

2019 -2020

HTML

Hyper Text Markup Language



Σπύρος Ζώγας

Δ.Π.Μ.Σ. «ΠΛΗΡΟΦΟΡΙΚΗ ΤΗΣ ΥΓΕΙΑΣ» ΝΟΣΗΛΕΥΤΙΚΗ ΕΚΠΑ

Γλώσσα χαρακτηρισμού υπερκειμένου

HTML

Περιεχόμενα

Εισαγωγή	3
Δίκτυα	3
Δικτυακές εφαρμογές	3
TCP/IP	4
World Wide Web (WWW)	5
Το πρωτόκολλο μεταφοράς υπερκειμένου HTTP	5
Client – Server μοντέλο στο Web	6
Υπερκείμενο (hypertext)	6
Ιστοσελίδες – web Site	7
Web browser	7
Γενικά για την HTML	8
HTML 5	10
URL (Uniform Resource Locator)	10
WYSIWYG HTML Editors	11
HTML Hyper Text Markup Language (Γλώσσα Χαρακτηρισμού Υπερκειμένου)	13
Ετικέτες (tags) ή element (έννοιες)	13
Δομή της HTML	14
Βασικές ετικέτες html	14
• Οι κύριες ετικέτες tags του HEAD:	14
• Οι κύριες ετικέτες tags (Elements) του BODY είναι:	15
Χαρακτηριστικά (Attributes) ετικετών (tags), Ιδιότητες (Properties) χαρακτηριστικών και Τιμές (Values). 17	
• Τα κύρια χαρακτηριστικά (attributes) της ετικέτας (tag) <HTML>	17
• Τα κύρια χαρακτηριστικά (attributes) της ετικέτας (tag) <body>	17
• Τα βασικότερο attribute του element <h>:	17
• Τα βασικότερα χαρακτηριστικά (attribute) της ετικέτας (tag) <p>:	18
Υπερσύνδεσμος - HTML links	19
• Τα βασικότερο χαρακτηριστικά (attribute) της ετικέτας (tag) <a>:	19
• Ετικέτα <name>	19
Εικόνα 	20
• Τα βασικότερο attribute του tag :	20

Λίστες	21
• Λίστα με bullets	21
• Αριθμημένη Λίστα	21
• Λίστα Ορισμών	21
Πίνακες	22
• Χαρακτηριστικά (Attributes) πινάκων.....	23
Φόρμες <form> και </form>	24
• Τα κυριότερα χαρακτηριστικά (attributes) της ετικέτας <form> είναι:.....	24
.....	25
Άλλες ετικέτες φορμών	25
• Η ετικέτα <input>	25
• Η ετικέτα <textarea>	27
• Η ετικέτα <select>	27
Κρυφά πεδία της Φόρμας	28
Αποστολή στοιχείων στον Server (Μέθοδος GET και POST).....	28
CSS Cascading Style Sheets (στυλ που μπορούμε να ορίσουμε για τις HTML σελίδες)	30
Βασικοί κανόνες σύνταξης CSS	31
Ορισμός των στυλ.....	31
Σύνταξη των στυλ	34
• Ετικέτες που περιέχουν το χαρακτηριστικό class	34
• Ετικέτες που περιέχουν το χαρακτηριστικό id.....	35
• Σύνταξη στυλ για ετικέτες με συγκεκριμένη ιδιότητα και τιμή	36
• Ψευδό-κλάσεις (Pseudo-classes)	36
• Βασικές CSS ιδιότητες (CSS properties).....	37

Εισαγωγή

Δίκτυα

Στην αρχή κάθε υπολογιστής ήταν ένα αυτόνομο μηχάνημα το οποίο για να χρησιμοποιηθεί από κάποιον, έπρεπε αυτός να βρίσκεται απέναντι του και εκτελούσε κάθε φορά ένα πρόγραμμα. Πολύ σύντομα με την βελτίωση της τεχνολογίας των υπολογιστών έκαναν την εμφάνισή τους γρήγορες δευτερεύουσες μνήμες και λειτουργικά συστήματα που υποστήριζαν πολυπρογραμματισμό. Δόθηκε λοιπόν η δυνατότητα του καταμερισμού του χρόνου της κεντρικής μονάδας επεξεργασίας μεταξύ περισσότερων προγραμμάτων κάτι που άνοιξε τον δρόμο για ταυτόχρονη πρόσβαση πολλών χρηστών μέσω τερματικών συσκευών στον ίδιο υπολογιστή.

Οι τερματικές συσκευές δεν ήταν αυτόνομοι υπολογιστές και δεν έκαναν τίποτε περισσότερο από το να συνδέονται στον κεντρικό υπολογιστή και να κάνουν διαθέσιμο αυτόν και τα περιφερειακά του στους χρήστες των τερματικών.

Αρχικά τα τερματικά βρίσκονταν κοντά στον υπολογιστή, αργότερα με την βοήθεια συσκευών modem και του τηλεφωνικού δικτύου, μπορούσαν να είναι διασκορπισμένα σε μεγάλη γεωγραφική περιοχή. Θα μπορούσε κανείς να μιλήσει για μια μορφή πρωτόγονων δικτύων.

Με την τεχνολογία των ολοκληρωμένων κυκλωμάτων το μέγεθος των Η/Υ μίκρυνε τόσο ώστε να μπορέσουν να αποτελέσουν μέρος ενός γραφείου. Το κόστος παραγωγής τους έπεσε κατακόρυφα και η απόκτηση ενός υπολογιστή έγινε δυνατή για τον καθένα. Συγχρόνως η ισχύς τους αυξήθηκε σημαντικά και αυξάνεται μέχρι και σήμερα με γεωμετρικούς ρυθμούς.

Σε μικρό χρονικό διάστημα πολλές επιχειρήσεις έφθασαν να έχουν πολλούς υπολογιστές γεωγραφικά κατανομημένους στα διάφορα τμήματα της επιχείρησής τους (δωμάτια, ορόφους, κ.τ.λ.). Προέκυψε λοιπόν η ανάγκη αυτοί οι υπολογιστές να επικοινωνούν μεταξύ τους για να **μοιράζονται** ακριβώς περιφερειακά (εκτυπωτές, κεντρικούς χώρους αποθήκευσης κ.α.) αλλά και για να **μεταφέρουν** πληροφορίες (εικόνα, κείμενο, ήχος). Δημιουργήθηκαν έτσι τα τοπικά δίκτυα υπολογιστών.

Η αυξημένη ανάγκη για επικοινωνίες και μεταφορά πληροφοριών έκανε ώστε να συνδεθούν αυτά τα μεμονωμένα δίκτυα μεταξύ τους, δημιουργώντας μεγαλύτερα δίκτυα με τελική κατάληξη το internet.

Είναι φανερό ότι τα δίκτυα υπολογιστών ήταν το σημείο συνάντησης της βιομηχανίας των τηλεπικοινωνιών και των ηλεκτρονικών υπολογιστών.

Δικτυακές εφαρμογές

Όλες οι **δικτυακές εφαρμογές** σε αντίθεση με τις **τοπικές εφαρμογές** είναι διαθέσιμες μέσω ενός δικτύου. Τα δίκτυα υπάρχουν για να καταστήσουν δυνατή την επικοινωνία μεταξύ δύο ή περισσότερων μερών. Έτσι οι δικτυακές εφαρμογές χωρίζονται σε δύο μέρη. Το πρώτο είναι αυτό

που **παρέχει την υπηρεσία (Server)** και το δεύτερο αυτό που την **δέχεται (Client)**. Μια δικτυακή εφαρμογή λοιπόν είναι «ένα σε δύο».

Η επικοινωνία μεταξύ των δύο τμημάτων λογισμικού μιας δικτυακής εφαρμογής μπορεί να απλοποιηθεί στα παρακάτω βήματα.

1. Ο χρήστης στέλνει μία αίτηση ή ένα ερώτημα, μέσω της εφαρμογής Client στον Server.
2. Ο Server ακούει την αίτηση του client. (Δεν ξεκινά πρώτος καμία ενέργεια. Αντίθετα περιμένει να φτάσουν οι αιτήσεις του Client μέσω δικτύου. Επίσης πρέπει να είναι σε θέση να δέχεται πολλές ταυτόχρονες αιτήσεις από πολλούς client (τεχνολογία socket)).
3. Από την στιγμή που ο Server δέχεται την αίτηση πρέπει να βεβαιωθεί ότι ο client είναι εξουσιοδοτημένος να λάβει την απάντηση που ζητά. Αν ο client δεν είναι εξουσιοδοτημένος, ο Server απορρίπτει την αίτηση και στέλνει ανάλογο μήνυμα στον client. Εάν ο client είναι εξουσιοδοτημένος, ο Server συνεχίζει και επεξεργάζεται την αίτηση.
4. Ο server επιστρέφει τα αποτελέσματα στον client.
5. Ο client δέχεται τα αποτελέσματα και τα παρουσιάζει στον χρήστη

TCP/IP

Το λογισμικό (πρωτόκολλο) που απαιτείτε να είναι εγκατεστημένο στους υπολογιστές που θέλουν να συμμετάσχουν στο διαδίκτυο και το οποίο δημιουργεί το δικτυακό περιβάλλον πάνω στο οποίο θα μπορέσουν να τρέξουν οι δικτυακές εφαρμογές είναι το **TCP/IP**. *(Κάτ. αντιστοιχία με τα λειτουργικά συστήματα προσωπικών υπολογιστών (πχ windows) που δημιουργούν ένα περιβάλλον εργασίας προσωπικού υπολογιστή πάνω στο οποίο εγκαθίστανται οι τοπικές εφαρμογές).*

Τα πρωτόκολλα είναι αναγκαία γιατί αρχικά υπολογιστές μόνο της ίδιας εταιρίας μπορούσαν να επικοινωνούν μεταξύ τους. Έτσι διάφορες εταιρίες δημιούργησαν τα δικά τους δίκτυα υπολογιστών. Τέτοια είναι το δίκτυο SNA (System Network Architecture) της IBM, το δίκτυο DNA (Digital Network Architecture) της DEC κ.α. Αυτά τα ξεχωριστά δίκτυα λειτουργούσαν με ειδικά προγράμματα που παρείχαν οι αντίστοιχες εταιρίες υλοποίησής τους. Όταν ήταν επιθυμητή η επικοινωνία μεταξύ τους, έπρεπε κάθε φορά να αναπτυχθεί κατάλληλο λογισμικό συνεννόησης, κάτι πολύ δύσκολο και δαπανηρό. Η μόνη λύση ήταν η αποδοχή και η υλοποίηση από τους διαφορετικούς προμηθευτές υπολογιστών, ενός κοινού συνόλου κανόνων. Έτσι δημιουργήθηκαν τα περίφημα πρότυπα (standards) από τους αρμόδιους διεθνείς οργανισμούς, προκειμένου να υπάρχει συνεννόηση μεταξύ διαφορετικών αρχιτεκτονικών.

Ωστόσο το έργο της επικοινωνίας μεταξύ υπολογιστών είναι πολύ σύνθετο ώστε να μελετηθεί σαν σύνολο. Το πρόβλημα πρέπει να χωρισθεί σε ξεχωριστά τμήματα καθένα από τα οποία θα προτυποποιηθεί ξεχωριστά. Πριν λοιπόν την ανάπτυξη συγκεκριμένων προτύπων έπρεπε να δημιουργηθεί η δομή που θα διέπει τις επικοινωνίες των υπολογιστών.

Οι παραπάνω λόγοι οδήγησαν τον οργανισμό ISO (International Standards Organization) το 1977 να ορίσει επιτροπή για την ανάπτυξη της αρχιτεκτονικής των δικτύων. Η επιτροπή δημιούργησε το μοντέλο διασύνδεσης ανοικτών συστημάτων OSI (Open System Interconnection) που αποτελεί και το πλαίσιο για τον καθορισμό προτύπων (συστάσεων) για την διασύνδεση ετερογενών υπολογιστών.

Πρέπει να υπογραμμισθεί ότι το OSI δεν είναι συστάσεις (πρωτόκολλο), αλλά απλά παρέχει το πλαίσιο στο οποίο θα αναπτυχθούν οι συστάσεις (πρωτόκολλα). Το OSI εξηγεί πώς πρέπει να δουλεύει ένα δίκτυο και αποτελεί το θεμέλιο της θεωρίας της δικτύωσης.

Το μοντέλο OSI πρακτικά δεν υλοποιήθηκε ποτέ στην πράξη, όμως έδωσε μία θεωρητική βάση για την δημιουργία πρωτοκόλλων.

Η αρχιτεκτονική που ακολούθησε το OSI ήταν η στρωμάτωση (Layering). Χώρισε την επικοινωνία δικτύων σε 7 στρώματα καθένα από τα οποία κτίζεται πάνω στο προηγούμενο. Κάθε στρώμα χρησιμοποιεί τις υπηρεσίες του κατωτέρου του και προσφέρει υπηρεσίες στο ανώτερο του.

Μεταξύ δύο υπολογιστών τα όμοια στρώματα επικοινωνούν μεταξύ τους με ένα σύνολο κανόνων που ονομάζονται πρωτόκολλα (protocols).

Αντίθετα με το OSI το TCP/IP είναι ένα υλοποιήσιμο πρωτόκολλο το οποίο προηγήθηκε του OSI και συνεπώς αν και όπως είναι φυσικό έχει παρόμοια λογική, όμως δεν δημιουργήθηκε με βάση αυτό. Είναι το δημοφιλέστερο πρωτόκολλο με διαφορά σε σχέση με άλλα. Αυτό οφείλεται στο γεγονός ότι πρόκειται για ένα ανοικτό πρωτόκολλο το οποίο δεν ελέγχεται από καμία εταιρία και επίσης είναι το πρωτόκολλο που διακινεί τα δεδομένα μέσω Internet.

