



## ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΤΜΗΜΑ ΠΑΤΡΑΣ

### Σχολή Διοίκησης και Οικονομίας

Τμήμα Λογιστικής

---

### ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Σχεδίαση και ανάπτυξη δικτυακού τόπου με χρήση συστήματος διαχείρισης περιεχομένου ανοιχτού κώδικα.

---

ΓΑΛΑΚΟΥΤΗΣ ΓΕΩΡΓΙΟΣ, ΔΕΜΗΣ ΠΑΝΑΓΙΩΤΗΣ, ΚΑΡΑΓΙΑΝΝΗΔΗΣ ΝΙΚΟΛΑΟΣ

Εισηγητής: ΤΣΙΡΑΚΗΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

Επιβλέπων:

ΠΟΥΛΑΣΤΙΔΗΣ ΦΙΛΙΠΠΟΣ

ΣΤΑΜΑΤΗΣ ΚΩΝΣΤΑΝΤΙΝΟΣ

©

2013

**Υπεύθυνη Δήλωση** : Βεβαιώνουμε ότι είμαστε συγγραφείς αυτής της πτυχιακής εργασίας και ότι κάθε βοήθεια την οποία είχαμε για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην πτυχιακή εργασία. Επίσης έχουμε αναφέρει τις όποιες πηγές από τις οποίες κάναμε χρήση δεδομένων, ιδεών ή λέξεων, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επίσης βεβαιώνουμε ότι αυτή η πτυχιακή εργασία προετοιμάστηκε από εμάς προσωπικά ειδικά για τις απαιτήσεις του προγράμματος σπουδών του Τμήματος Λογιστικής του Α.Τ.Ε.Ι. ΠΑΤΡΑΣ.

Γαλακούτης Γεώργιος, Καραγιαννίδης Νικόλαος, Δέμης Παναγιώτης.

Ημερ:04/09/2013

## ΠΕΡΙΛΗΨΗ

Στόχος μας σε αυτήν την πτυχιακή εργασία είναι να ολοκληρώσουμε την σχεδίαση και ανάπτυξη δικτυακού τόπου με χρήση συστήματος διαχείρισης περιεχομένου ανοιχτού κώδικα, που θα πληροί όλες τις λειτουργίες τις οποίες έχει ένας σύγχρονος διαδικτυακός ιστότοπος μιας ξενοδοχειακής μονάδας, με δυνατότητα διαδικτυακής κρατήσεις.

Στις επόμενες καλούμαστε σελίδες θα ανακαλύψουμε τα προγράμματα και τις γλώσσες προγραμματισμού που χρησιμοποιούνται για την κατασκευή της πλειονότητας των ιστότοπων του διαδικτύου. Θα μάθουμε πώς να γράφουμε και να τροποποιούμε κώδικα HTML και CSS προκειμένου να επιτύχουμε το επιθυμητό αποτέλεσμα. Θα ανακαλύψουμε εύκολες και ελεύθερες εφαρμογές για να δομήσουμε και να λειτουργήσουμε μια βάση δεδομένων.

Μετέπειτα θα δούμε το πώς λειτουργεί στην πράξη ο ιστότοπος και βήμα-βήμα πως μπορεί ένας επισκέπτης να πλοηγηθεί και να πραγματοποιήσει μια κράτηση. Στην συνέχεια θα επικεντρωθούμε στα άδρα του ιστότοπου, την βάση δεδομένων αυτή καθαυτή. Θα συναντήσουμε τους πίνακες και τις καταχωρίσεις που επιτρέπουν στον ιστότοπο να παύει να είναι ένα απλό μέσο διαφήμισης και να γίνει ένα αναντικατάστατο διαχειριστικό εργαλείο.

Τέλος εκφράζουμε τα συμπεράσματα και τις εμπειρίες μας, από αυτήν την προσπάθεια και προσπαθούμε να μαντέψουμε, τι μας επιφυλάσσει η τεχνολογία και ο παγκόσμιος ιστός στο μέλλον.

