

1. Εισαγωγή

Όπως αναφέραμε σε προηγούμενη άσκηση, για τη δημιουργία ενός ιστότοπου συνεργάζονται διαφορετικές τεχνολογίες. Η HTML καθορίζει το περιεχόμενο μίας σελίδας και παρ' ότι εμπεριέχει δυνατότητες για τον στοιχειώδη καθορισμό της εμφάνισης, αυτό ανατίθεται στην τεχνολογία Cascading Style Sheets (CSS). Η γλώσσα HTML ποτέ δεν είχε ως στόχο της να προσφέρει ετικέτες για τη μορφοποίηση ολόκληρου κειμένου, παρά μόνο για τη μορφοποίηση του περιεχομένου, π.χ. `<p>Αυτή είναι μια παράγραφος.</p>`. Όταν, όμως, στην έκδοση 3.2 της γλώσσας προστέθηκαν ετικέτες και δυνατότητες προσδιορισμού της γραμματοσειράς (font) και των χρωμάτων για τα γράμματα και το υπόβαθρο, η κατασκευή δικτυακών τόπων άρχισε να γίνεται πολύπλοκη. Ειδικά, εκείνων που αποτελούνταν από πολλές σελίδες. Η ανάγκη, λοιπόν, να αντιμετωπιστεί το πρόβλημα της ομοιόμορφης μορφοποίησης των σελίδων ενός δικτυακού τόπου οδήγησε στη δημιουργία των CSS.

Οι στόχοι της άσκησης αυτής είναι:

1. η κατανόηση της σύνταξης της CSS,
2. η κατανόηση των μηχανισμών σύνδεσης της HTML με τη CSS.

Αξίζει να σημειωθεί ότι, εφόσον η CSS ασχολείται με την εμφάνιση (χρώματα, διαστήματα, γραμματοσειρές), έχει έντονα γραφιστικά χαρακτηριστικά. Η σύνταξή της είναι αρκετά εύκολη, όπως θα δούμε παρακάτω, όμως η γνώση της δεν συνεπάγεται αυτόματα την ανάπτυξη άρτιων ως προς την εμφάνιση ιστοσελίδων. Αυτό είναι συνήθως το καθήκον ενός γραφίστα και ο προγραμματιστής αναλαμβάνει τη μετάφραση της γραφιστικής εργασίας, π.χ. από Photoshop σε μορφή (format) αναγνωρίσιμη από τους πλοηγούς (browsers), δηλαδή CSS/HTML. Επιπλέον, στο πλαίσιο μια ομάδας ανάπτυξης ιστοτόπων, ο γνώστης CSS ενημερώνει τους γραφίστες ως προς τη δυνατότητα υποστήριξης και εφαρμογής διαφόρων γραφιστικών τεχνικών.

2. Σύνταξη CSS

2.1 Αρχές σύνταξης

Η σύνταξη της CSS είναι πολύ απλή:

```
ΕΠΙΛΟΓΕΑΣ {ΙΔΙΟΤΗΤΑ: ΤΙΜΗ; ΙΔΙΟΤΗΤΑ:ΤΙΜΗ; ...;}
```

Ο ΕΠΙΛΟΓΕΑΣ (selector) προσδιορίζει το στοιχείο (element) της HTML, του οποίου η μορφοποίηση θα επηρεαστεί. Όπως και η HTML, έτσι και η CSS δεν διακρίνει μεταξύ πεζών και κεφαλαίων.

Παράδειγμα, ο κώδικας:

```
p{Border:1px solid red; color:yellow; background-color:green;}
```

καθορίζει ότι οι παράγραφοι του κειμένου (ό,τι δηλαδή σημαίνεται με `<p>` εντός της σελίδας) θα εμφανίζονται με κόκκινο περίγραμμα πάχους ενός pixel (px), κίτρινα γράμματα και πράσινο φόντο (όπως στην Εικόνα 12.1). Το τελευταίο ερωτηματικό μπορεί και να λείπει.



Εικόνα 12.1. Παράδειγμα του αποτελέσματος μορφοποίησης από τον προηγούμενο κώδικα.

Αξίζει να αναφέρουμε ότι υπάρχουν επικαλύψεις μεταξύ HTML και CSS. Για παράδειγμα, το φόντο μπορεί επίσης να καθοριστεί στην HTML μέσω της ιδιότητας bgcolor. Ο στόχος της CSS είναι οτιδήποτε έχει να κάνει με την εμφάνιση, να προσδιορίζεται από κώδικα CSS.