Το TCP/IP είναι ένα σύνολο πρωτοκόλλων (4 στρώματα) τα οποία έχουν στάνταρ τρόπους για την μεταξύ τους επικοινωνία. Το όνομά του είναι λίγο παραπλανητικό γιατί δύο από τα πρωτόκολλα που το αποτελούν, τα TCP και IP έδωσαν το όνομά τους σε ολόκληρη την σουίτα πρωτοκόλλων.

Σε αντίθεση με τα επτά επίπεδα του μοντέλου αναφοράς OSI, το TCP/IP χρησιμοποιεί μόνο τέσσερα επίπεδα, ορισμένα από τα οποία συγχωνεύουν περισσότερες από μία λειτουργίες των επιπέδων OSI, σε ένα επίπεδο TCP/IP.

World Wide Web (WWW)

Το World Wide Web (ή σκέτο **Web**) είναι μια υπηρεσία του ίντερνετ που μας επιτρέπει με ιδιαίτερα εύκολο τρόπο να έχουμε πρόσβαση σε μια τεράστια συλλογή ψηφιακών εγγράφων. Ο δομικός λίθος του www είναι οι ιστοσελίδες (Web Pages). Οι υπολογιστές για χρήση του web χρησιμοποιούν το πρωτόκολλο επικοινωνίας **HTTP**.

Το www είναι η πλέον αναπτυσσόμενη υπηρεσία του ίντερνετ, σε σημείο μάλιστα που πολλοί νομίζουν ότι οι όροι ίντερνετ και www είναι ταυτόσημοι, κάτι που δεν ισχύει.

Το πρωτόκολλο μεταφοράς υπερκειμένου HTTP

Με απλά λόγια το **HTTP** (Hyper Text Transfer Protocol) είναι ένα πρωτόκολλο (κατάλληλο λογισμικό) που κατασκευάστηκε προκειμένου να μπορεί να υλοποιηθεί η υπηρεσία Web. Έχει σχεδιαστεί και χρησιμοποιείται για την μεταφορά αρχείων υπέρ-κειμένου (hypertext files).

Για την λειτουργία του χρειάζονται:

- ένα πρόγραμμα πελάτη (http client program ή web client) από την μία μεριά, το οποίο παρέχει στον χρήστη το κατάλληλο περιβάλλον για την προσπέλαση μιας παγκόσμιας συλλογής από έγγραφα, που συνήθως ονομάζονται web σελίδες.
- και από την άλλη μεριά χρειάζεται ένα πρόγραμμα εξυπηρετητή (http Server) το οποίο διατηρεί ένα σύνολο από σελίδες που μπορούν να προσπελαστούν από τους χρήστες με την χρήση των web πελατών (client).

Το http καθορίζει αναλυτικά την μορφή που θα έχει το μήνυμα αίτησης (Request) που στέλνει ο Client στον Server και την μορφή του μηνύματος απάντησης (Response) που επιστρέφει ο server στον client.

Οι υπηρεσίες του πρωτοκόλλου http συνήθως παρέχονται πάνω σε δίκτυα που τρέχουν το TCP/IP με εξ'ορισμού (default) πόρτα λειτουργίας την 80. Όμως η http επικοινωνία μπορεί κάλλιστα να επιτευχθεί και πάνω σε άλλα πρωτόκολλα μεταφοράς στο ίντερνετ ή άλλα δίκτυα. Η μόνη προϋπόθεση για το http είναι το πρωτόκολλο του υποστρώματος να εξασφαλίζει αξιόπιστη μεταφορά. Είναι δηλαδή το HTTP ανεξάρτητο τόσο από το πρωτόκολλο όσο και από το δίκτυο πάνω στο οποίο μπορεί να δουλέψει.

Client – Server μοντέλο στο Web

Στην περίπτωση της Web διαδικτυακής υπηρεσίας ο τρόπος επικοινωνίας μεταξύ client και server καθορίζεται όπως προαναφέρθηκε από το πρωτόκολλο **HTTP (Hypertext Transfer Protocol)**. Μια εφαρμογή web, όπως όλες οι δικτυακές εφαρμογές, χωρίζεται σε δύο μέρη:

- **To Server λογισμικό:** Ένας web server (Apache, IIS κ.λπ.) που φιλοξενεί και στέλνει τις ιστοσελίδες.
- **To Client λογισμικό:** Συνήθως ο web browser (IE, Mozilla Firefox, Chrome κ.λπ.) που ζητάει, δέχεται και προβάλλει τις ιστοσελίδες.

Μια τυπική συναλλαγή μεταξύ client και server με το http έχει ως εξής:

1. **Σύνδεση:** Εγκαθίσταται σύνδεση μεταξύ client και server (TCP/IP, Port, Socket).
2. **Αίτηση:** Ο client στέλνει μια αίτηση προς τον server ζητώντας κάποια ιστοσελίδα.
3. **Απόκριση:** Ο server απαντά στον client με ένα μήνυμα απάντησης και την ιστοσελίδα.
4. **Κλείσιμο:** τερματίζεται η μεταξύ τους σύνδεση.

Υπερκείμενο (hypertext)

Το **υπερκείμενο** είναι ένας μη γραμμικός (μη σειριακός) τρόπος οργάνωσης πληροφοριών. Το υπερκείμενο θεωρείται μη γραμμικό κείμενο σε αντίθεση με το βιβλίο που θεωρείται γραμμικό. Η γραμμικότητα του βιβλίου έγκειται στο γεγονός ότι ο αναγνώστης οφείλει να διαβάσει τη μία σελίδα (ή παράγραφο) μετά την άλλη προκειμένου να κατανοήσει το περιεχόμενό του, δίχως να του δίδεται η δυνατότητα να «πλοηγείτε» ελεύθερα εντός αυτού. Σε ένα υπερκείμενο μέσω των **υπερσυνδέσμων** μπορούμε να πλοηγούμαστε ελεύθερα μεταξύ των ιστοσελίδων του υπερκειμένου.

Τα αρχεία υπερκειμένου (**ιστοσελίδες**) περιέχουν κείμενο και άλλες μορφές οπτικοακουστικής πληροφορίας και συνδέονται μεταξύ τους με μη γραμμικές και ασθενώς ιεραρχημένες σχέσεις που λέγονται υπερσύνδεσμοι (hyperlinks).

Με απλά λόγια υπερκείμενο (hypertext) ονομάζουμε κάθε ηλεκτρονικό κείμενο που έχει διασυνδεθεί με άλλα υπερκείμενα. Σαν διασύνδεση ορίζουμε την παροχή στον χρήστη της δυνατότητας να μεταφέρεται με παραπομπές σε άλλα κείμενα (συνήθως σχετικά με το τρέχον).



Εικόνα 1: Web

Στην παραπάνω εικόνα οι κόκκινες γραμμές παριστάνουν τους συνδέσμους με τους οποίους μεταβαίνουμε από μια σελίδα σε μια άλλη. Οι σύνδεσμοι, με λίγα λόγια, υλοποιούν την έννοια του υπερκειμένου.

Ιστοσελίδες – web Site

Οι ιστοσελίδες είναι αρχεία υπερκειμένου που βρίσκονται σε υπολογιστές που ονομάζονται **Web Servers** (οι οποίοι είναι 24 ώρες το 24ωρό συνδεδεμένοι με το ίντερνετ ώστε να μπορούμε οποιαδήποτε στιγμή να συνδεθούμε και να ανακτήσουμε μια ιστοσελίδα που βρίσκεται σε αυτούς), ενώ οι υπολογιστές που συνδέονται στους Web Servers για να ανακτήσουν τις ιστοσελίδες λέγονται **Web Clients**.

Κατά κανόνα ένας web server δεν περιέχει μόνο μία ιστοσελίδα αλλά μία ολόκληρη τοποθεσία ιστού (web site). Δηλαδή μία ομάδα ιστοσελίδων συνδεδεμένες μεταξύ τους, με θέματα που αφορούν έναν ιδιώτη, μία επιχείρηση, έναν οργανισμό ή οτιδήποτε άλλο.

Εκτός του ότι οι ιστοσελίδες ενός web site είναι συνδεδεμένες μεταξύ τους, σχεδόν πάντα συνδέονται και με άλλα web site άλλων web server.

Web browser

Οι χρήστες του Internet προκειμένου να μπορούν να δουν τις ιστοσελίδες που βρίσκονται στους Web Servers, χρησιμοποιούν κάποια προγράμματα περιήγησης (web browsers) όπως ο Microsoft Edge, ο Firefox, ο Chrome κ.α. Τέτοια προγράμματα εγκαθίστανται στους υπολογιστές των χρηστών

και υποστηρίζουν το πρωτόκολλο που κρύβεται πίσω από την υπηρεσία www, το http. Αυτές οι εφαρμογές αποτελούν το client κομμάτι μιας εφαρμογής web.

Δύο είναι οι κύριες υποχρεώσεις του Web browser:

- να μπορεί να προσπελάσει τις ιστοσελίδες μετά από μια αίτηση του χρήστη και
- να εμφανίζει τα περιεχόμενα των ιστοσελίδων.

Μία ιστοσελίδα κατασκευάζεται με μια γλώσσα που ονομάζεται **HTML**. Ο κώδικας html αποτελούνται από οδηγίες (**tags** ή **ετικέτες**). Αυτές οι ετικέτες καθορίζουν τον τρόπο που θα εμφανιστούν τα περιεχόμενα τους στον Web browser. Οι Web browsers διαβάζουν τις οδηγίες αυτές και εμφανίζουν τις σελίδες στην οθόνη μας.

Γενικά για την HTML

Η HTML είναι η γλώσσα με την οποία κατασκευάζουμε ιστοσελίδες (έγγραφα υπερκειμένου). Τα αρχικά HTML σημαίνουν **Hypertext Markup Language**. Είναι μία περιγραφική γλώσσα η οποία περιγράφει στον web browser τον τρόπο με τον οποίο θα τοποθετήσει τα περιεχόμενα (κείμενα, εικόνες, ήχους κ.τ.λ) επάνω στην σελίδα που θα εμφανιστεί στον τελικό χρήστη καθώς και τους συνδέσμους που έχει η σελίδα.

Η HTML είναι μια συγκεκριμένη εφαρμογή μιας πιο γενικής markup γλώσσας που ονομάζεται SGML (Standard Generalized Markup Language, ISO 8879) και είναι ένα πρότυπο για την περιγραφή markup γλωσσών.

Σαν markup γλώσσες χαρακτηρίζουμε τις γλώσσες εκείνες που βασίζονται σε οδηγίες (tags ή markups). Οι οδηγίες αυτές ενσωματώνονται στην web σελίδα και ουσιαστικά κατευθύνουν τον browser στο πώς να παρουσιάσει και να διαχειριστεί το έγγραφο του web.

Τα tags εκτός του να μορφοποιούν κατάλληλα ένα κείμενο μπορεί να αναφέρονται και σε εικόνες γραφικών, video, ήχο κ.τ.λ. που βρίσκονται σε αρχεία και καθοδηγούν τον browser πώς να τα ανακαλέσει και να τα παρουσιάσει στην σελίδα. Επίσης υλοποιούν την έννοια του υπερσυνδέσμου.

Η τελευταία έκδοση της γλώσσας HTML είναι η HTML5.

Οι ιστοσελίδες που επισκεπτόμαστε στο Internet δεν είναι τίποτε άλλο παρά αρχεία τα οποία περιέχουν κώδικα γραμμένο στην γλώσσα HTML.

Οι browsers (όπως οι Mozilla Firefox, Google Chrome, Opera, Internet Explorer και αρκετοί άλλοι) διαβάζουν τα αρχεία αυτά και εμφανίζουν το αποτέλεσμα του κώδικα HTML στην οθόνη μας.

