

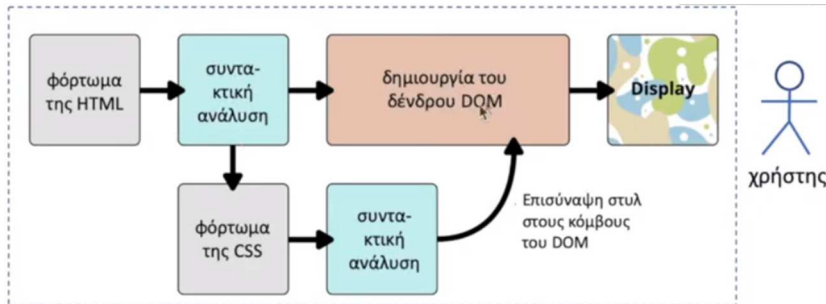
ΕΙΣΑΓΩΓΗ ΣΤΗΝ CSS3

1. Εισαγωγή: λειτουργία της CSS

-Η **CSS (Cascading Style Sheets)** είναι μια γλώσσα για τον προσδιορισμό του τρόπου **παρουσίασης** του περιεχομένου των **εγγράφων HTML** στους χρήστες - δηλ. του στυλ τους και της διάταξης τους, π.χ. τον καθορισμό της γραμματοσειράς, του μεγέθους και του χώρου του κειμένου, τη διαίρεση σε στήλες, την προσθήκη κινουμένων σχεδίων κ.λπ.

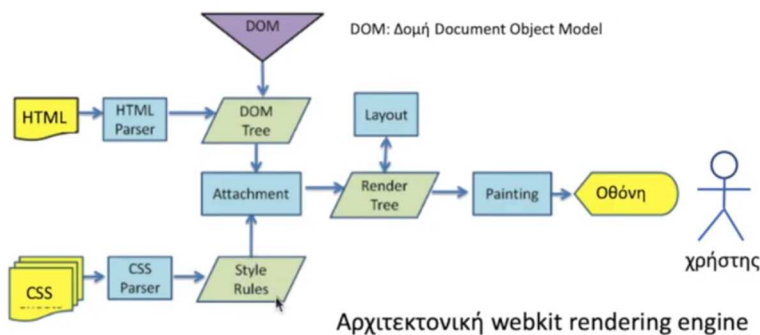
-Η **παρουσίαση** ενός εγγράφου αφορά στη μετατροπή του σε χρήσιμη μορφή για το χρήστη του. Οι Φυλλομετρητές, όπως οι Firefox, Chrome ή Edge, έχουν σχεδιαστεί για την οπτική παρουσίαση των στοιχείων του εγγράφου, στην οθόνη με βάση κανόνες CSS.

Η CSS συσχετίζεται με το DOM(τα στοιχεία του εγγράφου)



DOM και κανόνες CSS(style rules)

DOM και κανόνες CSS (style rules)



<https://www.html5rocks.com/en/tutorials/internals/howbrowserswork> (webkit: rendering engine)

κανόνες CSS

-Ορίζουν σύνολο από **ιδιότητες (properties)**, με αντίστοιχες τιμές που καθορίζουν τον τρόπο παρουσίασης του περιεχομένου

-Ένας **επιλογέας (selector)**, καθορίζει τα στοιχεία στα οποία η ιδιότητα θα εφαρμοστεί.

Τρόποι να εφαρμοστεί CSS σε ένα αρχείο HTML

1. εξωτερικό stylesheet.

ο προτεινόμενος τρόπος

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

2. εσωτερικό stylesheet

```
<style>  
  h1 { color : blue;      border : 1px solid black; }  
  p { color : red; }  
</style>
```

μέσα σε ένα στοιχείο
<style> στο <head>

3. inline stylesheet

```
<p style="color:red;">This is a CSS example</p>
```

-ΜΠΟΡΕΙ ΝΑ ΥΠΑΡΧΕΙ ΣΥΝΔΥΑΣΜΟΣ ΤΩΝ 3 ΠΕΡΙΠΤΩΣΕΩΝ ΤΑΥΤΟΧΡΟΝΑ ΣΤΟ ΕΓΓΡΑΦΟ ΜΑΣ
-ΥΠΑΡΧΕΙ ΣΕΙΡΑ ΠΡΟΤΑΙΡΑΙΟΤΗΤΑΣ.

ΠΡΩΤΑ ΛΑΜΒΑΝΕΤΑΙ ΥΠΟΨΗΝ η 3^η περίπτωση μετά η 2^η και τέλος η 1^η

<https://www.tutorialspoint.com/Rules-to-override-Style-Sheet-Rule-in-CSS>

π.χ.

| | |
|---|--|
| <pre><!DOCTYPE html>
<html>
 <head>
 <meta charset="utf-8">
 <link rel="stylesheet" href="mycss.css">
 </head>
 <body>
 <h1>Καλή σας μέρα άρχοντες</h1>
 </body>
</html></pre> | <pre>/*αρχείο mycss.css*/
h1 {
 font-family:Arial, Helvetica, sans-
 serif;
 color: blue;
 background-color: yellow;
 border: rgb(245, 31, 31) solid 2px;
 margin:10px;
 padding:10px 10px;
}</pre> |
|---|--|

Σύνταξη κανόνων CS

επιλογέας { δήλωση ; δήλωση ; }

h1 { color:blue; font-size:12px; }

{ ιδιότητα : τιμή ; ιδιότητα : τιμή ; }

- υπάρχουν περίπου 300 ιδιότητες
- κάθε ιδιότητα μπορεί να πάρει ορισμένες μόνο τιμές
- αν η ιδιότητα είναι άγνωστη ή η τιμή δεν είναι αποδεκτή για την ιδιότητα αυτή, ή αν δεν την υποστηρίζει ο browser, η δήλωση θεωρείται άκυρη και απλά δεν λαμβάνεται υπόψη
- οι **επιλογείς** ορίζουν τα στοιχεία του αρχείου HTML στα οποία θα εφαρμοστεί ο κανόνας.
- Ένας κανόνας μπορεί να περιέχει πολλαπλούς επιλογείς χωρισμένους με κόμματα π.χ. h1,h2 {...}
- Μπορούμε επίσης να ορίσουμε **αλυσίδα επιλογέων**, καθώς και συνθήκες για εφαρμογή του κανόνα (π.χ. όταν το ποντίκι ίπταται (hover) πάνω από το στοιχείο
- σχόλιο στον κώδικα css → /* αυτό είναι σχόλιο*/

ΠΑΡΑΔΕΙΓΜΑΤΑ

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Css example</title>
    <link rel="stylesheet" href="style.css">
  </head>
  <body>
    <h1>This is an h1 header</h1>
    <p>This is the first CSS example</p>
    <ul>
      <li>This is</li>
      <li>an unnamed list</li>
    </ul>
  </body>
</html>
```

```
/*αρχείο style.css*/
body{
  font-family: sans-serif;
}
h1{
  color: blue;
  background-color: yellow;
  border: 1px solid black;
}
p{
  color: red;
}
p,li{
  text-decoration: underline;
}
```

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Css example</title>
    <link rel="stylesheet" href="mountains.css">
  </head>
  <body>
    <p>High Greek Mountains are :
      <span>Mnt. Olympus,</span>
      <span>Mnt. Smolikas,</span>
      <span>Mnt. Kaimaktsalan,</span>
    </p>
  </body>
</html>
```

```
/*αρχείο mountains.css*/
span{
  font-family: sans-serif;
  border: solid black 1px;
  background-color: yellow;
}
/* άλλαξε το πάνω με το κάτω να
   δεις τις διαφορές
span{
  font-family: sans-serif;
  font-size: 2em;
  border: solid black 5px;
  background-color: yellow;
}*/
```

ΚΑΤΗΓΟΡΙΕΣ ΕΠΙΛΟΓΩΝ

- απλοί επιλογείς**: ταιριάζουν σε στοιχεία με βάση τον τύπο του στοιχείου, την κλάση ή την id επιλογέας στοιχείου
`p{color:red}` → όλα τα στοιχεία p θα εμφανιστούν κόκκινα
επιλογέας κλάσης
`.first{font-weight:bold;}` → όλα τα στοιχεία με `class="first"` είναι bold
κλήση από html `<li class="first">Create document`
επιλογέας id (τα id είναι μοναδικά στο έγγραφο html)
`#polite {font-family:cursive;}`
κλήση από html `<p id="polite"> -GOOD morning!</p>`

- πολλαπλοί επιλογείς** : οι επιλογείς χωρίζονται με κόμματα , ο ίδιος κανόνας εφαρμόζεται σε διαφορετικές κατηγορίες στοιχείων π.χ. `h1,h2 {color:red}`

- συνδυασμός**: π.χ. επιλογή παραγράφων `<p>` που είναι άμεσα παιδιά ενός `<div>` ή παράγραφοι `<p>` άμεσα παιδιά από επικεφαλίδες `<h1>`.

