава поотделно за всяка страна на елемента (горе, долу, ляво, дясно)
- или за всички измерения наведнъж (композитно property)

# Пример

```
h1 {  
    margin-top: 10px  
    margin-right: 20px;  
    margin-bottom: 30px;  
    margin-left: 40px;  
}
```

```
h1 {  
    margin: 10px 20px 30px 40px;  
}
```

# padding

- Това са отстъпите, които поставяме от *вътрешната* страна на елемента.
- Т.е. разстоянието на което ще отместим всички вложени елементи
- Може да се задава поотделно за всяка страна на елемента (горе, долу, ляво, дясно)
- или за всички измерения наведнъж (композитно property)

# Пример

```
h1 {  
    padding-top: 10px  
    padding-right: 20px;  
    padding-bottom: 30px;  
    padding-left: 40px;  
}
```

```
h1 {  
    padding: 10px 20px 30px 40px;  
}
```

# border

- Използва се за рисуване на рамка на елемента
- Тъй като има дебелина, също влияе на отстоянието от всички други елементи
- Може да се задава поотделно за всяка страна на елемента (горе, долу, ляво, дясно)
- пак е композитно property, но има повече компоненти (дебелина, цвят, стил)

# Пример

```
h1 {  
    border-top-width: 20px  
    border-right-width: 20px;  
    border-bottom-width: 20px;  
    border-left-width: 20px;  
    border-style: solid;  
    border-color: red;  
}
```

```
h1 {  
    border: 20px red solid;  
}
```

# Пример

<http://codepen.io/jenie/pen/EgLWEQ>

# Box sizing

# Концепцията за кутията

- Идеята за кутията идва от там че всички *html* елементи се представят като правоъгълници
- За съжаление обаче тази концепция не е съществувала още в началото, когато са били създадени първите *html* стандарти
- По онова време, фокусът е падал върху съдържанието (текстът) и не се е мислено много за обивката на съдържанието (кутията)
- Съответно когато се оразмеряват елементите (с *width* и *height*) всъщност се оразмерява само съдържанието, а не кутията
- Резултатът - елементите ни стават по-големи при ползването на отстъпи отвътре (*padding*) и се "чупи" layout-а на страницата