Τα αρχεία HTML έχουν επέκταση .html ή .htm (εκτός βέβαια κι' αν η ιστοσελίδα είναι δυναμική οπότε έχει επεκτάσεις όπως .php, .asp, .jsp κτλ.). Πρόκειται για απλά αρχεία κειμένου σε μορφή ASCII και δεν περιέχουν πληροφορίες για το περιβάλλον ή τα προγράμματα με τα οποία θα λειτουργήσουν. Μπορείτε να ανοίξετε και να δείτε τα αρχεία .htm ή .html με οποιονδήποτε επεξεργαστή κειμένου, π.χ. Σημειωματάριο (Notepad) των Windows.

```

<head>
<meta http-equiv="Content-Language" content="el">
<meta http-equiv="Content-Type" content="text/html; charset=windows-1253">
<title>Επιλογές</title>
</head>
<body topmargin="10" leftmargin="10" background="bugroundMenu.jpg" link="#453FA2" vlink="#453FA2">
<table id="AutoNumber1" style="BORDER-TOP-WIDTH: 0px; BORDER-BOTTOM-WIDTH: 0px; BORDER-COLLAPSE: collapse;
BORDER-RIGHT-WIDTH: 0px" borderColor="#ff0000" height="1" cellSpacing="0" cellPadding="0" width="96%" bgColor="#ffffff"
border="1">
<tr>
<td style="BORDER-RIGHT: medium none; BORDER-TOP: medium none; BORDER-LEFT: 1px solid; BORDER-BOTTOM: 1px solid"
width="100%" height="47">
<p align="center"><b><font face="Verdana" color="#ff0000">ΤΜΗΜΑ ΝΟΣΗΛΕΥΤΙΚΗΣ
</font></b>
<b><font face="Verdana"
color="#ff0000">ΠΑΝΕΠΙΣΤΗΜΙΟΥ
ΑΘΗΝΩΝ</font></b></td>
</tr>
<tr>
<td style="BORDER-RIGHT: medium none; BORDER-LEFT: 1px solid; BORDER-BOTTOM: 1px solid" vAlign="top" width="100%"
height="171">
<h1 style="PAGE-BREAK-AFTER: auto; MARGIN-LEFT: 21.3pt; TEXT-INDENT: -21.3pt; LINE-HEIGHT: normal; TEXT-ALIGN: center">
&nbsp;</h1>
<h1 style="PAGE-BREAK-AFTER: auto; MARGIN-LEFT: 10px; TEXT-INDENT: 5px; MARGIN-RIGHT:
10px; TEXT-ALIGN: center">
<font style="FONT-SIZE: 9pt" face="Verdana">
<span style="FONT-WEIGHT: normal; TEXT-DECORATION: underline">ΙΣΤΟΡΙΚΗ
ΑΝΑΔΡΟΜΗ</span></font></h1>
<p style="MARGIN: 5px 10px; TEXT-INDENT: 5px; LINE-HEIGHT: 150%" align="justify">
<font style="FONT-SIZE: 9pt" face="Verdana">Το Τμήμα Νοσηλευτικής ιδρύθηκε
το 1979 (Π.Δ. 184/79, Π.Δ. 562/80) ως Τμήμα τετραετούς φοίτησης στην Ιατρική
Σχολή του Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών και η λειτουργία
του Τμήματος άρχισε το ακαδημαϊκό έτος 1980-81.</font></p>
<p style="MARGIN: 5px 10px; TEXT-INDENT: 5px; LINE-HEIGHT: 150%" align="justify">
<font style="FONT-SIZE: 9pt" face="Verdana">Από το ακαδημαϊκό έτος 1983-84
το Τμήμα Νοσηλευτικής λειτουργεί ως Τμήμα ισότιμο με τα άλλα Τμήματα του
Πανεπιστημίου σύμφωνα με το Νόμο Πλαίσιο για τα ΑΕΙ (ν. 1268/82), την
2424/83 απόφαση του Συμβουλίου της Επικρατείας, την ΒΙ. 959/22-8-83
Υπουργική απόφαση και την από 29-7-83 απόφαση της Συγκλήτου του Εθνικού και
Καποδιστριακού Πανεπιστημίου Αθηνών.</font></td>
</tr>
</table>
</body>
<p align="center"></p>
<p align="center">&nbsp;</p>
</html>

```

Εικόνα 2: Κώδικας html

Όπως φαίνεται στην προηγούμενη εικόνα η ιστοσελίδα αποτελείται από κείμενο και tags. Μόνο που τα tags δεν εμφανίζονται μαζί με το κείμενο κατά την ανάγνωση της σελίδας από κάποιον browser αλλά τον καθοδηγούν στην εμφάνιση των δεδομένων και την απόδοση της δομής της σελίδας. Τα tags περικλείονται στους χαρακτήρες <,> και τα περισσότερα διακρίνονται σε start tag και stop tag. Τα πρώτα δηλώνουν την αρχή εφαρμογής της οδηγίας ενώ τα δεύτερα το τέλος της ισχύς της.

Μια web σελίδα μοιάζει εμφανισιακά με μία αντίστοιχη word σελίδα όμως έχει μία σημαντική διαφορά. Ενώ στην περίπτωση της word σελίδας συνήθως έχουμε ένα μόνο αρχείο στο οποίο εμπεριέχονται τόσο το κείμενο όσο και τα διάφορα γραφικά αυτού, μια web σελίδα αποτελείται από περισσότερα αρχεία. Το βασικό από αυτά έχει την κατάληξη .htm και εμπεριέχει το κείμενο και τον κώδικα html. Όμως οι εικόνες, τα γραφικά, οι ήχοι και τα video είναι ξεχωριστά αρχεία τα οποία

συνοδεύουν το html αρχείο. Η web σελίδα δημιουργείται δυναμικά από τον web client (browser) χρησιμοποιώντας όλα τα επιμέρους αρχεία που την αποτελούν (.htm, .gif, .mpg κ.αλ.)

Ενώ λοιπόν στην περίπτωση του word μιλάμε για ένα αρχείο, στην περίπτωση του www μιλάμε περισσότερο για έναν φάκελο (folder) με περισσότερα αρχεία.

HTML 5

Η html ξεκίνησε το 1990 σαν SGML DTD το 1992 και επηρεάστηκε από γλώσσες παρουσίασης (postscript). Στην αρχική της έκδοση περιείχε μόνο 20 απλά στοιχεία.








Η έκδοση HTML v4.0 περιέχει 80 περίπου στοιχεία.

Πλέον είμαστε στην **HTML5** που αποτελεί το τελευταίο πρότυπο για την HTML. Είναι ειδικά σχεδιασμένη για να προσφέρει πλούσιο περιεχόμενο χωρίς την ανάγκη για επιπλέον plugins. Η ανάπτυξη της ξεκίνησε το 2004 από μέλη της Apple, της Mozilla Foundation και της Opera software. Το 2008 κυκλοφόρησε η πρώτη έκδοση της με τους ειδικούς να λένε πως πρόκειται για μια εξελισσόμενη γλώσσα που μπορεί να μην ολοκληρωθεί ποτέ.

Ιστορική εξέλιξη της HTML

1991	1994	1996	1997	1998	2000	2002	2005	2009
HTML	HTML 2	CSS 1	HTML 4	CSS 2	XHTML 1	Tableless web	AJAX	HTML 5

Η τρέχουσα έκδοση προσφέρει τα πάντα, από κινούμενα σχέδια με τα γραφικά, μουσική για ταινίες, και μπορεί επίσης να χρησιμοποιηθεί για την κατασκευή πολύπλοκων εφαρμογών web.

HTML 5		<Storage> (Data can be stored on a users computer or mobile device, so web apps work without an internet connection.
		<type> Web pages can have flashier type with more fonts, Shadows, Colors and other effects.
		<Motion> . Objects move on web pages and react to the movement of a cursor.
		<Games> . Interactive games can run with just a web browser without installing other Software for Plug-ins.
		<Audio> . Audio is played without a plug-in. Browser makers have not agreed on formats.
		<3D> . A Technology called WebGL can create interactive 3D effects using a computer's graphics processor
		<Video> . Video can be embedded in a web page without a plug-in. Browser makers have not agreed on formats

HTML5 είναι επίσης cross-platform. Είναι σχεδιασμένη για να λειτουργεί αν χρησιμοποιείτε έναν υπολογιστή ή ένα Tablet, ένα Smartphone, ή Smart TV.

URL (Uniform Resource Locator)

Προκειμένου κάθε ιστοσελίδα να έχει μία μοναδική διεύθυνση μέσω της οποίας μπορεί να ανευρεθεί, υπάρχει ένα σύστημα ονοματολογίας αυτών, το URL (Uniform Resource Locator). Το URL

είναι αρκετό για να εντοπιστεί μία ιστοσελίδα που βρίσκεται σε έναν διακομιστή ιστού (web server) οπουδήποτε στον κόσμο. Συνήθως αποτελείται από 4 μέρη:

- Το πρωτόκολλο που χρησιμοποιείται για την μεταφορά,
- το όνομα του διακομιστή που την περιέχει,
- την τοπική διαδρομή του αρχείου μέσα στον διακομιστή και
- το όνομα της ιστοσελίδας.

Παράδειγμα: <http://www.nurs.uoa.gr/exetaseis/proptixiakoi/index.html>

- **http://**: εδώ φαίνεται ότι πρόκειται για υπηρεσία του web η οποία στηρίζεται στο πρωτόκολλο http. Το internet όπως είπαμε δεν παρέχει μόνο την υπηρεσία web έτσι το σύστημα ονοματολογίας επεκτείνεται πέρα από το web υποστηρίζοντας και τις άλλες υπηρεσίες του internet. Θα μπορούσαμε αντί http://, να είχαμε ftp://, file://, mailto:, gopher://, news:, κ.τ.λ.
- **www.nurs.uoa.gr**: είναι το όνομα του υπολογιστή στον οποίο βρίσκεται η ιστοσελίδα. Κάθε υπολογιστής στο internet έχει μία μοναδική διεύθυνση αποτελούμενη από μία τετράδα αριθμών (πχ: 195.134.100.90). Με το σύστημα DNS αυτοί οι αριθμοί αντιστοιχούνται σε ονόματα όπως το προηγούμενο.
- **/exetaseis/proptixiakoi**: είναι η ιεραρχική διαδρομή του αρχείου μέσα στον διακομιστή. Δεν αρχίζει με το σύμβολο του σκληρού δίσκου (πχ C:\program...) αλλά από τον μητρικό φάκελο ο οποίος έχει οριστεί από την εφαρμογή του web server σαν ο πρώτος ιεραρχικά που βγαίνει στο internet.
- **index.html**: το όνομα της ιστοσελίδας.

Το www κερδίζει οπαδούς με έναν ρυθμό τόσο ταχύ που όμοιος του δεν έχει εμφανιστεί ποτέ άλλοτε στην ανθρώπινη ιστορία. Τα τελευταία χρόνια έχει μεγαλώσει τόσο πολύ που πλέον έχει συμπεριλάβει το μεγαλύτερο ποσοστό της ανθρώπινης γνώσης συνολικά.

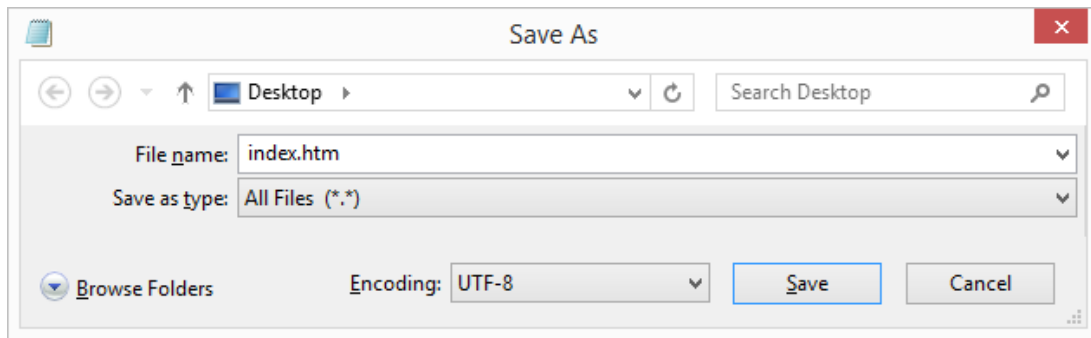
Επειδή ακριβώς το web είναι μία υπηρεσία τόσο δημοφιλής, με την πάροδο του χρόνου και για λόγους ευχρηστίας, οι εφαρμογές της μεριάς του client που υλοποιούν αυτήν την υπηρεσία, δηλαδή οι web browsers, άρχισαν να υποστηρίζουν εκτός του http και τα πρωτόκολλα των άλλων εφαρμογών του internet. Δηλαδή το FTP, NEWS, TELNET, κτλ. Οπότε με την κατοχή ενός web browser μπορεί κανείς να χρησιμοποιήσει και τις άλλες υπηρεσίες του internet χωρίς να χρειάζεται να έχει ξεχωριστή client εφαρμογή για κάθε μία από αυτές.

WYSIWYG HTML Editors

Για την δημιουργία αλλά και την εύκολη διαχείριση των ιστοσελίδων μας επιβάλλεται η χρήση ενός **WYSIWYG** (What You See Is What You Get) **προγράμματος επεξεργασίας ιστοσελίδων**. Υπάρχει πληθώρα τέτοιων προγραμμάτων στο Internet. Με την βοήθεια αυτών των προγραμμάτων μπορούμε πολύ εύκολα και γρήγορα να δημιουργήσουμε μια ιστοσελίδα εισάγοντας μορφοποιημένο κείμενο, λίστες, πίνακες, εικόνες, αρχεία flash και ότι άλλο μπορεί μια ιστοσελίδα να περιέχει, απλά και μόνο με την χρήση του μενού και των εργαλείων που παρέχουν τα προγράμματα αυτά.

Είναι αλήθεια ότι ακόμα και με την χρήση WYSIWYG επεξεργαστών Ιστοσελίδων θα χρειαστεί αρκετές φορές να προβεί κανείς σε αλλαγές στον πηγαίο κώδικα (HTML) λόγω αδυναμίας του προγράμματος να προσαρμόσει ένα στοιχείο που εισάγεται στην σελίδα έτσι ακριβώς όπως θέλετε κανείς να εμφανίζεται.

Για την εκμάθηση της HTML προτείνεται η χρήση ενός απλού text editor όπως το notepad. Γράφουμε τον HTML κώδικα με το Notepad και σώζουμε το αρχείο με κατάληξη “.html ή .htm” (πχ: index.html) και επιλέγουμε κωδικοποίηση (encoding) χαρακτήρων UTF-8.



Εικόνα 3

Με **διπλό κλικ** στο αρχείο που δημιουργήσαμε εκκινείτε κάποιος από τους web Browser, που έχουμε εγκατεστημένους και προβάλλει την ιστοσελίδα διαμορφωμένη.