2.2 Σύνοψη σύνταξης

Εάν κατανοήσει κανείς τα βασικά της σύνταξης και της λειτουργίας της CSS, έπειτα αρκεί μια απλή αναζήτηση στο διαδίκτυο για την εύρεση της κατάλληλης ιδιότητας. Συνεπώς, δεν έχει νόημα να παραθέσουμε λεπτομερώς όλες τις ιδιότητες, οι οποίες άλλωστε εξελίσσονται. Για παράδειγμα, στην έκδοση 3 προστίθενται ιδιότητες για σκίαση κειμένου.

Οι κυριότερες ιδιότητες οι οποίες καθορίζουν την εμφάνιση κειμένου είναι:

Ιδιότητα	Ορισμός	Παράδειγμα	Τιμές
font	Μέγεθος και γραμματοσειρά κειμένου	Font:14px Verdana	Verdana, Tahoma...
Border	Μέγεθος, τύπος και χρώμα περιγράμματος	Border:2px solid red	
Line-height	Διάστημα μεταξύ γραμμών	Line-height:120%	
Text-decoration	Υπογραμμισμένο ή διαγραμμένο κείμενο	Text-align:justify	Underline, none...

Για τη διάταξη των ετικετών στη σελίδα χρησιμοποιούμε τις ιδιότητες:

Ιδιότητα	Ορισμός	Παράδειγμα	Τιμές
float	Η ετικέτα τοποθετείται σε σχέση με την προηγούμενη ετικέτα (π.χ. αριστερά).	Float:left	Left, right, center
Clear	Η ετικέτα τοποθετείται κάτω από την προηγούμενη ετικέτα.	Clear:both	Both, right, left

Για το μέγεθος και τα περιθώρια μεταξύ ετικετών χρησιμοποιούμε:

Ιδιότητα	Ορισμός	Παράδειγμα	Τιμές
Width	Μήκος ετικέτας	Width:20%	Px ή ποσοστό επί της περικλείουσας ετικέτας
Height	Ύψος ετικέτας	Height:250px	
Padding	Εσωτερικό διάστημα	Padding-top:5px Padding:40px	
Margin	Εξωτερικό διάστημα	Margin:10px Margin-left:20px	

2.3 Responsive Design

Στις μέρες μας, η πρόσβαση στο διαδίκτυο δεν γίνεται πλέον αποκλειστικά μέσω PC αλλά μιας πληθώρας άλλων μέσων, όπως tablet ή έξυπνων τηλεφώνων, τα οποία έχουν διαφορετικό μέγεθος οθόνης και δυνατότητες. Ένας καλά σχεδιασμένος ιστότοπος θα πρέπει να εμφανίζεται ομοιόμορφα σε κάθε πιθανό μέσο. Η τεχνική η οποία ακολουθείται στο CSS για την επίτευξη αυτού του στόχου ονομάζεται Responsive Design (Αποκριτικός Σχεδιασμός). Μέρος της είναι ο βέλτιστος ορισμός ιδιοτήτων στη CSS, ούτως ώστε οι ιστοσελίδες να εμφανίζονται κατάλληλα σε κάθε μέσο. Ένας αρχικός τρόπος για να το πετύχουμε αυτό, είναι να χρησιμοποιούμε ποσοστά σε ιδιότητες, π.χ. Width:20% και όχι απόλυτες τιμές (όπως height:250px).

3. Ενοποίηση HTML με CSS

3.1 Εμβόλιμη (inline) σύνταξη

Με την παραπάνω σύνταξη μπορούμε να ορίσουμε τη μορφοποίηση μέσα στο άνοιγμα οποιασδήποτε ετικέτας HTML, αμέσως μετά το *στοιχείο* (element). Αυτό γίνεται με χρήση της ιδιότητας style η οποία είναι διαθέσιμη σε όλες τις ετικέτες, π.χ.:

```
<p style="color:green;line-height:12px"> ... </p>
```

Το αποτέλεσμα αυτής της σύνταξης είναι ότι τα γράμματα της παραγράφου θα είναι πράσινα και η απόσταση των γραμμών του κείμενου 12px.