- επιλογείς γνωρισμάτων**: Επιλέγουν στοιχεία με βάση τα γνωρίσματα τους ή τις τιμές των γνωρισμάτων τους. π.χ. `[class=vegetable]`

<code>[attr]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr.	<code>/* στοιχεία <a> που έχουν ιδιότητα με το όνομα title */ a[title] { color: purple;}</code>
<code>[attr=value]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr που έχουν ακριβώς την τιμή value.	<code>/* στοιχεία <a> που το href είναι ίσο με "https://example.org" */ a[href="https://example.org"] { color: green;}</code>
<code>[attr~=value]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr που οι τιμές τους είναι μία λίστα λέξεων χωρισμένες με whitespace (π.χ. κενό) μία εκ των οποίων έχει τιμή ακριβώς ίδια με το value.	<code>/* στοιχεία <a> elements whose των οποίων το class περιέχει την λέξη "logo" */ a[class~="logo"] {padding: 2px;}</code>
<code>[attr =value]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr που οι τιμές τους μπορούν να είναι ακριβώς το value ή μπορούν να ξεκινούν με το value-	
<code>[attr^=value]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr που οι τιμές τους ξεκινούν με value.	
<code>[attr\$=value]</code>	επιλέγουμε στοιχεία που έχουν ιδιότητα με το όνομα attr που οι τιμές τους τελειώνουν με value.	<code>/* στοιχεία <a> με href να τελειώνει σε ".org" */ a[href\$=".org"] {font-style: italic;}</code>
<code>[attr*=value]</code>	στοιχεία που έχουν ιδιότητα με το όνομα attr που οι τιμές τους περιέχει τουλάχιστον μία εμφάνιση του value μέσα στην τιμή της ιδιότητας.	<code>/* στοιχεία <a> που το href περιέχει "example" */ a[href*="example"] {font-size: 2em;}</code>
<code>[attr operator value i]</code>	Βάζοντας i ή I πριν την αγκύλη κάνει την value να συγκριθεί μη λαμβάνοντας υπόψιν την διαφορά κεφαλαίων πεζών γραμμάτων (case-insensitively) για χαρακτήρες ASCII.	
<code>[attr operator value s]</code>	Βάζοντας s ή S πριν την αγκύλη κάνει την value να συγκριθεί λαμβάνοντας υπόψιν την διαφορά κεφαλαίων πεζών γραμμάτων (case-sensitively) για χαρακτήρες ASCII.	

-**Ψευδο-κλάσεις:** επιλέγουν στοιχεία που βρίσκονται σε συγκεκριμένη κατάσταση (π.χ. σε αυτά που ίπταται πάνω τους το ποντίκι ή checkbox που έχει επιλεγεί, ή το πρώτο παιδί ενός στοιχείου στο DOM μερικές ψευδο-κλάσεις φαίνονται στον πιο κάτω πίνακα

:active	:checked	:default	:dir	:disabled
:empty	:enabled	:first	:first-child	:first-of-type
:fullscreen	:focus	:focus-within	:hover	:indeterminate
:in-range	:invalid	:lang	:last-child	:last-of-type
:left	:link	:matches()	:not	:nth-child
:nth-last-child	:nth-last-of-type	:nth-of-type	:only-child	:only-of-type
:optional	:out-of-range	:read-only	:read-write	:required
:right	:root	:scope	:target	:valid
:visited				

-Μια ψευδο-κλάση CSS δημιουργείται με την προσθήκη μιας λέξης κλειδιού στο τέλος ενός επιλογέα, μετά από τον χαρακτήρα : , υποδηλώνοντας ότι θα εφαρμοστεί το στυλ όταν το στοιχείο θα είναι σε ορισμένη κατάσταση

π.χ.

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      body {font-family: Arial, Helvetica, sans-serif;}
      p:nth-child(odd){
        background-color: black;
        color: white;
        padding: 10px;
        margin: 1px;
      }
      p:nth-child(even) {
        background-color: beige;
        color: rgb(19, 1, 1);
        padding: 10px;
        margin: 1px;
      }
      p:hover {background-color: red;
        color: black;}
    </style>
  </head>
  <body>
    <div class="tennis-players">
      <p>Roger Federer</p>
      <p>Rafa Nadal</p>
      <p>Nole Djokovic</p>
      <p>Stefanos Tsitsipas</p>
    </div>
  </body>
</html>
```

-Ψευδοστοιχεία: επιλέγουν περιεχόμενο που βρίσκεται σε ορισμένη θέση π.χ. η πρώτη λέξη μιας παραγράφου **::first-line ::first-letter ::before ::after**

π.χ.

::before και ::after

εισάγουν κάποιο κείμενο (ιδιότητα content) κάθε φορά που εμφανίζεται το στοιχείο

```
h2::before{ /*ορίζει ότι κάθε h2 θα ξεκινάει με το Κεφάλαιο: και μετά την τιμή που έχει δωθεί*/  
content:"Κεφάλαιο:";
```

```
color:red;}/*το χρώμα επηρεάζει μόνο το πρόθεμα Κεφάλαιο: που βάλαμε*/
```

ΟΝΟΜΑ	ΣΥΝΤΑΞΗ	ΕΠΙΛΕΓΕΙ
λίστα επιλογών	A,B	τα στοιχεία που ταιριάζουν στο A ή και στο B
Συνδυασμός αλυσίδας	A B	Κάθε στοιχείο που ταιριάζει το B που είναι απόγονος στοιχείου που ταιριάζει το A(παιδί του,εγγόνι του κτλ)
συνδυασμός άμεσο παιδί	A>B	Κάθε στοιχείο που ταιριάζει το B που είναι άμεσο παιδί ενός στοιχείου που ταιριάζει το A.
διπλανοί συγγενείς	A+B	Κάθε στοιχείο που ταιριάζει το B που είναι άμεσος επόμενος συγγενής του A (το επόμενο παιδί του πατέρα).
γενικοί συγγενείς	A~B	Κάθε στοιχείο B που είναι ένας από τους συγγενείς ίδιου επιπέδου A (παιδιά του ίδιου πατέρα).

Παράδειγμα

```
table td,table th{  
border:1px solid black;  
padding:0.5em 0.5em 0.4em;  
}
```

Εδώ επιλέγει όλα τα στοιχεία <td> μέσα σε στοιχείο <table> και όλα τα <th> μέσα σε <table>

-Το κόμμα **δεν είναι συνδυαστής**, αλλά μόνο μας επιτρέπει να στοχεύουμε αρκετούς επιλογείς με το ίδιο σύνολο κανόνων css.

-Το κόμμα **διαχωρίζει δύο διαφορετικές περιπτώσεις στοιχείων.**

Άρα στην πρώτη περίπτωση td που είναι απόγονοι table και στη δεύτερη th που είναι απόγονοι table επίσης.

Λίστες ζέβρα είναι οι λίστες εκείνες όπου τα στοιχεία εναλλάσσονται ως προς το χρώμα του υποβάθρου.

```
li:nth-of-type(2n+1) {  
color: lime; }
```

ΜΟΝΤΕΛΟ BOX

Το μοντέλο box είναι το θεμέλιο της διάταξης μιας ιστοσελίδας αφού κάθε στοιχείο της ορίζεται ως ένα ορθογώνιο κουτί, που εκτός από το περιεχόμενο του στοιχείου(**content**) περιλαμβάνει ένα περιτύλιγμα (πλαίσιο(**border**) και κενό πριν(**padding**) και μετά(**margin**) το πλαίσιο(border)).

παράμετροι ορισμού διαστάσεων του μοντέλου box

-width:ρυθμίζει το πλάτος του ορθογωνίου περιεχομένου

υπάρχουν επίσης οι **min-width** , **max-width** που ρυθμίζουν το ελάχιστο/μέγιστο πλάτος

-height:ρυθμίζει το ύψος του ορθογωνίου περιεχομένου

υπάρχουν επίσης οι **min-height** , **max-height** που ρυθμίζουν το ελάχιστο/μέγιστο ύψος

-padding:αναφέρεται στο εσωτερικό περιθώριο ενός πλαισίου CSS-μεταξύ περιεχομένου και πλαισίου Μπορεί να τοποθετηθεί και στις 4 πλευρές ταυτόχρονα με την ιδιότητα **padding** ή μία πλευρά κάθε φορά με τις ιδιότητες **padding-top**, **padding-right**, **padding-bottom**, **padding-left**

-margin:αναφέρεται στο εξωτερικό περιθώριο. Συμπεριφέρεται όπως το padding

Μπορεί να τοποθετηθεί και στις 4 πλευρές ταυτόχρονα με την ιδιότητα **margin** ή μία πλευρά κάθε φορά με τις ιδιότητες **margin-top**, **margin-right**, **margin-bottom**, **margin-left**

-border:είναι το πλαίσιο μεταξύ του εσωτερικού περιθωρίου και του εξωτερικού. Από προεπιλογή, το περιθώριο έχει **μέγεθος 0 (είναι αόρατο)**. Πρέπει να ρυθμίσουμε το πάχος ,το στυλ και το χρώμα του περιγράμματος ώστε να εμφανιστεί, για παράδειγμα **border:1px solid black**

-Υπάρχουν επίσης αντίστοιχα με τα υπόλοιπα οι **border-top**, **border-right**, **border-bottom**, **border-left** και τα **border-width**,**border-style**,**border-color** που ρυθμίζουν πάχος, στυλ ή το χρώμα μιας πλευράς του πλαισίου

-Τέλος υπάρχουν και οι **border-width** , **border-style**, **border-color** που ρυθμίζουν πάχος, στυλ ή το χρώμα για όλες τις πλευρές του πλαισίου.

π.χ.



πλάτος κουτιού=αριστερό margin + border*2+αριστερό padding+width+δεξιό padding+δεξιό margin
=70+10*2+20+100+20+70=300px

ύψος κουτιού=πάνω margin + border*2+πάνω padding+width+κάτω padding+κάτω margin
=70+10*2+20+100+20+70=300px

ΣΗΜΕΙΩΣΗ για τα margin,padding

Μπορούν να πάρουν από 1 έως και 4 παραμέτρους.

-ακολουθεί φορά ρολογιού από ΠΑΝΩ προς τα Αριστερά για τις 4 παραμέτρους

Παρακάτω υπάρχουν παραδείγματα και τι σημαίνει τι.



4 παράμετροι	margin: 25px 50px 75px 100px;	ΠΑΝΩ margin 25px ΔΕΞΙΑ margin 50px ΚΑΤΩ margin 75px ΑΡΙΣΤΕΡΑ margin 100px
3 παράμετροι	margin: 25px 50px 75px;	ΠΑΝΩ margin 25px ΔΕΞΙΑ & ΑΡΙΣΤΕΡΑ margins 50px ΚΑΤΩ margin 75px
2 παράμετροι	margin: 25px 50px;	ΠΑΝΩ & ΚΑΤΩ margins 25px ΔΕΞΙΑ & ΑΡΙΣΤΕΡΑ margins 50px
1 παράμετροι	margin: 25px;	ΠΑΝΩ margin 25px ΔΕΞΙΑ margin 25px ΚΑΤΩ margin 25px ΑΡΙΣΤΕΡΑ margin 25px

ότι ισχύει για τα margins ισχύει και για τα paddings

ΠΑΡΑΔΕΙΓΜΑ