Αν θελήσουμε να την **τροποποιήσουμε**, δηλαδή να αλλάξουμε τον κώδικα html, κάνουμε δεξί κλικ πάνω στο αρχείο και πατάμε «άνοιγμα με – open with». Από τα προγράμματα που θα μας προτείνει επιλέγουμε τον text editor “Notepad”. Με το Notepad προβάλλεται ο κώδικάς της σελίδας και μπορούμε να του κάνουμε αλλαγές.

Εκτός των εξειδικευμένων προγραμμάτων και πολλές άλλες μη εξειδικευμένες εφαρμογές μπορούν να σώζουν σε μορφή html πχ οι εφαρμογές του office.

HTML Hyper Text Markup Language (Γλώσσα Χαρακτηρισμού Υπερκειμένου).

Η HTML είναι μια γλώσσα χαρακτηρισμού υπερκειμένου με την οποία κατασκευάζουμε ιστοσελίδες. Η χρήση μιας γλώσσας χαρακτηρισμού σημαίνει ότι γράφετε πρώτα το κείμενο και έπειτα προστίθενται ειδικά σύμβολα γύρω από τις λέξεις ή από ολόκληρες προτάσεις ώστε να καθοριστεί η εμφάνιση τους στην οθόνη. Τα ειδικά σύμβολα στην HTML λέγονται **ετικέτες (tags)**. Η HTML διαθέτει ένα πεπερασμένο αριθμό ετικετών που μπορούμε να χρησιμοποιήσουμε. Ωστόσο ο αριθμός αυτός δεν παραμένει σταθερός.

Ετικέτες (tags) ή element (έννοιες)

ετικέτα αρχής <...> ... </...> ετικέτα τέλους

Οι ετικέτες ελέγχουν την δομή και την μορφή του κειμένου της ιστοσελίδας. Επίσης παρέχουν πληροφορίες προς τον Web Browser για την σελίδα που πρόκειται να εμφανίσουν, όπως ο τίτλος της σελίδας ή ο συγγραφέας της, κ.α.

Οι ετικέτες περιέχουν:

- Αρχή – Τέλος
- Ανοίγουν – Κλείνουν. Πχ: **<p> κείμενο </p>**
- Μπορούν να ενσωματωθούν ή μία στην άλλη. Πχ: **<i> κείμενο </i> **

Υπάρχουν ορισμένες ετικέτες που δεν έχουν ετικέτες τέλους. Στις ετικέτες αυτές, πριν από το σύμβολο > τοποθετούμε τον χαρακτήρα / . πχ: **
**

Οι ετικέτες μέσα σε ένα HTML αρχείο μπορούν να είναι γραμμένες είτε με μικρά γράμματα (πεζά), είτε με κεφαλαία. Η ετικέτα **** είναι ίδια με τη ετικέτα ****. Μόνο πρέπει να προσέχουμε τις ετικέτες τέλους, να είναι γραμμένες όπως η ετικέτες αρχής. Δηλαδή αν μια ετικέτα αρχής είναι γραμμένη με πεζά τότε και η ετικέτα τέλους πρέπει να γραφεί με πεζά.

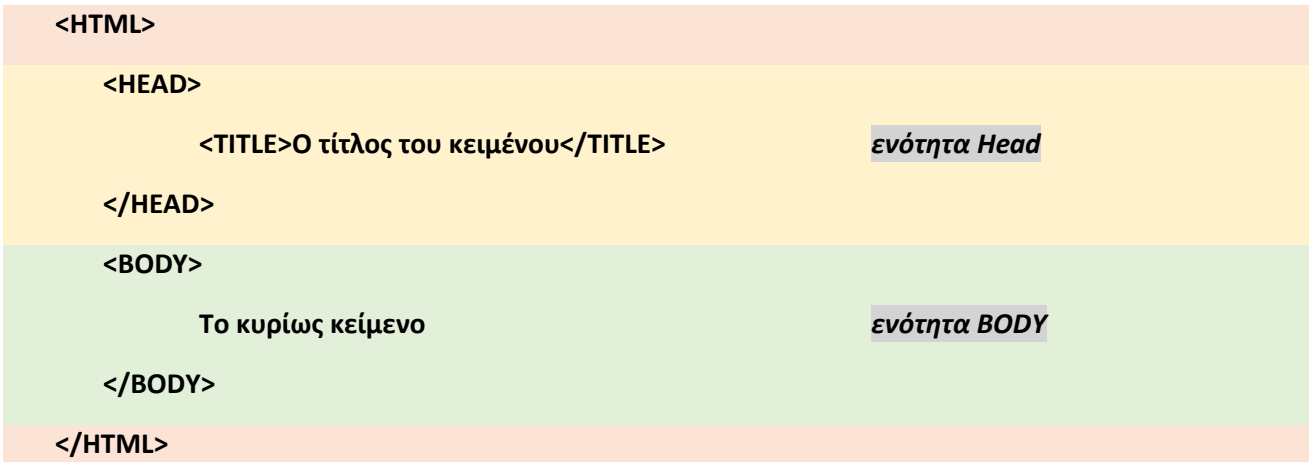
Ο Web Browser δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να παρουσιάσει το περιεχόμενο της σελίδας.

Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες.

Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, VBScript και Jscript τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML και από στατικές τις κάνουν διαδραστικές.

Δομή της HTML

Ένα αρχείο HTML αρχίζει πάντα με την ετικέτα `<html>` και αποτελείται από δύο ενότητες: την **κεφαλή (HEAD)** και το κυρίως **περιεχόμενο (BODY)** ή αλλιώς το "σώμα" της σελίδας όπως συνήθως το αποκαλούμε. Το αρχείο τελειώνει με την ετικέτα `</html>`.



Βασικές ετικέτες html

- `<html ... </html>` Με την ετικέτα `<html>` αρχίζουμε πάντα τον κώδικα μας και με την ετικέτα `</html>` τον τερματίζουμε.
- `<head> ... </head>` ορίζει την πρώτη ενότητα μιας HTML σελίδας. Οι ετικέτες που γράφονται στην ενότητα HEAD, αποτελούν τον πρόλογο για την HTML σελίδα. Ότι γράφουμε μέσα στην ενότητα HEAD δεν εμφανίζεται στην οθόνη του web browser. Υπάρχουν μόνο λίγες ετικέτες που εμπεριέχονται στην ενότητα αυτή. Η πιο βασικές από αυτές είναι η ετικέτα `<title>` και `<meta>`.
- `<body> ... </body>` είναι η δεύτερη ενότητα μιας HTML σελίδας και περιέχει το κυρίως περιεχόμενο της σελίδας μέσα στο οποίο γράφουμε το κείμενο που θέλουμε να εμφανιστεί μαζί με τις HTML ετικέτες που το μορφοποιούν. Στην ενότητα αυτή τοποθετούμε επίσης εικόνες, βίντεο και ότι άλλο θέλουμε να εμφανιστεί στην σελίδα. Περικλείει κάθε τι που τελικά προβάλλεται στον browser (κείμενα, συνδέσμους προς πολυμέσα, hyperlinks)

✚ Οι κύριες ετικέτες tags του HEAD:

- `<title> ... </title>` εμπεριέχεται στην ενότητα `<head>` και καθορίζει τον τίτλο της σελίδας ο οποίος εμφανίζεται στο πάνω μέρος του παραθύρου του web browser.
- `<meta> ... </meta>` εμπεριέχεται στην ενότητα `<head>` και παρέχει πληροφορίες για το αρχείο. Μια από τις λειτουργίες της ετικέτας είναι να ορίζει το σεντ των χαρακτήρων που θα χρησιμοποιήσουμε στην σελίδα.
Πχ για ελληνικά: `<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">`
- `<base>`: εμπεριέχεται στην ενότητα `<head>` και ορίζει την κύρια τοποθεσία του αρχείου και χρησιμεύει ως αναφορά για όλα τα pathnames και links.

- **<script>**: εμπεριέχεται στην ενότητα <head> και προστίθεται αν θέλουμε να χρησιμοποιήσουμε JavaScript, VBScript.
- **<style>**: εμπεριέχεται στην ενότητα <head> για ενσωμάτωση CSS (Cascading Style Sheets).

✚ Οι κύριες ετικέτες tags (Elements) του BODY είναι:

➤ Ετικέτες (tags) θέσης και στοίχισης

- **<p>...</p>** Αλλαγή παραγράφου. Πριν και μετά την παράγραφο εισάγεται αυτόματα στον browser μια κενή γραμμή.
- **<center> ... </center>** Το κείμενο εμφανίζεται με στοίχιση στο κέντρο της οθόνης.
- **<blockquote> ... </blockquote>** το κείμενο αρχίζοντας από δεξιότερη εσοχή απ' ότι το υπόλοιπο κείμενο.

➤ Ετικέτες (tags) διαμόρφωσης κειμένου

- **...** ** ... ** κείμενο με έντονη μορφή
- **<i>...</i>** κείμενο με πλάγιους χαρακτήρες
- ** ... ** κείμενο με πλάγιους και κάπως αχνά γραμμένους χαρακτήρες
- **<u> ... </u>** κείμενο υπογραμμισμένο
- **<big>...</big>** κείμενο με έντονη μορφή και ελαφρώς μεγαλύτερα γράμματα.
- **<small>...</small>** κείμενο μικρότερο σε μέγεθος από το κανονικό.
- **_{...}** κείμενο σαν μαθηματική βάση, δηλαδή σε χαμηλότερο σημείο απ' ότι βρίσκεται το υπόλοιπο κείμενο.
- ** ... ** Διαγραμμένο κείμενο
- **<mark> ... </mark>** κείμενο με χρώμα επισήμανσης
- **...** Καθορίζει το μέγεθος και το χρώμα του κειμένου.

➤ Διάφορες άλλες ετικέτες

- **<h>...</h>** Επικεφαλίδες. Το κείμενο που εμπεριέχουν εμφανίζετε με μεγάλα και έντονα γράμματα. Με την <h1> ορίζουμε την μεγαλύτερη ετικέτα ενώ με την <h6> την μικρότερη.
 - <h1>...</h1> ,
 - <h2>...</h2> ,
 - <h3>...</h3> μέχρι <h6>...</h6>
- **
** Αλλαγή γραμμής. Δεν χρειάζεται ετικέτα τερματισμού.
- **<hr>** Οριζόντια γραμμή. Η οριζόντια γραμμή είναι ένα απλό γραφικό που μπορούμε να χρησιμοποιήσουμε στην σελίδα μας (κυρίως σαν διαχωριστικό).
- **<!--σχόλια -->** Σχόλια
- **<a>...** Ορίζει hyperlink. Το κείμενο μεταξύ <a> και κρύβει ένα link που μεταφέρει τον χρήστη σε μια νέα σελίδα ή σε άλλο σημείο της ίδιας σελίδας. Χρησιμοποιείτε και με γραφικά.
- **<map>...</map>** Εισαγωγή χάρτη συντεταγμένων. Ο χάρτης περιγράφει κάθε τμήμα της εικόνας και δηλώνει ποια σελίδα πρέπει να κληθεί με κλικ σε καθένα από αυτά.
- **** Χρησιμοποιείτε για την εισαγωγή γραφικών σε html σελίδες.

- `...`, `...` Φτιάχνουν bulleting και αριθμητικές λίστες.
- `<table>...</table>` Εισαγωγή πίνακα.
- `<form>...</form>` Παρουσίαση φορμών με σκοπό την εισαγωγή δεδομένων από τον χρήστη.
- `<frameset>...</frameset>` Τα frames επιτρέπουν τον διαχωρισμό ενός παραθύρου web σε πολλαπλά τμήματα.
- **Κ.α.**

Παράδειγμα 1

```

<html>
  <head>
    <TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body>
    <h1> Δοκιμαστική Ιστοσελίδα </h1>
    <hr>
    Αυτή είναι η ιστοσελίδα του <b> Σπύρου Ζώγα </b>
    Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"
    <p> Στα πλαίσια του μαθήματος <i>Αρχές Πληροφορικής.</i> </p>
  <hr>
</body>
</html>

```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του **Σπύρου Ζώγα** Μεταπτυχιακού Φοιτητή
"Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής.*

Χαρακτηριστικά (Attributes) ετικετών (tags), Ιδιότητες (Properties) χαρακτηριστικών και Τιμές (Values)

Οι περισσότερες ετικέτες (tags) συνοδεύονται από κάποια **χαρακτηριστικά (Attribute)** τα οποία μπαίνουν μέσα στο start tag και του δίνουν μεγαλύτερη παραμετροποίηση, δηλαδή διαφοροποιούν ή επεκτείνουν την ετικέτα. Ένα χαρακτηριστικό (attribute) μπαίνει αμέσως μετά το όνομα της ετικέτας αρχής και αποτελείται από το όνομα του και μια **τιμή** μέσα σε διπλά εισαγωγικά.

`<tag attribute1="value" attribute2="value" > κείμενο </tag>`

πχ: Κόκκινη γραμματοσειρά

Συνήθως ο όρος **ιδιότητα** χρησιμοποιείται ως συνώνυμος του όρου χαρακτηριστικό. Κάτι που δεν είναι ακριβές. Τα χαρακτηριστικά (attributes) περιέχουν διάφορες **CSS ιδιότητες (properties)** που παίρνουν **CSS τιμές (values)**

`<tag attribute="property:value;">`

*πχ: *

✚ Τα κύρια χαρακτηριστικά (attributes) της ετικέτας (tag) <HTML>

- **Lang** Καθορίζει την γλώσσα του κειμένου της ιστοσελίδας.