3.2 Σύνταξη με κλάσεις (classes)

Η σύνταξη της προηγούμενης παραγράφου είναι μεν απλή αλλά όχι ευέλικτη. Ας φανταστούμε το παράδειγμα μιας σελίδας με πολλές παραγράφους. Θα μπορούσαμε να ορίσουμε το ίδιο style σε καθεμία από αυτές, επαναλαμβάνοντας τον παραπάνω κώδικα σε κάθε άνοιγμα της ετικέτας <p>. Αν, όμως, επιθυμούσαμε να αλλάξουμε τη μορφοποίηση (π.χ. το χρώμα από πράσινο σε μαύρο), αυτό θα απαιτούσε αλλαγή σε καθεμία παράγραφο χωριστά.

Ο μηχανισμός ορισμού κλάσεων της CSS μας επιτρέπει να ορίσουμε κλάσεις, δηλαδή σύνολα από χαρακτηριστικά CSS. Ο ορισμός της κλάσης γίνεται με τη μορφή:

```
.ONOMA ΚΛΑΣΗΣ {  
    ΙΔΙΟΤΗΤΑ: ΤΙΜΗ;  
    ΙΔΙΟΤΗΤΑ: ΤΙΜΗ;  
    ...  
}
```

Κάθε κλάση, δηλαδή, περικλείει πολλές ιδιότητες και τις τιμές τους σε έναν ορισμό, ο οποίος μπορεί να χρησιμοποιείται πολλαπλώς. Ένα παράδειγμα ορισμού κλάσης φαίνεται εδώ:

```
.maintxt  
{  
    color:red;  
    padding:10px;  
    border:1px solid red;  
}
```

Ο παραπάνω απλός κώδικας ορίζει μία κλάση με όνομα maintxt και ιδιότητες: κόκκινο χρώμα γραμμάτων (color:red), εσωτερικό περιθώριο 10 pixel (padding:10px) και περίγραμμα συνεχές, κόκκινου χρώματος και πάχους 1 pixel (border:1px solid red). Σε όποια ετικέτα HTML εφαρμοστεί αυτή η κλάση αποκτά αυτές τις ιδιότητες. Η εφαρμογή αυτή γίνεται πολύ απλά, π.χ. σε μία παράγραφο ή σε ένα κελί πίνακα:

```
<p class="maintxt">...</p>  
<td class="maintxt">...</td>
```

Το αποτέλεσμα εφαρμογής της κλάσης maintxt, που μόλις ορίσαμε, σε μια παράγραφο φαίνεται στην Εικόνα 12.2.

Αυτό είναι ένα παράδειγμα μορφοποίησης με κλάση.

Εικόνα 12.2 Παράδειγμα εφαρμογής της κλάσης `maintxt` σε παράγραφο.

Μπορούμε, επίσης, να ορίσουμε μορφοποιήσεις κεντρικά για κάθε τύπο ετικετών. Έτσι, αν για παράδειγμα θέλουμε όλες οι παράγραφοι του site μας να έχουν μπλε φόντο και εξωτερικό περιθώριο (`margin`) `2px`, τότε δεν ορίζουμε μία νέα κλάση, αλλά απλά δηλώνουμε και μορφοποιούμε το όνομα του στοιχείου, δηλαδή:

```
p
{
background-color:blue;
margin:2px;
}
```

Ο ορισμός των κλάσεων γίνεται είτε μέσα στην επικεφαλίδα `<HEAD>` με χρήση της ετικέτας `<STYLE>` ή σε ειδικά αρχεία (βλέπε επόμενη παράγραφο). Έχοντας ορίσει μια κλάση μπορούμε να την εφαρμόσουμε σε οποιαδήποτε ετικέτα.

```
<HEAD>
...
<STYLE>
    .bigletters {
        font-size:14px;
        font-weight:bold;
    }
    .footnote {
        font-size:10px;
        font-weight:normal;
        font-style:italic;
    }
</STYLE>
</HEAD>
```

Στον παραπάνω κώδικα ορίσαμε δύο κλάσεις: τη `bigletters`, που δηλώνει τονισμένα γράμματα μεγέθους 14 και την `footnote`, που δηλώνει πλάγια γράμματα μεγέθους 10.