```
<!DOCTYPE html>
<html>
  <!-- V2.2.1 -->
  <head>
    <meta charset="utf-8">
    <title>Box example</title>
    <style>
      body {
        font-size: 16px;
        font-family:sans-serif;
      }
      .box{
        padding: 0.5em;
        border:0.2em solid rgb(219,45,45);
        margin: 1em;
      }
    </style>
  </head>
  <body>
    <p>Πρώτη παράγραφος</p>
    <p class="box">Δεύτερη παράγραφος </p>
    <p> Τρίτη παράγραφος </p>
  </body>
</html>
```

Σημείωση

-em=font-size που ορίσαμε στο body=16 (0.5em=8...)

-Στο βίντεο που ακολουθεί δείχνει το ύψος του περιεχομένου της παραγράφου 2 να είναι 18px πως προκύπτει αυτό αφού δεν το ορίσαμε?

-Συνήθως οι περισσότεροι browser χρησιμοποιούν default τιμή το **1.2*em**

Δηλαδή εδώ 1.2*16 περίπου=18

-Αν αλλάξουμε το font-size=22; θα πάρουμε 1.2*22 περίπου=26px

ΟΡΙΣΜΟΣ ΕΞΩΤΕΡΙΚΩΝ ΔΙΑΣΤΑΣΕΩΝ BOX ΜΕ ΙΔΙΟΤΗΤΕΣ WIDTH , HEIGHT

- οι ιδιότητες **width,height** καθορίζουν τις **διαστάσεις του περιεχομένου** και **όχι** του συνολικού box
- Αν επιθυμούμε να ορίσουμε , με τις παραμέτρους **width,height** τις **εξωτερικές** διαστάσεις του **box** μέχρι το border **χωρίς το margin** τότε πρέπει να χρησιμοποιήσουμε την ιδιότητα **box-sizing:border-box;**
- ορισμένοι προγραμματιστές χρησιμοποιούν τον παρακάτω κώδικα

```
*{
    margin:0;
    padding:0;
    box-sizing:border-box;
}
```

για να αλλάζουν την προκαθορισμένη τιμή του box-sizing για όλη την εφαρμογή

ΔΙΑΧΕΙΡΙΣΗ ΠΕΡΙΘΩΡΙΟΥ ΜΕΤΑΞΥ ΣΤΟΙΧΕΙΩΝ

-Όταν δύο διαδοχικά στοιχεία block έχουν και τα 2 ορισμένα εξωτερικά περιθώρια (margin) το περιθώριο μεταξύ τους **δεν είναι** το άθροισμα των περιθωρίων **αλλά** το μεγαλύτερο από τα 2 (**vertical margin collapse**)

-Αν **δεν** επιθυμούμε αυτή την συμπεριφορά, πρέπει να εισάγουμε μεταξύ των στοιχείων ένα άδειο στοιχείο με **padding:1px;**

ΕΞΩΤΕΡΙΚΟ ΠΕΡΙΘΩΡΙΟ (margin) ΜΕΤΑΞΥ ΥΠΟΔΟΧΕΑ(container) ΚΑΙ ΠΕΡΙΕΧΟΜΕΝΩΝ ΣΤΟΙΧΕΙΩΝ

Αν δεν υπάρχουν border,padding να διαχωρίζει το margin ενός υποδοχέα από τα margin των στοιχείων που περιέχει τότε τα margins συντίθεται και το νέο margin καταλήγει εξωτερικά του στοιχείου-υποδοχέα

ΠΑΡΑΔΕΙΓΜΑΤΑ

```
<!DOCTYPE html>
<html>
  <!-- V2.2.2 -->
  <head>
    <meta charset="utf-8">
    <style>
      /*General styles */
      #wrapper > * {/*για όλους τους άμεσους απογόνους*/
        padding: 20px;
        font-size:20px;
        border:20px solid rgba(0,0,0,0.5);/*χρώμα μαύρο με 0.5 transparency*/
      }
      /*specific boxes*/
      #wrapper header,#wrapper footer{
        background-color: blue;
        color: white;
      }
    </style>
  </head>
  <body>
    <div id="wrapper">
      <header>Header</header>
      <main>Main content</main>
      <footer>Footer</footer>
    </div>
  </body>
</html>
```

```

<!DOCTYPE html>
<html>  <!-- V2.2.2 -->
  <head>
    <meta charset="utf-8">
    <style>
      * { font-family:sans-serif;}
      div{margin:1em;
          background: lavender;}
      p { margin: 5em;
          background: red;}
      .separator{padding: 1px;}
    </style>
  </head>
  <body>
    <div>υποδοχέας 1
      <p>Πρώτη παράγραφος</p>
      <p class="separator"></p>
      <p>Δεύτερη παράγραφος</p>
    </div>
    <div>υποδοχέας 2
      <p>Πρώτη παράγραφος</p>
      <p>Δεύτερη παράγραφος</p>
    </div>
    <div>υποδοχέας 3
      <p>Πρώτη παράγραφος</p>
      <p>Δεύτερη παράγραφος</p>
      <p class="separator"></p>
    </div>
  </body>
</html>

```

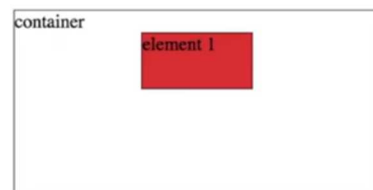
-margin:auto;

Για να στοιχίσουμε κάποιον στοιχείο στο μέσο του στοιχείου-υποδοχέα (στο πλάτος μόνο)

```

<html>  <!-- V2.2.2 -->
  <head>
    <style>
      body, html {
<!--για να μπει το 100% στο div πρέπει να οριστεί , στους γονείς -->
        height: 100%;}
      .element, .container{border:1px solid black;
        height:100%;}
      .element{height:30%;
        width:30%;
        margin:0 auto;
        background-color: crimson;}
    </style>
  </head>
  <body>
    <div class='container'>container
      <div class='element'>element 1</div>
    </div>
  </body>
</html>

```



Για αριστερή και δεξιά στοίχιση θα χρησιμοποιήσουμε την ιδιότητα float: left, right που θα δούμε στη συνέχεια.

ΔΙΑΧΕΙΡΙΣΗ ΠΕΡΙΕΧΟΜΕΝΟΥ OVERFLOW

ελέγχει την συμπεριφορά του περιεχομένου όταν αυτό δεν χωράει στον ορισμένο χώρο.

-auto: Το περιεχόμενο που δεν χωράει δεν εμφανίζεται και εμφανίζονται μπάρες κύλισης **αν αυτό είναι απαραίτητο** που επιτρέπουν στον χρήστη να δει το υπόλοιπο .

-scroll: Το περιεχόμενο που δεν χωράει δεν εμφανίζεται και εμφανίζονται μπάρες κύλισης που επιτρέπουν στον χρήστη να δει το υπόλοιπο

-hidden: Το περιεχόμενο που δεν χωράει κρύβεται

-visible: Το περιεχόμενο που δεν χωράει εμφανίζεται έξω από το κουτί(η προκαθορισμένη συμπεριφορά)

π.χ.

```
<!DOCTYPE html>
<html>  <!-- V2.2.3 -->
  <head>
    <meta charset="utf-8">
    <style>
      p{width:400px;height:2.5em;
padding:1em 1em 1em 1em;
border:1px solid black;}
      #autoscroll {overflow:auto;}
      #clipped {overflow:hidden;}
      #default {overflow:visible;}
    </style>
  </head>
  <body>
<p id="autoscroll">
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris
tempus turpis id ante mollis dignissim. Nam sed dolor non tortor
lacinia lobortis id dapibus nunc. Praesent iaculis tincidunt augue.
Integer efficitur sem eget risus cursus, ornare venenatis augue
hendrerit. Praesent non elit metus. Morbi vel sodales ligula.
</p>
<p id="clipped">
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris
tempus turpis id ante mollis dignissim. Nam sed dolor non tortor
lacinia lobortis id dapibus nunc. Praesent iaculis tincidunt augue.
Integer efficitur sem eget risus cursus, ornare venenatis augue
hendrerit. Praesent non elit metus. Morbi vel sodales ligula.
</p>
<p id="visible">
  Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris
tempus turpis id ante mollis dignissim. Nam sed dolor non tortor
lacinia lobortis id dapibus nunc. Praesent iaculis tincidunt augue.
Integer efficitur sem eget risus cursus, ornare venenatis augue
hendrerit. Praesent non elit metus. Morbi vel sodales ligula.
</p>
  </body>
</html>
```

ΟΡΙΣΜΟΣ ΥΠΟΒΑΘΡΟΥ background

-το background του box μπορεί να περιέχει χρώμα ή εικόνες το ένα πάνω από το άλλο (**background-color, background-image**)

-Το υπόβαθρο εκτείνεται μέχρι το εξωτερικό του περιγράμματος.

-Όμως με την ιδιότητα **background-clip** μπορεί να οριστεί αν το υπόβαθρο εκτείνεται στο border, στο padding ή μόνο στο περιεχόμενο.

-η συντομογραφία **background** επιτρέπει να ορισθούν με μια εντολή οι παρακάτω ιδιότητες:**background-clip,background-color,background-image,background-origin,background-position,background-repeat, background-size, και background-attachment**

π.χ.