Πχ: <html lang="en-US">

✚ Τα κύρια χαρακτηριστικά (attributes) της ετικέτας (tag) <body>

- bgcolor="color"
- text="color"
- link="color"
- vlink="color"
- alink="color"
- background="background.gif"
- marginwidth=0, marginheight=0, leftmargin=0, topmargin=0

πχ: <body bgcolor="blue" text="black" marginwidth=2px>

✚ Τα βασικότερο attribute του element <h>:

- **Style:** καθορίζει χρώμα γραμματοσειρά, μέγεθος κ.α

πχ: <h1 style="font-size:60px">heading</h1>

- Το font-size είναι: CSS ιδιότητα
- **Align:** Στοίχιση, τιμές: left, center, right, justify

πχ: <h1 align="right">κείμενο</h1>

✚ Τα βασικότερα χαρακτηριστικά (attribute) της ετικέτας (tag) <p>:

- **Style:** καθορίζει χρώμα γραμματοσειρά, μέγεθος κ.α.
πχ: `<p style="color:red">I am a paragraph</p>`
- **Title:** εμφανίζει πτυσσόμενο μήνυμα όταν το ποντίκι περνά πάνω από την παράγραφο.
Πχ: `<p title="tooltip">This is a paragraph.</p>`
- **Align:** τιμές: left, center, right, justify

Παράδειγμα 2

```
<html>
  <head>
    <TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body bgcolor="silver" text="blue">
    <h1 style="color:red"> Δοκιμαστική Ιστοσελίδα </h1>
    <hr>
    Αυτή είναι η ιστοσελίδα του <b style="color:green; font-size:20px "> Σπύρου Ζώγα </b>
    Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"
    <p align="center"> Στα πλαίσια του μαθήματος <i > <b>Αρχές Πληροφορικής.</b></i> </p>
    <hr>
  </body>
</html>
```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του **Σπύρου Ζώγα** Μεταπτυχιακού Φοιτητή
"Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής.*

Υπερσύνδεσμος - HTML links

Με την ετικέτα `<a>` δημιουργούμε έναν σύνδεσμο (link) προς μια άλλη σελίδα χρησιμοποιώντας το χαρακτηριστικό `href`. ` κείμενο `

Παράδειγμα: `This is a link`

✚ Τα βασικότερα χαρακτηριστικά (attribute) της ετικέτας (tag) `<a>`:

- `href` είναι το βασικότερο attribute του element `<a>`
- **Download:** ``
- **Target:** `Νοσηλευτική`
Ορίζει σε ποιο παράθυρο ή μέρος του παραθύρου θα εμφανιστεί η σελίδα που ανοίγουμε πατώντας τον σύνδεσμο. Τιμές: `_blank`, `_self`, `_parent`, `_top`, `framename`.

✚ Ετικέτα `<name>`

Μια παραπλήσια δυνατότητα με την δημιουργία υπερσυνδέσμου είναι η δημιουργία παραγράφων. Μπορούμε να χωρίσουμε το κείμενο μας σε παραγράφους χρησιμοποιώντας την ετικέτα `<name>`

`κείμενο`

Για να προσθέσουμε link προς την παράγραφο αυτή γράφουμε σε κάποιο άλλο σημείο του κειμένου: `Πήγαινε στην πρώτη παράγραφο.`

Παράδειγμα 3

```
<html>
  <head>
    <TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body bgcolor="silver" text="blue">
    <h1 style="color:red" align="center"> Δοκιμαστική Ιστοσελίδα </h1>
    <hr>
    Αυτή είναι η ιστοσελίδα του <b style="color:green; font-size:20px "> Σπύρου Ζώγα </b>
    Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"
    <p> Στα πλαίσια του μαθήματος <i > <b>Αρχές Πληροφορικής.</b></i> </p>
    <hr>
    Οργάνωση Μεταπτυχιακού: <a href="http://www.nurs.uoa.gr" target="blank"> Τμήμα
    Νοσηλευτικής</a>
  </body>
</html>
```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του **Σπύρου Ζώγα** Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής*.

Οργάνωση Μεταπτυχιακού: [Τμήμα Νοσηλευτικής](#)

Εικόνα

Με την ετικέτα εμφανίζουμε μια εικόνα μέσα στην ιστοσελίδα μας χρησιμοποιώντας το χαρακτηριστικό src.

```

```

✚ Τα βασικότερο attribute του tag :

- **src**, Υποχρεωτικό χαρακτηριστικό. Καθορίζει την διεύθυνση του αρχείου εικόνας

- **alt**, Προσδιορίζει το κείμενο που θα πρέπει να εμφανιστεί σε browsers που δεν υποστηρίζουν γραφικά.

- **width, height** Καθορίζουν το πλάτος και το ύψος της εικόνας σε pixel.

- **align**: Ορίζει την στοίχιση της εικόνας με το κείμενο που βρίσκετε δίπλα της.
Τιμές: left, right, middle, top, bottom;

- **Border**: πάχος περιγράμματος εικόνας κ.α

Παράδειγμα 4

```
<html>
  <head>
    <TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
    <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
  </head>
  <body bgcolor="silver" text="blue">
    <h1 style="color:red" align="center"> Δοκιμαστική Ιστοσελίδα </h1>
    <hr>
    Αυτή είναι η ιστοσελίδα του <b style="color:green; font-size:20px "> <u> Σπύρου Ζώγα </u>
    </b>
    Μεταπτυχιακού Φοιτητή <mark>"Πληροφορικής Υγείας"</mark>
    <p> Στα πλαίσια του μαθήματος <i > <b>Αρχές Πληροφορικής.</b></i> </p>
    <hr>
    <p align="center"> Οργάνωση Μεταπτυχιακού: <a href="http://www.nurs.uoa.gr"
    target="blank"> Τμήμα Νοσηλευτικής</a></p>
    <center>  </center>

  </body>
</html>
```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του Σπύρου Ζώγα Μεταπτυχιακού Φοιτητή
"Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής*.

Οργάνωση Μεταπτυχιακού: Τμήμα Νοσηλευτικής



Λίστες

✚ Λίστα με bullets

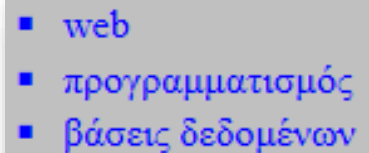
Η ετικέτα `` εισάγει μια μη αριθμημένη λίστα στην σελίδα μας. Το ul είναι τα δύο πρώτα γράμματα από το Unordered List.

Η ετικέτα `` προσθέτει γραμμές στην λίστα.

Το **χαρακτηριστικό type** της ετικέτας `` ορίζει το σύμβολο που μπαίνει μπροστά από κάθε γραμμή της λίστας. Οι τιμές που παίρνει η ιδιότητα type είναι οι εξής: **disc, circle, square**

Η ετικέτα `<dir>` κάνει την ίδια δουλειά με την ετικέτα ``

- `<ul type="square">`
 - `web`
 - `προγραμματισμός`
 - `βάσεις δεδομένων`
- ``



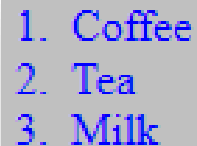
```


- web
- προγραμματισμός
- βάσεις δεδομένων

```

✚ Αριθμημένη Λίστα

- ``
 - `Coffee`
 - `Tea`
 - `Milk`
- ``



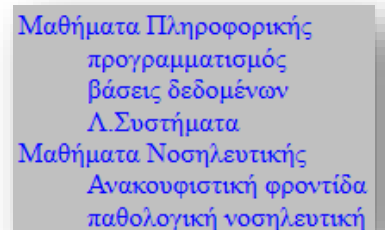
```


1. Coffee
2. Tea
3. Milk

```

✚ Λίστα Ορισμών

- `<dl>`
 - `<dt>Μαθήματα Πληροφορικής</dt>`
 - `<dd>προγραμματισμός</dd>`
 - `<dd>βάσεις δεδομένων</dd>`
 - `<dd>λειτουργικά συστήματα</dd>`
 - `<dt>Μαθήματα Νοσηλευτικής</dt>`
 - `<dd>Ανακουφιστική φροντίδα</dd>`
 - `<dd>Παθολογική Νοσηλευτική</dd>`
- `</dl>`



```


- Μαθήματα Πληροφορικής
- προγραμματισμός
- βάσεις δεδομένων
- Λ.Συστήματα
- Μαθήματα Νοσηλευτικής
- Ανακουφιστική φροντίδα
- παθολογική νοσηλευτική


```

Πίνακες

Με την ετικέτα `<table>` ορίζουμε έναν πίνακα. Το ζεύγος των ετικετών `<tr>...</tr>` ορίζει μια γραμμή του πίνακα, ενώ το ζεύγος των ετικετών `<td>...</td>` ορίζουν ένα κελί στην γραμμή του πίνακα.

- `<table>`
 - `<tr>`
 - `<td>` γραμμή 1, στήλη 1 `</td>`
 - `<td>` γραμμή 1, στήλη 2 `</td>`
 - `</tr>`
 - `<tr>`
 - `<td>` γραμμή 2, στήλη 1 `</td>`
 - `<td>` γραμμή 2, στήλη 2 `</td>`
 - `</tr>`
- `</table>`

γραμμή 1, στήλη 1 γραμμή 1, στήλη 2
 γραμμή 2, στήλη 1 γραμμή 2, στήλη 2

Για να προσθέσω μέγεθος πλαισίου (border), τίτλο (caption) και επικεφαλίδες (th) στον πίνακα:

- `<table border="1">`
- `<caption>`Τίτλος του Πίνακα`</caption>`
 - `<tr>`
 - `<th>`Επικεφαλίδα 1`</th>`
 - `<th>`Επικεφαλίδα 2`</th>`
 - `</tr>`
 - `<tr>`
 - `<td>`γραμμή 1, στήλη 1`</td>`
 - `<td>`γραμμή 1, στήλη 2`</td>`
 - `</tr>`
 - `<tr>`
 - `<td>`γραμμή 2, στήλη 1`</td>`
 - `<td>`γραμμή 2, στήλη 2`</td>`
 - `</tr>`
- `</table>`

Παράδειγμα 5

```
<table border="2">
<caption>Πρόγραμμα</caption>
  <tr>
    <th> Δευτέρα </th>
    <th> Τρίτη</th>
  </tr>
  <tr>
    <td>Σχολή</td>
    <td>Νοσοκομείο</td>
  </tr>
  <tr>
    <td>Αργία</td>
    <td>Σχολή</td>
  </tr>
```

Δευτέρα	Τρίτη
Σχολή	Νοσοκομείο
Αργία	Σχολή

✚ Χαρακτηριστικά (Attributes) πινάκων

- Το χαρακτηριστικό **border** της ετικέτας `<table>` ορίζει το πάχος του περιγράμματος του πίνακα.
- μήκος (**width**) και ύψος (**height**) ενός πίνακα
- Στοίχιση του πίνακα μέσα στην σελίδα(**align**): `<table border="1" align="center">`
- Στοίχιση κειμένου μέσα στα κελιά: `<td align="left"> Νοσοκομείο </td>`
Τιμές για στοίχιση: **left, center και right**
- Χρώμα σε όλο τον πίνακα, στην γραμμή, στο κελί: **Bgcolor**
 - `<table border="1" bgcolor="red">`
 - `<tr bgcolor="red">`
 - `<td bgcolor="red">Σχολή</td>`
- Εικόνα σαν φόντο σε όλο τον πίνακα, στην γραμμή, στο κελί (**Background**):
 - `<table border="1" background="bird.jpg">`
 - `<tr background="bird.jpg">`
 - `<td background="bird.jpg">Σχολή</td>`
- **Cellspacing**: ορίζουμε το πάχος των σκιασμένων γραμμών που διαχωρίζουν τα κελιά.
- **Cellpadding**: ορίζουμε τον κενό χώρο μεταξύ του πλαισίου των κελιών και του περιεχομένου αυτών. `<table width="264" border="1" cellspacing="0" cellpadding="0">`
- **Rowspan**: Κάθετη συγχώνευση κελιών, ορίζει τον αριθμό των γραμμών που θα επεκτείνεται το κελί `<td rowspan="2" bgcolor="#66CCFF">`

Παράδειγμα 6

```
<table border="1" align="center" bgcolor="white">
<caption>Πρόγραμμα</caption>
```

```
<tr bgcolor="yellow" align="center">
  <th> Δευτέρα </th>
  <th> Τρίτη</th>
</tr>
<tr>
  <td>Σχολή</td>
  <td>Νοσοκομείο</td>
</tr>
<tr>
  <td>Αργία</td>
  <td bgcolor="red">Σχολή</td>
</tr>
```

Πρόγραμμα	
Δευτέρα	Τρίτη
Σχολή	Νοσοκομείο
Αργία	Σχολή

Φόρμες <form> και </form>

Αν και ο κύριος σκοπός μιας ιστοσελίδας είναι η δημοσίευση πληροφοριών στο Internet όλο και περισσότερα websites χρησιμοποιούν την αμφίδρομη επικοινωνία, δηλαδή δεν παρέχουν απλά πληροφορίες προς τους επισκέπτες, αλλά ζητούν από αυτούς να επιλέγουν ή να πληκτρολογούν στοιχεία τα οποία προωθούντε στον WebServer της σελίδας. Αυτό επιτυγχάνεται με την δυνατότητα δημιουργίας φορμών της HTML.

Με το πάτημα ενός κουμπιού της φόρμας τα δεδομένα που συμπλήρωσε ο επισκέπτης στέλνονται στον Server μέσω του πρωτοκόλλου HTTP. Έπειτα ο Server επεξεργάζεται με κάποιο πρόγραμμα script τις τιμές των πεδίων και επιστρέφει στον web browser του επισκέπτη μια νέα HTML σελίδα. Τα περιεχόμενα της σελίδας αυτής μπορεί να είναι οτιδήποτε, από μια απλή απάντηση μέχρι επιστροφή αποτελέσματος σε ερώτηση προς μια περίπλοκη Βάση Δεδομένων.

