

## Σχεδιασμός Ιστοσελίδων II

### Μαθήματα CSS - Μέρος 1 - Εισαγωγή

#### *Εισαγωγή στα CSS*

Τα Διαδοχικά Φύλλα Στυλ (CSS, Cascading Style Sheets) αποτελούν ένα πολύ καλό εργαλείο για να μπορούμε να αλλάζουμε την εμφάνιση και τη διάταξη (layout) των ιστοσελίδων μας. Μπορούν να μας γλυτώσουν από πολύ χρόνο και κόπο και μας δίνουν τη δυνατότητα να σχεδιάζουμε τις ιστοσελίδες μας με μια εντελώς καινούργια φιλοσοφία. Η κατανόηση των CSS απαιτεί να υπάρχει κάποια βασική εμπειρία με την HTML.

Για να δουλέψουμε με τα CSS δεν χρειάζεται να χρησιμοποιήσουμε κάποιο πρόγραμμα δημιουργίας ιστοσελίδων, όπως είναι τα γνωστά FrontPage, DreamWeaver ή και το Word, καθώς θα σταθούν εμπόδιο στην κατανόηση των CSS. Αυτό που χρειαζόμαστε είναι ένας απλός επεξεργαστής κειμένου, όπως είναι το Σημειωματάριο (Notepad) των Windows.

Μπορούμε να χρησιμοποιήσουμε όποιον φυλλομετρητή (browser) επιθυμούμε για να βλέπουμε πώς θα εμφανίζονται οι ιστοσελίδες που θα δημιουργούμε. Συνεπώς αυτό που χρειαζόμαστε είναι ένας φυλλομετρητής (browser) και ένας απλός επεξεργαστής κειμένου (text editor).

Στο ερώτημα τι μπορούμε να κάνουμε με τα CSS, θα πρέπει να γνωρίζουμε ότι τα CSS είναι μια γλώσσα στυλ (style language) που ορίζουν τη διάταξη (layout) των HTML εγγράφων. Για παράδειγμα, τα CSS έχουν να κάνουν με γραμματοσειρές (fonts), με χρώματα (colours), με περιθώρια (margins), με εικόνες φόντου (background images) και με πολλά άλλα. Με την HTML θα δυσκολευτούμε να αλλάξουμε τη διάταξη των ιστοσελίδων μας, αλλά τα CSS προσφέρουν πολλές επιλογές και είναι πολύ πιο συγκεκριμένα στις λεπτομέρειες. Επιπλέον, υποστηρίζονται απ' όλους τους φυλλομετρητές.

Στο ερώτημα ποια είναι η διαφορά ανάμεσα στα CSS και την HTML, μπορούμε να πούμε ότι η HTML χρησιμοποιείται για να δομήσει το περιεχόμενο (content), ενώ τα CSS χρησιμοποιούνται για τη διαμόρφωση ή μορφοποίηση (formatting) του δομημένου περιεχομένου. Σύντομα θα γίνει σαφές το τι εννοούμε. Από το ξεκίνημά της, η HTML χρησιμοποιείτο μόνο για να μπορούμε να προσθέτουμε δομή στο κείμενο, όπως για να επισημαίνουμε ποια είναι κεφαλίδα (headline) ή ποια είναι παράγραφος (paragraph) με τα γνωστά tags (ετικέτες) της HTML, όπως είναι τα <h1> και <p>.

Όμως, καθώς η δημοτικότητα του Web αύξανε συνεχώς, οι σχεδιαστές των ιστοσελίδων άρχισαν να ψάχνουν για επιπλέον δυνατότητες, όπως προσθήκη διάταξης (layout) στα έγγραφα. Για να γίνει αυτό, οι φυλλομετρητές επινόησαν καινούργια HTML tags, όπως για παράδειγμα το <font>, τα οποία διέφεραν από τα αρχικά HTML tags καθώς όριζαν τη διάταξη και όχι τη δομή μιας ιστοσελίδας.

Στο ερώτημα τι έχουν να μου προσφέρουν τα CSS, μπορούμε να πούμε ότι τα CSS αποτέλεσαν μια

επανάσταση στον κόσμο του Web design και τα μεγάλα πλεονεκτήματά τους είναι τα εξής :

Δυνατότητα για τον ταυτόχρονο έλεγχο της διάταξης (layout) πολλών εγγράφων από ένα μόνο φύλλο στυλ (style sheet).

Πιο ακριβής έλεγχος της διάταξης (layout) των ιστοσελίδων.

Εφαρμογή διαφορετικής διάταξης σε διαφορετικές εξόδους, όπως screen, print κ.ά.

Αναρίθμητες προηγμένες και εξεζητημένες τεχνικές.

### ***Η Βασική Σύνταξη των CSS***

Θα δούμε τώρα πώς μπορούμε να δημιουργήσουμε το πρώτο μας φύλλο στυλ (style sheet). Θα πρέπει να έχουμε υπόψη μας ότι πολλές από τις ιδιότητες που χρησιμοποιούνται στα CSS είναι παρόμοιες μ' αυτές της HTML. Ας υποθέσουμε ότι θέλουμε να έχουμε ένα ωραίο κόκκινο χρώμα για το φόντο (background) μιας ιστοσελίδας. Με την HTML θα είχαμε γράψει τα εξής :

```
<body bgcolor="#FF0000">
```

Με τα CSS μπορούμε να επιτύχουμε το ίδιο αποτέλεσμα ως εξής :

```
body {  
background-color: #FF0000;  
}
```

Όπως μπορούμε να δούμε, οι κωδικοί είναι περίπου οι ίδιοι και για την HTML και για τα CSS. Από το παραπάνω παράδειγμα μπορούμε να δούμε και το βασικό μοντέλο των CSS :

```
selector (επιλογέας) {  
property (ιδιότητα): value (τιμή);  
}
```

selector	property	value
Σε ποια tags εφαρμόζεται η ιδιότητα	Μια ιδιότητα είναι το χρώμα φόντου	Η τιμή της ιδιότητας, όπως π.χ. κόκκινο για το χρώμα φόντου
(π.χ. body)	(background color)	

## ***Εφαρμογή των CSS σ' ένα HTML Έγγραφο***

Υπάρχουν τρεις τρόποι που μπορούμε να χρησιμοποιήσουμε για να εφαρμόσουμε ένα CSS σ' ένα HTML έγγραφο.

### **Μέθοδος 1 : In-line (attribute style)**

Ο πρώτος τρόπος για να εφαρμόσουμε ένα CSS σ' ένα HTML κείμενο είναι να χρησιμοποιήσουμε την ιδιότητα (attribute) style της HTML. Το είδαμε ήδη προηγουμένως με το κόκκινο χρώμα φόντου και μπορεί να εφαρμοσθεί ως εξής :

```
<html>
<head>
<title>Παράδειγμα</title>
</head>
<body style="background-color: #FF0000;">
<p>Αυτή είναι μια κόκκινη σελίδα</p>
</body>
</html>
```

### **Μέθοδος 2 : Εσωτερικό-Internal (tag style)**

Ένας άλλος τρόπος είναι να γράψουμε τον κώδικα για τα CSS με το HTML tag <style>, ως εξής :

```
<html>
<head>
<title>Παράδειγμα</title>
<style type="text/css">
body {
background-color: #FF0000;
}
</style>
```

```
</style>
</head>
<body>
<p>Αυτή είναι μια κόκκινη σελίδα</p>
</body>
</html>
```

### Μέθοδος 3 : Εξωτερικό-External (link to a style sheet)

Ο τρίτος τρόπος, που είναι αυτός που συνιστάται να χρησιμοποιούμε, είναι ένας σύνδεσμος (link) προς ένα εξωτερικό αρχείο φύλλου στυλ (external style sheet). Αυτή η μέθοδος θα χρησιμοποιηθεί σ' όλα τα παραδείγματα που ακολουθούν. Ένα εξωτερικό αρχείο φύλλου στυλ (external style sheet) είναι απλά ένα αρχείο κειμένου (text file) που έχει επέκταση (extension) .css. Μπορούμε να το τοποθετήσουμε (upload) στον Web server, όπως όλα τα άλλα αρχεία.

Για παράδειγμα, ας υποθέσουμε ότι το αρχείο μας style sheet έχει όνομα style.css και βρίσκεται στον φάκελο style. Για να δημιουργήσουμε έναν σύνδεσμο (link) από το HTML έγγραφο, που είναι το default.htm, προς το αρχείο του style sheet, που είναι το style.css, θα πρέπει να γράψουμε τον εξής κώδικα :

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

Η διαδρομή (path) προς το style sheet προσδιορίζεται με την ιδιότητα (attribute) href. Η παραπάνω γραμμή κώδικα πρέπει να τοποθετηθεί στο τμήμα header του HTML κώδικα, δηλ. ανάμεσα στα tags <head> και </head>, ως εξής :

```
<html>
<head>
<title>Κύρια Ιστοσελίδα</title>
<link rel="stylesheet" type="text/css" href="style/style.css" />
</head>
<body>
```

Ο παραπάνω σύνδεσμος (link) λέει στον φυλλομετρητή ότι θα πρέπει να χρησιμοποιήσει τη διάταξη (layout) που υπάρχει στο συγκεκριμένο CSS αρχείο όταν πρόκειται να εμφανίσει το περιεχόμενο του τρέχοντος HTML αρχείου.

Αυτό που είναι πολύ καλό με τη μέθοδο αυτή είναι ότι μπορούμε να συνδέσουμε πολλά HTML

αρχεία με το ίδιο αρχείο style sheet. Μ' άλλα λόγια, ένα αρχείο CSS μπορεί να χρησιμοποιηθεί για να μπορούμε να ελέγχουμε το layout πολλών HTML εγγράφων ταυτόχρονα.

Για να το δούμε αυτό και στην πράξη, μπορούμε να δημιουργήσουμε δύο αρχεία, ένα HTML και ένα CSS, με το εξής περιεχόμενο :

default.htm

```
<html>
<head>
<title>Κύρια Ιστοσελίδα</title>
<link rel="stylesheet" type="text/css" href="style.css" />
</head>
<body>
<h1>Το Πρώτο μας Φύλλο Στυλ</h1>
</body>
</html>
```

style.css

```
body {
background-color: #FF0000;
}
```

Θα πρέπει να δημιουργήσουμε τα αρχεία και να τα αποθηκεύσουμε με τις σωστές επεκτάσεις .htm και .css και φυσικά να τα τοποθετήσουμε και στον ίδιο φάκελο. Μετά, μόλις ανοίξουμε την ιστοσελίδα default.htm με τον φυλλομετρητή μας, θα διαπιστώσουμε ότι θα έχει κόκκινο φόντο.

### ***Αλλαγή Χρώματος και Φόντου***

Θα δούμε τώρα το πώς μπορούμε να εφαρμόσουμε χρώματα (colors) και χρώματα φόντου (background colors) στις ιστοσελίδες μας. Θα δούμε επίσης και προηγμένες μεθόδους για την τοποθέτηση και τον έλεγχο των εικόνων φόντου (background images).