---



## ΠΕΡΙΕΧΟΜΕΝΑ

---

<b>Περιεχόμενα</b>	<b>04</b>
<b>ΚΕΦΑΛΑΙΟ 1 – ΣΚΟΠΟΣ-ΕΙΣΑΓΩΓΗ</b>	<b>07</b>
1.1 Πότε Ανακαλύφθηκε το διαδίκτυο	09
1.2 Ηλεκτρονικό εμπόριο - Τουρισμός και Παγκόσμιος Ιστός	10
1.3 Ηλεκτρονικές Υπηρεσίες και Εφαρμογές στον Τουρισμό	12
1.4 Οφέλη μέσω της προβολής μίας τουριστικής μονάδας με ηλεκτρονικό σύστημα κρατήσεων στο διαδίκτυο.	14
1.5 Οι υπηρεσίες του σήμερα και του αύριο στην ηλεκτρονική προβολή τουριστικών καταλυμάτων.	14
1.6 Ανάλυση των γενικών χαρακτηριστικών που διέπουν μια επιτυχημένη προβολή στο διαδίκτυο.	14
1.7 Ερευνά αγοράς.	15
<b>ΚΕΦΑΛΑΙΟ 2 – ΠΡΟΡΑΜΜΑΤΑ ΚΑΙ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ</b>	<b>19</b>
2.1 Τι είναι η HTML	20
2.1.1 Οι διάφορες εκδόσεις HTML	21
2.1.2 Δημιουργία αρχείων HTML	22
2.1.2.1 Tag <head> </head>	23
2.1.2.2 Tag <body> </body>	24
2.2 Τι είναι το CSS (Cascading Style Sheets)	27
2.2.1 Η αλληλουχία εφαρμογής των φύλλων στυλ	28
2.2.2 Ποια πλεονεκτήματα έχει η χρήση CSS έναντι της μορφοποίησης μέσω HTML attributes	29
2.2.3 Βασικοί κανόνες σύνταξης	30
2.2.4 Πως εισάγουμε CSS στη σελίδα μας	33
2.2.5 Βασικές ιδιότητες CSS (CSS properties)	35
2.3 Τι Είναι η Php (εν συντομία)	43
2.3.1 Το πρώτο παράδειγμα σε PHP	44
2.3.2 Το πρώτο πρόγραμμα σε PHP	45
2.3.3 PHP στον εξυπηρετητή ιστοσελίδων	48
2.4 Javascript	52
<b>ΚΕΦΑΛΑΙΟ 3 - Σχεδιασμός και λειτουργία</b>	<b>54</b>
3.1 Περιγραφή σελίδας.	55
3.1.1 Γενική χαρακτηριστικά και δομή.	56

3.1.2 Τομείς ιστότοπου	57
3.1.3 Μπάρες πλοήγησης (Navigation Bars)	58
3.1.3.1 Κύρια σελίδα (Homepage).	59
3.1.3.2 Σχετικά με το ξενοδοχίο μας(About our hote).	60
3.1.3.3 Μεσσηνία (Messinia)	61
3.1.3.4 Έκθεση (Photogallery).	62
3.1.3.5 Επικοινωνία (Contacts)	63
3.2 Λειτουργία συστήματος κρατήσεων και βάσης δεδομένων	64
3.2.1 Η προσέγγιση μας.	64
4.2.2 Παρουσίαση του ιστότοπου στην πράξη.	64

---

#### **ΚΕΦΑΛΑΙΟ 4 – Εφαρμογές και ανάλυση βάσης δεδομένων.**

4.1 Βάση δεδομένων	71
4.1.1 MY SQL εν συντομία	71
2.4.2 Τι είναι ο Apache HTTP	72
2.4.3 Τι είναι το phpMyAdmin	73
4.1.4 Εμείς χρησιμοποιούμε EASYRHP	75
4.2 Βάση δεδομένων.	76
4.2.1 Συνοπτική περιγραφή πίνακα Bookin	78
4.2.2 Συνοπτική περιγραφή πίνακα images	79
4.2.3 Συνοπτική περιγραφή πίνακα Rooms	81
4.3 Διάγραμμα και περιγραφή Δραστηριοτήτων	82
4.3.1 περιγραφή δραστηριότητας κράτησης	82
4.3.2 Σχήμα διαγραμματικής απεικόνισης διαδικασία μιας κράτησης	83
4.3.3 Κανονικοποίηση	84
4.3.4 Περιγραφή Συστήματος	89