```
<!DOCTYPE html>
<html>  <!-- V2.2.4 -->
  <head>
    <meta charset="utf-8">
    <style>
      div{
        width:500px;
        height:500px;
        color: white;
        background:black no-repeat center/20% url("https://i.picsum.photos/id/736/200/200.jpg");
      }
    </style>
  </head>
  <body>
    <div>mydiv</div>
  </body>
</html>
```

Η ΙΔΙΟΤΗΤΑ OUTLINE

Με την ιδιότητα **outline** ορίζεται ένα περίγραμμα, που μοιάζει με το border όμως αυτό σχεδιάζεται στο εξωτερικό περιθώριο (**margin**) και χρησιμοποιείται για λόγους προσβασιμότητας για να δώσει έμφαση στα ενεργά στοιχεία μιας ιστοσελίδας.

box-shadow

box-shadow: 4px 4px 5px 5px #bbb;

δέχεται 5 παραμέτρους (2 υποχρεωτικές):

1. Η **οριζόντια μετατόπιση** της σκιάς από το αρχικό πλαίσιο (**υποχρεωτική παράμετρος**) μπορεί να πάρει τις περισσότερες διαθέσιμες μονάδες μήκους και μεγέθους CSS συνήθως μετριέται σε px.
2. Η **κατακόρυφη μετατόπιση** της σκιάς απ' το αρχικό πλαίσιο (**υποχρεωτική παράμετρος**) μπορεί να πάρει τις περισσότερες διαθέσιμες μονάδες μήκους και μεγέθους CSS συνήθως μετριέται σε px.
3. Η **ακτίνα θόλωσης** - μια υψηλότερη τιμή σημαίνει ότι η σκιά διασκορπίζεται ευρύτερα. Αν αυτή η τιμή δεν οριστεί, **η προεπιλεγμένη τιμή είναι 0**, που σημαίνει ότι δεν υπάρχει θόλωση.
4. Η **ακτίνα διασποράς** - θετικές τιμές κάνουν την σκιά να επεκτείνεται, ενώ αρνητικές να συρρικνώνεται (συνήθως σε px)
5. Το **βασικό χρώμα** της σκιάς, το οποίο μπορεί να λάβει οποιαδήποτε μονάδα χρώματος CSS. Εάν δεν συμπεριλαμβάνεται, η προεπιλογή είναι **black**.

παραδείγματα

```
box-shadow:10px 10px;  
box-shadow:10px 10px 30px;  
box-shadow:10px 10px 30px 10px;  
box-shadow:10px 10px 30px 10px rgba(120,120,200,0.5);
```

ΣΚΙΑ ΧΩΡΙΣ ΚΑΤΕΥΘΥΝΣΗ ΦΩΤΕΙΝΗΣ ΠΗΓΗΣ

```
box-shadow:0px 0px 35px -10px rgba(0,0,0,0.75)
```

```
<!DOCTYPE html>  
<html>  <!-- V2.2.5 -->  
  <head>  
    <meta charset="utf-8">  
    <style>  
      div{  
        width:100px;  
        height:100px;  
        margin:30px;  
      }  
      #box1 {box-shadow:10px 10px;}  
      #box2 {box-shadow:10px 10px 30px;}  
      #box3 {box-shadow:10px 10px 30px 10px;}  
      #box4 {box-shadow:10px 10px 30px 10px rgba(120,120,200,0.5);}  
      #box5 {box-shadow:0px 0px 35px -10px rgba(0,0,0,0.75);}  
    </style>  
  </head>  
  <body>  
    <div id="box1">mydiv1</div>  
    <div id="box2">mydiv2</div>  
    <div id="box3">mydiv3</div>  
    <div id="box4">mydiv4</div>  
    <div id="box5">mydiv5</div>  
  </body>  
</html>
```

ΣΤΡΟΓΓΥΛΕΥΟΝΤΑΣ ΓΩΝΙΕΣ border-radius:τιμή;

Η border-radius απαιτεί 1 έως 4 τιμές

1 τιμή	η ίδια τιμή και για τις 4 γωνίες	border-radius:10px;
2 τιμές	πάνω-αριστερά & κάτω δεξιά πάνω-δεξιά & κάτω-αριστερά	border-radius:10px 5%;
3 τιμές	πάνω αριστερά πάνω-δεξιά & κάτω-αριστερά κάτω-δεξιά	border-radius:10px 2px 4px;
4 τιμές	πάνω αριστερά πάνω-δεξιά κάτω-δεξιά κάτω-αριστερά	border-radius:1px 0px 5px 3px;

border-radius: 20%;



Ζωγραφίζουμε ένα κύκλο (ο κύκλος σχηματίζεται μόνο σε τετραγωνικά στοιχεία με width=height)

border-radius: 50%;

border-radius: 0% 30%;



Display: inline ή block

Η ιδιότητα display ορίζει τη συμπεριφορά του box σε σχέση με το υπόλοιπο περιεχόμενο.

Η ιδιότητα display μπορεί να πάρει δύο τιμές ή block ή inline. Ανάλογα με την τιμή η συμπεριφορά του στοιχείου που έχει αυτή την ιδιότητα αλλάζει.

Όταν παίρνει την τιμή **block** τότε το box αρχίζει με νέα γραμμή και μπορεί να πάρει τις ιδιότητες width και height.

Όταν η παράμετρος display πάρει την τιμή inline τότε περιέχεται μέσα στο στοιχείο-πατέρα, δεν μπορεί να πάρει τις τιμές width και height και αγνοεί την τιμή του margin-top και margin-bottom, δεν λαμβάνονται υπόψη.

>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Mauris tempus turpis id ante mollis dignissim. Nam sed dolor non tortor lacinia lobortis id dapibus nunc.

επιλογέας { display: inline; }

>Lorem ipsum dolor sit amet, consectetur adipiscing elit.
Mauris tempus turpis id ante mollis dignissim.
Nam sed dolor non tortor lacinia lobortis id dapibus nunc.

επιλογέας { display: block; }

παράδειγμα

```
<!DOCTYPE html>
<html><!-- v2.2.7 -->
  <head>
    <meta charset="utf-8">
    <title>Box sample</title>
    <style>
      body {font-size: 16px;
            font-family: sans-serif;}
      .box {padding: 0.5em;
            border: 0.2em solid rgb(219,45,45);
            margin:1em;
            }
      .block{display:block;}
      .inline{display:inline;}
    </style>
  </head>
  <body>
    <div>
      Πρώτη παράγραφος
      <div class="box">
        Δεύτερη παράγραφος
      </div>
      Τρίτη παράγραφος
    </div>
    <div>
      Πρώτη παράγραφος1
      <div class="box inline">
        Δεύτερη παράγραφος1
      </div>
      Τρίτη παράγραφος1
    </div>
  </body>
</html>
```

Μία 3^η περίπτωση είναι η display να πάρει τιμή inline-block

Στην περίπτωση αυτή έχουμε την δυνατότητα να ορίσουμε το **width** και **height** καθώς και **margin-top**, **margin-bottom** του στοιχείου αν και το στοιχείο είναι inline, **δηλαδή δεν** προστίθεται αλλαγή γραμμής πριν και μετά από αυτό.

παράδειγμα

