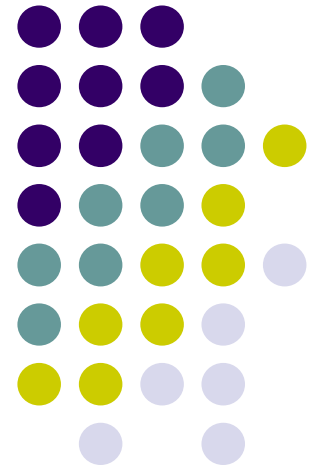
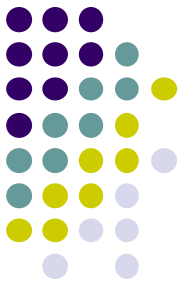


# Προγραμματισμός & Συστήματα στον Παγκόσμιο Ιστό

Η «γλώσσα» HTML  
& CSS





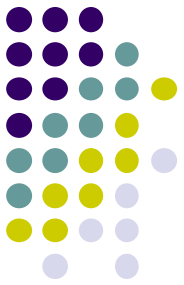
# Cascading Style Sheets

- Η αρχική HTML δεν προοριζόταν για την custom μορφοποίηση των περιεχόμενων μιας ιστοσελίδας
  - Περιγραφή (markup) των περιεχόμενων
  - Οι ετικέτες και ιδιότητες μορφοποίησης (π.χ. <font>) εμφανίστηκαν στην HTML 3.2
  - Μεγάλος πονοκέφαλος για τους διαχειριστές μεγάλων ιστοτόπων με πολλές ιστοσελίδες.
- Η λύση: CSS
- Καθορίζουν το style, τη δομή και τη θέση των στοιχείων μέσα στο αρχείο
- Κάνουν εφικτό το διαχωρισμό της εμφάνισης των δεδομένων και των ίδιων των δεδομένων
- Ο valid τρόπος για μορφοποίηση!

# Τρόποι εισαγωγής style sheet

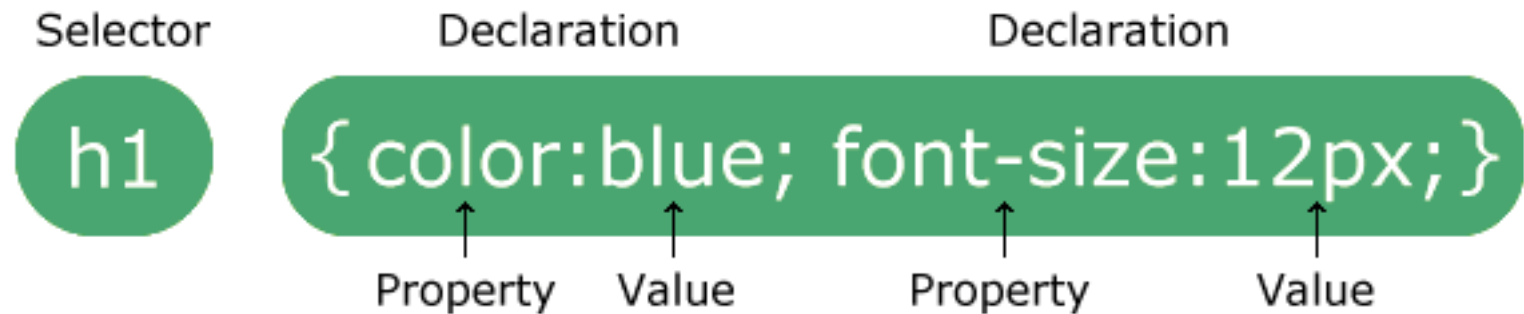


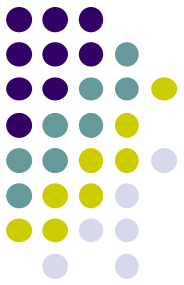
- Inline styles
  - Καθορίζει το style για συγκεκριμένα elements
  - Χρήση του attribute “style” μέσα σε tag
  - Μπορεί να καθορίσει πολλά properties
- Internal styles sheets
  - Ορίζεται μέσα στο <head> με το tag <style>
  - Επηρεάζει τα elements στα οποία αναφέρεται
- External style sheets
  - Δημιουργία ξεχωριστού αρχείου css με το οποίο συνδέεται το html αρχείο
  - Ιδανικό όταν το ίδιο style sheet εφαρμόζεται σε περισσότερα από ένα αντικείμενα



# CSS Syntax

- Σύνταξη
  - selector {property1: value1; property2: value2}
    - selector : html tag – {body, p, h1,...}, class name, id ...
    - property : το χαρακτηριστικό που θέλουμε να αλλάξουμε
      - color, font-family, text-align





# Inline styles

`<p>This text does not have any style applied to it.</p>`

`<p style = "font-size: 20pt">This text has the <em>font-size</em> style applied to it, making it 20pt.</p>`

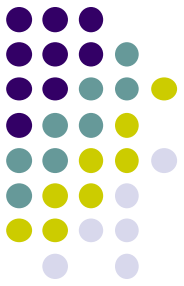
`<p style = "font-size: 20pt; color: #0000ff">This text has the <em>font-size</em> and <em>color</em> styles applied to it, making it 20pt. and blue.</p>`

Μονάδα μέτρησης pt (point): Ισούται με το 1/72 της ίντσας

[View page](#)

# Internal style sheets

1



```
<head>
```

```
  <style type = "text/css">
```

```
    em    { background-color: #8000ff; color: white }
```

```
    h1    { font-family: arial, sans-serif }
```

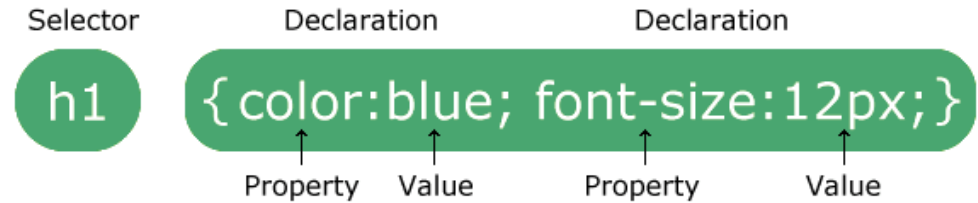
```
    p     { font-size: 14pt }
```

```
  </style>
```

```
</head>
```

[View page](#)

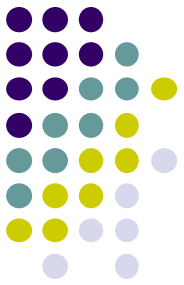
# CSS Selectors



- **Element selector**
  - Όλα τα στοιχεία με tag `<element>`
- **ID selector**
  - Όλα τα στοιχεία με tag attribute `id="id"`
- **Class selector**
  - Όλα τα στοιχεία με tag attribute `class="class"`
- **Universal selector**
  - Όλα τα στοιχεία
- **Grouping selector**
  - Πολλαπλά elements με την ίδια μορφοποίηση

Ο selector είναι το βασικό στοιχείο της CSS που μας επιτρέπει να επιλέγουμε και να μορφοποιούμε συγκεκριμένα HTML στοιχεία, κάνοντάς τα να φαίνονται και να συμπεριφέρονται με τον τρόπο που θέλουμε.

# CSS Selectors



- Element selector

```
<!DOCTYPE html>
<html>

<head>

  <style>
    p {
      text-align: center;
      color: red;
    }
  </style>

</head>

<body>

  <h1>Test!</h1>
  <hr>
  <p>Hello World!</p>
  <p>This paragraph is also affected by the style </p>
  While this text is not.

</body>
</html>
```

Yellow arrows point from the `p` selector in the `<style>` block to the `<p>` tags in the `<body>` block.

## Test!

Hello World!