## Η Ιδιότητα color

Η ιδιότητα color περιγράφει το χρώμα προσκηνίου (foreground color) ενός στοιχείου. Για παράδειγμα, ας υποθέσουμε ότι θέλουμε όλες οι επικεφαλίδες (headlines) ενός εγγράφου να έχουν χρώμα κόκκινο (red). Οι επικεφαλίδες χαρακτηρίζονται με το tag <h1> και ο παρακάτω κώδικας ορίζει σε κόκκινο το χρώμα των στοιχείων <h1> :

```
h1 {  
  
    color: #ff0000;  
  
}
```

Τα χρώματα μπορούν να γραφούν με τις δεκαεξαδικές τους τιμές, όπως στο παραπάνω παράδειγμα (#ff0000), ή μπορούμε να χρησιμοποιήσουμε τα ονόματα των χρωμάτων ("red") ή και τις τιμές RGB : rgb(255, 0, 0).

## Η Ιδιότητα background-color

Η ιδιότητα background-color περιγράφει το χρώμα φόντου (background color) ενός στοιχείου. Ως γνωστόν, το στοιχείο <body> περιέχει όλο το κείμενο ενός HTML εγγράφου και έτσι για να αλλάξουμε το χρώμα φόντου μιας ιστοσελίδας, θα πρέπει να εφαρμόσουμε την ιδιότητα background-color στο στοιχείο <body>.

Μπορούμε επίσης να εφαρμόσουμε χρώμα φόντου και σ' άλλα στοιχεία, όπως σε επικεφαλίδες (headlines) και σε απλό κείμενο (text). Στο παρακάτω παράδειγμα, εφαρμόζουμε διαφορετικά χρώματα φόντου στα στοιχεία <body> και <h1>.

```
body {  
  
    background-color: #FFCC66;  
  
}  
  
h1 {  
  
    color: #990000;  
  
    background-color: #FC9804;  
  
}
```

Βλέπουμε ότι για να εφαρμόσουμε δύο ιδιότητες σ' ένα στοιχείο, όπως το <h1>, τις ξεχωρίζουμε με το σύμβολο ;.

## Η Ιδιότητα background-image

Η ιδιότητα background-image χρησιμοποιείται για να εισάγουμε μια εικόνα φόντου (background image) σε μια ιστοσελίδα. Απλά εφαρμόζουμε την ιδιότητα background-image στο tag <body> και προσδιορίζουμε το όνομα και τη θέση του αρχείου της εικόνας.

```
body {  
  
    background-color: #FFCC66;  
  
    background-image: url("florina1.jpg");  
  
}  
  
h1 {  
  
    color: red;  
  
    background-color: blue;  
  
}
```

Βλέπουμε ότι προσδιορίζουμε το όνομα και τη θέση της εικόνας με την έκφραση url("florina1.jpg"). Αυτό βέβαια προϋποθέτει ότι η εικόνα βρίσκεται στον ίδιο φάκελο με το φύλλο στυλ (style sheet).

Μπορούμε να αναφερόμαστε και σε εικόνες που βρίσκονται σ' άλλους φακέλους με την έκφραση url("../images/florina1.jpg") ή που βρίσκονται ακόμη και στο Internet, αρκεί να ορίσουμε την πλήρη διεύθυνση του αρχείου με την έκφραση url("http://dide.flo.sch.gr/florina1.jpg").

## Η Ιδιότητα background-repeat

Στο παραπάνω παράδειγμα, παρατηρούμε ότι εξ' ορισμού η εικόνα επαναλαμβάνεται και προς την οριζόντια και προς την κατακόρυφη κατεύθυνση για να καλύψει ολόκληρη την οθόνη. Με την ιδιότητα background-repeat μπορούμε να ελέγχουμε την επανάληψη μιας εικόνας φόντου.

Η ιδιότητα background-repeat μπορεί να έχει τις εξής τέσσερις τιμές :

Τιμή

Περιγραφή

background-repeat: repeat-x

Η εικόνα επαναλαμβάνεται μόνο οριζόντια

background-repeat: repeat-y

Η εικόνα επαναλαμβάνεται μόνο κατακόρυφα

background-repeat: repeat

Η εικόνα επαναλαμβάνεται και οριζόντια και κατακόρυφα (εξ' ορισμού)

background-repeat: no-repeat  
Η εικόνα δεν επαναλαμβάνεται

Για παράδειγμα, για να αποφύγουμε την επανάληψη μιας εικόνας φόντου (background image), θα πρέπει να γράψουμε τον εξής κώδικα :

```
body {  
  
    background-color: #FFCC66;  
  
    background-image: url("florina1.jpg");  
  
    background-repeat: no-repeat;  
  
}  
  
h1 {  
  
    color: green;  
  
    background-color: #FC9804;  
  
}
```

## Η ιδιότητα background-attachment

Με την ιδιότητα background-attachment μπορούμε να καθορίσουμε αν μια εικόνα φόντου θα είναι σταθερή (fixed) ή θα κυλάει (scrolls) μαζί με τα περιεχόμενα της ιστοσελίδας. Μια εικόνα φόντου που είναι σταθερή (fixed) δεν θα μετακινείται μαζί με το κείμενο όταν ο χρήστης κυλάει τα περιεχόμενα της ιστοσελίδας, ενώ μια μη κλειδωμένη εικόνα φόντου θα κυλάει μαζί με τα περιεχόμενα της ιστοσελίδας (εξ ορισμού).

Ο παρακάτω πίνακας παρουσιάζει τις δύο διαφορετικές τιμές που μπορεί να λάβει η ιδιότητα background-attachment :

Τιμή  
Περιγραφή

background-attachment: scroll  
Η εικόνα κυλάει μαζί με την ιστοσελίδα – μη κλειδωμένη

background-attachment: fixed  
Η εικόνα είναι κλειδωμένη

Ο παρακάτω κώδικας κάνει την εικόνα φόντου της ιστοσελίδας να μην επαναλαμβάνεται (no-repeat) και να είναι κλειδωμένη (fixed) :

```

body {

    background-color: #FFCC66;

    background-image: url("florina1.jpg");

    background-repeat: no-repeat;

    background-attachment: fixed;

}

h1 {

    color: #990000;

    background-color: #FC9804;

}

```

## Η ιδιότητα background-position

Εξ ορισμού, μια εικόνα φόντου τοποθετείται στην πάνω αριστερή γωνία της οθόνης. Με την ιδιότητα background-position μπορούμε να αλλάξουμε αυτή τη ρύθμιση και να τοποθετήσουμε την εικόνα φόντου οπουδήποτε θέλουμε στην οθόνη.

Υπάρχουν τρεις διαφορετικοί τρόποι για να προσδιορίσουμε τη θέση μιας εικόνας φόντου μέσα στην οθόνη. Για παράδειγμα, μπορούμε να ορίσουμε τις συντεταγμένες της εικόνας με τις τιμές '100px 200px', οπότε η εικόνα φόντου θα τοποθετηθεί σε απόσταση 100px (pixels) από την αριστερή πλευρά της οθόνης και 200px (pixels) από την κορυφή της οθόνης, στην ουσία του παραθύρου του φυλλομετρητή.

Ένας άλλος τρόπος είναι να δηλώσουμε την απόσταση σε ποσοστά, όπου οι τιμές 0% 0% σημαίνουν την πάνω αριστερή γωνία της οθόνης, οι τιμές 50% 50% το κέντρο της οθόνης και οι τιμές 100% 100% την κάτω δεξιά γωνία της οθόνης. Ένας τρίτος τρόπος είναι να χρησιμοποιήσουμε τις δεσμευμένες λέξεις top, bottom, center, left και right και να κάνουμε συνδυασμούς, όπως top left, center center, top right, bottom right κ.ά.

Ο παρακάτω πίνακας έχει μερικά χαρακτηριστικά παραδείγματα :

Τιμή

Περιγραφή

background-position: 2cm 2cm

Η εικόνα φόντου θα τοποθετηθεί 2 cm από τα αριστερά και 2 cm από την κορυφή

background-position: 50% 25%

Η εικόνα φόντου θα τοποθετηθεί στο κέντρο οριζόντια και κατά το ένα τέταρτο από την κορυφή

background-position: top right

Η εικόνα φόντου θα τοποθετηθεί στην πάνω δεξιά γωνία (top-right) της οθόνης

Ο παρακάτω κώδικας θα τοποθετήσει την εικόνα φόντου στην κάτω δεξιά γωνία (bottom right) της οθόνης :

```
body {  
  
    background-color: #FFCC66;  
  
    background-image: url("florina1.jpg");  
  
    background-repeat: no-repeat;  
  
    background-attachment: fixed;  
  
    background-position: right bottom;  
  
}  
  
h1 {  
  
    color: #990000;  
  
    background-color: #FC9804;  
  
}
```

## Η ιδιότητα background

Η ιδιότητα background αποτελεί μια συντόμευση γι' όλες τις ιδιότητες του αντικειμένου background που είδαμε προηγουμένως και μας δίνει τη δυνατότητα να γράψουμε όλες τις ιδιότητες σε μια γραμμή κώδικα.

Για παράδειγμα, ας δούμε τις επόμενες πέντε γραμμές κώδικα :

```
background-color: #FFCC66;  
  
background-image: url("florina1.jpg");  
  
background-repeat: no-repeat;  
  
background-attachment: fixed;  
  
background-position: right bottom;
```

Με τη χρήση της ιδιότητας background μπορούμε να πετύχουμε το ίδιο αποτέλεσμα σε μία μόνο γραμμή κώδικα :

```
background: #FFCC66 url("florina1.jpg") no-repeat fixed right bottom;
```

Η σειρά αναγραφής των ιδιοτήτων είναι η εξής :

```
[background-color] | [background-image] | [background-repeat] | [background-attachment] |  
[background-position]
```

Αν παραλείψουμε κάποια ιδιότητα, θα εκληφθεί η προκαθορισμένη (default) τιμή της. Για παράδειγμα, αν παραλειφθούν οι δύο τελευταίες ιδιότητες, background-attachment και background-position, θα έχουμε το εξής :

```
background: #FFCC66 url("florina1.jpg") no-repeat;
```

Για τις δύο ιδιότητες που δεν δώσαμε τιμή, θα εκληφθούν οι προκαθορισμένες τιμές, που είναι οι scroll και top left.

Είδαμε ήδη αρκετές νέες τεχνικές μορφοποίησης με τη βοήθεια των CSS που θα ήταν αδύνατο να τις κάνουμε με την απλή HTML.

## ***Οι Γραμματοσειρές (Fonts)***

Θα δούμε τώρα τις γραμματοσειρές (fonts) και πώς μπορούμε να τις εφαρμόσουμε με τη βοήθεια των CSS.