Εάν θέλουμε μια ετικέτα να ακολουθήσει τις ιδιότητες μιας κλάσης που ορίστηκε με αυτόν τον τρόπο, χρησιμοποιούμε την ιδιότητα `class` με τιμή το όνομα της κλάσης (όπως δηλαδή και πριν):

```
<ΣΤΟΙΧΕΙΟ class="ΟΝΟΜΑ ΚΛΑΣΗΣ">
```

Για παράδειγμα:

```
<a class="footnote" href="paper1.pdf"> Papers </a>
```

Με αυτόν τον τρόπο μπορούμε να ορίσουμε την ίδια κλάση σε περισσότερες από μια ετικέτες. Εάν επιθυμούμε να αλλάξουμε μία από τις ιδιότητες, αρκεί να αλλάξουμε την ίδια την κλάση. Με τον τρόπο αυτό απαιτείται μία μόνο αλλαγή για όλο το αρχείο.

Ακόμη, μπορούμε να ορίσουμε ότι συγκεκριμένα μόνο στοιχεία HTML θα επηρεάζονται από μια κλάση (δηλώνοντάς την π.χ. ως `p.maintxt`). Τέλος, μπορούμε να χρησιμοποιήσουμε την ιδιότητα `id` της HTML και να ορίσουμε μια κλάση ως:

```
#mytext {
  text-align:center;
  color:red;
}
```

Και να την καλέσουμε ως:

```
<p id="mytext"> Hello World! </p>
```

Προσοχή: Τα ονόματα των κλάσεων δεν πρέπει να ξεκινούν με αριθμό.

3.3 Σύνταξη μέσω σύνδεσης με αρχείο .css

Παρότι ο τρόπος της προηγούμενης παραγράφου είναι ευέλικτος για αλλαγές, δεν είναι αρκετός, όταν αναπτύσσουμε έναν ιστότοπο με πολλαπλές ιστοσελίδες. Μια κλάση η οποία ορίζεται μέσα σε ένα αρχείο ιστοσελίδας (τύπου .htm), δεν είναι ορατή από τα υπόλοιπα άλλα αρχεία του ιστότοπου. Αν θέλουμε η μορφοποίηση σε αυτά να είναι ίδια, πρέπει να ορίσουμε την κλάση εκ νέου σε καθένα από αυτά.

Για να έχουμε μια κλάση ορατή από πολλές ιστοσελίδες, δεν την ορίζουμε μέσα στο αρχείο HTML, άλλα χρησιμοποιούμε ένα (ή περισσότερα) αρχεία με κατάληξη .css. Τα αρχεία αυτά είναι στην ουσία αρχεία κειμένου, στα οποία διαφοροποιείται η κατάληξη. Στην αρχή κάθε χωριστού αρχείου HTML χρειάζεται να γίνει αναφορά στα αρχεία CSS, ώστε ο browser να γνωρίζει πού να βρει τους ορισμούς των κλάσεων. Για όποιον είναι εξοικειωμένος με γλώσσες προγραμματισμού, όπως η C, πρόκειται για μια διαδικασία αντίστοιχη με τη σύνδεση ενός αρχείου κώδικα με εξωτερικές βιβλιοθήκες, π.χ. μέσω της εντολής INCLUDE. Η αναφορά των εξωτερικών αρχείων CSS και η σύνδεση με αυτά γίνονται στην επικεφαλίδα κάθε αρχείου HTML ως εξής:

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

Στο παράδειγμά μας το αρχείο style.css περιέχει τους ορισμούς των κλάσεων που θα χρησιμοποιηθούν στο αρχείο .html.

3.4 Εύρος εφαρμογής

Το CSS ακολουθεί μορφή «καταρράκτη» (cascade). Αυτό σημαίνει ότι μία μορφοποίηση για μία ετικέτα ισχύει για όλες τις εσωτερικές ετικέτες, εκτός αν για κάποια εσωτερική ετικέτα οριστεί άλλη μορφοποίηση, π.χ.

```
<p style="font-size:14px;color:blue"> Hello <a href="aa.htm"
style="color:red"> Go </a> </p>
```

Η λέξη Hello θα εμφανιστεί με μέγεθος 14 pixels και χρώμα μπλε. Ο υπερσύνδεσμος με τη λέξη Go θα εμφανιστεί με το ίδιο μέγεθος (14 pixels), αφού είναι στο περιεχόμενο της παραγράφου. Θα είναι, όμως, με κόκκινα γράμματα, αφού αυτό ορίζει το CSS του.