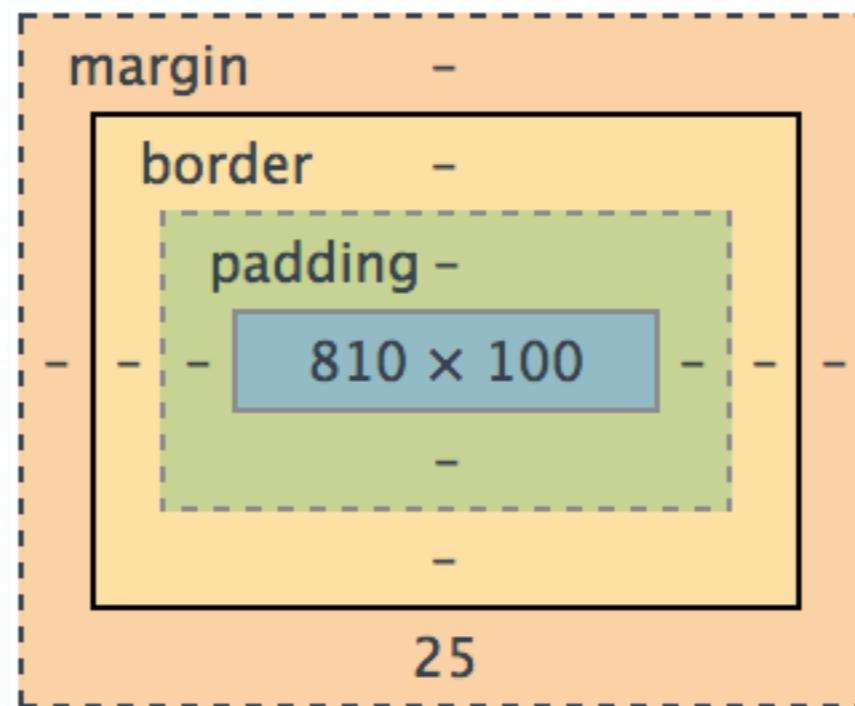


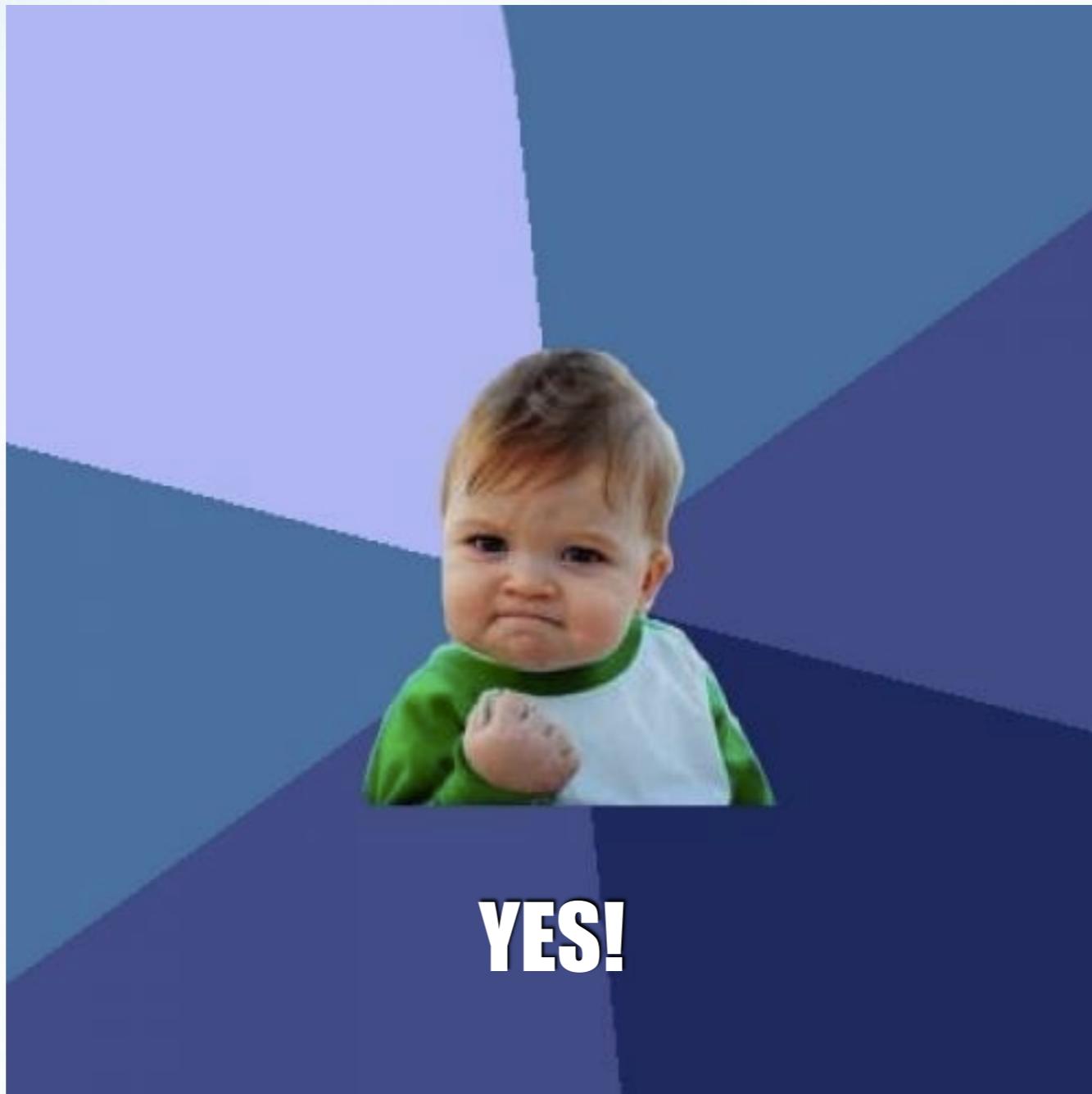
**SO WHAT YOU SAY IS  
THAT IF I SET BOTH WIDTH AND PADDING  
ON A DIV..**

**IT'S SIZE WILL BE UNEXPECTED?**

# Think out of the box!

\* { box-sizing: **border-box**; }





# Упражнение

- ▶ Добавете CSS към проекта от предишната задача
- ▶ За да създадете css файл в brackets, просто трябва да изберете **File -> new**, след това с клавишната комбинация **Ctrl+S**, да го запазите в папката на проекта (hw04).  
Именувайте го: **style.css**
- ▶ В html документа трябва да линк-нете новосъздадения файл
- ▶ В новосъздадения css файл напишете правила, които да направят навигацията цветна и оразмерена (както е на картинката от началото на презентацията)