Το script είναι ένα πρόγραμμα το οποίο τρέχει στον Server και το οποίο μεταξύ των άλλων μπορεί να δέχεται σαν είσοδο δεδομένα τα οποία λαμβάνει ο Server από τον web browser του επισκέπτη (συνήθως από μια Φόρμα). Το script , χρησιμοποιώντας μια script γλώσσα προγραμματισμού (π.χ. asp , php κτλ.) , επεξεργάζεται τα δεδομένα αυτά και έπειτα επιστρέφει στον browser μια HTML σελίδα. Η επεξεργασία αυτή μπορεί να είναι από απλές πράξεις μεταξύ των δεδομένων έως και αναζήτηση σε μια περίπλοκη Βάση Δεδομένων με βάση τα δεδομένα αυτά (ένα τέτοιο script περιέχει κατάλληλες εντολές ώστε να συνδεθεί σε μια Βάση Δεδομένων και να προσπελάσει αλλά και να προσθέσει και να διαγράψει εγγραφές).

Για να δημιουργήσουμε μια φόρμα χρησιμοποιούμε τις ετικέτες <form> και </form>.

✚ Τα κυριότερα χαρακτηριστικά (attributes) της ετικέτας <form> είναι:

- **Name:** Με το χαρακτηριστικό (attribute) **name** δίνουμε ένα όνομα στην φόρμα ώστε να μπορούμε να αναφερόμαστε σε αυτή μέσα από τον κώδικα ενός script. Καλό είναι πάντα να δίνουμε όνομα στις Φόρμες που κατασκευάζουμε, ιδιαίτερα στις περιπτώσεις που έχουμε περισσότερες από μια Φόρμες στην ίδια σελίδα.
- **Action:** Το χαρακτηριστικό (attribute) **action** περιέχει το URL του αρχείου script στον Server το οποίο θα επεξεργαστεί τα στοιχεία της Φόρμας.
- **Method:** Το χαρακτηριστικό (attribute) **method** καθορίζει τον τρόπο με τον οποίο στέλνονται τα δεδομένα της Φόρμας στον Server που βρίσκεται το πρόγραμμα script που θα τα επεξεργαστεί και μπορεί να πάρει τις τιμές **get** ή **post**.
 - Με την μέθοδο **get** τα δεδομένα προσθέτονται στο τέλος του URL που "δείχνει" η ιδιότητα action και χωρίζονται από το σύμβολο &. Η GET έχει κάποιους περιορισμούς και σε μεγάλες ποσότητες δεδομένων, μπορεί κάποια από αυτά να χαθούν. Επίσης, έχει αποδειχθεί ιδιαίτερα ευάλωτη σε επιθέσεις hackers.
 - Με την μέθοδο **post** τα δεδομένα στέλνονται ξεχωριστά από το URL. Στην περίπτωση post το αρχείο script παίρνει τα δεδομένα της Φόρμας μέσω της στάνταρ εισόδου.

Μέσα στα όρια των ετικετών <form> και </form> εκτός από τα στοιχεία της Φόρμας μπορούμε επίσης να προσθέσουμε κείμενο (συνήθως Λεζάντες των στοιχείων) μαζί με ετικέτες μορφοποίησης.

Όλα τα στοιχεία μιας Φόρμας, είτε αυτά είναι πεδία κειμένου, είτε είναι περιοχές κειμένου, είτε είναι κουμπιά επιλογών καλό είναι να έχουν μία **λεζάντα**.

Παράδειγμα 7

```
<form name="form1" method="post" action="http://www.wlearn.gr/script.php">
  Ηλικία: <input type="text" name="age" value="0" />
</form>
```



Άλλες ετικέτες φόρμών

✚ Η ετικέτα <input>

Με την ετικέτα <input> εισάγουμε τα περισσότερα στοιχεία της φόρμας.

- Πεδίο Κειμένου (**text**)
- κωδικός (**password**)
- Κουμπί Επιλογών (**radio**)
- Κουτί Πολλαπλών Επιλογών (**checkbox**)
- Κουμπί υποβολής φόρμας (**submit**)
- Καθαρισμός φόρμας (**button**)

Η ετικέτα <input> δεν έχει ετικέτα τέλους.

➤ Τα κυριότερα *attributes* της ετικέτας αυτής είναι:

- **type**: καθορίζει τον τύπο του στοιχείου της Φόρμας, (πεδίο Κειμένου - **text**, κωδικός - **password**, κουμπί Επιλογών - **radio**, κουτί Πολλαπλών Επιλογών - **checkbox**, κουμπί υποβολής φόρμας - **submit**, καθαρισμός φόρμας - **button**).
- **name**: δίνουμε ένα μοναδικό όνομα στο στοιχείο της φόρμας (δεν πρέπει να υπάρχουν στοιχεία φόρμας με τα ίδια ονόματα)
- **value**: δίνουμε μια αρχική τιμή στο στοιχείο της φόρμας. Στην περίπτωση που πρόκειται για κουμπί επιλογών η value ορίζει την τιμή που θα στείλει η φόρμα στον Server αφού πατήσουμε το κουμπί αποστολής.
- **Size**: καθορίζει τον αριθμό των ορατών χαρακτήρων που μπορεί να χωρέσει το πεδίο κειμένου.
- Η **maxlength** καθορίζει τον αριθμό χαρακτήρων που μπορεί να πληκτρολογήσει ο επισκέπτης στην περίπτωση που έχουμε πεδίο κειμένου.
- Η **checked** χρησιμοποιείται για να ορίσουμε ποια από τα ομαδοποιημένα κουμπιά θα είναι εξορισμού επιλεγμένο όταν φορτώνεται η σελίδα. Η ιδιότητα αυτή δεν παίρνει τιμή.

Παράδειγμα 8

```

<form name="form1" method="post" action="http://www.wlearn.gr/script.php">

    <em>Όνομα:</em>
    <input type="text" name="pedio_keimenou" value="Πληκτρολογήστε το όνομα σας εδώ" size="35"
    maxlength="30" />
    <br>
    <hr>
    Password: <input type="password" name="kodikos" size="20" maxlength="20" />
    <br />
    <hr>
    Φύλο: <input type="radio" name="sex" value="male" checked="checked" > Ανδρας
          <input type="radio" name="sex" value="female" /> Γυναίκα
    <br />
    <hr>
    <input type="checkbox" name="sports" value="1" /> Αθλητισμός<br />
    <input type="checkbox" name="tv" value="1" /> Τηλεόραση<br />
    <input type="checkbox" name="travel" value="1" /> Ταξίδια<br />

    <center>
    <input type="Submit" name="Submit" value="Αποστολή" />
    </center>
</form>

```

Όνομα:

Password:

Φύλο: Ανδρας Γυναίκα

Αθλητισμός
 Τηλεόραση
 Ταξίδια

✚ Η ετικέτα < textarea >

- Με την ετικέτα < **textarea** > εισάγουμε **περιοχή Κειμένου** στην Φόρμα μας.

➤ Τα κυριότερα χαρακτηριστικά (attributes) της ετικέτας είναι:

- Η ιδιότητα **cols** ορίζει τον αριθμό των ορατών χαρακτήρων (αυτοί δηλαδή που φαίνονται χωρίς οριζόντιο scrolling) που θα χωράει κάθε γραμμή στην Περιοχή Κειμένου,
- η ιδιότητα **rows** ορίζει τον αριθμό των ορατών γραμμών (αυτές δηλαδή που φαίνονται χωρίς κάθετο scrolling) που θα έχει η Περιοχή Κειμένου.

Η χρήση και των δύο αυτών ετικετών είναι απαραίτητη. Όταν ο επισκέπτης πατάει το Κουμπί Υποβολής της Φόρμας αυτό που θα σταλεί στον Server είναι ολόκληρο το κείμενο που πληκτρολόγησε στη Περιοχή κειμένου.

Πχ: < **textarea** **name**="textarea2" **cols**="30" **rows**="3"></ **textarea** >

✚ Η ετικέτα < select >

- Με την ετικέτα < **select** > εισάγουμε **Λίστα Επιλογών** στην Φόρμα μας.
Η ετικέτα < **select** > έχει το χαρακτηριστικό **name**
- Η ετικέτα < **select** > εμπεριέχει την ετικέτα < **option** > που ορίζει τις διάφορες επιλογές της λίστας. Όσες επιλογές επιθυμούμε να περιλαμβάνει η Λίστα Επιλογών τόσες ετικέτες < **option** > πρέπει να προσθέσουμε. Μεταξύ των ετικετών < **option** > και < /**option** > γράφουμε το κείμενο που θέλουμε να εμφανίζεται στην λίστα.
- Η ετικέτα < **option** > έχει το χαρακτηριστικό (attribute) **value** μέσω του οποίου ορίζουμε την τιμή που θα σταλεί στον Server.

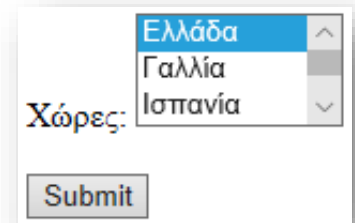
Παράδειγμα 9

```
<form name="form1" method="post" action="example5.php">
  <select name="countries">
    <option value="GR" selected >Ελλάδα</option>
    <option value="FR">Γαλλία</option>
    <option value="ES">Ισπανία</option>
    <option value="GE">Γερμανία</option>
    <option value="GE">Ιταλία</option>
    <option value="GE">Ολλανδία</option>
    <option value="GE">Πορτογαλλία</option>
    <option value="GE">Αγγλία</option>
  </select>
</form>
```

- Η χρήση της ιδιότητας **size** της ετικέτας **<select>** εμφανίζει την Λίστα Επιλογών σαν ένα **πλαίσιο επιλογών** με οριζόντια μπάρα ολίσθησης. Η ιδιότητα **size** παίρνει σαν τιμή της έναν αριθμό ο οποίος ορίζει τον αριθμό γραμμών που θα εμφανίζονται στο πλαίσιο επιλογών.

Παράδειγμα 10

```
<form name="form1" method="post" action="example5.php">
  <select name="countries" size="3">
    <option value="GR" selected="selected" >Ελλάδα</option>
    <option value="FR">Γαλλία</option>
    <option value="ES">Ισπανία</option>
    <option value="GE">Γερμανία</option>
    <option value="GE">Ιταλία</option>
    <option value="GE">Ολλανδία</option>
    <option value="GE">Πορτογαλλία</option>
    <option value="GE">Αγγλία</option>
  </select>
</form>
```



Σε ορισμένες περιπτώσεις επιθυμούμε ο επισκέπτης να έχει την δυνατότητα να **επιλέξει παραπάνω από μια επιλογή**. Για να μετατρέψουμε το προηγούμενο πλαίσιο επιλογών σε πλαίσιο πολλαπλών επιλογών απλά προσθέτουμε το χαρακτηριστικό (attribute) **multiple** στην ετικέτα **<select>**.

Το χαρακτηριστικό **multiple** δεν παίρνει τιμές. Η πολλαπλή επιλογή επιτυγχάνεται κρατώντας πατημένο το **ctrl**.

Κρυφά πεδία της Φόρμας

Για να εισάγουμε Κρυφά Πεδία στην φόρμα μας χρησιμοποιούμε την ετικέτα **<input>** ορίζοντας την τιμή **hidden** στο χαρακτηριστικό **type**.

Τα κρυφά πεδία δεν εμφανίζονται στον browser. Χρησιμοποιούμε κρυφά πεδία όταν θέλουμε να περάσουμε στο αρχείο script μια τιμή μαζί με τις υπόλοιπες τιμές που πληκτρολογεί ή επιλέγει ο επισκέπτης.

Αποστολή στοιχείων στον Server (Μέθοδος GET και POST)

Η αποστολή των τιμών από την φόρμα του browser προς το αρχείο script του Server που θα τα επεξεργαστεί, γίνεται με βάση το attribute **method** (μέθοδος) της ετικέτας **<form>** το οποίο μπορεί να πάρει τις τιμές **get** και **post**.

Στην περίπτωση της μεθόδου **GET** κάθε όνομα ενός στοιχείου της φόρμας συνοδεύεται με την τιμή που πληκτρολόγησε ή επέλεξε ο χρήστης και αυτό το "πακέτο" (όνομα στοιχείου-τιμή στοιχείου) προστίθεται στο τέλος του URL που δείχνει η ιδιότητα action της ετικέτας <form>.

Παράδειγμα 11

```
<form name="forma1" method="get" action="http://www.nurs.uoa.gr/script.php">
  Όνομα: <input type="text" name="onoma" /> </br>
  Επώνυμο:<input type="text" name="eponimo" /></br>
  <center><input type="Submit" name="Submit" value="Αποστολή" /> </center>
</form>
```

Όνομα:

Επώνυμο:

Στο παράδειγμα 11 αν ο χρήστης πληκτρολογήσει όνομα «Μαρία» και επώνυμο «Λιάσκου» με το πάτημα του κουμπιού αποστολή στο URL που δείχνει το χαρακτηριστικό action, θα προσδεθούν τα δεδομένα αυτά και το URL που θα σταλεί στο web Server θα έχει την μορφή:

<http://www.nurs.uoa.gr/script.php?onoma=Μαρία&eponimo=Λιάσκου&Submit=Αποστολή>

Ο Server αποθηκεύει όλη την συμβολοσειρά του URL μετά το λατινικό ερωτηματικό (?) στην μεταβλητή περιβάλλοντος QUERY_STRING. Έπειτα η συμβολοσειρά αυτή περνάει μέσα στο script το οποίο με προγραμματιστικό κώδικα διαχωρίζει τα πακέτα τιμών (**όνομα_στοιχείο = τιμή**) μεταξύ τους.