### **Η ιδιότητα font-family**

Η ιδιότητα font-family χρησιμοποιείται για να ορίσει μια λίστα προτεραιότητας από γραμματοσειρές που θα χρησιμοποιηθούν για να εμφανισθεί ένα συγκεκριμένο στοιχείο ή μια ιστοσελίδα. Αν η πρώτη γραμματοσειρά που υπάρχει στη λίστα δεν είναι εγκατεστημένη στον υπολογιστή από τον οποίο γίνεται η πρόσβαση στο site, θα δοκιμασθεί η επόμενη γραμματοσειρά κοκ μέχρι να βρεθεί κάποια που να ταιριάζει. Υπάρχουν δύο είδη ονομάτων που χρησιμοποιούνται για να κατηγοριοποιηθούν οι γραμματοσειρές : τα family-names και τα generic families.

Παραδείγματα ενός family-name, που τα ξέρουμε και ως font, είναι για παράδειγμα τα "Arial", "Times New Roman", "Tahoma" κ.ά. Τα generic families είναι ομάδες από family-names με παρόμοια εμφάνιση. Ένα παράδειγμα είναι το sans-serif, που είναι μια συλλογή από γραμματοσειρές χωρίς πόδια (feet).

Στο generic family serif ανήκουν οι γραμματοσειρές Times New Roman, Garamond και Georgia, όπου όλες έχουν πόδια (feet). Στο generic family sans-serif ανήκουν οι γραμματοσειρές Trebuchet, Arial και Verdana, όπου όλες δεν έχουν πόδια (feet). Στο generic family monospace ανήκουν οι γραμματοσειρές Courier, Courier New και Andale Mono, όπου όλοι οι χαρακτήρες τους έχουν το ίδιο πλάτος.

Όταν εμφανίζουμε τις γραμματοσειρές για την ιστοσελίδα μας, θα πρέπει να ξεκινάμε με τις πιο

δημοφιλείς γραμματοσειρές και μετά με κάποιες εναλλακτικές. Αποτελεί καλή πρακτική να ολοκληρώνουμε τη λίστα με μια γραμματοσειρά από generic family. Έτσι, η ιστοσελίδα θα εμφανισθεί με μια γραμματοσειρά της ίδιας οικογένειας αν δεν βρεθεί κάποια από τις προηγούμενες γραμματοσειρές.

Ένα παράδειγμα χρήσης της ιδιότητας font-family μπορεί να είναι το εξής :

```
h1 {  
  
font-family: arial, verdana, sans-serif;  
  
}  
  
h2 {  
  
font-family: "Times New Roman", serif;  
  
}
```

Σύμφωνα με τον παραπάνω κώδικα, οι επικεφαλίδες (headlines) που είναι μαρκαρισμένες με το tag <h1> θα εμφανισθούν με τη γραμματοσειρά "Arial". Αν αυτή η γραμματοσειρά δεν είναι εγκατεστημένη στον υπολογιστή του χρήστη, θα χρησιμοποιηθεί η "Verdana" και αν καμία απ' αυτές δεν είναι διαθέσιμη, θα χρησιμοποιηθεί μια γραμματοσειρά από την οικογένεια sans-serif, ώστε να εμφανισθεί σωστά το κείμενο των επικεφαλίδων.

Επειδή το όνομα της γραμματοσειράς "Times New Roman" περιέχει κενούς χαρακτήρες, χρησιμοποιήσαμε διπλά εισαγωγικά για να το προσδιορίσουμε.

### **Η ιδιότητα font-style**

Η ιδιότητα font-style κάνει τη μορφοποίηση της επιλεγμένης γραμματοσειράς να είναι normal, italic ή oblique. Στο επόμενο παράδειγμα, όλες οι επικεφαλίδες που είναι μαρκαρισμένες με το tag <h2> θα εμφανισθούν με πλάγια γράμματα.

```
h1 {  
  
font-family: arial, verdana, sans-serif;  
  
}  
  
h2 {  
  
font-family: "Times New Roman", serif; font-style: italic;  
  
}
```

### **Η ιδιότητα font-variant**

Η ιδιότητα font-variant χρησιμοποιείται για να επιλέξουμε ανάμεσα στις παραλλαγές normal ή small-caps μιας γραμματοσειράς. Μια γραμματοσειρά του τύπου small-caps χρησιμοποιεί μικρού

μεγέθους κεφαλαία γράμματα αντί για μικρά (πεζά) γράμματα. Στην περίπτωση που η ιδιότητα `font-variant` είναι ορισμένη στην τιμή `small-caps` και δεν υπάρχει διαθέσιμη κάποια γραμματοσειρά που να την υποστηρίζει, ο φυλλομετρητής είναι πολύ πιθανό να εμφανίσει το κείμενο με κανονικά κεφαλαία γράμματα.

Ακολουθούν παραδείγματα :

```
h1 {  
font-variant: small-caps;  
}
```

```
h2 {  
font-variant: normal;  
}
```

## **Η ιδιότητα `font-weight`**

Η ιδιότητα `font-weight` περιγράφει πόσο έντονα θα πρέπει να εμφανίζεται μια γραμματοσειρά. Μια γραμματοσειρά μπορεί να είναι είτε `normal` ή `bold`. Μερικοί φυλλομετρητές υποστηρίζουν ακόμη και αριθμούς από το 100 έως το 900 για να προσδιορίσουν με ακρίβεια το πόσο έντονη θέλουμε να είναι η γραμματοσειρά.

Ακολουθούν παραδείγματα :

```
p {  
font-family: arial, verdana, sans-serif;  
}
```

```
td {  
font-family: arial, verdana, sans-serif; font-weight: bold;  
}
```

## **Η ιδιότητα `font-size`**

Η ιδιότητα `font-size` ορίζει το μέγεθος μιας γραμματοσειράς. Υπάρχουν πολλές μονάδες, όπως `pixels`, `points` και `em`, που μπορούμε να χρησιμοποιήσουμε για να προσδιορίσουμε το μέγεθος μιας γραμματοσειράς.

Ακολουθούν παραδείγματα :

```
h1 {  
  
font-size: 30px;  
  
}
```

```
h2 {  
font-size: 12pt;  
}
```

```
h3 {  
font-size: 120%;  
}
```

```
p {  
font-size: 1em;  
}
```

Με τις μονάδες px και pt προσδιορίζουμε το μέγεθος με απόλυτες τιμές, ενώ με τις μονάδες % και em προσδιορίζουμε το μέγεθος με σχετικές τιμές.

## Η ιδιότητα font

Με την ιδιότητα font μπορούμε να προσδιορίσουμε όλες τις ιδιότητες font σε μία μόνο γραμμή. Για παράδειγμα, οι επόμενες τέσσερις γραμμές κώδικα περιγράφουν τις ιδιότητες γραμματοσειράς του tag <p> :

```
p {  
  
    font-style: italic;  
  
    font-weight: bold;  
  
    font-size: 30px;  
  
    font-family: arial, sans-serif;  
  
}
```

Με την ιδιότητα font μπορούμε να επιτύχουμε το ίδιο αποτέλεσμα, αλλά με μία μόνο γραμμή κώδικα :

```
p {  
  
    font: italic bold 30px arial, sans-serif;  
  
}
```

Η σειρά αναγραφής των τιμών των ιδιοτήτων για την ιδιότητα font είναι η εξής :

font-style | font-variant | font-weight | font-size | font-family

Και μην ξεχνάτε : η τεχνολογία των Διαδοχικών Φύλλων Στυλ (CSS) σάς γλυτώνει από χρόνο και κάνει τη ζωή σας ευκολότερη.

## Μέρος 2 - Κείμενο, Links, Class και Id

### *To Κείμενο (Text)*

Η μορφοποίηση και η προσθήκη στυλ σ' ένα κείμενο (text) αποτελούν ένα πολύ βασικό κομμάτι στο Web design. Θα δούμε τις εκπληκτικές δυνατότητες που μας δίνουν τα CSS για να μπορέσουμε να προσθέσουμε διάταξη (layout) στο κείμενο μιας ιστοσελίδας.

### **Η ιδιότητα text-indent**

Με την ιδιότητα text-indent μπορούμε να κάνουμε το κείμενο των παραγράφων να είναι πιο εμφανίσιμο εφαρμόζοντας μια εσοχή (indent) στην πρώτη γραμμή της παραγράφου. Στο επόμενο παράδειγμα, εφαρμόζουμε μια απόσταση 30px στην πρώτη γραμμή του κειμένου όλων των παραγράφων που σημειώνονται με το tag <p> :

```
p {  
  
    text-indent: 30px;  
  
}
```

### **Η ιδιότητα text-align**

Η ιδιότητα text-align αντιστοιχεί με την ιδιότητα align που χρησιμοποιείται σε παλαιότερες εκδόσεις της HTML. Ένα κείμενο μπορεί να είναι στοιχισμένο (aligned) στα αριστερά (left) ή στα δεξιά (right) ή και στο κέντρο (center). Υπάρχει και η τιμή justify για να έχουμε ταυτόχρονα δεξιά και αριστερή, δηλ. πλήρη στοίχιση.

Στο επόμενο παράδειγμα, το κείμενο που περιέχεται στις επικεφαλίδες ενός πίνακα (table headings <th>) είναι στοιχισμένο στα δεξιά ενώ τα δεδομένα του πίνακα (table data <td>) είναι κεντραρισμένα. Επίσης, το κείμενο των κανονικών παραγράφων είναι πλήρως στοιχισμένο (justified).

```
th {  
  
    text-align: right;  
  
}  
  
td {  
  
    text-align: center;
```

```
}  
p {  
    text-align: justify;  
}
```

## Η ιδιότητα text-decoration

Με την ιδιότητα text-decoration μπορούμε να προσθέσουμε διαφορετική διακόσμηση ή και εφέ σ' ένα κείμενο. Για παράδειγμα, μπορούμε να υπογραμμίσουμε (underline) ένα κείμενο, να βάλουμε μια γραμμή διαμέσου (line-through) ή πάνω από το κείμενο (overline) κλπ.

Στο επόμενο παράδειγμα, τα <h1> είναι κεφαλίδες με υπογράμμιση (underlined headlines), τα <h2> είναι κεφαλίδες με μια γραμμή πάνω από το κείμενο (overline) και τα <h3> είναι κεφαλίδες με μια γραμμή διαμέσου του κειμένου (line-through).

```
h1 {  
    text-decoration: underline;  
}  
h2 {  
    text-decoration: overline;  
}  
h3 {  
    text-decoration: line-through;  
}
```

## Η ιδιότητα letter-spacing

Η απόσταση ανάμεσα στους χαρακτήρες του κειμένου μπορεί να ορισθεί με την ιδιότητα letter-spacing, η τιμή της οποίας είναι το επιθυμητό πλάτος. Για παράδειγμα, αν θέλουμε μια απόσταση 3px ανάμεσα στα γράμματα σε μια παράγραφο κειμένου <p> και 6px ανάμεσα στα γράμματα στις κεφαλίδες <h1>, θα πρέπει να γράψουμε τον εξής κώδικα :

```
h1 {  
    letter-spacing: 6px;
```

```
}
```

```
p {
```

```
    letter-spacing: 3px;
```

```
}
```

## Η ιδιότητα text-transform

Με την ιδιότητα text-transform μπορούμε να ελέγχουμε τον τρόπο γραφής ενός κειμένου όσον αφορά το αν τα γράμματα θα εμφανίζονται ως πεζά ή κεφαλαία (capitalization). Μπορούμε να επιλέξουμε να είναι κεφαλαίο μόνο το πρώτο γράμμα της κάθε λέξης (capitalize) ή όλα τα γράμματα κεφαλαία (uppercase) ή όλα τα γράμματα πεζά (lowercase) ανεξάρτητα από το πώς είναι το αρχικό κείμενο στον HTML κώδικα.