# Пример

<http://codepen.io/jenie/pen/QKAdmb?editors=1100>

CSS floating  
CSS презентиране (display)  
CSS позициониране

# Плаващи и преливащи елементи

- **float** - кара елементите да се нареждат един до друг вляво или вдясно, като те наместват според това дали има достатъчно място по хоризонталата
  
- **.left { float: left; }**  
**.right { float: right; }**
- **overflow** е css property, което се грижи за това, дали съдържанието на даден елемент прелива.  
Стойности: **hidden, auto, scroll**
- [http://www.w3schools.com/css/css\\_float.asp](http://www.w3schools.com/css/css_float.asp)

# Примери

<http://codepen.io/jenie/pen/EgLWEQ>

<http://codepen.io/jenie/pen/yajrdq>

# block vs inline

- Както вече казахме, *html* елементите могат да се представят като правоъгълници
- Тези правоъгълници се разполагат по 2 основни начина на экрана: или се подреждат един след друг по хоризонтала, или един след друг по вертикала
- Правоъгълниците, които имат блоков характер (*блокови елементи*) се подреждат един под друг (заемат 100% от ширината на экрана)  
Пример: **div, p, h1, h2, .., h6, ul** и тн.
- Правоъгълниците с *inline* характер (*inline елементи*) се подреждат един до друг, стига да има място. Те заемат толкова пространство, колкото е тяхното съдържание  
Пример: **img, span, a**

# CSS display на HTML елементите

- Благодарение на CSS вече дори *native* поведението на *html* елементите може да бъде променено. Като например поведението им като блокови или *inline* елементи
- Това става благодарение на пропъртито *display*
- <https://css-tricks.com/almanac/properties/d/display/> (да се чете заедно със следващия линк по отношение на *display-table*)
- <http://colintoh.com/blog/display-table-anti-hero>
- <http://j4n.co/blog/Creating-your-own-css-grid-system>

# CSS positioning

- fixed
- absolute
- relative
- static
- z-index
- [http://www.w3schools.com/css/css\\_positioning.asp](http://www.w3schools.com/css/css_positioning.asp)

# **Let's code !!!**



# Задача: електронна визитка

Направете страница, която да има:

1. Следните секции:
  - A. Представяне (основна информация за вас: име, снимка, настояща професия, мото или настояща мисия)
  - B. Портфолио (ваши завършени проекти и постижения с които се гордеете)
  - C. Контакти
2. Навигация между секциите
3. Хедър и футър

# Пояснения към задачата:

- Използвайте валиден HTML синтаксис и семантика
- Секциите трябва да са разположени една под друга, а навигацията да е отстрани или отгоре. Всяка секция трябва да има най-отдолу линк, който да ви връща отново горе
- Ако нямате никаква идея какво да сложите в Портфолио секцията, може да я заместите със секция Резюме (кратко професионално резюме: къде сте учили по години; какви квалификации сте придобили; кога, къде и какво сте работили)
- Можете да ползвате HTML елемента за таблици: `<table>` където намерите за уместно

# Примери:

STEVENBROWN

Home Portfolio Blog Resume Contact



## WEB DESIGNER

Hello there! My name's Steven and I'm a web designer from London, UK. I believe in clean and minimalistic design and also think it is very important to make a web site compatible with all modern devices, like phones and tablets.

Feel free to check out my work or view my resume. I am currently available for hire, so if I sound fit for your next project, don't hesitate to contact me!

I'm a social person. Or at least that's what I keep telling myself. Either way, you can find me at the following places:

 @stevenbrown

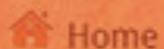
 Steven Brown

 Steven Brown

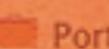
 me@brown.com

# WILLIAM DAVIS

php programmer, html developer and just coffee lover



Home



Portfolio



About



Contact

## HERE IS ME:



## SOME WORDS ABOUT:

Name: William Davis

Birthday: May 30, 1990

Address: 10 South Park Avenue  
San Francisco, CA 94108  
USA

Feel free to contact me.