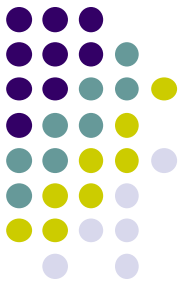
This paragraph is also affected by the style

While this text is not.

The element selector selects HTML elements based on the element name.



# CSS Selectors



- ID selector

```
<!DOCTYPE html>
<html>

<head>

  <style>
    #para1 {
      text-align: center;
      color: red;
    }
  </style>

</head>

<body>

  <p id="para1">Hello World!</p>
  <p>This paragraph is not affected by the style </p>

</body>
</html>
```

Two yellow arrows originate from the CSS code. One arrow points from the '#para1' selector in the style block to the 'id="para1"' attribute in the first paragraph of the body. The second arrow points from the same style block to the second paragraph, which does not have an ID attribute.

Hello World!

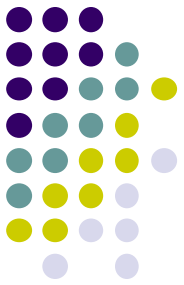
This paragraph is not affected by the style.

The ID selector matches an element based on the value of the element's id attribute. In order for the element to be selected, its id attribute must match exactly the value given in the selector.

Syntax

**#id\_value** { style properties }

# CSS Selectors



- Class selector

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .center_red {
      text-align: center;
      color: red;
    }
  </style>
</head>
<body>
  <h1>Test!</h1>
  <hr>
  <p class = "center_red">Hello World!</p>
  <p>This paragraph is not affected by the style </p>
  <h2>This heading is not affected.</h2>
  <h2 class = "center_red">This heading is affected.</h2>
</body>
</html>
```

## Test!

Hello World!

This paragraph is not affected by the style

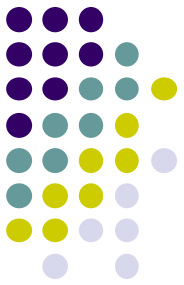
**This heading is not affected.**

**This heading is affected.**

The CSS class selector matches elements based on the contents of their class attribute.

Syntax

- `class_name { style properties }`



# CSS Selectors

- Universal selector

```
<!DOCTYPE html>
<html>

<head>

  <style>
    * {
      text-align: center;
      color: red;
    }
  </style>

</head>

<body>

  <h1>Test!</h1>
  <hr>
  <p>Hello World!</p>
  <p>This paragraph is not affected by the style </p>
  <h2>This heading is not affected.</h2>
  <h2>This heading is affected.</h2>

</body>
</html>
```

**Test!**

Hello World!

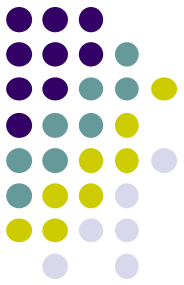
This paragraph is not affected by the style

**This heading is not affected.**

**This heading is affected.**

The universal selector (\*) selects all HTML elements on the page.

# CSS Selectors



- Grouping selector

```
<!DOCTYPE html>
<html>

<head>

  <style>
    em,a{
      padding-right:10px;
      padding-left:10px;
      color: white;
      background-color: black;
    }
  </style>

</head>

<body>

  <h1>Test!</h1>
  <hr>
  <p>Hello World!</p>
  <p>This <em>paragraph</em> is affected by the <a
href="#">style</a> specified in the head section.</p>

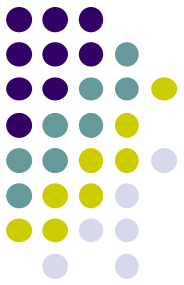
</body>
</html>
```

## Test!

Hello World!

This **paragraph** is affected by the **style** specified in the head section.

When multiple selectors share the same declarations, they can be grouped together into a comma-separated list.

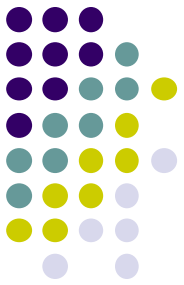


# Μερικά παραδείγματα

```
<style type = "text/css">
  a { text-decoration: none }
  a:hover { text-decoration: underline;
            color: red; background-color: #ccffcc }
  li em    { color: red;
            font-weight: bold }
  ul       { margin-left: 75px }
  ul ul    { text-decoration: underline;
            margin-left: 15px }
</style>
```

Px (pixel): Είναι σχετικό για κάθε σύστημα

[View page](#)



# Pseudo-classes

```
<!DOCTYPE html>
<html>
<head>
<style>
/* unvisited link */
a:link {
  color: red;
}

/* visited link */
a:visited {
  color: green;
}

/* mouse over link */
a:hover {
  color: hotpink;
}

/* selected link */
a:active {
  color: blue;
}
</style>
</head>
<body>

<h2>Styling a link depending on state</h2>

<p><b><a href="default.asp" target="_blank">This is a
link</a></b></p>
<p><b>Note:</b> a:hover MUST come after a:link and a:visited
in the CSS definition in order to be effective.</p>
<p><b>Note:</b> a:active MUST come after a:hover in the CSS
definition in order to be effective.</p>

</body>
</html>
```

## Styling a link depending on state

### This is a link

**Note:** a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective.

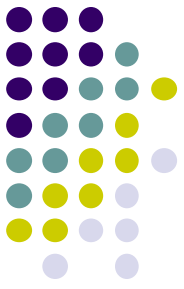
**Note:** a:active MUST come after a:hover in the CSS definition in order to be effective.

In CSS, a pseudo-class selector targets elements depending on their state rather than on information from the document tree. For example, the selector a:visited applies styles only to links that the user has already followed.

syntax

```
selector:pseudo-class {
  property: value;
}
```

[https://www.w3schools.com/css/tryit.asp?filename=trycss\\_pseudo-class\\_links](https://www.w3schools.com/css/tryit.asp?filename=trycss_pseudo-class_links)



# External style sheets

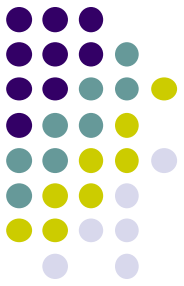
<head>

<title>External Style Sheets</title>

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

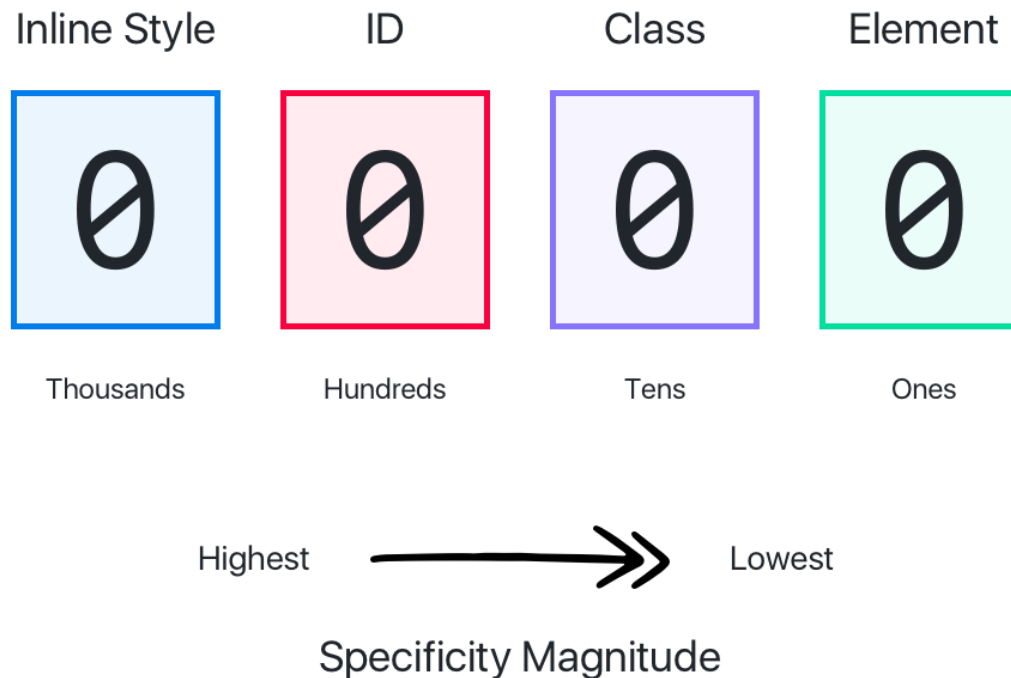
</head>

[View page](#)



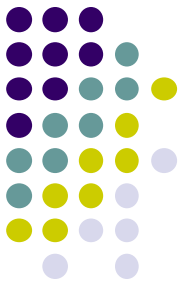
# Προτεραιότητα

- CSS Specificity:
  - Το μεγαλύτερο «σκορ» κερδίζει





# Προτεραιότητα