Για παράδειγμα, αν έχουμε τη λέξη "florina", μπορούμε να επιλέξουμε να την εμφανίσουμε ως "FLORINA" ή και ως "Florina". Υπάρχουν τέσσερις δυνατές τιμές για την ιδιότητα text-transform :

capitalize

Κάνει κεφαλαίο μόνο το πρώτο γράμμα της κάθε λέξης. Για παράδειγμα, το "florina per sempre" θα γίνει "Florina Per Sempre".

uppercase

Κάνει όλα τα γράμματα κεφαλαία (uppercase), δηλ. το "florina per sempre" θα γίνει "FLORINA PER SEMPRE".

lowercase

Κάνει όλα τα γράμματα πεζά (lowercase), δηλ. το "FLORINA PER SEMPRE" θα γίνει "florina per sempre".

none

Δεν κάνει κανέναν μετασχηματισμό, δηλ. το κείμενο εμφανίζεται όπως και στον HTML κώδικα.

Μπορούμε να χρησιμοποιήσουμε αυτήν την ιδιότητα για να κάνουμε τα ονόματα κάποιων ατόμων να είναι με το πρώτο γράμμα κεφαλαίο. Θα πρέπει να έχουμε υπόψη μας ότι το αρχικό κείμενο στον HTML κώδικα παραμένει όπως ήταν, άσχετα από το πώς εμφανίζεται στην ιστοσελίδα.

Ακολουθεί ένα χαρακτηριστικό παράδειγμα.

```
h1 {
```

```
    text-transform: uppercase;
```

```
}  
  
li {  
  
    text-transform: capitalize;  
  
}
```

### ***Οι Σύνδεσμοι (Links)***

Μπορούμε να εφαρμόσουμε αυτά που μάθαμε έως τώρα και στο κείμενο των συνδέσμων (links), όπως να αλλάξουμε χρώματα, γραμματοσειρές (fonts), να υπογραμμίσουμε κ.ά. Με τη βοήθεια των CSS μπορούμε να επηρεάσουμε αυτές τις ιδιότητες ανάλογα με το αν το link δεν έχει ακόμα γίνει κλικ (unvisited) ή έχει γίνει κλικ (visited) ή είναι ενεργό (active) ή τέλος αν ο δρομέας βρίσκεται πάνω από το link χωρίς να έχουμε κάνει κλικ ακόμα (hover).

Μπορούμε έτσι να προσθέσουμε πολύ ωραία εφέ στις ιστοσελίδες μας. Για να μπορέσουμε να ελέγξουμε αυτά τα εφέ χρησιμοποιούμε τις λεγόμενες ψευδο-τάξεις (pseudo-classes). Μια ψευδο-τάξη (pseudo-class) μάς δίνει τη δυνατότητα να λάβουμε υπόψη μας διαφορετικές καταστάσεις (conditions) ή συμβάντα (events) όταν ορίζουμε μια ιδιότητα (property) για ένα tag της HTML.

Θα δούμε ένα παράδειγμα. Όπως ήδη γνωρίζουμε, οι σύνδεσμοι (links) ορίζονται στην HTML με το tag <a>, το οποίο μπορούμε να χρησιμοποιήσουμε ως επιλογή (selector) στα CSS :

```
a {  
  
color: blue;  
  
}
```

Ένα link μπορεί να έχει διαφορετικές καταστάσεις. Για παράδειγμα, μπορεί να το έχουμε επισκεφθεί (visited) ή να μην το έχουμε επισκεφθεί (not visited). Μπορούμε να χρησιμοποιήσουμε τα pseudo-classes για να εκχωρήσουμε διαφορετικά styles στα visited και unvisited links, ως εξής :

```
a:link {  
  
    color: blue;  
  
}  
  
a:visited {  
  
    color: red;  
  
}
```

Χρησιμοποιούμε τα a:link και a:visited για τα unvisited και visited links αντίστοιχα. Τα links που

είναι ενεργά (active) έχουν την pseudo-class a:active και το a:hover είναι όταν ο δρομέας βρίσκεται πάνω από το link.

Θα δούμε τώρα και τις τέσσερις ψευδο-τάξεις με παραδείγματα και επεξηγήσεις.

### **Pseudo-class: link**

Η pseudo-class :link χρησιμοποιείται για τα links που οδηγούν σε ιστοσελίδες που ο χρήστης δεν έχει επισκεφθεί ακόμα (not visited). Στο επόμενο παράδειγμα, τα unvisited links θα έχουν χρώμα ελαφρύ μπλε (light blue).

```
a:link {  
  
    color: #6699CC;  
  
}
```

### **Pseudo-class: visited**

Η pseudo-class :visited χρησιμοποιείται για τα links που οδηγούν σε ιστοσελίδες που ο χρήστης έχει επισκεφθεί (visited). Στο επόμενο παράδειγμα, ο κώδικας θα κάνει όλα τα visited links να έχουν χρώμα σκούρο μωβ (dark purple).

```
a:visited {  
  
    color: #660099;  
  
}
```

### **Pseudo-class: active**

Η pseudo-class :active χρησιμοποιείται για τα links που είναι ενεργά (active), δηλ. ο χρήστης έχει κάνει κλικ πάνω τους και δεν έχει ακόμα απομακρύνει το ποντίκι. Το επόμενο παράδειγμα κάνει τα active links να έχουν ένα κίτρινο χρώμα φόντου (yellow background color). Θα πρέπει να έχουμε υπόψη μας ότι το χρώμα φόντου γίνεται κίτρινο τη στιγμή που κάνουμε κλικ στο link.

```
a:active {  
  
    background-color: #FFFF00;  
  
}
```

## Pseudo-class: hover

Η pseudo-class :hover χρησιμοποιείται όταν ο δείκτης του ποντικιού βρίσκεται πάνω από ένα link (hover) και μπορούμε έτσι να δημιουργήσουμε πολύ ωραία εφέ. Για παράδειγμα, αν θέλουμε τα links να είναι πορτοκαλί (orange) και με πλάγια γράμματα (italics) όταν ο δρομέας βρεθεί από πάνω τους, ο κώδικας των CSS θα πρέπει να είναι ως εξής :

```
a:hover {  
  
    color: orange;  
  
    font-style: italic;  
  
}
```

Θα δούμε μερικά ακόμα ενδιαφέροντα παραδείγματα όταν ο δρομέας βρεθεί πάνω από ένα link. Θα χρησιμοποιήσουμε την ιδιότητα letter-spacing για να αλλάξουμε την απόσταση ανάμεσα στους χαρακτήρες του κειμένου του link αλλά και τις ιδιότητες font-weight και color, ως εξής :

```
a:hover {  
  
    letter-spacing: 10px;  
  
    font-weight:bold;  
  
    color:red;  
  
}
```

Στο επόμενο παράδειγμα θα χρησιμοποιήσουμε την ιδιότητα text-transform για να κάνουμε τα γράμματα του κειμένου του link κεφαλαία (uppercase) όταν ο δρομέας βρεθεί πάνω από ένα link, ως εξής :

```
a:hover {  
  
    text-transform: uppercase;  
  
    font-weight:bold;  
  
    color:blue;  
  
    background-color:yellow;  
  
}
```

## Αφαίρεση της Υπογράμμισης από τους Συνδέσμους

Κάτι που ζητάνε πολλοί χρήστες είναι η αφαίρεση της υπογράμμισης από τους συνδέσμους

(underlining of links). Βέβαια, θα πρέπει να έχουμε υπόψη μας ότι αν αφαιρέσουμε την υπογράμμιση από τα links, υπάρχει ο φόβος να μην μπορούν εύκολα οι επισκέπτες της ιστοσελίδας μας να αντιληφθούν τα σημεία όπου βρίσκονται οι σύνδεσμοι.

Για να αφαιρέσουμε την υπογράμμιση από τους συνδέσμους θα χρησιμοποιήσουμε την ιδιότητα `text-decoration`, στην οποία θα δώσουμε την τιμή `none`, ως εξής :

```
a {  
  
    text-decoration:none;  
  
}
```

Μια καλή λύση είναι να δώσουμε την τιμή `none` στην ιδιότητα `text-decoration` αλλά διαφορετικά χρώματα για τις τέσσερις καταστάσεις των ψευδο-τάξεων, ως εξής :

```
a:link {  
  
    color: blue;  
  
    text-decoration:none;  
  
}  
  
a:visited {  
  
    color: purple;  
  
    text-decoration:none;  
  
}  
  
a:active {  
  
    background-color: yellow;  
  
    text-decoration:none;  
  
}  
  
a:hover {  
  
    color:red;  
  
    text-decoration:none;  
  
}
```

## To Class και το Id

Μερικές φορές θέλουμε να εφαρμόσουμε ένα ειδικό στυλ (style) σ' ένα συγκεκριμένο στοιχείο (element) ή σε μια συγκεκριμένη ομάδα στοιχείων (group of elements). Θα δούμε το πώς μπορούμε να χρησιμοποιήσουμε το class και το id για να ορίσουμε ιδιότητες για επιλεγμένα στοιχεία.

Για παράδειγμα, θα μας απασχολήσουν θέματα όπως το πώς μπορούμε να χρωματίσουμε μια συγκεκριμένη κεφαλίδα (headline) διαφορετικά από τις άλλες κεφαλίδες της ίδιας ιστοσελίδας ή το πώς μπορούμε να ομαδοποιήσουμε τα links σε διαφορετικές κατηγορίες και να δώσουμε σε κάθε κατηγορία ένα ειδικό στυλ.

### Ομαδοποίηση Στοιχείων με το Class

Ας υποθέσουμε ότι έχουμε δύο λίστες από links για διαφορετικά σταφύλια που χρησιμοποιούνται για άσπρο κρασί (white wine) και για κόκκινο κρασί (red wine).