```
<!DOCTYPE html>
<html><!-- v2.2.7 -->
  <head>
    <meta charset="utf-8">
    <title>Box sample</title>
    <style>
      body {font-size: 16px;
            font-family: sans-serif;}
      .box {padding: 0.5em;
           border: 0.2em solid rgb(219,45,45);
           margin: 1em;
           height: 3em;
           }
      .block{display:block;}
      .inline{display:inline-block;}
    </style>
  </head>
  <body>
    <div>
      Πρώτη παράγραφος
      <div class="box inline">
        Δεύτερη παράγραφος
      </div>
      Τρίτη παράγραφος
    </div>
  </body>
</html>
```

Κάτι που θα πρέπει να τονίσουμε είναι ότι αν δεν χρησιμοποιήσω καθόλου την παράμετρο **display**, η html ορίζει προκαθορισμένη συμπεριφορά αν θα είναι **block** ή **inline** για διαφορετικά στοιχεία.

-Έτσι για παράδειγμα στοιχεία που έχουν προκαθορισμένη συμπεριφορά **block** είναι π.χ. τα στοιχεία **<p>**, **<h1>**...**<h6>** και λοιπά.

-Αντίστοιχα στοιχεία που έχουν προκαθορισμένα συμπεριφορά **inline** είναι π.χ. τα στοιχεία ****, **** που είναι **στοιχεία που δίνουν έμφαση** σε ένα τμήμα του κειμένου **χωρίς** να επιβάλλουν στην ουσία αλλαγή παραγράφου.

Μονάδες μέτρησης της CSS

α) απόλυτες μονάδες μέτρησης

-τα **pixel** (τα **εικονοστοιχεία**): Τα pixel τίθενται δίπλα στον αριθμό κολλημένα με το px που σημαίνει pixel. Θα πρέπει να προσέξουμε αν θέλω να δώσω 0 pixel τότε η μονάδα το pixel μπορεί και να παραληφθεί. π.χ. `margin 0 auto`

-Άλλες απόλυτες μονάδες μέτρησης είναι τα **mm**, **cm** ίντσες, **points**, **picas** και λοιπά

β) Σχετικές μονάδες μέτρησης

-**em**: είναι σχετική μονάδα μέτρησης με βάση την τρέχουσα τιμή μεγέθους γραμματοσειράς (ύψος M) **1em** είναι το μέγεθος της γραμματοσειράς του τρέχοντος στοιχείου (πλάτος/ύψος ενός κεφαλαίου M)

-Το **προεπιλεγμένο μέγεθος** γραμματοσειράς που δίνεται σε ιστοσελίδες από φυλλομετρητές πριν την εφαρμογή του στυλ **CSS** είναι **16px** άρα **1em=16px** για ένα στοιχείο από προεπιλογή.

-Ωστόσο τα μεγέθη γραμματοσειρών **κληρονομούνται από τους γονείς τους**, έτσι το ισοδύναμο ενός em μπορεί να γίνει περίπλοκο.

-rem, ex, ch

rem με βάση το μέγεθος κεφαλαίου M του **root element** (ενώ το em παίρνει σαν βάση τον πατέρα του)

ex με βάση το ύψος του x της τρέχουσας γραμματοσειράς

ch με βάση το πλάτος του μηδέν (0)

-ποσοστό άλλου στοιχείου

% ποσοστό του υποδοχέα που περιέχει το στοιχείο

vw (viewport width), % ποσοστό από το πλάτος του παράθυρου,

vh (viewport height) % ποσοστό από το ύψος του παράθυρου

vmin (viewport min) % ποσοστό του μεγέθους της μικρότερης διάστασης του παράθυρου

vmax (viewport max) ποσοστό του μεγέθους της μεγαλύτερης διάστασης του παράθυρου

παράδειγμα

```
<!DOCTYPE html>
<html><!-- v2.2.8 -->
  <head>
    <meta charset="utf-8">
    <title>Box sample</title>
    <style>
      .boxes {margin:10px;
        font-size: 2em;
        color: white;
        height: 150px;
        border: 2px solid black;}
      .boxes:nth-child(1){background-color: red;
        width: 400px;}
      .boxes:nth-child(2){background-color: blue;
        width: 75%;}
    </style>
  </head>
  <body>
    <div>
      <div class="boxes">box 1</div>
      <div class="boxes">box 2</div>
    </div>
  </body>
</html>
```

ΧΡΩΜΑΤΑ

-μία ιστοσελίδα αν δεν ορίσουμε το χρώμα της θα έχει υπόβαθρο background-color λευκό, τα γράμματα θα είναι μαύρα, οι υπερσύνδεσμοι θα είναι μπλε και θα αλλάξουν σε κάποια απόχρωση του μωβ εάν τους έχουμε ήδη επισκεφτεί, είναι δηλαδή σε κατάσταση visited.

-Το χρωματικό μοντέλο της html css επιτρέπει την εμφάνιση πολλών αποχρώσεων στηρίζεται σε τρία κύρια χρώματα.

-**RGB**(Είναι το κόκκινο, το πράσινο και το μπλε red green blue).

κάθε χρώμα παίρνει τιμές από 0-255 (από έχω ένα συνδυασμό 256 X 256 X 256 διαφορετικούς συνδυασμούς. Σύνολο πάνω από 16 εκατομμύρια αποχρώσεις.

-αν το **R** έχει τιμή **255** και **G=0 B=0** τότε έχω το απόλυτο **κόκκινο 255,0,0** ομοίως για πράσινο και μπλε

ιδιότητες στοιχείων που συνδέονται με το χρώμα

color:red;	χρώμα των γραμμάτων
background-color:red;	χρώμα υποβάθρου
border-color:red;	χρώμα του border
font-color:red;	χρώμα του font

Τρόποι ορισμού χρώματος

Λεκτικά (υπάρχουν 165 keywords)	p{background-color:red;}
Δεκαεξαδική τιμή χρώματος	p{background-color:#ff0000;}
rgb(r,g,b) r,g,b(0-255)	p{background-color:rgb(255,0,0);}
rgba(r,g,b,a) r,g,b(0-255) a=ποσοστό αδιαφάνειας	p{background-color:rgba(255,0,0,0.5);}

keyword	Hex
black	#000000
green	#008000
silver	#C0C0C0
lime	#00FF00
grey	#808080
olive	#808000
white	#FFFFFF
yellow	#FFFF00

keyword	Hex
maroon	#800000
navy	#000080
red	#FF0000
blue	#0000FF
purple	#800080
teal	#008080
fuchsia	#FF00FF
aqua	#00FFFF

ΔΙΑΦΑΝΕΙΑ ΧΡΩΜΑΤΟΣ rgba()

-τιμές a από 0 έως 1 a=0 τελείως διαφανές a=1 τελείως αδιαφανές

-εναλλακτικός τρόπος να οριστεί η διαφάνεια ενός στοιχείου είναι η ιδιότητα **opacity**(αδιαφάνεια)

Ορίζει την διαφάνεια όλων των επιλεγμένων στοιχείων και των παιδιών τους.

π.χ. p {background-color:rgba(255,0,0,0.5);}

γίνεται

p{background-color:rgba(255,0,0); opacity:0.5; }

ΣΗΜΕΙΩΣΗ:η διαφορά είναι ότι με τον δεύτερο τρόπο επηρεάζονται και τα παιδιά του στοιχείου

παράδειγμα

```
<!DOCTYPE html>
<html><!-- v2.2.9 -->
  <head>
    <style>
      body {
        background-color: beige;
      }
      p{border: black solid 1px;
        margin:1em;
        padding:1em;
      }
      p:nth-child(1){
        background-color: rgba(255,0,0,0.5);}
      p:nth-child(2){
        background-color:rgb(255,0,0);
        opacity:0.5;
      }
    </style>
  </head>
  <body>
    <div>
      <p id="box">box 1</div>
      <p id="box">box 2</div>
    </div>
  </body>
</html>
```

ΕΜΦΑΝΙΣΗ ΚΕΙΜΕΝΟΥ

Το κείμενο τοποθετείται μέσα στο content ενός στοιχείου box και συμπεριφέρεται σαν μια σειρά γραμμικών inline στοιχείων έως ότου φτάσει το τέλος της γραμμής ή εκτός αν η γραμμή διακοπεί από
.

Το κείμενο εμφανίζεται με βάση το **στυλ γραμματοσειράς** (εμφάνιση χαρακτήρων) και **στυλ διάταξης** κειμένου (αποστάσεις, θέση κειμένου).

Ιδιότητες εμφάνισης κειμένου

- color χρώμα
- font-family γραμματοσειρά
- font-size μέγεθος
- font-style normal/italic/oblique στυλ
- font-weight ένταση χαρακτήρων
- text-transform none/uppercase/lowercase/capitalize/full-width
- text-decoration none/underline/overline/line-through
- text-shadow σκίαση

color

-δίνει χρώμα στο κείμενο

-Η ιδιότητα color **επηρεάζει** ακόμη το χρώμα της υπογράμμισης ή υπερ-γράμμισης (underline, overline) που τοποθετείται στο κείμενο μέσω ιδιότητας text-decoration

π.χ. pre { color:red;}

font-family

-ορίζει μία γραμματοσειρά

→p {font-family:arial;}

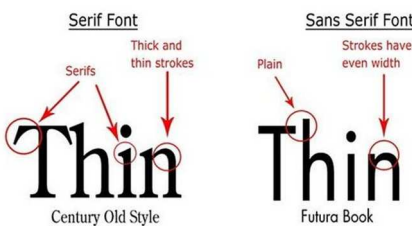
-ορίζει μία ακολουθία γραμματοσειρών

→p{font-family: "Trebuchet MS",Verdana,sans-serif;}

με την σειρά που τις βάζω η πρώτη γραμματοσειρά που θα βρεθεί διαθέσιμη αυτή και θα χρησιμοποιηθεί

Βασικές οικογένειες γραμματοσειρών

Term	Definition	Example
serif	Γραμματοσειρές με serifs (έντονοι άκρες στα typefaces)	serif font
sans-serif	Γραμματοσειρές χωρίς serifs.	sans-serif font
monospace	Γραμματοσειρές με χαρακτήρες ίσου πλάτους.	monospace font
cursive	Γραμματοσειρές που προσομοιάζουν χειρόγραφο γραφή	cursive font



web safe fonts

Υπάρχουν γραμματοσειρές διαθέσιμες σε όλα τα συστήματα . Ονομάζονται **web safe fonts**.

Γραμματοσειρά	Οικογένεια	Εναλλακτικές γραμματοσειρές
Arial	sans-serif	Η Arial είναι ευρέως διαδεδομένη, Επίσης η γραμματοσειρά Helvetica που είναι σχεδόν ίδια με την Arial
Courier New	monospace	Σε μερικά λειτουργικά συναντάται ακόμη η παλαιότερη γραμματοσειρά Courier. Καλό είναι να ορίζονται και οι δύο Courier και Courier New .
Georgia	serif	
Times New Roman	serif	Σε μερικά λειτουργικά συναντάται η παλαιότερη γραμματοσειρά Times. Καλή πρακτική να χρησιμοποιούνται και οι δύο ως εναλλακτικές γραμματοσειρές.
Trebuchet MS	sans-serif	Όχι διαθέσιμη σε κινητές συσκευές
Verdana	sans-serif	

ΜΕΓΕΘΟΣ font-size

Το μέγεθος γραμματοσειράς ενός στοιχείου κληρονομείται από το γονικό στοιχείο του.

Όλα αυτά ξεκινούν με το στοιχείο ρίζας ολόκληρου του εγγράφου <html> το μέγεθος γραμματοσειράς του οποίου έχει οριστεί σε **16px** ως πρότυπο σε όλους τους φυλλομετρητές.

Μονάδες μεγέθους γραμματοσειράς.

px (pixels): απόλυτη τιμή ύψους χαρακτήρων σε px

em: 1 em είναι ίσο με το ύψος του γονικού στοιχείου του τρέχοντος στοιχείου, το em χρησιμοποιείται ως μονάδα μέτρησης απόστασης όχι μόνο κειμένου

rem: όπως το em, αφορά όμως το ριζικό στοιχείο του εγγράφου και όχι το γονικό

παραδείγματα