```
<!DOCTYPE html>
<html>
<head>
<style>
    .bc{
        color:black;
    }
    #myid {
        color: red;
    }
</style>

<link rel="stylesheet" href="mystyle.css">
</head>

<body>
<h1>Some heading</h1>
<h1 class="bc">This is a heading</h1>
<h1 class="bc" id="myid">And another</h1>
<h1 class="bc" id="myid" style="color:green;">Green heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

Some heading

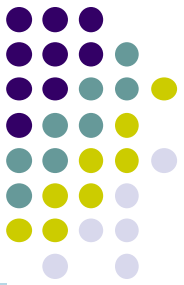
This is a heading

And another

Green heading

This is a paragraph.

# Προτεραιότητα



```
<!DOCTYPE html>
<html>
<head>
<style>
  .bc{
    color:black;
  }
  #myid {
    color: red;
  }
</style>

<link rel="stylesheet" href="mystyle.css">
</head>

<body>
<h1>Some heading</h1>
<h1 class="bc">This is a heading</h1>
<h1 class="bc" id="myid">And another</h1>
<h1 class="bc" id="myid" style="color:green;">Green heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

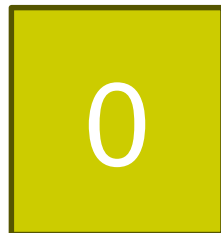
**Some heading**

**This is a heading**

**And another**

**Green heading**

This is a paragraph.



inline



id

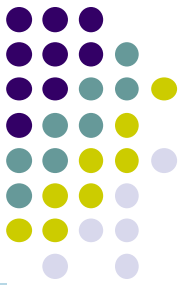


class



element

# Προτεραιότητα



```
<!DOCTYPE html>
<html>
<head>
<style>
  .bc{
    color:black;
  }
  #myid {
    color: red;
  }
</style>

<link rel="stylesheet" href="mystyle.css">
</head>

<body>
<h1>Some heading</h1>
<h1 class="bc">This is a heading</h1>
<h1 class="bc" id="myid">And another</h1>
<h1 class="bc" id="myid" style="color:green;">Green heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

**Some heading**

**This is a heading**

**And another**

**Green heading**

This is a paragraph.

class="bc"

0

inline

0

id

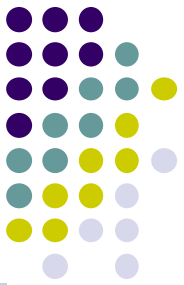
1

class

0

element

# Προτεραιότητα



```
<!DOCTYPE html>
<html>
<head>
<style>
  .bc{
    color:black;
  }
  #myid {
    color: red;
  }
</style>

<link rel="stylesheet" href="mystyle.css">
</head>

<body>
<h1>Some heading</h1>
<h1 class="bc">This is a heading</h1>
<h1 class="bc" id="myid">And another</h1>
<h1 class="bc" id="myid" style="color:green;">Green heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

**Some heading**

**This is a heading**

**And another**

**Green heading**

This is a paragraph.

class="bc" id="myid"

0

1

1

0

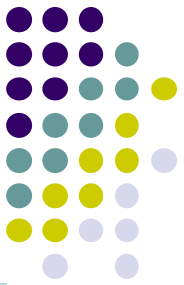
inline

id

class

element

# Προτεραιότητα



```
<!DOCTYPE html>
<html>
<head>
<style>
  .bc{
    color:black;
  }
  #myid {
    color: red;
  }
</style>

<link rel="stylesheet" href="mystyle.css">
</head>

<body>
<h1>Some heading</h1>
<h1 class="bc">This is a heading</h1>
<h1 class="bc" id="myid">And another</h1>
<h1 class="bc" id="myid" style="color:green;">Green heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

**Some heading**

**This is a heading**

**And another**

**Green heading**

This is a paragraph.

style=".." class="bc" id="myid"

1

inline

1

id

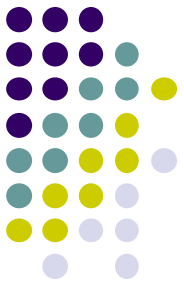
1

class

0

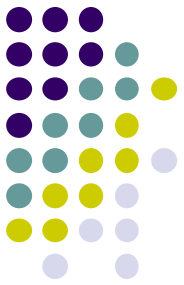
element

# CSS Units



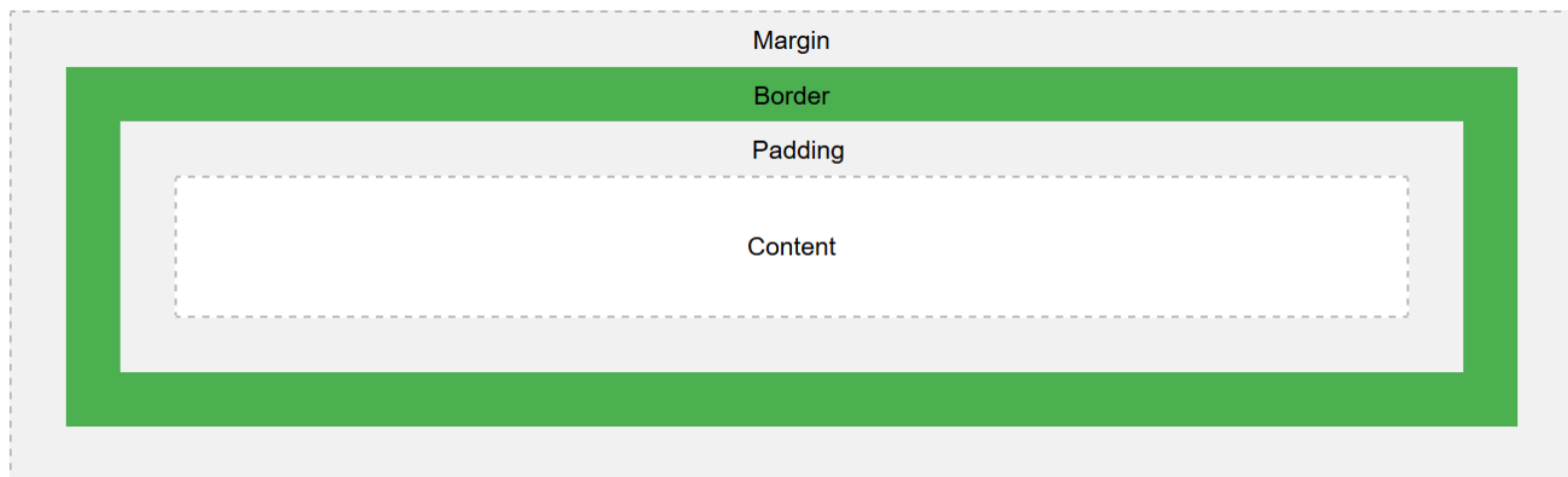
Unit	Description
%	percentage
in	inch
cm	centimeter
mm	millimeter
em	1em is equal to the current font size. 2em means 2 times the size of the current font. E.g., if an element is displayed with a font of 12 pt, then '2em' is 24 pt. The 'em' is a very useful unit in CSS, since it can adapt automatically to the font that the reader uses
ex	one ex is the x-height of a font (x-height is usually about half the font-size)
pt	point (1 pt is the same as 1/72 inch)
pc	pica (1 pc is the same as 12 points)
px	pixels (a dot on the computer screen)