Ο HTML κώδικας θα είναι ως εξής :

```
<p>Σταφύλια για άσπρο κρασί : </p>
<ul>
<li><a href="ri.htm">Riesling</a></li>
<li><a href="ch.htm">Chardonnay</a></li>
<li><a href="pb.htm">Pinot Blanc</a></li>
</ul>
<p>Σταφύλια για κόκκινο κρασί : </p>
<ul>
<li><a href="cs.htm">Cabernet Sauvignon</a></li>
<li><a href="me.htm">Merlot</a></li>
<li><a href="pn.htm">Pinot Noir</a></li>
</ul>
```

Θέλουμε τώρα οι σύνδεσμοι για το άσπρο κρασί (white wine links) να έχουν κίτρινο χρώμα (yellow), οι σύνδεσμοι για το κόκκινο κρασί (red wine links) να έχουν κόκκινο χρώμα (red) και οι υπόλοιποι σύνδεσμοι της ιστοσελίδας να παραμείνουν μπλε (blue).

Για να το επιτύχουμε αυτό, θα διαιρέσουμε τα links σε δύο κατηγορίες, εκχωρώντας μια τάξη (class) σε κάθε link κάνοντας χρήση της ιδιότητας (attribute) class, ως εξής :

```
<p>Σταφύλια για άσπρο κρασί : </p>
```

```
<ul>
```

```
<li><a href="ri.htm" class="whitewine">
```

```
Riesling</a></li>
```

```
<li><a href="ch.htm" class="whitewine">
```

```
Chardonnay</a></li>
```

```
<li><a href="pb.htm" class="whitewine">
```

```
Pinot Blanc</a></li>
```

```
</ul>
```

```
<p>Σταφύλια για κόκκινο κρασί : </p>
```

```
<ul>
```

```
<li><a href="cs.htm" class="redwine">
```

```
Cabernet Sauvignon</a></li>
```

```
<li><a href="me.htm" class="redwine">
```

```
Merlot</a></li>
```

```
<li><a href="pn.htm" class="redwine">
```

```
Pinot Noir</a></li>
```

```
</ul>
```

Μπορούμε τώρα να ορίσουμε ξεχωριστές ιδιότητες για τα links που ανήκουν στις τάξεις whitewine και redwine, αντίστοιχα, ως εξής :

```
a {
```

```
    color: blue;
```

```
}
```

```
a.whitewine {
```

```
    color: #FFBB00;
```

```
}  
  
a.redwine {  
  
    color: #800000;  
  
}
```

Όπως βλέπουμε από το παραπάνω παράδειγμα, μπορούμε να ορίσουμε τις ιδιότητες των στοιχείων που ανήκουν σε μια συγκεκριμένη τάξη (class) χρησιμοποιώντας το επίθεμα `.classname` στο style sheet του εγγράφου.

### **Αναγνώριση Στοιχείου με το Id**

Εκτός από την ομαδοποίηση των στοιχείων, μπορούμε να αναγνωρίσουμε και ένα στοιχείο μεμονωμένα με την ιδιότητα (attribute) `id`. Εκείνο που θα πρέπει να έχουμε υπόψη μας είναι ότι δεν μπορούν να υπάρχουν δύο στοιχεία (elements) στην ίδια ιστοσελίδα με το ίδιο `id`.

Δηλαδή το κάθε `id` θα πρέπει να είναι μοναδικό. Αν αυτό αποτελεί πρόβλημα, μπορούμε να χρησιμοποιήσουμε την ιδιότητα (attribute) `class`. Ακολουθεί ένα παράδειγμα χρήσης του `id` :

```
<h1>Κεφάλαιο 1</h1>
```

...

```
<h2>Κεφάλαιο 1.1</h2>
```

...

```
<h2>Κεφάλαιο 1.2</h2>
```

...

```
<h1>Κεφάλαιο 2</h1>
```

...

```
<h2>Κεφάλαιο 2.1</h2>
```

...

```
<h3>Κεφάλαιο 2.1.2</h3>
```

...

Ο παραπάνω κώδικας μπορεί να χρησιμοποιηθεί για τις επικεφαλίδες (headings) ενός εγγράφου που είναι χωρισμένο σε κεφάλαια ή παραγράφους.

Έχουμε τη δυνατότητα να εκχωρήσουμε ένα id σε κάθε κεφάλαιο για να μπορούμε να το ξεχωρίσουμε, ως εξής :

```
<h1 id="c1">Κεφάλαιο 1</h1>
```

...

```
<h2 id="c1-1">Κεφάλαιο 1.1</h2>
```

...

```
<h2 id="c1-2">Κεφάλαιο 1.2</h2>
```

...

```
<h1 id="c2">Κεφάλαιο 2</h1>
```

...

```
<h2 id="c2-1">Κεφάλαιο 2.1</h2>
```

...

```
<h3 id="c2-1-2">Κεφάλαιο 2.1.2</h3>
```

...

Αν θέλουμε τώρα να κάνουμε μόνο την επικεφαλίδα του Κεφαλαίου 1.2 να έχει χρώμα κόκκινο (red) και όχι όλες τις επικεφαλίδες του επιπέδου h2, θα πρέπει να γράψουμε το εξής CSS :

```
#c1-2 {  
  
    color: red;  
  
}
```

Όπως βλέπουμε από το παραπάνω παράδειγμα, μπορούμε να ορίσουμε τις ιδιότητες ενός συγκεκριμένου στοιχείου (element) κάνοντας χρήση του #id στο stylesheet του εγγράφου.

### ***Η Ομαδοποίηση των Στοιχείων (Span και Div)***

Τα στοιχεία <span> και <div> χρησιμοποιούνται για να ομαδοποιήσουν και να δομήσουν ένα έγγραφο και συνδυάζονται συχνά με τις ιδιότητες class και id.

Θα δούμε από κοντά τη χρήση των <span> και <div> καθώς αυτά τα δύο στοιχεία της HTML έχουν μεγάλη σπουδαιότητα για τα CSS.

## Ομαδοποίηση με το Span

Το στοιχείο `<span>` είναι αυτό που μπορούμε να αποκαλέσουμε ουδέτερο στοιχείο καθώς δεν προσθέτει τίποτα στο ίδιο το έγγραφο. Αλλά με τα CSS, το `<span>` μπορεί να χρησιμοποιηθεί για να προσθέσουμε οπτικά χαρακτηριστικά σε συγκεκριμένα κομμάτια κειμένου των εγγράφων μας.

Ακολουθεί ένα παράδειγμα :

```
<p>Στην Περιφέρεια Δυτικής Μακεδονίας ανήκουν οι Νομοί Γρεβενών, Καστοριάς, Κοζάνης και Φλώρινας.</p>
```

Ας υποθέσουμε ότι θέλουμε στο παραπάνω κομμάτι κειμένου να εμφανίζονται οι τέσσερις νομοί της Δυτικής Μακεδονίας με κόκκινο χρώμα (red). Για να γίνει αυτό, μπορούμε να απομονώσουμε τους νομούς με το `<span>` και σε κάθε span να αντιστοιχίσουμε μια τάξη (class), στην οποία μπορούμε να προσδώσουμε ένα χρώμα με το style sheet.

```
<p> Στην Περιφέρεια Δυτικής Μακεδονίας ανήκουν οι Νομοί
```

```
<span class="nomos"> Γρεβενών</span> ,
```

```
<span class="nomos"> Καστοριάς</span> ,
```

```
<span class="nomos"> Κοζάνης</span> και
```

```
<span class="nomos"> Φλώρινας</span>.</p>
```

Το CSS θα είναι το εξής :

```
span.nomos {  
    color:red;  
}
```

Ένας άλλος τρόπος για να πετύχουμε το παραπάνω αποτέλεσμα, θα ήταν να αποδώσουμε ένα id σε κάθε νομό, αλλά αυτό θα πρέπει να είναι μοναδικό για τον κάθε νομό, οπότε σε περίπτωση που έχουμε πολλά τέτοια στοιχεία, δεν αποτελεί και την καλύτερη δυνατή λύση η χρήση του id.

## Ομαδοποίηση με το Div

Ενώ το `<span>` χρησιμοποιείται μέσα σ' ένα στοιχείο επιπέδου μπλοκ, το `<div>` χρησιμοποιείται για να ομαδοποιήσει ένα ή περισσότερα στοιχεία επιπέδου μπλοκ. Εκτός απ' αυτή τη μικρή διαφορά, τα δύο αυτά στοιχεία κάνουν την ίδια περίπου δουλειά.

Θα δούμε ένα παράδειγμα με τις δύο λίστες των νομών που ανήκουν στις περιφέρειες Θεσσαλίας και Ηπείρου :

```
<div id="thessaly">
```

```
<ul>
```

```
<li>Καρδίτσα</li>
```

```
<li>Λάρισα</li>
```

```
<li>Μαγνησία</li>
```

```
<li>Τρίκαλα</li>
```

```
</ul>
```

```
</div>
```

```
<div id="epirus">
```

```
<ul>
```

```
<li>Άρτα</li>
```

```
<li>Θεσπρωτία</li>
```

```
<li>Ιωάννινα</li>
```

```
<li>Πρέβεζα</li>
```

```
</ul>
```

```
</div>
```

Θέλουμε τώρα το φόντο της λίστας με τους νομούς της Θεσσαλίας να έχει χρώμα μπλε και το φόντο της λίστας με τους νομούς της Ηπείρου να έχει χρώμα κόκκινο. Ο κώδικας για το style sheet θα πρέπει να είναι ο εξής :

```
#thessaly {
```

```
    background:blue;
```

```
}
```

```
#epirus {
```

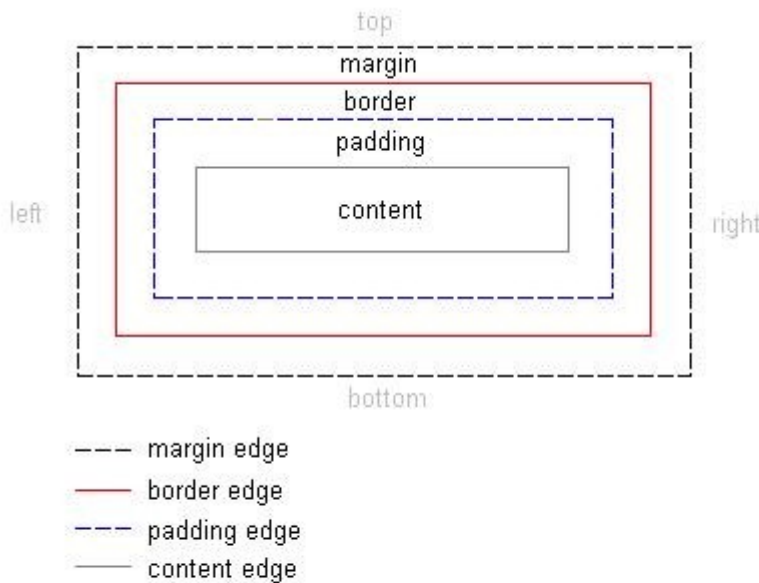
```
    background:red;
```

```
}
```

## Μέρος 3 - Το Box Model, τα Περιθώρια και τα Περιγράμματα

### *To Box Model*

Το box model στα CSS περιγράφει τα πλαίσια (boxes) που δημιουργούνται για τα στοιχεία (elements) της HTML. Το box model περιέχει επίσης λεπτομέρειες σχετικά με τη ρύθμιση των τιμών των εξής στοιχείων : margin (περιθώριο), border (περίγραμμα), padding (γέμισμα) και content (περιεχόμενο). Το παρακάτω σχεδιάγραμμα δείχνει πώς είναι δομημένο το box model μιας ιστοσελίδας :



Για να μπορέσουμε να κατανοήσουμε το παραπάνω σχεδιάγραμμα, θα δούμε ένα πρακτικό παράδειγμα με μια επικεφαλίδα (headline) και λίγο κείμενο (text). Ο HTML κώδικας που θα χρησιμοποιήσουμε θα περιέχει τη διακήρυξη των Δικαιωμάτων του Ανθρώπου και θα είναι ο εξής :

```
<h1>Article 1:</h1>
```

```
<p>All human beings are born free
```

and equal in dignity and rights.