```
h1 {font-size:40px;}
```

```
p {font-size:14px;}
```

```
h1 {font-size:2.5em;}/*40px/16=2.5em*/
```

```
p {font-size:0.875em;}/*14px/16=0.875em*/
```

```
body {font-size:100%;}
```

-Καλή πρακτική είναι να ορίσουμε font-size:100% και εν συνεχεία να χρησιμοποιούμε em

ΣΤΥΛ ΚΕΙΜΕΝΟΥ font-style

-**normal**: όρθια γραφή, καταργεί πλάγια (italics) αν έχει χρησιμοποιηθεί.

-**italic**: χρησιμοποιεί την πλάγια έκδοση της γραμματοσειράς αν υπάρχει, αν δεν υπάρχει προσομειώνει την γραμματοσειρά με oblique

-**oblique**: Προσομείωση του italic font με κλίση της όρθιας γραφής.

```
p.normal {font-style:normal;}
```

```
p.italic {font-style:italic;}
```

```
p.oblique {font-style:oblique;}
```

Καλημέρα

Καλημέρα

Καλημέρα

ΕΝΤΑΣΗ ΓΡΑΦΗΣ font-weight

-**normal**: κανονική γραφή

-**bold**: **έντονη** γραφή

-**lighter**: η ένταση μειώνεται ένα βήμα από την τρέχουσα τιμή

-**bolder**: η ένταση αυξάνει ένα βήμα από την τρέχουσα τιμή

-**100-900**: τιμές για πιο λεπτομερή έλεγχο της έντασης αν απαιτείται

π.χ. p.bold {font-weight:bold;}

text-transform

-**none**: κανένας μετασχηματισμός.

-**uppercase**: μετασχηματίζει το κείμενο **ΣΕ ΚΕΦΑΛΑΙΑ**.

-**lowercase**: μετασχηματίζει το κείμενο σε **πεζά**.

-**capitalize**: μετασχηματίζει το κείμενο ώστε όλες οι λέξεις να αρχίζουν με κεφαλαίο γράμμα

π.χ.