Στην περίπτωση της μεθόδου post τα δεδομένα στέλνονται στον server ξεχωριστά από το URL. Ο Server δέχεται τα δεδομένα αυτά και τα αποθηκεύει σε ένα προσωρινό αρχείο και έπειτα τα περνάει μέσα στο script για να τα επεξεργαστεί.

Η μέθοδος get δεν συνιστάται από το τελευταίο επίσημο πρότυπο της HTML.

CSS Cascading Style Sheets

(στυλ που μπορούμε να ορίσουμε για τις HTML σελίδες)

Γράφοντας τις σελίδες μας μόνο με HTML κώδικα, μπορούμε να ορίσουμε το χρώμα και το μέγεθος του κειμένου αλλά και άλλων στοιχείων της σελίδας (όπως πίνακες, links, λίστες κτλ). Για να αλλάξουμε το χρώμα κάποιου κειμένου ή το χρώμα ενός πίνακα, θα πρέπει να βρούμε το χρώμα αυτό μέσα στον κώδικα και να το αλλάξουμε. Η διαδικασία αυτή μπορεί να φαντάζει εύκολη όταν έχουμε να διαχειριστούμε μια μόνο σελίδα. Όμως ένα website αποτελείται από δεκάδες σελίδες τις οποίες χρειάζεται να διαχειριζόμαστε εύκολα και γρήγορα.

Φανταστείτε, για παράδειγμα, πόσο χρονοβόρο θα είναι αν θελήσουμε κάποια στιγμή να αλλάξουμε τα χρώματα στο κύριο μενού του site μας, το οποίο επαναλαμβάνεται σε όλες τις σελίδες. Σε μια τέτοια περίπτωση θα χρειαζόταν να ανοίγουμε κάθε σελίδα του site και να αλλάζουμε τα χρώματα του φόντου και των links του μενού, διαδικασία που εκτός από χρονοβόρα είναι και κουραστική.

Ένα φύλο στυλ είναι μία ομάδα κανόνων που ορίζονται συνήθως σε ξεχωριστό αρχείο με επέκταση .cs. Οι κανόνες αυτοί μπορούν να καθορίζουν τα πάντα, από τις γραμματοσειρές για τα διάφορα στοιχεία της html, μέχρι τα χρώματα του κειμένου, του φόντου και των γραφικών, τα περιθώρια, τα διαστήματα, το στυλ γραφής του κειμένου και πολλά άλλα. Οτιδήποτε αφορά την εμφάνιση του εγγράφου.

Κατόπιν μπορούμε να βάλουμε σε ένα έγγραφο html μία παραπομπή που να λέει στον browser ποιο φύλο στυλ να χρησιμοποιήσει για να την μορφοποιήσει όλη ή ένα μέρος της. Επίσης μπορούμε να το ορίσουμε έτσι ώστε να επιλέγεται το κατάλληλο φύλλο στυλ ανάλογα με τον χρήστη.

Εκτός από την ευκολία στην διαχείριση ενός site, ένα άλλο σημαντικό πλεονέκτημα της χρήσης CSS στις σελίδες είναι ο "καθαρότερος" κώδικας, χωρίς πολλές ιδιότητες στις ετικέτες οι οποίες τον κάνουν δυσανάγνωστο. Επιπλέον κάνει γρηγορότερη την πλοήγηση καθώς το αρχείο, μέσα στο οποίο ορίζονται τα στυλ, "διαβάζεται" από τον browser μόνο μια φορά και έπειτα αποθηκεύεται στην cache memory, μειώνοντας έτσι το μέγεθος της πληροφορίας που γίνεται download από τους browsers.

Το CSS είναι μια απλή γλώσσα που μας βοηθάει να ορίσουμε με σαφήνεια και ιδιαίτερη ευελιξία τον τρόπο με τον οποίο θα εμφανίζονται τα διάφορα στοιχεία στην ιστοσελίδα μας.

Ποια πλεονεκτήματα έχει η χρήση CSS έναντι της μορφοποίησης μέσω HTML attributes?

- Πολύ μεγαλύτερη ευελιξία. Το CSS κατέστησε εφικτές μορφοποιήσεις οι οποίες ήταν αδύνατες ή πολύ δύσκολες με την κλασική HTML. Ευκολότερη συντήρηση των ιστοσελίδων. Η εμφάνιση ενός ολόκληρου site μπορεί να ελέγχεται από ένα μόνο εξωτερικό αρχείο CSS. Έτσι, κάθε αλλαγή στο στυλ της ιστοσελίδας μπορεί να γίνεται με μια μοναδική αλλαγή σε αυτό το αρχείο, αντί για την επεξεργασία πολλών σημείων σε κάθε σελίδα που υπάρχει στο site

- Μικρότερο μέγεθος αρχείου, δεδομένου ότι ο κάθε κανόνας μορφοποίησης γράφεται μόνο μια φορά και όχι σε κάθε σημείο που εφαρμόζεται.
- Καλύτερο SEO (Search engine optimization). Οι μηχανές αναζήτησης δεν «μπερδεύονται» ανάμεσα σε περιεχόμενο και τη μορφοποίηση του, αλλά έχουν πρόσβαση στο περιεχόμενο σκέτο, οπότε είναι πολύ ευκολότερο να το καταγράψουν και να το αρχειοθετήσουν (indexing).
- Γρηγορότερες σελίδες. Όταν χρησιμοποιούμε εξωτερικό αρχείο CSS, ο browser την πρώτη φορά που θα φορτώσει κάποια σελίδα του site μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του site μας.

Βασικοί κανόνες σύνταξης CSS

Ένας «κανόνας» CSS αποτελείται από 2 μέρη:

- τον **επιλογέα (CSS selector)** ο οποίος αφορά το **ΤΙ** θα μορφοποιηθεί,
- τις **ιδιότητες (CSS properties)** οι οποίες αφορούν το **ΠΩΣ** αυτό θα μορφοποιηθεί και
- τις **τιμές (values)** των ιδιοτήτων του επιλογέα.

Η σύνταξη είναι η εξής:

```

επιλογέας
{
    ιδιότητα1: τιμή1;
    ιδιότητα2: τιμή2;
    ...
}
  
```

Τα σχόλια στον κώδικά CSS εισάγονται ανάμεσα στα σύμβολα: `/*...*/`

Ορισμός των στυλ

Ο κώδικας CSS μπορεί να βρίσκεται είτε μέσα στην σελίδα html είτε σε ξεχωριστό αρχείο:

- **Ορισμός των στυλ ανάμεσα στον κώδικά της σελίδας (Inline CSS):**
Αν επιθυμούμε να μορφοποιήσουμε ένα στοιχείο μόνο, και δεν πρόκειται να χρειαστούμε αυτό το είδος μορφοποίησης για τίποτε άλλο στο site, μπορούμε να γράψουμε «χύμα» τις ιδιότητες CSS μέσα στον κώδικά html χρησιμοποιώντας το χαρακτηριστικό (attribute) style, το οποίο το διαθέτει σχεδόν κάθε στοιχείο html.
Στην περίπτωση αυτή δεν ορίζουμε επιλογέα (CSS selector) και δεν βάζουμε αγκύλες.

Παραδείγματα:

- `<p style="color:red;"> κείμενο </p>`
- `<em style="font-size: 12px; font-family: Arial, Helvetica, color: #FFA07A; font-weight: bold;">κείμενο`

- **Ορισμός των στυλ μέσα στην σελίδα (Internal Style Sheet)**

CSS για μια συγκεκριμένη σελίδα: Στο <head>...</head>

Πολλές φορές, μπορεί να θέλουμε να εφαρμόσουμε κάποιους κανόνες CSS μόνο για τη συγκεκριμένη σελίδα και όχι για όλο το website.

Ένας τρόπος να το κάνουμε αυτό, για να μην δημιουργήσουμε ξεχωριστό αρχείο CSS, είναι να εισάγουμε εντός των tags <head>...</head> της σελίδας τους κανόνες CSS μας οριοθετημένα μεταξύ των tag <style>.

Παράδειγμα 12

```
<html>
<head>
<TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<style type="text/css">
  body
  {
    color:gray;
    margin-left: 25;
    margin-right: 25;
  }
  p
  {
    color: blue;
    font-family: Arial;
  }
  h1
  {
    color: red;
    text-align:center;
  }
</style>

</head>
<body>
  <h1> Δοκιμαστική Ιστοσελίδα </h1>
  <hr>
  Αυτή είναι η ιστοσελίδα του <b> Σπύρου Ζώγα </b> Μεταπτυχιακού Φοιτητή
  "Πληροφορικής Υγείας"
  <p> Στα πλαίσια του μαθήματος <i>Αρχές Πληροφορικής.</i> </p>
  <hr>
</body>
</html>
```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του **Σπύρου Ζώγα** Μεταπτυχιακού Φοιτητή
"Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής.*

- **Ορισμός των στυλ σε εξωτερικό αρχείο (External Style Sheet) CSS:**

Η πιο «σωστή» χρήση του CSS και αυτή που θα έπρεπε να είναι η πρώτη λύση στην οποία θα καταφύγουμε, είναι η χρήση εξωτερικού αρχείου CSS. Με τον τρόπο αυτό μπορούμε να εισάγουμε τα στυλ, που περιέχονται σε ένα αρχείο .css, μέσα σε όσες σελίδες html επιθυμούμε, δημιουργώντας έτσι ένα ενιαίο στυλ για τις σελίδες αυτές.

Για να τη χρησιμοποιήσουμε:

- γράφουμε τους κανόνες CSS μας σε ένα αρχείο με επέκταση css (πχ **main.css**) και
- στο head της κάθε σελίδας του site μας εισάγουμε την εγγραφή:

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

Αξίζει να σημειωθεί ότι μπορούμε να έχουμε πολλά αρχεία css στην ίδια σελίδα, και μεγαλύτερη προτεραιότητα έχει πάντα αυτό που έχει γραφτεί τελευταίο.

Παράδειγμα 13

```
<html>
<head>
  <TITLE>Προσωπική πρώτη ιστοσελίδα </TITLE>
  <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

  <link rel="stylesheet" type="text/css" href="main1.css" />

</head>
<body>
  <h1> Δοκιμαστική Ιστοσελίδα </h1>
  <hr>
  Αυτή είναι η ιστοσελίδα του <b> Σπύρου Ζώγα </b>
  Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"
  <p> Στα πλαίσια του μαθήματος <i>Αρχές Πληροφορικής.</i> </p>
  <hr>
</body>
</html>
```

Αρχείο main1.css

```
body
{
  color:gray;
  margin-right:25;
  margin-left: 25;
}
p
{
  color:blue;
  font-family:Arial;
}
h1
{
  color:green;
  text-align:center;
}
```

Δοκιμαστική Ιστοσελίδα

Αυτή είναι η ιστοσελίδα του **Σπύρου Ζώγα** Μεταπτυχιακού Φοιτητή "Πληροφορικής Υγείας"

Στα πλαίσια του μαθήματος *Αρχές Πληροφορικής.*

Έχουμε την δυνατότητα να ορίσουμε το **ίδιο στυλ για περισσότερες από μια HTML ετικέτες**. Αυτό μπορούμε να το πετύχουμε τοποθετώντας στην θέση του επιλογέα τις ετικέτες χωρίζοντας τες με κόμμα.

```
p,td,th {
  font-family: "Courier New";
  color: #FF3333;
  text-align: center;
}
```

Σύνταξη των στυλ

Ο **επιλογέας** συνήθως είναι μια ετικέτα της HTML στην οποία θέλουμε να εφαρμόσουμε το στυλ.

Στα προηγούμενα παραδείγματα ο επιλογέας ήταν πάντα κάποια ετικέτα.

Γενικά μπορούμε να ορίσουμε στυλ για:

- **συγκεκριμένες ετικέτες.**
- **ετικέτες που περιέχουν την ιδιότητα class**, η τιμή της οποίας χρησιμοποιείται στα στυλ ώστε αυτά να εφαρμοστούν μόνο στις ετικέτες που έχουν την ίδια τιμή στην class.
- **ετικέτες που περιέχουν την ιδιότητα id**, η τιμή της οποίας χρησιμοποιείται στα στυλ ώστε αυτά να εφαρμοστούν μόνο στις ετικέτες που έχουν την ίδια τιμή στην id.

```

επιλογέας
{
    ιδιότητα1: τιμή1;
    ιδιότητα2: τιμή2;
    ...
}

```

✚ Ετικέτες που περιέχουν το χαρακτηριστικό class

Μπορούμε να ορίσουμε διαφορετικά στυλ για ίδιες ετικέτες χρησιμοποιώντας για επιλογή το όνομα της ετικέτας μαζί με την τιμή που περιέχει το χαρακτηριστικό class της ετικέτας.