They are endowed with reason and conscience

and should act towards one another in a

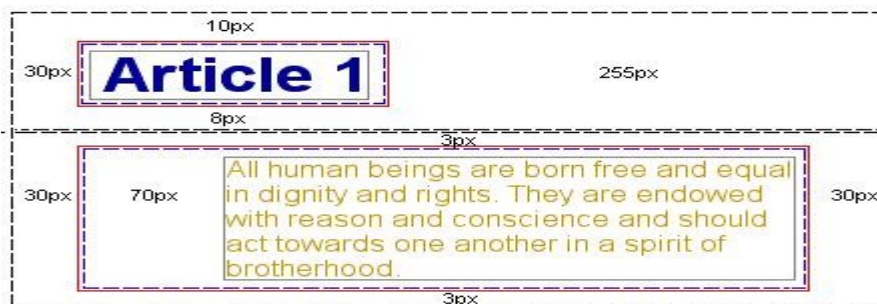
spirit of brotherhood</p>

Προσθέτοντας λίγο χρώμα και ορίζοντας γραμματοσειρές, το κείμενο θα μπορεί να εμφανισθεί ως εξής :

## Article 1

All human beings are born free and equal in dignity and rights. They are endowed with reason and conscience and should act towards one another in a spirit of brotherhood.

Το παραπάνω παράδειγμα περιέχει δύο στοιχεία, το <h1> και το <p>, και το box model γι' αυτά τα δύο στοιχεία μπορεί να απεικονισθεί ως εξής :



Από το παραπάνω σχεδιάγραμμα μπορούμε να δούμε πώς το κάθε στοιχείο της HTML περιβάλλεται από πλαίσια (boxes), τα οποία μπορούμε να ρυθμίσουμε με τα CSS.

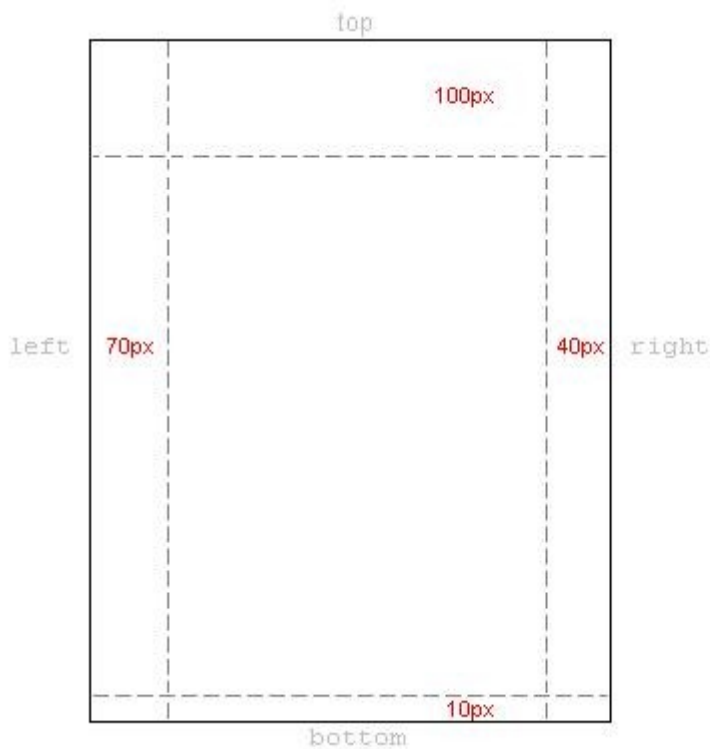
### *To Margin και το Padding*

Είδαμε νωρίτερα το box model και θα δούμε τώρα το πώς μπορούμε να αλλάξουμε την εμφάνιση των στοιχείων, δίνοντας τιμές στις ιδιότητες margin και padding.

## Η Ιδιότητα Margin

Ένα στοιχείο (element) έχει τέσσερις πλευρές : right (δεξιά), left (αριστερά), top (πάνω) και bottom (κάτω). Το margin (περιθώριο) είναι η απόσταση από την κάθε πλευρά μέχρι το γειτονικό στοιχείο ή μέχρι τα όρια (borders) του εγγράφου (document).

Ως πρώτο παράδειγμα, θα δούμε το πώς μπορούμε να ορίσουμε περιθώρια (margins) για το ίδιο το έγγραφο, δηλ. για το στοιχείο <body>. Το επόμενο σχεδιάγραμμα δείχνει τις τιμές που θέλουμε να έχουν τα περιθώρια στις ιστοσελίδες μας.



Ο CSS κώδικας για να δοθούν οι παραπάνω τιμές είναι ο εξής :

```
body {  
  
    margin-top: 100px;  
  
    margin-right: 40px;  
  
    margin-bottom: 10px;  
  
    margin-left: 70px;  
  
}
```

Μπορούμε να χρησιμοποιήσουμε και τον εξής κώδικα :

```
body {  
  
    margin: 100px 40px 10px 70px;  
  
}
```

Με τον ίδιο τρόπο μπορούμε να ορίσουμε περιθώρια (margins) για κάθε στοιχείο της HTML. Για παράδειγμα, μπορούμε να ορίσουμε περιθώρια για το κείμενο όλων των παραγράφων μιας ιστοσελίδας, δηλ. των στοιχείων που είναι μαρκαρισμένα με το tag <p> :

```
body {  
  
    margin: 100px 40px 10px 70px;  
  
}  
  
p {  
  
    margin: 5px 50px 5px 50px;  
  
}
```

### ***H Ιδιότητα Padding***

Το padding μπορούμε να πούμε ότι είναι το γέμισμα (filling) και είναι μια ιδιότητα που δεν επηρεάζει την απόσταση ανάμεσα σε κάποια στοιχεία αλλά ορίζει την εσωτερική απόσταση ανάμεσα στο περίγραμμα (border) και τα περιεχόμενα (content) του στοιχείου.

Θα δούμε ένα απλό παράδειγμα όπου όλες οι επικεφαλίδες έχουν κάποιο χρώμα φόντου για να μπορούμε να τις ξεχωρίσουμε :

```
h1 {  
  
    background: yellow;  
  
}  
  
h2 {  
  
    background: orange;  
  
}
```

Δίνοντας τιμές στην ιδιότητα padding, από μια για top, right, bottom και left, για τα στοιχεία h1 και h2, μπορούμε να ορίσουμε πόσο γέμισμα (filling) θα υπάρχει γύρω από το κείμενο της κάθε επικεφαλίδας :

```

h1 {
    background: yellow;
    padding: 20px 20px 20px 80px;
}
h2 {
    background: orange;
    padding-left:120px;
}

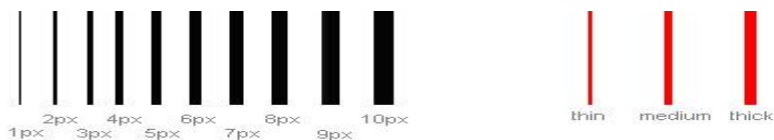
```

### ***Τα Περιγράμματα (Borders)***

Τα περιγράμματα (borders) μπορούν να χρησιμοποιηθούν σε πολλές περιπτώσεις, όπως για παράδειγμα ως διακοσμητικά στοιχεία ή για την υπογράμμιση του διαχωρισμού δύο πραγμάτων. Με τα CSS έχουμε πάρα πολλές επιλογές για να χρησιμοποιήσουμε τα περιγράμματα στις ιστοσελίδες μας.

### ***Το πλάτος (border-width)***

Το πλάτος (width) των περιγραμμάτων ορίζεται με την ιδιότητα border-width, η οποία μπορεί να πάρει μια τις τιμές thin, medium και thick ή και μια αριθμητική τιμή σε pixels. Το παρακάτω σχήμα δείχνει τις πιθανές τιμές της ιδιότητας border-width :



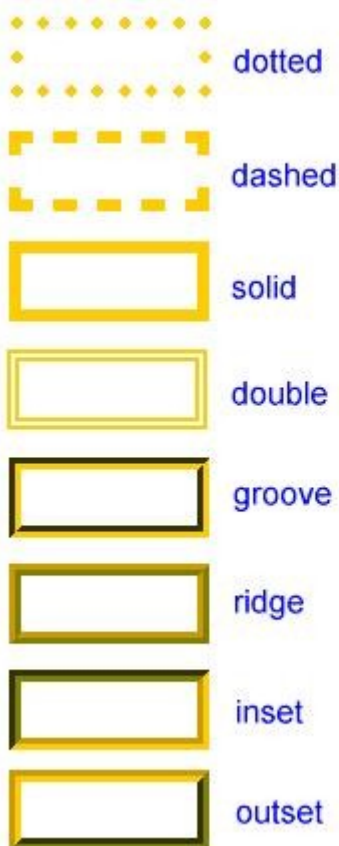
## *To χρώμα (border-color)*



Η ιδιότητα border-color ορίζει το χρώμα του περιγράμματος και οι τιμές που μπορεί να πάρει είναι οι γνωστές τιμές των χρωμάτων, όπως για παράδειγμα "#123456", "rgb(123, 123, 123)" ή "yellow".

## *To στυλ (border-style)*

Υπάρχουν διαφορετικοί τύποι (στυλ) περιγραμμάτων που μπορούμε να επιλέξουμε και παρακάτω φαίνονται 8 απ' αυτούς όπως τους ερμηνεύει ο Internet Explorer 5.5. Τα περιγράμματα αυτά έχουν χρώμα "gold" και πάχος "thick" αλλά μπορούν φυσικά να έχουν διαφορετικά χρώματα και διαφορετικά πάχη. Αν δεν θέλουμε να έχει περίγραμμα η ιστοσελίδα μας, μπορούμε να χρησιμοποιήσουμε μια από τις τιμές none ή hidden.