---

#### **ΣΥΜΠΕΡΑΣΜΑΤΑ**

---

#### **ΑΝΑΦΟΡΕΣ**

---

#### **Συνοδευτικό DVD με τα αρχεία του ιστότοπου**

## ΕΥΧΑΡΙΣΤΙΕΣ

Θα θέλαμε να ευχαριστήσουμε όλα τα άτομα τα οποία κατά την διάρκεια της πτυχιακής μας εργασίας ήταν δίπλα μας και ιδιαίτερα τον επιβλέπων καθηγητή μας Τσιράκη Νικόλαο, που με τις πολύτιμες συμβουλές τους μας καθοδήγησαν για την διεκπεραίωση της εργασίας.

# ΚΕΦΑΛΑΙΟ 1

## ΣΚΟΠΟΣ ΕΙΣΑΓΩΓΗ



## 1.1 Πότε Ανακαλύφθηκε ο παγκόσμιος ιστός

Στις 12 Νοεμβρίου του 1990, ο επονομαζόμενος και «πατέρας» του διαδικτύου. Τιμ Μπέρνερς Λι ανακαλύπτει τον Παγκόσμιο Ευρύ Ιστό, γνωστότερο με τα αρχικά... WWW, πάνω στον οποίο βασίζεται η λειτουργία του διαδικτύου. Ο Τιμ Μπέρνερς Λι γεννήθηκε στο Λονδίνο στις 8 Ιουνίου 1955. Φοίτησε στο Queens College της Οξφόρδης. Εκεί κατασκεύασε τον πρώτο του υπολογιστή, με μόνα υλικά παλιά ηλεκτρονικά εξαρτήματα και ένα κολλητήρι.



Στο Πανεπιστήμιο, όπου σπούδασε Φυσική, συνελήφθη ως χάκερ με συνέπεια να αποκλειστεί από τη χρήση του πανεπιστημιακού υπολογιστή. Από την Οξφόρδη αποφοίτησε το 1976 και εργάστηκε σε επιχειρήσεις ως προγραμματιστής / σύμβουλος μέχρι το 1980, οπότε και μετακλήθηκε από το CERN, το Ευρωπαϊκό Κέντρο Φυσικής Στοιχειωδών Σωματιδίων, ως Σύμβουλος Μηχανικός Προγραμματισμού. Εκεί επινόησε και το πρόγραμμα Enquire, τον πρόδρομο του Παγκόσμιου Ιστού, που τον βοηθούσε να παρακολουθεί τον τεράστιο αριθμό ερευνητών και προγραμμάτων (projects) του Ιδρύματος. Αυτό το πρόγραμμα ποτέ δε δόθηκε για χρήση στο κοινό. Έφυγε από το CERN σε άλλες εργασίες, για να επιστρέψει εκεί το 1984 και να αναλάβει τα κατανεμημένα συστήματα μεταφοράς επιστημονικών δεδομένων και ελέγχου. Σύντομα βρέθηκε ξανά αντιμέτωπος με το παλιό πρόβλημα χειρισμού του τεράστιου όγκου ερευνητών και προγραμμάτων του Ιδρύματος συν το μάλλον ιδιότροπο σύστημα του CERN για τη διακίνηση επιστημονικών πληροφοριών ανάμεσα στα μέλη του. Άρχισε να οραματίζεται ένα παγκόσμιο σύστημα διακίνησης πληροφοριών, ταχύτερο και ολοσχερώς αποκεντρωμένο, ανεξάρτητο της πλατφόρμας του



κάθε υπολογιστή, πολύγλωσσο και χωρίς γραφειοκρατικούς περιορισμούς και καθυστερήσεις. Υπέβαλε ένα υπόμνημα σχετικά με το σχέδιό του στη διοίκηση, αλλά, τότε, δεν πήρε καμία απάντηση. Περιμένοντας να ξεπεραστούν τα γραφειοκρατικά προβλήματα της διοίκησης, άρχισε να εργάζεται πάνω στις λεπτομέρειες του συστήματος που είχε σκεφθεί.