Οι ιδιότητες της CSS ορίζουν, λοιπόν, την εμφάνιση της ετικέτας για την οποία ορίζονται συμπεριλαμβανομένων και ετικετών στο εσωτερικό.

```
<p style="font-weight :bold"> Click for
<a href="paper1.pdf">Papers</a>
</p>
```

Η λέξη papers εντός του υπερσυνδέσμου θα εμφανιστεί με τονισμένα (bold) γράμματα ακολουθώντας τον ορισμό της ετικέτας <p>, αφού είναι στο εσωτερικό της.

Τι συμβαίνει, όμως, όταν μια ετικέτα περικλείεται από μια άλλη και έχουν αντικρουόμενες ιδιότητες; Τότε υπερισχύει ο ορισμός που βρίσκεται πιο κοντά στην ετικέτα. Χρησιμοποιώντας το προηγούμενο παράδειγμα, εάν θέλουμε η λέξη papers να μην είναι τονισμένη, θα γράψουμε:

```
<p style="font-weight :bold"> Click for  
<a style="font-weight:normal" href="paper1.pdf">Papers</a>  
</p>
```

Η ιδιότητα ισχύει, λοιπόν, και στο εσωτερικό των ετικετών, εκτός και αν βρει εκεί νέα τιμή. Αυτή είναι η εφαρμογή της διαδοχής ή του αποτελέσματος καταρράκτη (Cascading) που εμφανίζεται στον τίτλο του CSS.

4. Παραδείγματα

4.1 Σελίδα με διαμορφωμένη εικόνα

Για να δημιουργηθεί το αποτέλεσμα της επόμενης εικόνας, χρησιμοποιούμε συνδυασμό κώδικα HTML και CSS, όπως παρακάτω.

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



Εικόνα 12.3 Παράδειγμα συνδυασμού CSS και HTML για μορφοποίηση σελίδας.

CSS

```
.floatright  
{  
float: right;  
margin: 0 0 10px 10px;  
border: 1px solid #666;  
padding: 2px;  
}
```

HTML

```
<p>  
 Lorem ipsum dolor sit amet, consectetur...  
Lorem ipsum dolor sit amet, consectetur... Lorem ipsum dolor sit amet,  
consectetur... Lorem ipsum dolor sit amet, consectetur... Lorem ipsum  
dolor sit amet, consectetur... Lorem ipsum dolor sit amet,  
consectetur... Lorem ipsum dolor sit amet, consectetur... Lorem ipsum  
dolor sit amet, consectetur... Lorem ipsum dolor sit amet,
```

```
consectetuer... Lorem ipsum dolor sit amet, consectetuer... Lorem ipsum
dolor sit amet, consectetuer... Lorem ipsum dolor sit amet,
consectetuer... Lorem ipsum dolor sit amet, consectetuer... Lorem ipsum
dolor sit amet, consectetuer... Lorem ipsum dolor sit amet,
consectetuer... Lorem ipsum dolor sit amet, consectetuer... </p>
```

Παρατηρούμε ότι μια εμφανίσιμη σελίδα δεν χρειάζεται μεγάλου μήκους κώδικα!
Επίσης, ο σχεδιαστής ιστοσελίδων μπορεί να χρειαστεί εικόνες ελεύθερης διανομής, για να εμπλουτίσει τις σελίδες που κατασκευάζει. Στο διαδίκτυο υπάρχουν πολλές πηγές τέτοιων εικόνων, όπου οι καλλιτέχνες διαθέτουν ελεύθερα τις φωτογραφίες τους. Μια τέτοια πηγή μπορείτε να βρείτε στο [1].