```
p.uppercase {text-transform:uppercase;}
```

```
p.lowercase {text-transform:lowercase;}
```

```
p.capitalize {text-transform:capitalize;}
```

text-decoration (συντομογραφία της text-decoration-line)

- overline** πάνω υπογράμμιση
- line-through** διαγραφή κειμένου
- underline** υπογράμμιση
- none** τίποτα
- συνδυασμός των παραπάνω** π.χ. underline overline

π.χ.

h1 {text-decoration:overline;}	This is heading 1
h2 {text-decoration:line-through;}	This is heading 2
h3 {text-decoration:underline;}	<u>This is heading 3</u>
p {text-decoration:none;}	

text-decoration-style(στυλ γραμμής) και text-decoration-color(χρώμα της γραμμής)

-**solid , double ,dotted ,dashed ,wavy**

χρησιμοποιείται συνδυαστικά με την text-decoration

σκιά text-shadow

π.χ text-shadow: 4px 4px 5px #bbb

δέχεται 4 παραμέτρους .

1. Η **οριζόντια μετατόπιση** της σκιάς από το αρχικό κείμενο αυτή μπορεί να πάρει τις περισσότερες διαθέσιμες μονάδες μήκους και μεγέθους CSS, αλλά συνήθως μετριέται σε px (**υποχρεωτική παράμετρος**).
2. Η **κατακόρυφη μετατόπιση** της σκιάς από το αρχικό κείμενο. Συμπεριφέρεται όπως και η οριζόντια μετατόπιση, εκτός από το ότι μετακινεί τη σκιά πάνω / Κάτω, όχι αριστερά / δεξιά. (**υποχρεωτική παράμετρος**).
3. Η **ακτίνα θόλωσης** - μια υψηλότερη τιμή σημαίνει ότι η σκιά διασκορπίζεται ευρύτερα. Αν αυτή η τιμή δεν οριστεί, η προεπιλεγμένη τιμή είναι 0, που σημαίνει ότι δεν υπάρχει θόλωση. Αυτό μπορεί να πάρει τις περισσότερες διαθέσιμες μονάδες μήκους και μεγέθους CSS.
4. Το **βασικό χρώμα** της σκιάς, το οποίο μπορεί να λάβει οποιαδήποτε μονάδα χρώματος CSS. Εάν δεν συμπεριλαμβάνεται, η προεπιλογή είναι μαύρη.

στοίχιση κειμένου: text-align

Καθορίζει τη στοίχιση του κειμένου μέσα στον υποδοχέα που ευρίσκεται

-left: αριστερή στοίχιση.

-right: δεξιά στοίχιση

-center: στοίχιση στο κέντρο

-justify: αμφίπλευρη στοίχιση, συχνά συνδυάζεται με συλλαβισμό hyphens ώστε λέξεις να χωρίζονται σε διαφορετικές γραμμές.

Υψος γραμμής line-height

-Η ιδιότητα αυτή καθορίζει το ύψος κάθε γραμμής κειμένου.

-Μετρείται σε μονάδες μήκους ή σε αδιάστατο αριθμό χωρίς μονάδες, οπότε στην περίπτωση αυτή θεωρείται πολλαπλάσιο του τρέχοντος ύψους γραμματοσειράς.

letter-spacing/word-spacing

Οι παράμετροι αυτές επιτρέπουν στο χρήστη να ορίσει την απόσταση μεταξύ των γραμμάτων και των λέξεων σε ένα κείμενο. Έτσι μπορούμε να αραιώσουμε ένα τμήμα του κειμένου.

text-indent(εσοχή)

Η παράμετρος text-indent καθορίζει το κατά πόσο με μονάδα μέτρησης συνήθως px ή em, κατά πόσο θα είναι πιο μέσα, την εσοχή δηλαδή της πρώτης γραμμής του κειμένου.

white-space

-καθορίζει πώς χειριζόμαστε τους κενούς χαρακτήρες και τους χαρακτήρες αλλαγής γραμμής.

-normal οι ακολουθίες κενών χώρων εμφανίζονται ως ένα κενό. Το κείμενο αναδιπλώνεται μόνο όταν είναι απαραίτητο όχι με '\n'. (προεπιλογή)

-nowrap Οι ακολουθίες κενών χώρων εμφανίζονται ως ένα κενό. Το κείμενο δεν αναδιπλώνεται.

white-space: normal;

"ΑΠΟΛΕΙΠΕΙΝ Ο ΘΕΟΣ ΑΝΤΩΝΙΩΝ" Σαν έξαφνα, ώρα μεσάνυχτ',
ακουσθεί αόρατος θίασος να περνά με μουσικές ερμείσεις, με
φωνές— την τύχη σου που ενδίδει πια, τα έργα σου που
απέτυχαν, τα σχέδια της ζωής σου που βγήκαν όλα πλάνες, μη
ανωφέλετα θρηνήσεις. Σαν έτοιμος από καιρό, σα θαρραλέος,
αποχαιρέτα την, την Αλεξάνδρεια που φεύγει. Προ πάντων να μη
γελασθείς, μην πεις πως ήταν ένα όνειρο, πως απατήθηκεν η
ακοή σου— μάταιες ελπίδες τέτοιες μην καταδεχθείς. Σαν έτοιμος
από καιρό, σα θαρραλέος, σαν που ταιριάζει σε που αξιώθηκες
μια τέτοια πόλι, πλησίασε σταθερά προς το παράθυρο, κι άκουσε
με συγκίνηση, αλλ' όχι με των δειλών τα παρακάλια και
παράπονα, ως τελευταία απόλαυσι τους ήχους, τα ερμείσια
όργανα του μυστικού θιάσου, κι αποχαιρέτα την, την Αλεξάνδρεια
που χάνεις. Επιμέλεια Γ. Π. Σαββίδη. Τα Ποιήματα, Τ. Α' 1897 -
1918, Ίκαρος 1963

white-space: nowrap;

"ΑΠΟΛΕΙΠΕΙΝ Ο ΘΕΟΣ ΑΝΤΩΝΙΩΝ" Σαν έξαφνα, ώρα μεσάνυχτ', ακουσθεί α

-pre Τα κενά διατηρούνται. Το κείμενο αλλάζει γραμμή με '\n' (όπως η ετικέτα <pre> της HTML)

-pre-wrap Τα κενά διατηρούνται. Αλλαγή γραμμής μόνο με '\n' καθώς και όπου είναι απαραίτητο λόγω διαστάσεων του υποδοχέα.

white-space: pre;

"ΑΠΟΛΕΙΠΕΙΝ Ο ΘΕΟΣ ΑΝΤΩΝΙΩΝ"
Σαν έξαφνα, ώρα μεσάνυχτ', ακουσθεί
αόρατος θίασος να περνά
με μουσικές ερμείσεις, με φωνές—
την τύχη σου που ενδίδει πια, τα έργα σου
που απέτυχαν, τα σχέδια της ζωής σου
που βγήκαν όλα πλάνες, μη ανωφέλετα θρηνήσεις.
Σαν έτοιμος από καιρό, σα θαρραλέος,
αποχαιρέτα την, την Αλεξάνδρεια που φεύγει.
Προ πάντων να μη γελασθείς, μην πεις πως ήταν
ένα όνειρο, πως απατήθηκεν η ακοή σου—
μάταιες ελπίδες τέτοιες μην καταδεχθείς.
Σαν έτοιμος από καιρό, σα θαρραλέος,
σαν που ταιριάζει σε που αξιώθηκες μια τέτοια πόλι,
πλησίασε σταθερά προς το παράθυρο,
κι άκουσε με συγκίνηση, αλλ' όχι
με των δειλών τα παρακάλια και παράπονα,
ως τελευταία απόλαυσι τους ήχους,
τα ερμείσια όργανα του μυστικού θιάσου,
κι αποχαιρέτα την, την Αλεξάνδρεια που χάνεις.
Επιμέλεια Γ. Π. Σαββίδη. Τα Ποιήματα, Τ. Α' 1897 - 1918, Ίκαρος

white-space: pre-wrap;

"ΑΠΟΛΕΙΠΕΙΝ Ο ΘΕΟΣ ΑΝΤΩΝΙΩΝ"
Σαν έξαφνα, ώρα μεσάνυχτ', ακουσθεί
αόρατος θίασος να περνά
με μουσικές ερμείσεις, με φωνές—
την τύχη σου που ενδίδει πια, τα έργα σου
που απέτυχαν, τα σχέδια της ζωής σου
που βγήκαν όλα πλάνες, μη ανωφέλετα θρηνήσεις.
Σαν έτοιμος από καιρό, σα θαρραλέος,
αποχαιρέτα την, την Αλεξάνδρεια που φεύγει.
Προ πάντων να μη γελασθείς, μην πεις πως ήταν
ένα όνειρο, πως απατήθηκεν η ακοή σου—
μάταιες ελπίδες τέτοιες μην καταδεχθείς.
Σαν έτοιμος από καιρό, σα θαρραλέος,
σαν που ταιριάζει σε που αξιώθηκες μια τέτοια πόλι,
πλησίασε σταθερά προς το παράθυρο,
κι άκουσε με συγκίνηση, αλλ' όχι
με των δειλών τα παρακάλια και παράπονα,
ως τελευταία απόλαυσι τους ήχους,
τα ερμείσια όργανα του μυστικού θιάσου,
κι αποχαιρέτα την, την Αλεξάνδρεια που χάνεις.
Επιμέλεια Γ. Π. Σαββίδη. Τα Ποιήματα, Τ. Α' 1897 - 1918, Ίκαρος
1963

-pre-line Ακολουθίες κενών χαρακτήρων εμφανίζονται ως ένα κενό. Το κείμενο αναδιπλώνεται όταν είναι απαραίτητο, ή με '\n'

white-space: pre-line;



hyphens

Η ιδιότητα αυτή ορίζει το συλλαβισμό των λέξεων στο τέλος της γραμμής. Μπορεί να αποφεύγεται (**none**), σε ορισμένα σημεία (**manual**), ή αυτόματα από τον φυλλομέτρητη (**auto**).

hyphens: none;

hyphens: manual;

hyphens: auto;

Προσοχή: οι κανόνες συλλαβισμού εξαρτώνται από τη γλώσσα.

Στην HTML η γλώσσα ορίζεται από την παράμετρο **lang**

Ο φυλλομετρητής θα συλλαβίσει το κείμενο αν έχει οριστεί η τιμή της παραμέτρου και υπάρχει λεξικό συλλαβισμού.

text-overflow

-Η ιδιότητα αυτή καθορίζει πως οι χρήστες ενημερώνονται για κείμενο που δεν έχει εμφανιστεί λόγω έλλειψης χώρου.

-Μπορεί: να αποκοπεί να δείξουμε '...' (ellipsis), ή να δείξουμε κάποια άλλη συμβολοσειρά, πχ '---'.

-Η ιδιότητα *text-overflow* δεν υποχρεώνει το κείμενο να υπερχειλίσει.

Η ιδιότητα *overflow* καθορίζει αν το κείμενο θα υπερχειλίσει

π.χ.

text-overflow: none;

text-overflow: ellipsis;

text-overflow: "---";

ΕΜΦΑΝΙΣΗ ΛΙΣΤΩΝ

Ιδιότητες εμφάνισης λιστών

list-style-type: Ορίζει τον τύπο κουκίδων που θα χρησιμοποιηθούν στη λίστα, για παράδειγμα, στρογγυλές ή κυκλικές κουκίδες για μια μη-ταξινομημένη λίστα ή αριθμούς, γράμματα ή ρωμαϊκούς αριθμούς για μια ταξινομημένη λίστα. π.χ. `ul{list-style-type:circle;}`

`list-style-type:disc; /*default*/`

`list-style-type:circle;`

`list-style-type:square;`

`list-style-type:decimal;`

`list-style-type:lower-greek;`

`list-style-type:lower-latin;`

`list-style-type:lower-roman;`

`list-style-type:none; /*keyword value*/`

list-style-position: Καθορίζει αν οι κουκκίδες εμφανίζονται μέσα (**inside**) στα στοιχεία της λίστας ή εκτός(**outside**) αυτών πριν από την έναρξη κάθε στοιχείου.

<code>ol {list-style-type:lower-roman; list-style-position:inside;} </code>	i. bananas ii. apples and oranges iii. pears
<code>ol {list-style-type:lower-roman; list-style-position:outside;} </code>	i. bananas ii. apples and oranges iii. pears

list-style-image: Μας επιτρέπει να χρησιμοποιήσετε μια ειδική εικόνα για την κουκκίδα και όχι ένα απλό τετράγωνο ή κύκλο.

<pre><!DOCTYPE html> <html lang="el"> <head> <meta charset="UTF-8"> <title>list-title</title> <style> h1,li {font-family: sans-serif;} li{list-style-image: url("sqpurple.gif"); font-size: 2em;} </style> </head> <body> <h1>Ποτά</h1> καφέ τσάι χυμός </body> </html></pre>	<p>Ποτά</p> <ul style="list-style-type: none">■ καφέ■ τσάι■ χυμός
--	---

Σημείωση: υπάρχει και συντομογραφία `list-style` π.χ

```
ul{
  list-style:square url("sqpurple.gif") inside;
}
```

ΕΜΦΑΝΙΣΗ ΥΠΕΡΣΥΝΔΕΣΜΩΝ

ΔΕΝ ΕΙΝΑΙ ΚΑΛΗ ΠΡΑΚΤΙΚΗ ΝΑ ΑΛΛΑΖΟΥΜΕ ΤΟΝ ΤΡΟΠΟ ΠΟΥ ΕΧΟΥΝ ΣΥΝΗΘΙΣΕΙ ΟΙ ΧΡΗΣΤΕΣ ΝΑ ΒΛΕΠΟΥΝ ΤΟΥΣ ΥΠΕΡΣΥΝΔΕΣΜΟΥΣ

Με χρήση ψευδοκλάσεων μπορούμε να ορίσουμε την εμφάνιση υπερσυνδέσμων

```
a:link {color:#FF0000;} /*unvisited link*/  
a:visited {color:#00FF00;} /*visited link*/  
a:hover {color:#FF00FF;} /*mouse over link*/  
a:active {color:#0000FF;} /*selected link*/
```