[W3C CSS units examples](#)

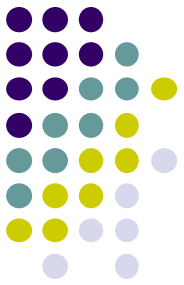


# CSS Box Model

- Όλα τα HTML στοιχεία θεωρούνται ως boxes.
- Ιεραρχία HTML μεταξύ των στοιχείων στο CSS
- [Demonstrating the box model](#)



# Τοποθέτηση στοιχείων (positioning)



- Static: τοποθέτηση πάντα με βάση την κανονική ροή της σελίδας. Η προκαθορισμένη τοποθέτηση για τα html στοιχεία.
- Fixed: τοποθέτηση σε σχέση με το παράθυρο του browser. Πχ navigation bars
- Absolute: τοποθέτηση σε σχέση με το πρώτο parent element που έχει τοποθέτηση διαφορετική από static, αλλιώς σε σχέση με το <html> πχ dropdown menus
- Relative: τοποθέτηση σε σχέση με την κανονική θέση του στοιχείου.
- Αφού δοθεί τιμή στο positioning τα στοιχεία τοποθετούνται με τις ιδιότητες top, bottom, left, right



# Απόλυτη τοποθέτηση στοιχείων

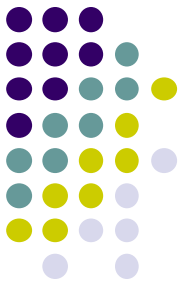


```
<p><img src = "images/i.gif" style = "position: absolute;  
top: 0px; left: 0px; z-index: 1" alt = "First image"></p>
```

```
<p style = "position: absolute; top: 50px; left: 50px;  
z-index: 3; font-size: 20pt;">Positioned Text</p>
```

```
<p><img src = "images/circle.gif" style = "position: absolute;  
top: 25px; left: 100px; z-index: 2"  
alt = "Second image"></p>
```

[View page](#)



# Σχετική τοποθέτηση στοιχείων

`.super` { `position: relative`; `top: -1ex` }

`.sub` { `position: relative`; `bottom: -1ex` }

`.shiftright` { `position: relative`; `left: -1ex` }

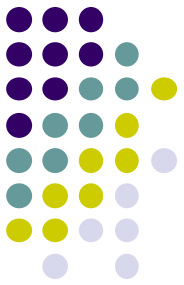
`.shiftright` { `position: relative`; `right: -1ex` }

em: the 'font-size' of the relevant font

ex: the 'x-height' of the relevant font



[View page](#)



# Διαστάσεις αντικειμένων

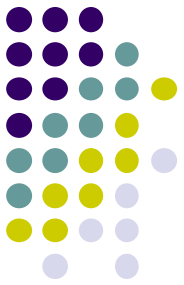
```
<style type = "text/css">  
  div { background-color: #ffccff;  
        margin-bottom: .3cm }  
</style>
```

.....

```
<div style = "width: 80%; text-align: center">
```

Here is some CENTERED text that goes in a box  
which is set to stretch across eighty percent of  
the width of the screen.</div>

[View page](#)



# Borders

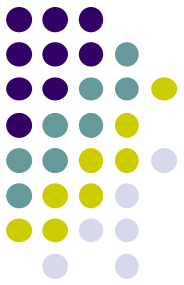
border-width: {thick (παχύ), medium, thin (λεπτό) }

border-style: {groove(αυλάκι), dotted, solid,... }

border-color

\*border-{top, left, right, bottom}-width

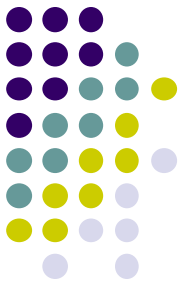
[View page](#)



# Responsive Web Design

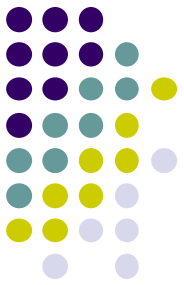
- Η πρακτική της δημιουργίας ενός website κατάλληλο για οποιαδήποτε συσκευή, ανεξάρτητα από το μέγεθος της οθόνης
- Responsive vs Mobile: ο 2<sup>ος</sup> όρος αναφέρεται στη δημιουργία άλλης έκδοσης του website για κινητές συσκευές. Στο responsive design έχουμε ένα website που η εμφάνισή του προσαρμόζεται στα χαρακτηριστικά της συσκευής.

# Τεχνικές



- Flexible layout
  - The practice of building the layout of a website with a flexible grid, capable of dynamically resizing to any width. Flexible grids are built using relative length units, most commonly percentages or em units.
- Media Queries
  - Media queries provide the ability to specify different styles for individual browser and device circumstances, the width of the viewport or device orientation for example.
- Flexible Media
  - Images, videos, and other media types need to be scalable, changing their size as the size of the viewport changes.

# Flexible Layout



## Flexible Grid Demo

100% WIDE CONTAINER

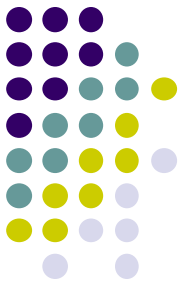


75% WIDE CONTAINER



50% WIDE CONTAINER





# Media Queries

- @media: all, screen, print, tv, 3d-glasses (html5)

1. `@media all and (min-width: 800px) and (max-width: 1024px) {...}` **AND OPERATOR**

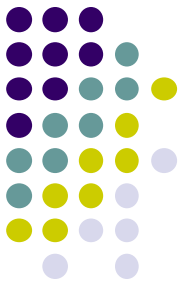
1. `@media all and (min-width: 320px) and (max-width: 780px) {...}` **HEIGHT & WIDTH**

1. `@media all and (orientation: landscape) {...}` **ORIENTATION**

- [Παραδείγματα](#)



# Flexible Media



```
img, video, canvas {  
  max-width: 100%;  
}
```

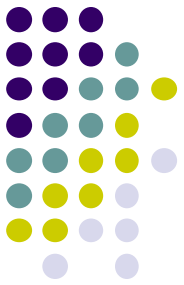
75% WIDE CONTAINER



50% WIDE CONTAINER

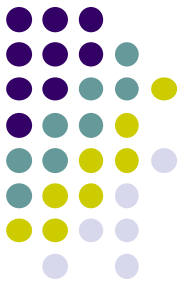


# Παράδειγμα



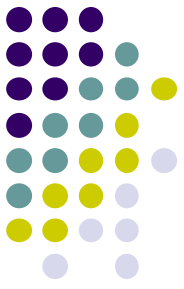
- Tutorial: <http://learn.shayhowe.com/advanced-html-css/responsive-web-design>

# Παράδειγμα non-responsive website



- <https://dequeuniversity.com/library/responsive/1-non-responsive>
- Οριζόντιο scrolling