## *Παραδείγματα Ορισμού Περιγραμμάτων*

Οι τρεις ιδιότητες που είδαμε παραπάνω μπορούν να ορισθούν όλες μαζί για το κάθε στοιχείο και να δημιουργήσουν έτσι ενδιαφέροντα περιγράμματα. Θα δούμε ένα παράδειγμα για ένα έγγραφο όπου θα ορίσουμε διαφορετικά περιγράμματα για τα στοιχεία <h1>, <h2>, <p> και <ul> :

```
h1 {  
  
    border-width: thick;  
  
    border-style: dotted;  
  
    border-color: gold;  
  
}  
  
h2 {  
  
    border-width: 20px;  
  
    border-style: outset;  
  
    border-color: red;  
  
}  
  
p {  
  
    border-width: 1px;  
  
    border-style: dashed;  
  
    border-color: blue;  
  
}  
  
ul {  
  
    border-width: thin;  
  
    border-style: solid;  
  
    border-color: orange;  
  
}
```

Μπορούμε επίσης να ορίσουμε διαφορετικές ιδιότητες για τις τέσσερις πλευρές ενός περιγράμματος, δηλ. `border-top`, `border-bottom`, `border-right` και `border-left`, όπως φαίνεται παρακάτω :

```
h1 {  
  
    border-top-width: thick;  
  
    border-top-style: solid;  
  
    border-top-color: red;  
  
  
    border-bottom-width: thick;  
  
    border-bottom-style: solid;  
  
    border-bottom-color: blue;  
  
  
    border-right-width: thick;  
  
    border-right-style: solid;  
  
    border-right-color: green;  
  
  
    border-left-width: thick;  
  
    border-left-style: solid;  
  
    border-left-color: orange;  
  
}
```

Όπως ισχύει σε πολλές άλλες ιδιότητες, μπορούμε να συνδυάσουμε πολλές ιδιότητες των περιγραμμάτων σε μία μόνο γραμμή κώδικα.

Για παράδειγμα, ο παρακάτω κώδικας :

```
p {  
  
    border-width: 1px;  
  
    border-style: solid;  
  
    border-color: blue;  
  
}
```

Μπορεί να γραφεί και ως εξής :

```
p {  
  
    border: 1px solid blue;  
  
}
```

### ***To Ύψος (Height) και το Πλάτος (Width)***

Θα δούμε τώρα το πώς μπορούμε να ορίσουμε τις διαστάσεις, δηλ. το ύψος (height) και το πλάτος (width) ενός στοιχείου (element).

Ορισμός του πλάτους (width)

Με την ιδιότητα width μπορούμε να ορίσουμε ένα συγκεκριμένο πλάτος για ένα στοιχείο. Ο παρακάτω κώδικας δημιουργεί ένα πλαίσιο (box) μέσα στο οποίο μπορεί να υπάρχει κάποιο κείμενο :

```
div.box {  
  
    width: 200px;  
  
    border: 1px solid black;  
  
    background: orange;  
  
}
```

Ο παραπάνω κώδικας γράφεται σ' ένα αρχείο css, ενώ ο HTML κώδικας μπορεί να είναι κάπως έτσι :

```
<div class="box">... το κείμενο γράφεται εδώ ... </div>
```

Ορισμός του ύψους (height)

Στο προηγούμενο παράδειγμα το ύψος του πλαισίου ορίζεται από το περιεχόμενο που υπάρχει στο πλαίσιο. Μπορούμε να προσδιορίσουμε το ύψος ενός στοιχείου με την ιδιότητα height, όπως φαίνεται παρακάτω :

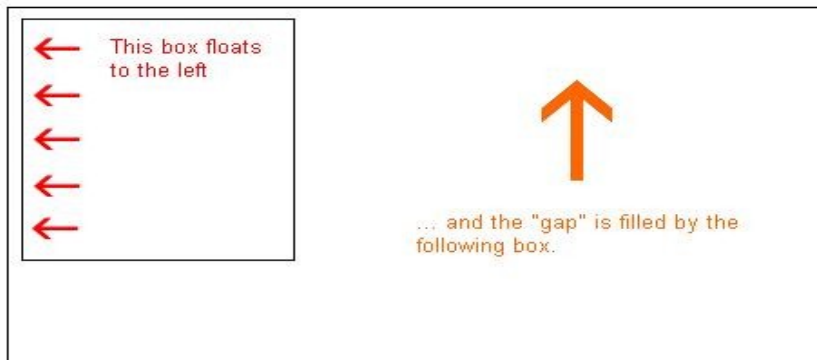
```
div.box {  
  
    height: 500px;  
    width: 200px;  
    border: 1px solid black;  
    background: orange;  
  
}
```

## Μέρος 4 – Positioning, Layers και Web Standards

### Τα Επιπλέοντα Στοιχεία

Ένα στοιχείο μπορεί να τοποθετηθεί στα δεξιά (right) ή στα αριστερά (left) με χρήση της ιδιότητας float. Αυτό πρακτικά σημαίνει ότι το πλαίσιο του στοιχείου (box) μαζί με τα περιεχόμενά του (contents) μπορεί να τοποθετηθεί είτε δεξιά είτε αριστερά σ' ένα έγγραφο.

Το επόμενο σχήμα εξηγεί τα παραπάνω :



Αν, για παράδειγμα, θελήσουμε να έχουμε ένα κείμενο που να περιβάλλει μια εικόνα, θα έχουμε το εξής αποτέλεσμα :



**A floating image**

Iste quidem veteres inter ponetur honeste, qui vel mense brevi vel toto est iunior anno. Utor permissio, caudaeque pilos ut equinae paulatim vello unum, demo etiam unum, dum cadat elusus Interdum volgus rectum videt, est ubi peccat. Si veteres ita miratur laudatque poetas, ut nihil anteferat, nihil illis comparet, errat. Si quaedam nimis antique, si peraque dure Interdum volgus rectum videt, est ubi peccat. Si veteres ita miratur laudatque poetas, ut nihil anteferat, nihil illis comparet, errat. Si quaedam nimis antique si peraque dure

Ο HTML κώδικας για το παραπάνω παράδειγμα μπορεί να είναι ο εξής :

```
<div id="picture">  
      
</div>  
<p>causas naturales et antecedentes, idcirco etiam nostrarum voluntatum...</p>
```

Για να κάνουμε την εικόνα να επιπλέει στα αριστερά και το κείμενο να την περιβάλλει, αρκεί να ορίσουμε το πλάτος (width) του πλαισίου (box) που περικλείει την εικόνα και να δώσουμε την τιμή left στην ιδιότητα float, ως εξής :

```
#picture {  
    float: left;  
    width: 100px;  
}
```

### ***Δημιουργία Στηλών (Columns)***

Η ιδιότητα float μπορεί επίσης να χρησιμοποιηθεί για τις στήλες (columns) σ' ένα έγγραφο. Για να δημιουργήσουμε στήλες, θα πρέπει να δομήσουμε τον HTML κώδικα με το tag <div>, ως εξής :

```
<div id="column1">  
<p>Haec disserens qua de re agatur  
    et in quo causa consistat non videt...</p>  
</div>  
<div id="column2">  
    <p>causas naturales et antecedentes,  
    idcirco etiam nostrarum voluntatum...</p>  
</div>  
<div id="column3">  
    <p>nam nihil esset in nostra
```

```
potestate si res ita se haberet...</p>
```

```
</div>
```

Τώρα ορίζουμε το πλάτος των στηλών ως ποσοστό, π.χ. 33%, και αφήνουμε την κάθε στήλη να πάει στα αριστερά με την τιμή left στην ιδιότητα float, ως εξής :

```
#column1 {  
    float:left;  
    width: 33%;  
}
```

```
#column2 {  
    float:left;  
    width: 33%;  
}
```

```
#column3 {  
    float:left;  
    width: 33%;  
}
```

Η ιδιότητα float μπορεί να έχει μια από τις τιμές left, right ή none.

### ***Η Ιδιότητα Clear***

Η ιδιότητα clear χρησιμοποιείται για να ελέγχει το πώς θα συμπεριφέρονται τα επόμενα στοιχεία των στοιχείων που επιπλέουν. Εξ ορισμού, τα επόμενα στοιχεία μετακινούνται για να γεμίσουν τον διαθέσιμο χώρο ο οποίος θα ελευθερωθεί όταν ένα πλαίσιο (box) τοποθετείται σε μια πλευρά.

Η ιδιότητα clear μπορεί να λάβει μια από τις τιμές left, right, both ή none.

```
<div id="picture">
```

```
    
```

```
</div>
```

```
<h1>Bill Gates</h1>
```

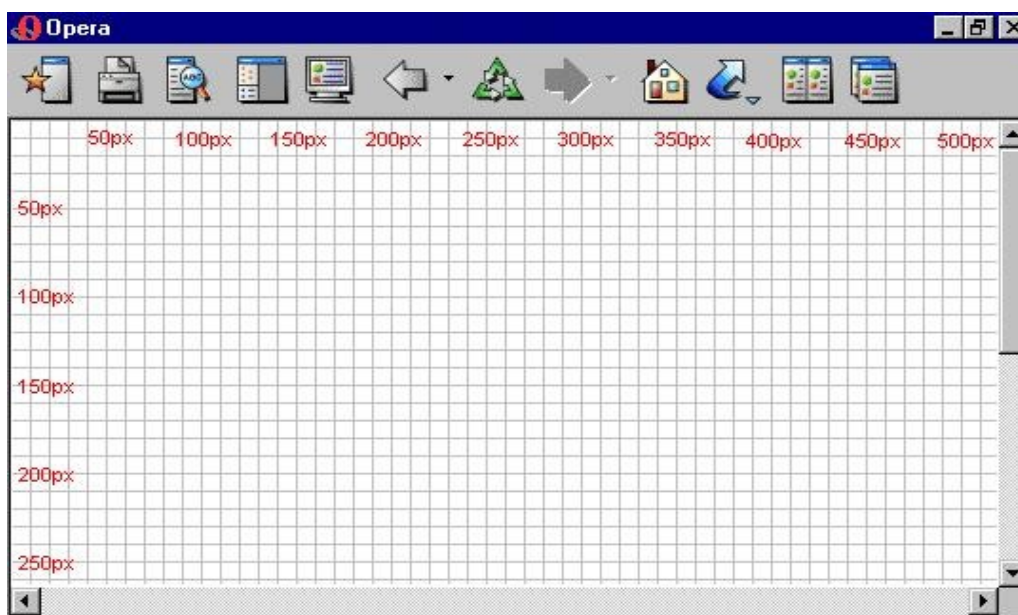
```
<p class="floatstop">causas naturales et antecedentes, idcirco etiam nostrarum voluntatum...</p>
```

Για να εμποδίσουμε το κείμενο από το να πάει δίπλα στην εικόνα, μπορούμε να προσθέσουμε τα εξής στον κώδικα των CSS :

```
#picture {  
  
    float:left;  
  
    width: 100px;  
  
}  
  
.floatstop {  
  
    clear:both;  
  
}
```