4.2 Δημιουργία μενού

Όμοια, για να δημιουργηθεί η Εικόνα 12.4 πρέπει να γράψετε τον παρακάτω κώδικα. Παρατηρήστε ότι ο κώδικας HTML χρησιμοποιεί λίστες. Η διαμόρφωση μέσω CSS μετατρέπει την απλή λίστα σε μενού.

```
<!DOCTYPE html>
<HTML>
<HEAD>
<TITLE> Αυτό είναι το πρώτο μου CSS μενού! </TITLE>
<STYLE>

#navlist
{
padding: 0;
margin: 0;
list-style-type: none;
float: left;
width: 100%;
color: #fff;
background-color: #036;
}

#navlist li { display: inline; }

#navlist li a
{
float: left;
width: 5em;
color: #fff;
background-color: #036;
padding: 0.2em 1em;
text-decoration: none;
border-right: 1px solid #fff;
}

#navlist li a:hover
{
background-color: #369;color: #fff;
}
```

```

</STYLE>
</HEAD>

<BODY>

<ul id="navlist">
  <li><a href="#">Item one</a></li>
  <li><a href="#">Item two</a></li>
  <li><a href="#">Item three</a></li>
</ul>

```

<p> Στο παραπάνω παράδειγμα, κάνουμε τα στοιχεία ul και a να "επιπλέουν" (float) προς τα αριστερά. Τα στοιχεία li παρουσιάζονται εμβόλιμα (inline) χωρίς δηλαδή να υπάρχει αλλαγή γραμμής πριν ή μετά από αυτά. Με τον τρόπο αυτό επιβάλλεται, η λίστα να βρίσκεται όλη σε μια γραμμή. Το στοιχείο ul έχει πλάτος 100% και κάθε υπερσύνδεσμος στη λίστα έχει πλάτος 6em (που σημαίνει 6 φορές το μέγεθος της τρέχουσας γραμματοσειράς). Επίσης, προσθέτουμε κάποια χρώματα για να γίνει πιο όμορφο.</p>

<p> Τυπικά, στη θέση των # στα href θα μπουν οι υπερσύνδεσμοι προς τις αντίστοιχες ιστοσελίδες του site σας και το κείμενο "Item one" (και τα υπόλοιπα) θα έχουν τους τίτλους των μενού σας. Για παράδειγμα, Αρχική σελίδα, Εικόνες, Επικοινωνία, κ.λπ. </p>

<p> Αλλάξτε το μέγεθος του παραθύρου παρουσίασης για να δείτε πως συμπεριφέρεται το μενού.</p>

```

</BODY>
</HTML>

```

Item one	Item two	Item three
----------	----------	------------

Στο παραπάνω παράδειγμα, κάνουμε τα στοιχεία ul και a να "επιπλέουν" (float) προς τα αριστερά. Τα στοιχεία li παρουσιάζονται εμβόλιμα (inline) χωρίς δηλαδή να υπάρχει αλλαγή γραμμής πριν ή μετά από αυτά. Με τον τρόπο αυτό επιβάλλεται, η λίστα να βρίσκεται όλη σε μια γραμμή. Το στοιχείο ul έχει πλάτος 100% και κάθε υπερσύνδεσμος στη λίστα έχει πλάτος 6em (που σημαίνει 6 φορές το μέγεθος της τρέχουσας γραμματοσειράς). Επίσης, προσθέτονται κάποια χρώματα για να γίνει πιο όμορφο.

Τυπικά, στη θέση των # στα href θα μπουν οι υπερσύνδεσμοι προς τις αντίστοιχες ιστοσελίδες του site σας και το κείμενο "Item one" (και τα υπόλοιπα) θα έχουν τους τίτλους των μενού σας. Για παράδειγμα, Αρχική σελίδα, Εικόνες, Επικοινωνία, κ.λπ.

Αλλάξτε το μέγεθος του παραθύρου παρουσίασης για να δείτε πως συμπεριφέρεται το μενού.

Εικόνα 12.4 Παράδειγμα δημιουργίας μενού μέσω τροποποίησης στοιχείων της HTML με CSS.

4.3 Σελίδα με τρία πλαίσια

Οι δυνατότητες των CSS είναι τέτοιες που μπορούμε να δημιουργήσουμε σύνθετες σελίδες, όπως για παράδειγμα μια σελίδα με τρία πλαίσια και ανεξάρτητο τίτλο και υποσέλιδο (Εικόνα 12.5).

Τίτλος		
Lor...	Επικεφαλίδα Lorem ipsum dolor sit amet, cons....	Lor...
Υποσέλιδο		

Εικόνα 12.5 Μια σελίδα με τρία πλαίσια κατασκευασμένη με CSS.