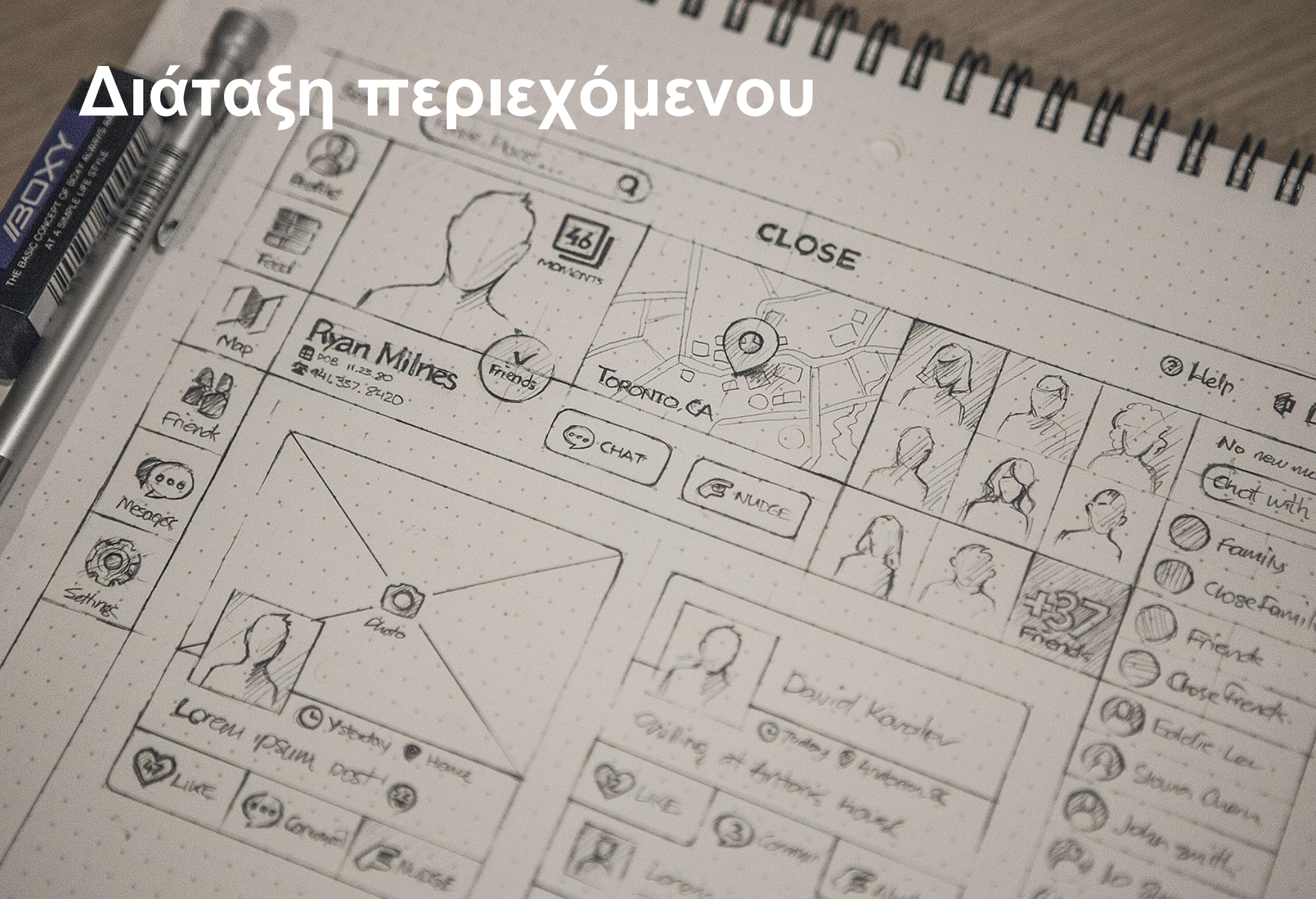


# Προσβασιμότητα

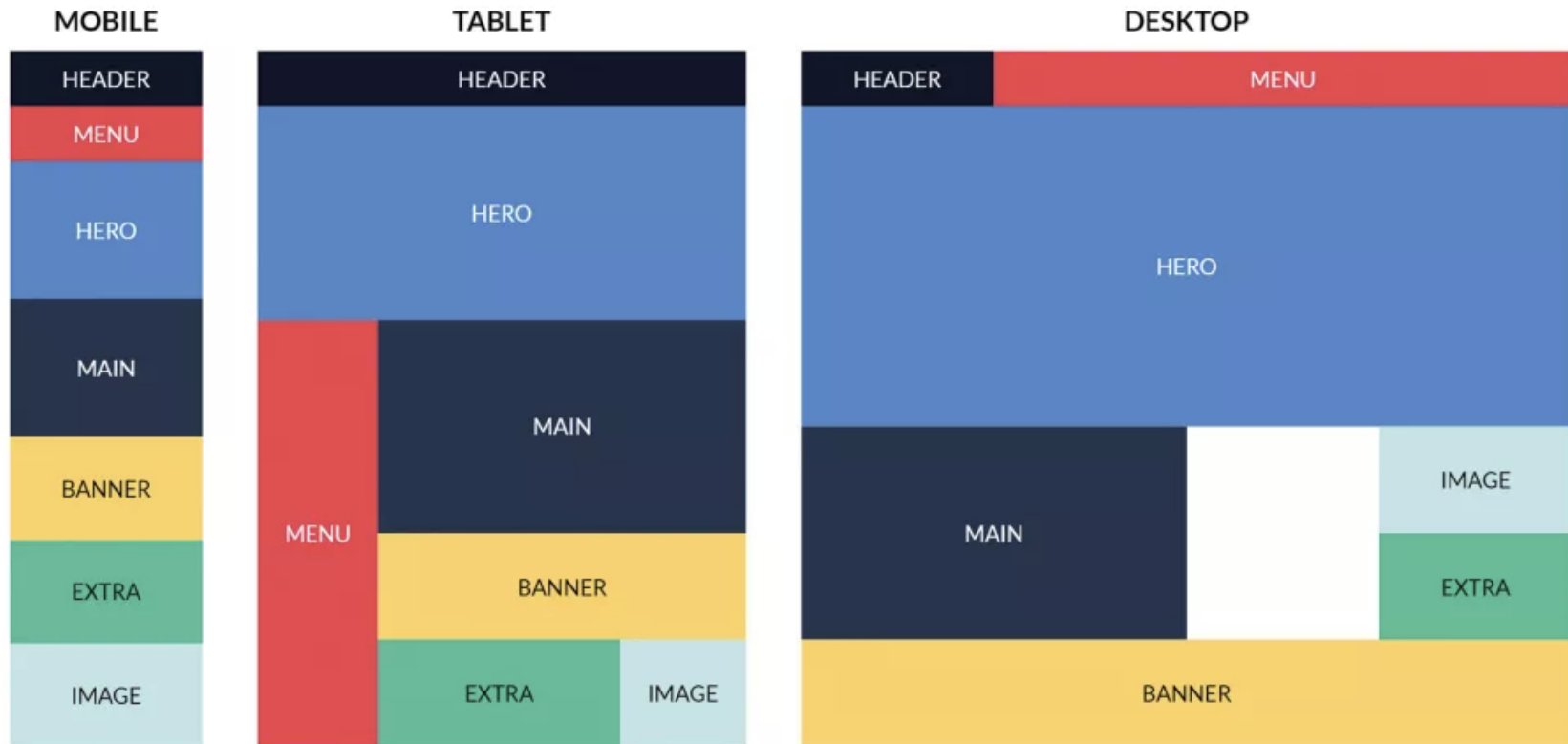
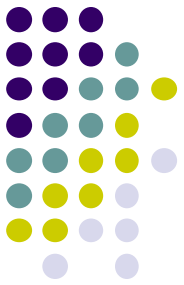
- Σχεδίαση websites ώστε να μπορούν να χρησιμοποιηθούν από άτομα με αναπηρίες
- [W3 Web Accessibility Initiative \(WAI\)](#)
- [Web Content Accessibility Guidelines \(WCAG\)](#)
- Ν.4591/2019: ενσωμάτωση οδηγίας της Ε.Ε. για την προσβασιμότητα ιστότοπων και mobile apps των δημόσιων φορέων
- Accessibility checkers: π.χ. [WebAIM WAVE](#)