### ***Η Τοποθέτηση (Positioning) των Στοιχείων***

Με την ιδιότητα position των CSS, μπορούμε να τοποθετήσουμε ένα στοιχείο ακριβώς εκεί που θέλουμε να βρίσκεται μέσα στην ιστοσελίδα. Μαζί με την ιδιότητα float, η ιδιότητα position μάς δίνει πολλές δυνατότητες για να δημιουργήσουμε μια προηγμένη και ακριβή διάταξη (layout). Για να μπορέσουμε να κατανοήσουμε την ιδιότητα position, θα πρέπει να φανταστούμε ένα παράθυρο φυλλομετρητή ως ένα σύστημα συντεταγμένων :



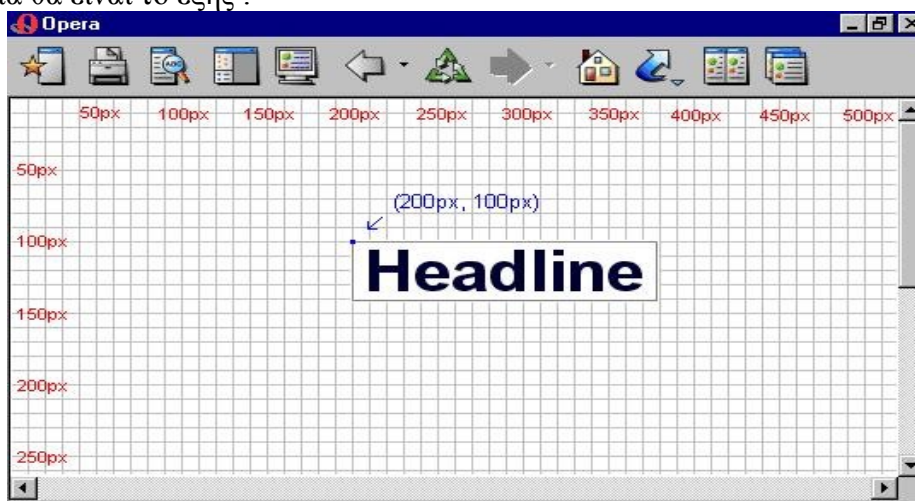
Η βασική αρχή της ιδιότητας position των CSS είναι ότι μπορούμε να τοποθετήσουμε ένα πλαίσιο (box) οπουδήποτε σ' αυτό το σύστημα συντεταγμένων. Ας υποθέσουμε ότι θέλουμε να τοποθετήσουμε μια επικεφαλίδα (headline). Με το box model, η επικεφαλίδα θα εμφανίζεται ως εξής :

**Headline**

Αν θέλουμε αυτή η επικεφαλίδα να βρίσκεται 100px από την κορυφή (top) του εγγράφου και 200px από την αριστερή πλευρά (left) του εγγράφου, θα πρέπει να γράψουμε τα εξής στον κώδικα CSS :

```
h1 {  
  
    position:absolute;  
  
    top: 100px;  
  
    left: 200px;  
  
}
```

Το αποτέλεσμα θα είναι το εξής :



Και ένα παράδειγμα με ακριβή τοποθέτηση μιας παραγράφου :

```
<html>
<style>
#p1 {
        position:absolute;
        top: 200px;
        left: 300px;
}
</style>
<body>
<div id="p1">
<p> Florina per sempre</p>
</div>
</body>
</html>
```

Όπως μπορούμε να δούμε, η τοποθέτηση των στοιχείων με τη χρήση των CSS αποτελεί μια τεχνική με μεγάλη ακρίβεια στην τοποθέτηση των αντικειμένων και είναι πολύ καλύτερη από τη χρήση πινάκων (tables) ή διαφανών εικόνων (transparent images).

### ***Η Απόλυτη Τοποθέτηση (Absolute Positioning)***

Ένα στοιχείο που τοποθετείται απόλυτα (absolute positioning) δεν καταλαμβάνει κάποιον χώρο στο έγγραφο. Για να τοποθετήσουμε ένα έγγραφο απόλυτα, θα πρέπει να δώσουμε την τιμή absolute στην ιδιότητα position και μετά μπορούμε να χρησιμοποιήσουμε τις τιμές left, right, top και bottom για να τοποθετήσουμε το πλαίσιο (box).

Στο παρακάτω παράδειγμα τοποθετήσαμε 4 πλαίσια, από ένα σε κάθε γωνία του εγγράφου :

```
#box1 {
        position:absolute;
```

```
        top: 50px;
        left: 50px;
    }
#box2 {
    position: absolute;
    top: 50px;
    right: 50px;
}
#box3 {
    position: absolute;
    bottom: 50px;
    right: 50px;
}
#box4 {
    position: absolute;
    bottom: 50px;
    left: 50px;
}
```

### ***Η Σχετική Τοποθέτηση (Relative Positioning)***

Για να τοποθετήσουμε ένα στοιχείο σε σχετική θέση (relative positioning), θα πρέπει να δώσουμε την τιμή relative στην ιδιότητα position. Η θέση ενός στοιχείου που τοποθετείται σχετικά υπολογίζεται από την αρχική θέση που είχε στο έγγραφο.

Αυτό σημαίνει ότι μπορούμε να μετακινήσουμε το στοιχείο προς τα δεξιά (right), προς τα αριστερά (left), προς τα πάνω (up) ή προς τα κάτω (down). Θα τοποθετήσουμε τώρα τρεις εικόνες σχετικά με την αρχική τους θέση στην ιστοσελίδα :

```
#dog1 {
```

```

        position:relative;

        left: 350px;

        bottom: 150px;
    }

    #dog2 {

        position:relative;

        left: 150px;

        bottom: 500px;

    }

    #dog3 {

        position:relative;

        left: 50px;

        bottom: 700px;

    }

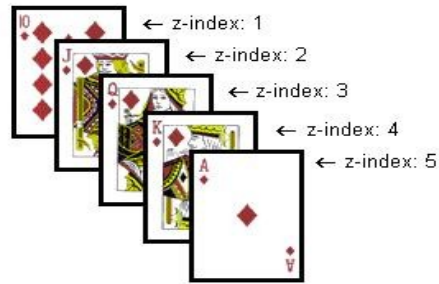
```

### ***Η Ιδιότητα z-index***

Τα CSS δουλεύουν σε τρεις διαστάσεις : ύψος (height), πλάτος (width) και βάθος (depth). Έχουμε ήδη δει τις δύο πρώτες διαστάσεις νωρίτερα και τώρα θα δούμε το πώς μπορούμε να τοποθετήσουμε κάποια στοιχεία το ένα πάνω από το άλλο και να δημιουργήσουμε έτσι επίπεδα (layers).

Για να γίνει αυτό, θα πρέπει να εκχωρήσουμε σε κάθε στοιχείο έναν αριθμό, που είναι γνωστός με τον όρο z-index. Αυτό που ισχύει είναι ότι ένα στοιχείο που έχει μεγαλύτερο αριθμό z-index θα επικαλύπτει (overlaps) ένα άλλο στοιχείο που έχει μικρότερο αριθμό z-index.

Ας υποθέσουμε ότι έχουμε 5 τραπουλόχαρτα σε αντίστοιχες εικόνες. Θέλουμε να δημιουργήσουμε την εξής διάταξη στην ιστοσελίδα μας :



Ο κώδικας θα είναι ως εξής :

```
#ten_of_diamonds {  
    position: absolute;  
    left: 100px;  
    bottom: 100px;  
    z-index: 1;  
}
```

```
#jack_of_diamonds {  
    position: absolute;  
    left: 115px;  
    bottom: 115px;  
    z-index: 2;  
}
```

```
#queen_of_diamonds {  
    position: absolute;  
    left: 130px;  
    bottom: 130px;
```

```

        z-index: 3;
    }

    #king_of_diamonds {
        position: absolute;
        left: 145px;
        bottom: 145px;
        z-index: 4;
    }

    #ace_of_diamonds {
        position: absolute;
        left: 160px;
        bottom: 160px;
        z-index: 5;
    }

```

Η μέθοδος είναι σχετικά απλή αλλά μας δίνει πολλές δυνατότητες. Μπορούμε να τοποθετήσουμε εικόνες πάνω από κείμενο ή κείμενο πάνω από ένα άλλο κείμενο κοκ.

Ακολουθεί ένα παράδειγμα με τοποθέτηση μιας εικόνας πάνω από μια άλλη εικόνα :

```

<html>

<style>

#img1 {

        position: absolute;

        top: 100px;

        left: 200px;

        z-index: 2;

    }

#img2 {

```

```

        position:absolute;

        top: 100px;

        left: 200px;

        z-index: 1;

    }

</style>

<body>

<h1 align=center>Hallo from Florina!</h1>

<div id="img1">

</div>

<div id="img2">

</div>

</body>

</html>

```

### ***Ta Web Standards και το Validation***

Το W3C (World Wide Web Consortium) είναι ένας ανεξάρτητος οργανισμός που διαχειρίζεται την κωδικοποίηση του Web, δηλ. τις γλώσσες προγραμματισμού και μορφοποίησης HTML, CSS, XML κ.ά. Η εταιρεία Microsoft, το Mozilla Foundation και πολλοί άλλοι οργανισμοί αποτελούν μέρος του W3C και συμφωνούν για τη μελλοντική ανάπτυξη των προτύπων (standards).

Αν έχετε ακόμα και μικρή εμπειρία στο Web design, θα γνωρίζετε ότι μπορεί να υπάρχουν μεγάλες διαφορές στο πώς εμφανίζεται μια ιστοσελίδα σε διάφορους φυλλομετρητές (browsers). Σίγουρα δεν αποτελεί εύκολη και ευχάριστη δουλειά να προσπαθήσετε να κάνετε μια ιστοσελίδα που να μπορεί να ειπωθεί το ίδιο καλά στους διάφορους φυλλομετρητές, όπως είναι οι Mozilla, Internet Explorer, Opera, Netscape κ.ά.

Η βασική ιδέα πίσω από τα standards είναι να υπάρξει μια συμφωνία στη χρήση των τεχνολογιών του Web. Αυτό σημαίνει ότι ένας Web developer θα είναι σίγουρος ότι η εργασία του θα

εμφανίζεται σωστά στις διάφορες πλατφόρμες.

Τα πρότυπα (standards) των CSS υπάρχουν στην ιστοσελίδα :

<http://www.w3.org/TR/REC-CSS2/>.

Το W3C έχει δημιουργήσει τον λεγόμενο Validator στην ιστοσελίδα <http://jigsaw.w3.org/css-validator/>, ο οποίος μπορεί να διαβάσει μια ιστοσελίδα και να επιστρέψει μια λίστα με λάθη (errors) και προειδοποιήσεις (warnings) αν ο CSS κώδικας δεν είναι έγκυρος.

Αν ο Validator δεν βρει λάθη, θα εμφανισθεί η επόμενη εικόνα, η οποία μπορεί να τοποθετηθεί στο Web site για να δείχθει ότι ο κώδικας είναι επικυρωμένος (validated) :

Ο Validator μπορεί να βρεθεί και στην εξής ιστοσελίδα :

<http://jigsaw.w3.org/css-validator/>