Email: info@williamdavis.com

Phone: +123 456 789

Website: williamdavis.com

## LET'S GET SOCIAL!

You can find me in these social places. Don't be shy, contact me.



© williamdavis, 2012. All rights reserved

# Amanda Smith

v f in g+ o



## MY PROFILE



### I'm England Developer and Business Lady :)

I'm Graphic designer and occasional author who specialises in designing brand identities. People hire me because I help make their businesses more profitable.

Perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae.

NAME Amanda Smith

AGE 25

ADDRESS England, London, Street 27, UK

E-MAIL amandasmith@mail.com

PHONE +1 256 254 84 56

FREELANCE Available



## RESUME



## MY PORTFOLIO



## CONTACT INFO



#### MY SKILLS

HTML5



CSS3



jQuery



Photoshop



PHP



Illustrator



#### About me

Nam pharetra mollis erat quis pulvinar. Donec ut metus in libero elementum pulvinar vitae quis sapien. Proin nec diam purus, at imperdiet diam. Vivamus auctor, purus et vestibulum ullamcorper, tellus dolor iaculis risus, eu interdum urna est at tortor.

*Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi volutpat sollicitudin neque sit amet fermentum. Morbi mollis, ante*

Nam pharetra mollis erat quis pulvinar. Donec ut metus in libero elementum pulvinar vitae quis sapien. Proin nec diam purus, at imperdiet diam. Vivamus auctor, purus et vestibulum ullamcorper, tellus dolor iaculis risus, eu interdum urna est at tortor..

#### Services

Webpage layout design (Photoshop PSD)

Third-party graphic layout to webpage

Custom Graphic Design Services

HTML email design

Website maintenance, update and promotion

Logo design (logotype design)

Social Profiles



# ANDERSON SMITH

GRAPHIC & WEB DESIGNER

I like to make cool and creative designs. My design stash is always full of refreshing ideas. Feel free to take a look around my Vcard.

Name	Anderson smith
Birthday	March 13, 1988
Address	Melbourne Victoria 3000 Australia
Email	Anderson.smith@gmail.com
Phone	+123 456 789 111
Website	www.Andersonsmith.com



Profile



Resume



Portfolio



Contact

Copyright © 2013 John smith

o

f g s in @ v

 PROFILE  



**SARAH JEFFERSON**  
Designer

Name: Sarah Jefferson  
Date of birth: April 13, 1988  
Address: San Francisco, CA 94102, US  
Phone: +123-456-789-111  
Email: wpspacetemplates@gmail.com  
Website: <http://themeforest.net/user/wpspace>

Lorem ipsum dolor sit amet, consectetur adipisicing 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.

	RESUME	<input checked="" type="checkbox"/>
	PORTFOLIO	<input checked="" type="checkbox"/>
	CONTACTS	<input checked="" type="checkbox"/>

WP Space © 2013 All Rights Reserved

# John Doe

Web Developer

## About Me



Lorem ipsum dolor sit amet, consectetur adipisicing 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.

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. Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae.

Facebook

Twitter

LinkedIn

Flickr

Vimeo

Dribbble



© Copyright 2011. Squares by Joel Bessada

HOME

RESUME

SKILLS

PORTFOLIO

BLOG

CONTACT



# Hi, I'm John Rex!

Web Designer & Developer

Founder of DotRex. Professional UI/UX designer and web developer based on London. Sometimes works as a freelancer.

email: john@dotrex.co / phone +123 456 789

111



## RESUME

[Download my resume](#)

[Get in touch](#)

### Education

#### Graphic Design

St. Patrick University (2 Years Course) - Graduation May 2013

Lorem ipsum is simply dummy text of the printing and typesetting industry. Lorem ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book.

#### Master of Web Design

St. Patrick University (2 Years Course) - Graduation May 2013

**John Smith**  
developer & designer

about

resume

portfolio

contact



<http://wp-themes.pro/templates/circlus/>

**SWIFTACADEMY**

НИЕ ВЯРВАМЕ ВЪВ ВАШЕТО БЪДЕЩЕ

Front-End Development

# Въпроси?

# Примери

<http://zenlabs.pro/courses/sa-fe-04/lessons/lesson4/examples.zip>

# Домашно

<http://zenlabs.pro/courses/sa-fe-04/lessons/lesson4/hw-task.html>