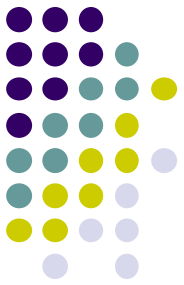
# Διάταξη περιεχομένου



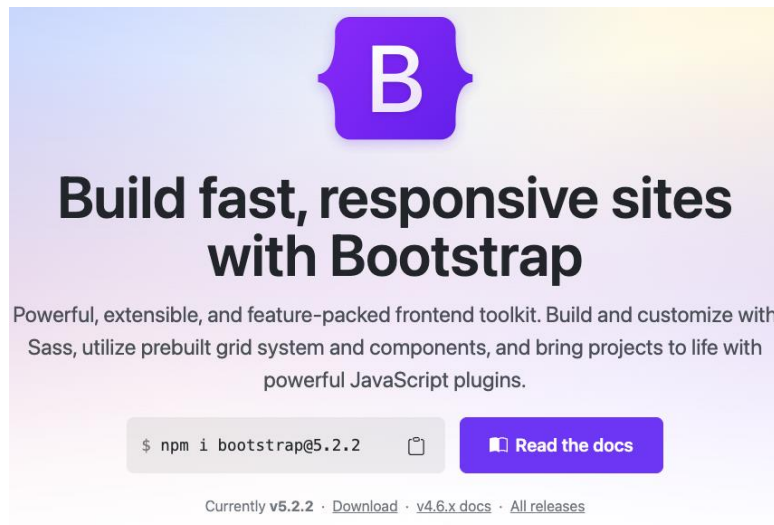
# Διάταξη περιεχομένου



# Bootstrap

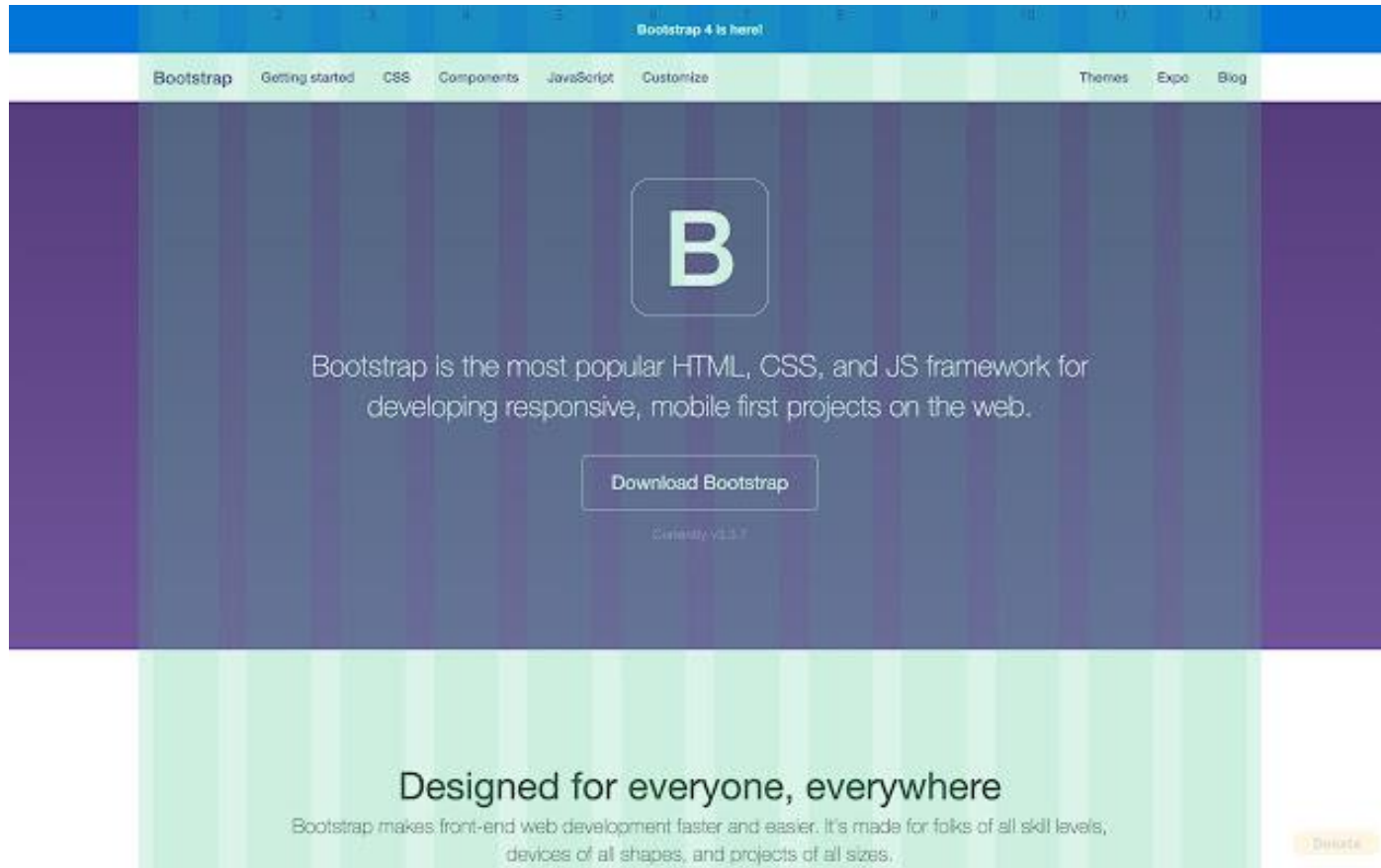
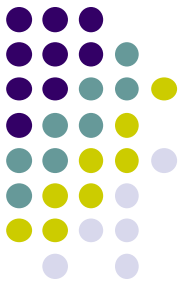


- Από τα δημοφιλέστερα UI toolkits
- Δημιουργήθηκε από 2 υπαλλήλους στην Twitter το 2011
- Λόγος:
  - Υπήρχε η ανάγκη για ένα πρότυπο στην front-end εργαλειοθήκη που θα ακολουθούσαν όλοι στην εταιρία



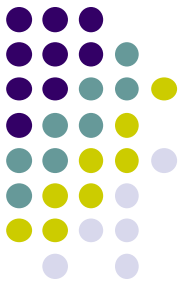


# Bootstrap layout system





# Τι περιλαμβάνει



Bootstrap/

└─ css/

└─ bootstrap.css

└─ bootstrap.min.css

└─ bootstrap.theme.css

└─ bootstrap.theme.min.css

└─ js/

└─ bootstrap.js

└─ bootstrap.min.js

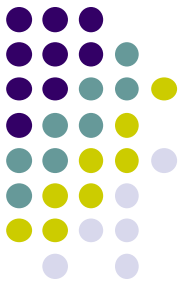


**JavaScript**  
(client side code)

**CSS (Cascading Style Sheet)**  
(client side presentation)

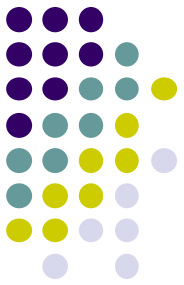
**HTML**  
(page content)





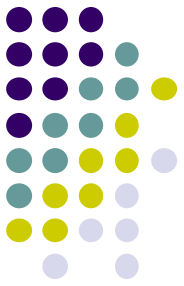
# Χαρακτηριστικά (1/3)