Ακολουθεί ο κώδικας διαμόρφωσης αυτής της σελίδας.

```

<!DOCTYPE html>
<HTML>
<HEAD>
<TITLE> Αυτό είναι ένα παράδειγμα με τρία πλαίσια με χρήση CSS. </TITLE>
<STYLE>

#container {
    width: 90%; margin: 10px auto;
    background-color: #fff; color: #333;
    border: 1px solid gray;
    line-height: 130%;
}

#top {
    padding: .5em; background-color: #ddd;
    border-bottom: 1px solid gray;
}

#top h1 { padding: 0; margin: 0; }

#leftnav {
    float:left; width:60px; margin:0;
    padding:1em;
    max-width:12em;
}

#rightnav {
    float:right; width:60px; margin:0;
    padding: 1em;
    max-width:12em;
}

#content {
    margin-left:200px;
    border-left:1px solid gray;
    border-right:1px solid gray;
    padding:1em; max-width:36em;
}

#footer {

```

```

        clear: both; margin: 0; padding: .5em;
        color: #333;background-color: #ddd;
        border-top: 1px solid gray;
    }

    #leftnav p { margin: 0 0 1em 0; }
    #content h2 { margin: 0 0 .5em 0; }

</STYLE>
</HEAD>

<BODY>

<div id="container">
    <div id="top"> <h1> Τίτλος </h1> </div>

    <div id="leftnav"> <p>Lor...</p> </div>

    <div id="rightnav"> <p>Lor...</p> </div>

    <div id="content"> <h2> Επικεφαλίδα </h2> <p> Lorem ipsum dolor
sit amet, cons.... </p> </div>

    <div id="footer"> Υποσέλιδο </div>
</div>

</BODY>
</HTML>

```

Γενικά, διατάξεις, όπως αυτή, είναι αρκετά πολύπλοκες και απαιτούν αρκετό πειραματισμό κι εμπειρία για την κατασκευή τους. Η πολυπλοκότητα διαμόρφωσης τέτοιων πολύστηλων δομών, αλλά και η ανάγκη για τη χρήση τους, π.χ. σε blogs και sites εφημερίδων, οδήγησε στη δημιουργία ειδικών στοιχείων HTML και CSS που χειρίζονται πολλές στήλες ευκολότερα.

Για παράδειγμα, ένα στυλ όπως το παρακάτω δημιουργεί τρεις στήλες με συγκεκριμένο πλάτος και απόσταση. Από τον ορισμό του φαίνεται ότι υπάρχουν συγκεκριμένες δεσμευμένες λέξεις για διαφορετικούς browsers, ενώ υπάρχουν κάποιοι, όπως ο Internet Explorer 9 και οι παλαιότερες εκδόσεις του, που δεν τις υποστηρίζουν [2].

```

.newspaper {
    -webkit-column-count: 3; /* Chrome, Safari, Opera */
    -moz-column-count: 3; /* Firefox */
    column-count: 3;
    -webkit-column-gap: 40px; /* Chrome, Safari, Opera */
    -moz-column-gap: 40px; /* Firefox */
    column-gap: 40px;
}

```

5. Πρόσθετες πηγές και εκπαίδευση

Το διαδίκτυο αποτελεί έναν πολύ χρήσιμο συνοδοιπόρο στην επίλυση προβλημάτων και στην εκμάθηση νέων δεξιοτήτων. Οι ονομαζόμενες *παιδικές χαρές* (playgrounds) είναι ιστότοποι οι οποίοι επιτρέπουν τη δημιουργία κώδικα και την άμεση εμφάνιση του αποτελέσματος της εκτέλεσής του. Ορισμένα χρήσιμα είναι:

- Ο δικτυακός τόπος w3schools.com που αποτελεί μια εξαιρετική πηγή γνώσης και χώρο εκπαίδευσης στις τεχνολογίες του ιστού [2].
- Ο δικτυακός τόπος <http://liveweave.com/> στον οποίο επίσης προσφέρεται περιβάλλον ανάπτυξης και δοκιμής κώδικα HTML, CSS3 και JavaScript.

Επίσης, υπάρχουν ορισμένα ενδιαφέροντα sites τα οποία επιτρέπουν να σχεδιάσεις γραφικά το CSS και σου επιστρέφουν τις παραμέτρους του κώδικα που πρέπει να χρησιμοποιήσεις, για να έχεις το αποτέλεσμα που βλέπεις. Τέτοια είναι τα:

- <http://css3generator.com/>
- <http://css3.mikeplate.com/>