Παράδειγμα. Ορίζουμε μέσα στην ετικέτα <head> ή σε ξεχωριστό αρχείο τον επόμενο κώδικά CSS.

```

p.xarakteristiko1
{
    color:blue;
    font-family:Arial;
}
p.xarakteristiko 2
{
    color:red;
    font-family:Verdana;
}
p.xarakteristiko 3
{
    color:Green;
    font-family:calibri;
}

```

Μέσα στον κώδικα της html μπορούμε να χρησιμοποιήσουμε κατά περίπτωση όποια μορφοποίηση παραγράφου από τις παρακάτω θέλουμε:

- `<p class="xarakteristiko1"> κείμενο</p>`
- `<p class="xarakteristiko2"> κάποιο άλλο κείμενο</p>`
- `<p class="xarakteristiko3"> κάποιο άλλο κείμενο</p>`

Έτσι το στυλ **p.xarakteristiko1** θα εφαρμοστεί σε όλες τις ετικέτες `<p>` οι οποίες περιέχουν το χαρακτηριστικό class με την τιμή "xarakteristiko1".

Ενώ το στυλ **p.xarakteristiko2** θα εφαρμοστεί σε όλες τις ετικέτες `<p>` οι οποίες περιέχουν το χαρακτηριστικό class με την τιμή "xarakteristiko2" κ.ο.κ.

Στην περίπτωση που θέλουμε κάποιο χαρακτηριστικό να χρησιμοποιείται σε όλες τις ετικέτες παραλείπουμε το όνομα της ετικέτας στον επιλογέα ενός στυλ.

Παράδειγμα

```
.xaraktiristikoX
{
  color:red;
  font-family:Verdana;
}
```

Το χαρακτηριστικό **xaraktiristikoX** μπορεί τώρα να εισαχθεί σε όποια ετικέτα θέλουμε.

- `<h1 class="xaraktiristikoX"> επικεφαλίδα</h1>`
- `<p class="xaraktiristikoX"> παράγραφος</p>`
- `<li class="xaraktiristikoX"> στοιχείο λίστας`

Ετικέτες που περιέχουν το χαρακτηριστικό id

Ακολουθείτε η ίδια λογική που αναφέρθηκε και με το χαρακτηριστικό class για τις ετικέτες που έχουν το χαρακτηριστικό id.

Παράδειγμα. Ορίζουμε μέσα στην ετικέτα `<head>` ή σε ξεχωριστό αρχείο τον επόμενο κώδικά CSS.

```
p#xaraktiristiko1
{
  color:blue;
  font-family:Arial;
}
p#xaraktiristiko 2
{
  color:red;
  font-family:Verdana;
}
```

Μέσα στον κώδικα της html μπορούμε να χρησιμοποιήσουμε κατά περίπτωση όποια μορφοποίηση παραγράφου από τις παρακάτω θέλουμε:

- `<p id="xaraktiristiko1"> κείμενο</p>`
- `<p id="xaraktiristiko2"> κάποιο άλλο κείμενο</p>`

Έχουμε επίσης την δυνατότητα, παραλείποντας το όνομα της ετικέτας πριν το σύμβολο #, να εφαρμόσουμε ένα στυλ σε κάθε ετικέτα που έχει id ίδιο με αυτό που ακολουθεί μετά το #.

Και αυτός ο τρόπος σύνταξης (όπως και η σύνταξη στυλ με την χρήση class) είναι ο περισσότερο χρησιμοποιούμενος, καθώς μας δίνει την ευελιξία να διαχωρίσουμε ίδιες ετικέτες μεταξύ τους και να εφαρμόσουμε σε αυτές διαφορετικά στυλ.

✚ Σύνταξη στυλ για ετικέτες με συγκεκριμένη ιδιότητα και τιμή

Εφαρμόζει στυλ σε HTML ετικέτες με συγκεκριμένο χαρακτηριστικό και τιμή.

Παράδειγμα. Ορίζουμε μέσα στην ετικέτα <head> ή σε ξεχωριστό αρχείο τον επόμενο κώδικά CSS.

```
p[align="center"]
{
  color:blue;
  font-family:Arial;
}
p[align="left"]
{
  color:red;
  font-family:Verdana;
}
```

Το κάθε στυλ θα εφαρμοστεί στην αντίστοιχη παράγραφο.

- `<p align="center">Αυτή είναι μία ιστοσελίδα </p>`
- `<p align="left"> Στα πλαίσια του μαθήματος </p>`

✚ Ψευδο-κλάσεις (Pseudo-classes)

Οι ψευδοκλάσεις της CSS χρησιμοποιούνται για να προσθέσουν εφέ σε ορισμένα στοιχεία της σελίδας μας.

Αποτελούν κάποιες λέξεις κλειδιά που γράφουμε μετά από έναν επιλογέα του τύπου στοιχείο και ξεκινούν με :

```
Επιλογέας:Ψευδοκλάση
{
  ιδιότητα1: τιμή1;
  ιδιότητα2: τιμή2;
  ...
}
```

Παραδείγματα

<pre>a.example1 { color: Red; font-weight : bold; font-size : 14px; }</pre>	<pre>a.example1:visited { color: green; font-weight : bold; font-size : 14px; }</pre>	<pre>a.example1:hover { color: green; font-weight : bold; font-size : 14px; }</pre>
---	---	---

`Wlearn.gr link`

:link, εφαρμόζεται το στυλ με την ψευδοκλάση αυτή, όταν ο χρήστης δεν έχει επισκεφθεί παλαιότερα τον σύνδεσμο.

:visited, εφαρμόζεται το στυλ με την ψευδοκλάση αυτή, όταν ο χρήστης έχει επισκεφθεί παλαιότερα τον σύνδεσμο.

:hover, εφαρμόζεται το στυλ με την ψευδοκλάση αυτή, όταν ο χρήστης τοποθετεί τον δείκτη του ποντικιού του επάνω από τον σύνδεσμο

:active, εφαρμόζεται το στυλ με την ψευδοκλάση αυτή, όταν ο χρήστης έχει πατημένο το αριστερό κλικ του ποντικιού του επάνω στον σύνδεσμο

✚ Βασικές CSS ιδιότητες (CSS properties)

➤ Για μορφοποίηση κειμένου

- **Color:** Αφορά το χρώμα του κειμένου. Τα χρώματα μπορούν να εισαχθούν είτε σε μορφή RGB (πχ `color: rgb(255,128,30);`), είτε σε μορφή hex (πχ `color: #ff801e;`) είτε με τη μορφή κάποιου keyword (πχ `color:orange;`).
- **font-size:** Αφορά το μέγεθος της γραμματοσειράς. Σε pixels (πχ `font-size: 12px;`) και σε στιγμές (πχ `font-size:10pt`)
- **font-family:** Η ιδιότητα αυτή μας επιτρέπει να ορίσουμε ένα πλήθος γραμματοσειρών που θα χρησιμοποιηθούν για το κείμενο, κατά σειρά προτίμησης.
- **font-style:** Αφορά μόνο την περίπτωση όπου το κείμενο θα είναι πλάγιο. Οι πιθανές τιμές της είναι **normal**, και **italic**.
- **font-weight:** Χρησιμοποιείται για να ορίσει αν το κείμενο μας θα είναι έντονο ή όχι. Οι τιμές που συνήθως χρησιμοποιούνται σε αυτή την ιδιότητα είναι οι **normal** και **bold**.
- **text-decoration:** Μας επιτρέπει να εφαρμόσουμε στο κείμενο μας διάφορα εφέ, συμπεριλαμβανόμενης και της υπογράμμισης. Οι τιμές που δέχεται είναι οι εξής:
 - **none:** Καμία διακόσμηση
 - **underline:** Υπογράμμιση
 - **overline:** Γραμμή πάνω από το κείμενο (ουσιαστικά το αντίθετο της υπογράμμισης)
 - **line-through:** Διαγράμμιση
 - **blink:** Κάνει το κείμενο να αναβοσβήνει.
- **text-align:** Μας επιτρέπει να καθορίσουμε τη στοίχιση του κειμένου μας. Όπως θα περίμενε κανείς, οι πιθανές τιμές είναι `left`, `center`, `right` και `justify`.
- **font:** Μας επιτρέπει να καθορίσουμε αρκετές ιδιότητες κειμένου με μια μόνο ιδιότητα (συγκεκριμένα τις `font-size`, `font-family`, `font-weight` και `font-style`)
 - πχ: `font: 12px Calibri, Trebuchet MS, Verdana, sans-serif;`

➤ Ιδιότητες που αφορούν το φόντο

- **background-color:** Χρησιμοποιείται για να ορίσει χρώμα φόντου στα στοιχεία που αφορά ο επιλογέας. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιγράφηκαν για την ιδιότητα `color`.
- **background-image:** Χρησιμοποιείται για να ορίσει μια εικόνα φόντου. Η τιμή που δέχεται είναι της μορφής `url(διεύθυνση_εικόνας)`.
- **background-repeat:** Αφορά τον τρόπο που θα επαναλαμβάνεται η εικόνα που ορίσαμε στην παραπάνω ιδιότητα. Οι τιμές που δέχεται είναι:
 - **no-repeat:** Καμία επανάληψη
 - **repeat-x:** Η εικόνα επαναλαμβάνεται στον οριζόντιο άξονα
 - **repeat-y:** Η εικόνα επαναλαμβάνεται στον κάθετο άξονα
 - **repeat:** Η εικόνα επαναλαμβάνεται και οριζόντια και κάθετα (η τιμή `repeat` είναι και η προεπιλεγμένη τιμή της ιδιότητας `background-repeat`).

- **background-position:** Η ιδιότητα αυτή μας επιτρέπει να ορίσουμε τη θέση που θα τοποθετηθεί η εικόνα φόντου που ορίσαμε με την ιδιότητα `background-image` τόσο κάθετα, όσο και οριζόντια. Συνήθως ως τιμές της χρησιμοποιούμε κάποια keywords, τα οποία είναι τα εξής:
 - `left`, `center`, `right` όσον αφορά την οριζόντια θέση της εικόνας
 - `top`, `center`, `bottom` όσον αφορά την κάθετη θέση της εικόνας
- **background:** Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να ορίσουμε με μία μόνο ιδιότητα όλες τις παραπάνω.
 - πχ: `background: #CCCCCC url(background.gif) no-repeat top center;`

➤ **Ιδιότητες που αφορούν το περίγραμμα**

- **border-color:** Ρυθμίζει το χρώμα περιγράμματος.
- **border-width:** Ρυθμίζει το πάχος του περιγράμματος
- **border-style:** Ορίζει το στυλ του περιγράμματος. Οι τιμές που πέρνει είναι οι εξής:
 - **solid:** «Συμπαγές» περίγραμμα, δηλαδή χωρίς κάποια διακόσμηση, μια ενιαία γραμμή
 - **dashed:** Περίγραμμα που αποτελείται από παύλες.
 - **dotted:** Περίγραμμα που αποτελείται από τελείες.
 - **double:** Σαν την τιμή `solid`, μόνο που δημιουργεί δύο περιγράμματα,
- Υπάρχουν και οι τιμές **groove**, **ridge**, **inset**, **outset** οι οποίες δημιουργούν διαφόρων ειδών «ψευδό-3D» περιγράμματα.
- **border:** Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να ορίσουμε με μία μόνο ιδιότητα.
 - Πχ: `border: 10px solid #999999`

➤ **Διάφορες άλλες CSS ιδιότητες**

- **padding:** Ορίζει το κενό που θα υπάρχει μεταξύ των ορίων ενός στοιχείου και των περιεχομένων του.
- **margin:** Ορίζει τον χώρο μεταξύ των ορίων ενός στοιχείου και όσων το περιβάλλουν.
- **width, height:** Οι δύο αυτές ιδιότητες ορίζουν το πλάτος και το ύψος ενός στοιχείου, είτε σε ποσοστό, είτε σε κάποια μονάδα μήκους.
- **float:** Δέχεται τις τιμές `right`, `left` και `none` και επιτρέπει τα στοιχεία που περιβάλλουν το στοιχείο εκείνο στο οποίο εφαρμόζουμε την ιδιότητα αυτή να «ρέουν» τριγύρω του.

➤ **Γενικά οι κανόνες προτεραιότητας στο CSS**

- Οι ιδιότητες CSS που βρίσκονται μέσα στην ιδιότητα `style` ενός στοιχείου έχουν πάντα μεγαλύτερη προτεραιότητα από οποιοδήποτε άλλες (εκτός από όσες χρησιμοποιούν `!important`)
- Οι κανόνες CSS που βρίσκονται μέσα στα tags `<style>...</style>` έχουν μεγαλύτερη προτεραιότητα από αυτούς που βρίσκονται σε ένα εξωτερικό αρχείο CSS.

- Όταν δύο ή περισσότερα αρχεία CSS συμπεριλαμβάνονται στην ίδια ιστοσελίδα, οι κανόνες αυτού που γράφτηκε τελευταίο στη σελίδα έχουν μεγαλύτερη προτεραιότητα από αυτούς του πρώτου.
- Ένας κανόνας CSS έχει μεγαλύτερη προτεραιότητα από όσους κανόνες CSS βρίσκονται γραμμένοι πιο πάνω από αυτόν αν δεν συντρέχει άλλος λόγος για να μετρήσει παραπάνω κάποιος από αυτούς.
- Οι κανόνες με επιλογή του τύπου `.όνομα_κλάσης` έχουν μεγαλύτερη προτεραιότητα από κανόνες με επιλογή του τύπου `στοιχείο`, μιας και είναι πιο συγκεκριμένοι.
- Οι επιλογείς του τύπου `#όνομα_id` έχουν μεγαλύτερη προτεραιότητα από επιλογείς του τύπου `.όνομα_κλάσης`, μιας και είναι πιο συγκεκριμένοι.
- Σε κάθε περίπτωση, αν επιθυμούμε μια ιδιότητα ενός κανόνα να έχει μεγαλύτερη προτεραιότητα από αυτή που της αντιστοιχεί, μπορούμε πριν το `;` να γράψουμε **!important**.