- Responsive
- Mobile first\*
- Σύνολα από HTML στοιχεία και συνδυασμούς τους
- Διάφορα θέματα (templates)
- Ενσωματωμένος μηχανισμός για την τοποθέτηση HTML στοιχείων σε πλέγμα
- Παρέχει την Javascript
- Επιταχύνει την διαδικασία δημιουργία σελίδας



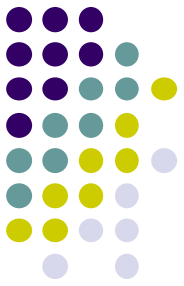
# Mobile first

- Η διαδικασία κατά την οποία χτίζουμε κάτι με βασικό κριτήριο να εμφανίζεται και να παίζει καλά πρώτα σε μικρές οθόνες.
- Όσο μεγαλώνει η οθόνη, πρόσθετοι κανόνες δημιουργούνται για να υποστηρίξουν και τα μεγαλύτερα μεγέθη.
  - Πρόσθετοι κανόνες με χρήση των **media queries**



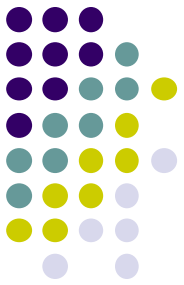
# Χαρακτηριστικά (2/3)

- Παρέχει CSS κλάσεις για :
  - Alignment
  - Transformation
  - Blockquotes
  - Description
  - Code
  - Tables
  - Forms
  - Buttons
  - Images
  - ...



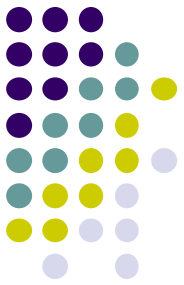
# Χαρακτηριστικά (3/3)

- Παρέχει τα παρακάτω components:
  - Dropdowns
  - Input groups
  - Navbar
  - Carousel
  - Alerts
  - Progress bars
  - List group
  - Responsive embed
  - ...



# Πότε αξίζει η χρήση Bootstrap

- Όταν πιέζει ο χρόνος
- Όταν ο σχεδιασμός απαιτεί την χρήση grid (πλέγμα)
- Όταν είναι απαραίτητο να υποστηρίξει όλα τα μεγέθη οθονών
- Όταν θέλω compatibility με browsers χωρίς να με απασχολεί ποιος θα χρησιμοποιηθεί

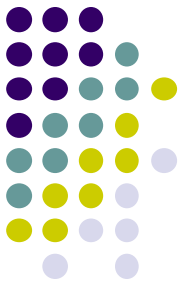


# Χρήση Bootstrap (1/2)

- Χρήση στην σελίδα/εφαρμογή μας:

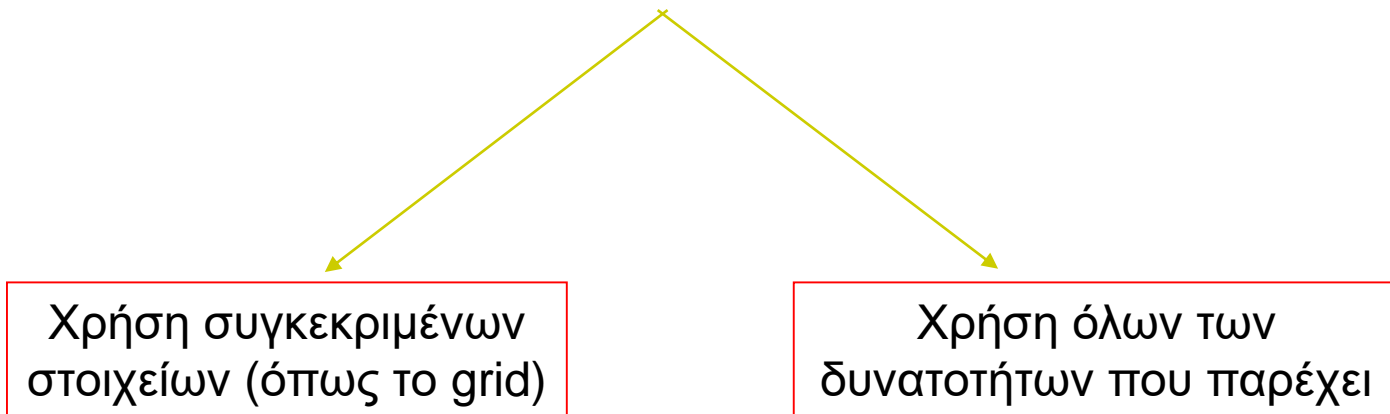
```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1 user-scalable=no">
    <style rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css" />
    <title>Page/App title</title>
  </head>
  <body>
    <!-- Το σώμα της εφαρμογής/σελίδας -->
  </body>
</html>
```

Γιατί βάζουμε το CSS μέσα στο <HEAD> στοιχείο?



# Χρήση Bootstrap (1/2)

Υπάρχουν 2 τρόποι που μπορεί να γίνει χρήση:

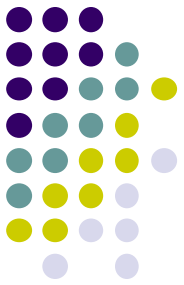


**Γενικά ξεκινάμε με τα πολύ βασικά και χτίζουμε και βελτιώνουμε όσο υλοποιούμε**





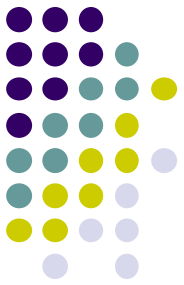
# Χρήση του Grid (2/4)



- Υπάρχουν 6 συστήματα πλέγματος στο Bootstrap, με μόνο στοιχείο διαφοροποίησης το viewport
- Breakpoint
  - πλάτος οθόνης που καθορίζει πως συμπεριφέρεται ένα responsive website
    - δηλ. τότε αποφασίζεται να αλλάξει το layout.

Breakpoint	Class infix	Dimensions
Extra small	<i>None</i>	<576px
Small	<i>sm</i>	≥576px
Medium	<i>md</i>	≥768px
Large	<i>lg</i>	≥992px
Extra large	<i>xl</i>	≥1200px
Extra extra large	<i>xxl</i>	≥1400px

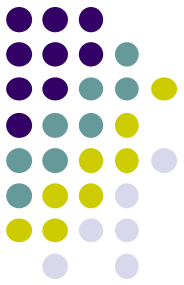
# Χρήση του Grid (3/4)



- Κλάσεις γραμμών
  - row (δημιουργεί οριζόντια σελ στηλών)
- Κλάσεις Στηλών: **col-[SIZE]-[SPAN]**
  - col-xs-1, col-sm-4, col-md-3, col-lg-10
  - Size: το breakpoint στο οποίο αλλάζει η συμπεριφορά
  - Span: πόσες στήλες δεσμεύει το div (το άθροισμα πρέπει να είναι 12!)
- Οι στήλες **πρέπει** να είναι το πρώτο παιδί μιας γραμμής.

```
<body>
  <div class="container">
    <div class="row">
      <div class="col-*-*"></div>
      <div class="col-*-*"></div>
    </div>
    <div class="row">
      ...
    </div>
  </div>
</body>
```

# Παράδειγμα



```
<body>

<div class="jumbotron text-center">
  <h1>My First Bootstrap Page</h1>
  <p>Resize this responsive page to see the effect!</p>
</div>

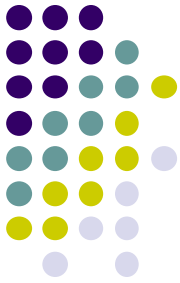
<div class="container">
  <div class="row">
    <div class="col-sm-4">
      <h3>Column 1</h3>
      <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit...</p>
      <p>Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 2</h3>
      <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit...</p>
      <p>Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...</p>
    </div>
    <div class="col-sm-4">
      <h3>Column 3</h3>
      <p>Lorem ipsum dolor sit amet, consectetur adipisicing elit...</p>
      <p>Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...</p>
    </div>
  </div>
</div>

</body>
</html>
```

Breakpoint	Class infix	Dimensions
Extra small	None	<576px
Small	sm	≥576px
Medium	md	≥768px

Ο κανόνας  
δέσμευσης ισχύει  
όταν το πλάτος γίνει  
≥sm (576px)

# Παράδειγμα



## My First Bootstrap Page

Resize this responsive page to see the effect!