```
a:link {background-color:#B2FF99;}  
a:link {text-decoration:none;}
```

ΕΜΦΑΝΙΣΗ ΠΛΑΙΣΙΩΝ(border)

π.χ. border:solid black 1px;

none: Defines no border

dotted: Defines a dotted border

dashed: Defines a dashed border

solid: Defines a solid border

double: Defines two borders. The width of the two borders are the same as the border-width value

groove: Defines a 3D grooved border. The effect depends on the border-color value

ridge: Defines a 3D ridged border. The effect depends on the border-color value

inset: Defines a 3D inset border. The effect depends on the border-color value

outset: Defines a 3D outset border. The effect depends on the border-color value

ΤΡΟΠΟΙ ΓΙΑ ΝΑ ΑΠΟΚΡΥΨΟΥΜΕ ΕΝΑ ΑΝΤΙΚΕΙΜΕΝΟ

-Με την χρήση της ιδιότητας **display:none**

εκτός από τις τιμές `block`,`inline`,`inline-block` μπορεί να πάρει την τιμή **none**

στην περίπτωση αυτή το στοιχείο γίνεται **αόρατο** και **δεν** διατηρεί το χώρο **ούτε** μπορεί να δεχτεί events(να επιλεγεί) π.χ. `img{display:none;}/*` η εικόνα εξαφανίζεται και δεν διατηρεί τον χώρο της*/

-Με την χρήση της ιδιότητας **visibility:hidden**

Το `visibility` μπορεί να πάρει τιμές

visibility:visible; **ορατό**

visibility:hidden; **αόρατο**(αλλά διατηρεί τον χώρο του) ενώ δεν μπορεί να δεχτεί events

π.χ. `img{visibility:hidden;}/*` η εικόνα εξαφανίζεται αλλά διατηρεί το χώρο της*/

-Με την χρήση της ιδιότητας **opacity:0**

Η ιδιότητα αυτή παίρνει τιμές από 0 έως 1

`opacity=0;` σημαίνει **αόρατη**

`opacity=1;` σημαίνει **ορατή**

π.χ. `img {opacity:0;}/*` η εικόνα δεν φαίνεται (είναι αόρατη) αλλά διατηρεί τον χώρο της και δέχεται events*/

	<code>display: none;</code>	<code>visibility: hidden;</code>	<code>opacity: 0;</code>
καταλαμβάνει χώρο;	✗	✓	✓
δέχεται events;	✗	✗	✓

ΘΕΣΗ ΣΤΟΙΧΕΙΩΝ ΕΚΤΟΣ ΡΟΗΣ**Η ιδιότητα position**

Ιδιότητα που καθορίζει τη θέση ενός στοιχείου στην σελίδα εκτός της φυσικής του θέσης,παίρνει τιμές:

absolute/relative/fixed/static/sticky σε συνδυασμό με τις **top,bottom,left,right**

position:static; (default τιμή)

Το τοποθετεί στην **φυσιολογική ροή** της σελίδας. **Οι τιμές των ιδιοτήτων top,left,right,bottom αγνοούνται.**

position:relative;

Το στοιχείο τοποθετείται σε κάποια σχετική θέση ως προς την φυσιολογική θέση του **με βάση** τις τιμές των ιδιοτήτων **top,left,right,bottom**. Τα υπόλοιπα στοιχεία όμως παραμένουν στη θέση που καθορίζει η ροή της ιστοσελίδας

position:fixed;

Το στοιχείο παραμένει σε σταθερή θέση ως προς το viewport. Η θέση ορίζεται από τις ιδιότητες **top,bottom,left,right** ακόμα και αν το υπόλοιπο περιεχόμενο της σελίδας κυλάει σε διαφορετική θέση(scrolling page)

position:absolute;

Το στοιχείο τοποθετείται **εκτός της ροής** του αρχείου σε θέση που ορίζεται από τις ιδιότητες **top,left,right,bottom** του αντικειμένου αναφοράς στο οποίο εντάσσεται.

Αν δεν εντάσσεται σε κάποιο άλλο στοιχείο τότε τοποθετείται σε σχετική θέση ως προς το στοιχείο `<body>`

Σημείωση: Οι ιδιότητες **top, right, bottom, left** ορίζουν παράλληλες αποστάσεις από τις αντίστοιχες πλευρές του στοιχείου που περιέχονται.

ΕΠΙΚΑΛΥΨΗ ΑΝΤΙΚΕΙΜΕΝΩΝ

z-index:ακέραιος;

Επιτρέπει να δημιουργηθούν πολλαπλά επίπεδα.

Τα αντικείμενα με μεγαλύτερη τιμή του **z-index** εμφανίζονται μπροστά από εκείνα με χαμηλότερο **z-index** default τιμή το z-index:0;

ΕΠΑΛΛΗΛΙΑ ΚΑΝΟΝΩΝ (cascade)/ ΚΛΗΡΟΝΟΜΙΚΟΤΗΤΑ

-CSS(Cascading Style Sheets - επάλληλα φύλλα στυλ) που σημαίνει ότι επάλληλοι-πολλαπλοί κανόνες μπορεί να ισχύουν και πρέπει να εφαρμοστούν για το ίδιο στοιχείο.

-Όταν υπάρχουν πολλαπλοί κανόνες επικρατούν κάποιοι με βάση τους εξής παράγοντες:

α)Σημασία: (!important)

π.χ. CSS: `p { background-color:gray; color: red !important ;}`
`#thing { color: green;}`

HTML: `<p id="thing">Θα είναι κόκκινο.</p>`

Σημείωση:ισχύει μόνο για την ιδιότητα που μπαίνει λίγο πριν το ;.
Αν θέλω παραπάνω ιδιότητες τότε το βάζω και σε αυτές.

β)εξειδίκευση (inline >id>class>element)

inline →ορισμός μέσα στο στοιχείο π.χ. `<p style="....">`

γ)Σειρά (μετέπειτα επιλογείς επικρατούν των παλαιότερων)

ΔΙΑΧΕΙΡΙΣΗ ΚΛΗΡΟΝΟΜΙΚΟΤΗΤΑΣ ΚΑΝΟΝΩΝ

-inherit:

Ορίζει η τιμή της ιδιότητας που εφαρμόζεται σε ένα επιλεγμένο στοιχείο να είναι ίδια με εκείνη του γονικού στοιχείου.

-initial:

Ρυθμίζει την τιμή ιδιότητας να είναι ίδια με την τιμή που έχει οριστεί για το στοιχείο στο προεπιλεγμένο φύλλο στυλ του φυλλομετρητή.

Εάν δεν έχει οριστεί τιμή από τον φυλλομετρητή τότε η τιμή της ιδιότητας κληρονομείται.

-unset:

Επαναφέρει την ιδιότητα στη φυσική της τιμή πράγμα που σημαίνει ότι εάν η ιδιότητα κληρονομείται φυσικά ενεργεί όπως η **inherit** διαφορετικά λειτουργεί όπως η **initial**

```
body { color: green; font-size:2em;} • Default link color
.my-class-1 a { color: inherit; } • Inherit the link color
.my-class-2 a { color: initial; } • Initial link color
.my-class-3 a { color: unset; } • Unset the link color
```

```
<ul>
  <li>Default <a href="#">link</a> color</li>
  <li class="my-class-1">Inherit the <a href="#">link</a> color</li>
  <li class="my-class-2">Initial <a href="#">link</a> color</li>
  <li class="my-class-3">Unset the <a href="#">link</a> color</li>
</ul>
```

ΜΕΤΑΒΛΗΤΕΣ ΣΤΗΝ CSS

οι μεταβλητές CSS είναι οντότητες που περιέχουν τιμές τις οποίες επιθυμούμε να επαναχρησιμοποιήσουμε στο έγγραφο CSS (**ΟΧΙ ΣΤΟΝ INTERNET EXPLORER**).

Είναι ιδιότυπες ιδιότητες που ορίζονται από τον προγραμματιστή.

Αρχίζουν με τους χαρακτήρες -- π.χ. `:root { --main-color:black;}`

Έχουμε πρόσβαση σε μια μεταβλητή μέσω της συνάρτησης `var()`

π.χ. `.container { color:var(--main-color);}`

ΕΜΒΕΛΕΙΑ

-Οι μεταβλητές CSS έχουν εμβέλεια στο στοιχείο που ορίζονται και στα παιδιά του.

-Έχουν **καθολική** εμβέλεια όταν ορίζονται στο ψευδο-στοιχείο `:root`

π.χ. `:root { --bg-color: #efefef;}`

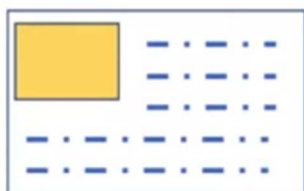
ΔΙΑΤΑΞΗ ΠΕΡΙΕΧΟΜΕΝΟΥ ΜΕ float

Η ιδιότητα float μπορεί καθορίζει τη θέση ενός στοιχείου στα δεξιά ή αριστερά ενός υποδοχέα.

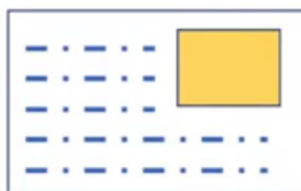
Το στοιχείο τότε δεν ακολουθεί την προκαθορισμένη ροή

float:left ή float:right το κείμενο και τα υπόλοιπα inline περιστοιχίζουν το στοιχείο

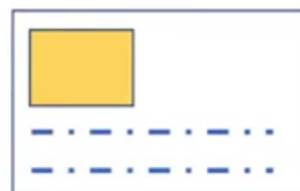
float:none τότε το κείμενο ακολουθεί



float:left



float:right



float:none

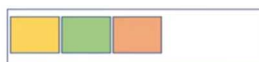
ΟΡΙΖΟΝΤΙΑ ΔΙΑΤΑΞΗ ΣΤΟΙΧΕΙΩΝ

1. **χωρίς την ιδιότητα float**, τα στοιχεία τύπου block στοιβάζονται το ένα κάτω από το άλλο **ακόμη και αν** το μέγεθος τους επιτρέπει διάταξη δίπλα-δίπλα.

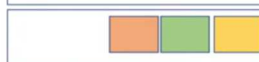


2. Η ιδιότητα float μας επιτρέπει να επιτύχουμε **οριζόντια διάταξη στοιχείων**.

`float: left;`



`float: right;`



ΠΩΣ ΘΑ ΕΠΙΤΡΕΨΟΥΜΕ ΣΕ ΕΝΑ ΥΠΟΔΟΧΕΑ ΝΑ ΠΕΡΙΕΧΕΙ ΤΑ floats παιδιά του

Στην περίπτωση που ένα στοιχείο υποδοχέας περιέχει floats για να του επιτρέψουμε να περιέχει τα παιδιά του, πρέπει να του δώσουμε την ιδιότητα `display:flow-root` η οποία επιτρέπει τη δημιουργία ενός νέου **block formatting content**

π.χ. αν το στοιχείο `.container` είναι μαύρο και τα floats `.div1`, `.div2` cyan και red, το στοιχείο `.container` πρέπει να έχει την ιδιότητα `display:flow-root`

```
<div id="container">καλή σας μέρα  
  <div id="div1"></div>  
  <div id="div2"></div>  
</div>
```



https://developer.mozilla.org/en-US/docs/Learn/CSS/CSS_layout/Floats