### Column 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

### Column 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

### Column 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

## My First Bootstrap Page

Resize this responsive page to see the effect!

### Column 1

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

### Column 2

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

### Column 3

Lorem ipsum dolor sit amet, consectetur adipiscing elit...

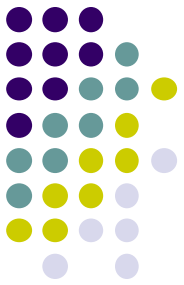
Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris...

`<div class="col-sm-4">`

W=1024px (>sm)  
Σέβεται τον  
κανόνα  
(δέσμευση 4  
στηλών / div)

Breakpoint	Class infix	Dimensions
Extra small	None	<576px
Small	sm	≥576px
Medium	md	≥768px

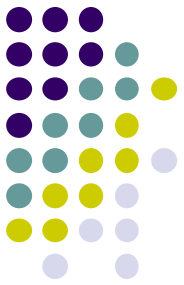
W=575px  
Δεν υπάρχει  
κανόνας  
δέσμευσης. Τα  
divs απλά  
πάνε το ένα  
πάνω στο  
άλλο.



# Χρήση του Grid (4/4)

- Μπορεί να γίνει συνδυασμός κλάσεων για υποστήριξη περισσότερων οθονών:

```
<body>
  <div class="container">
    <div class="row">
      <div class="col-md-6 col-sm-10">
        This div fills left half of the page
      </div>
      <div class="col-md-6 col-sm-10">
        This div fills right half of the page
      </div>
    </div>
  </div>
</body>
```



# Πίνακες

- Οι πίνακες πρέπει να έχουν την κλάση "table" για να αναγνωριστούν από το bootstrap

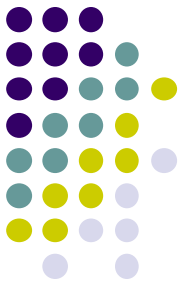
```
<body>
  <div class="container">
    <table class="table">
      <thead>
        <tr>
          <td>Όνομα</td>
          <td>Τηλέφωνο</td>
        </tr>
      </thead>
      <tbody>
        <tr>
          <td>John</td>
        </tr>
        <tr>
          <td>555-1234689</td>
        </tr>
      </tbody>
    </table>
  </div>
</body>
```

Κλάσεις μορφοποίησης  
πίνακων:

- table-striped
- table-bordered
- table-hover
- table-condensed

Μετατροπή σε responsive πίνακα:

```
<div class="table-responsive">
<table class="table">
</table> </div>
```

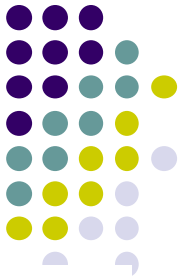


# Forms

- Υπάρχουν 3 είδη φόρμας : Basic, Inline και horizontal

```
<form class="form-horizontal">
  <div class="form-group">
    <label for="inputEmail">Email address</label>
    <input type="email" class="form-control" id="inputEmail" placeholder="Email">
  </div>
  <div class="form-group">
    <label for="inputPassword">Password</label>
    <input type="password" class="form-control" id="inputPassword" placeholder="Password">
  </div>
  <div class="form-group">
    <label for="inputFile">File input</label>
    <input type="file" id="inputFile">
    <p class="help-block">Example block-level help text here.</p>
  </div>
  <div class="checkbox">
    <label><input type="checkbox"> Check me</label>
  </div>
  <button type="submit" class="btn btn-default">Submit</button>
</form>
```





### Bootstrap Vertical Form (default)

**Email:**

**Password:**

☐ Remember me



### Bootstrap Vertical Form (default)

**Email:**

**Password:**

☐ Remember me

### Bootstrap Inline Form

**Email:**  **Password:**  ☐ Remember me



### Bootstrap Inline Form

**Email:**

**Password:**

☐ Remember me

### Bootstrap Horizontal Form

**Email:**

**Password:**

☐ Remember me



### Bootstrap Horizontal Form

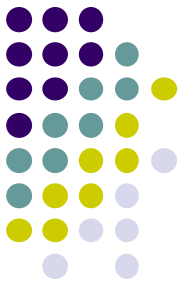
**Email:**

**Password:**

☐ Remember me

Μεγάλες οθόνες

Μικρές οθόνες



# Buttons

- Τα κουμπιά μπορεί να είναι:
  - Στοιχεία button : <button>
  - Στοιχεία link : <a>
  - Στοιχεία input: <input>

## EXAMPLE

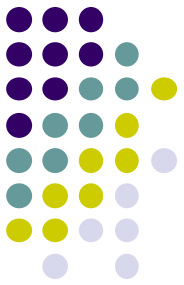
Link

Button

Input

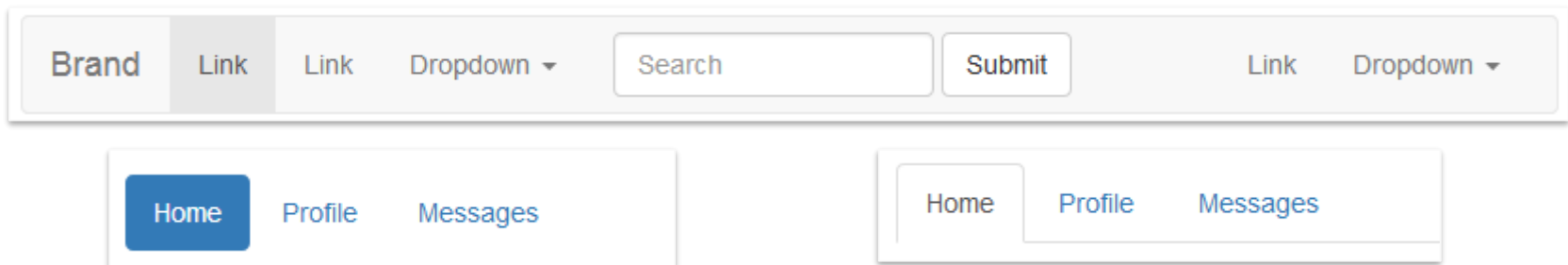
Submit

```
<a class="btn btn-default" href="#" role="button">Link</a>
<button class="btn btn-default" type="submit">Button</button>
<input class="btn btn-default" type="button" value="Input">
<input class="btn btn-default" type="submit" value="Submit">
```

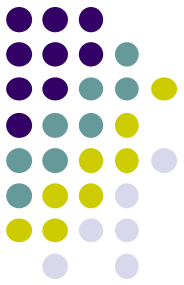


# Στοιχεία μενού








- Το bootstrap παρέχει στοιχεία που μπορούν να χρησιμοποιηθούν ως μενού/υπομενού και γενικότερα navigation:
  - [Dropdowns](#)
  - [Tabs](#)
  - [Pills](#)
  - [Navbars](#)



# Εικονίδια γραμματοσειράς (font icons) deprecated



- Περιέχει τα [glyphicon](#) font icons

						
glyphicon glyphicon- glass	glyphicon glyphicon- music	glyphicon glyphicon- search	glyphicon glyphicon- heart	glyphicon glyphicon- star	glyphicon glyphicon- star- empty	glyphicon glyphicon- user

- Χρήση:



# Τέλος

Ερωτήσεις?