Έτσι δημιούργησε το πρωτόκολλο http (hypertext transfer protocol), δηλαδή τη «γλώσσα» επικοινωνίας των υπολογιστών στο διαδίκτυο και, παράλληλα, επινόησε ένα τρόπο αναγνώρισης κάθε «εγγράφου», αποδίδοντάς του ένα μοναδικό παγκόσμιο αναγνωριστικό (Universal Resource Identifier), μαζί με ένα αναγνωριστικό διεύθυνσης. Τα δύο αυτά χαρακτηριστικά συνδυάστηκαν και σήμερα αποτελούν το URL Ενιαίο Χαρακτηριστικό Εντοπισμού (Uniform Resource Locator).

Το 1990 και συγκεκριμένα το Νοέμβριο, στις 12, ολοκλήρωσε τη δημιουργία του πρώτου προγράμματος περιήγησης (browser), έπρεπε όμως να δημιουργήσει και ένα πρόγραμμα εξυπηρέτησης (server) και μια γλώσσα για την περιγραφή του εγγράφου. Έτσι, επινόησε τη γλώσσα HTML (HyperText Markup Language). Το 1991 τα είχε ετοιμάσει όλα και, μη έχοντας ακόμη επίσημη απάντηση από το CERN, δημιούργησε τον πρώτο διακομιστή (server), τον info.cern.ch., διαθέτοντας παράλληλα ελεύθερα το πρόγραμμα περιήγησης και το λογισμικό του διακομιστή μέσω του Διαδικτύου. Παράλληλα, άρχισε να «διαφημίζει» το δημιούργημά του μέσω των Ομάδων Νέων (Newsgroups). Σύντομα άρχισε να επικοινωνεί με χρήστες και να βελτιώνει τη δημιουργία του, χρησιμοποιώντας τις υποδείξεις τους.

Στο μεταξύ, οι χρήστες της νέας υπηρεσίας, που ο ίδιος είχε ονομάσει Παγκόσμιο Ιστό (World Wide Web), άρχισαν να αυξάνονται αλματωδώς, ενώ οι ερευνητές διαπίστωσαν ότι ήταν πολύ πιο εύκολο και εξυπηρετικό να «δημοσιεύουν» τις πληροφορίες τους σε έναν ιστότοπο, αντί να απαντούν σε πολυάριθμα μηνύματα ηλεκτρονικού ταχυδρομείου ή να τις δημοσιεύουν στις ομάδες νέων, των οποίων η κίνηση πολλές φορές τις «έθαβε» (με την έννοια ότι περνούσαν απαρατήρητες). Ενδιαφέρον επίσης έδειξαν για το επίτευγμα του Λι και κυβερνητικές υπηρεσίες. Σύντομα, η κίνηση της νέας υπηρεσίας έγινε τόση, ώστε προέκυψε η ανάγκη δημιουργίας νέου λογισμικού περιήγησης.

Ο Marc Andreessen, ένας φοιτητής του Πανεπιστημίου του Ιλλινόις δημιούργησε το λογισμικό Mosaic, τον πρόγονο του σημερινού λογισμικού περιήγησης Netscape Navigator. Σημαντική συμβολή στο όλο εγχείρημα είχε, επίσης, και η συνεχώς αυξανόμενη δημοτικότητα των Windows της Microsoft με το γραφικό τους περιβάλλον. Η υπηρεσία του Παγκόσμιου Ιστού είναι σήμερα η δημοφιλέστερη στο Διαδίκτυο, με περισσότερες από 25 δισεκατομμύρια δημοσιευμένες σελίδες και περίπου 1,3 δισ. χρήστες παγκοσμίως, αριθμός που αυξάνεται μέρα με την ημέρα.

Σήμερα ο Τιμ Μπέρνερς Λι είναι ο Πρόεδρος του W3C Κονσόρτσιοθμ του Παγκόσμιου Ιστού, ερευνητής στο MIT και έχει πολλές τιμητικές διακρίσεις, μεταξύ πολλών άλλων και Ιππότης της Βρετανικής Αυτοκρατορίας (2004) από την Βασίλισσα της Αγγλίας και παράσημο εξαίρετων υπηρεσιών (Order of Merit) από την ίδια το 2007. Το περιοδικό Time τον κατέταξε ανάμεσα σε ένα από τα 100 λαμπρότερα πνεύματα του αιώνα. Τον

Δεκέμβριο του 2004 ονομάστηκε επίτιμος Καθηγητής στο τμήμα Επιστήμης Υπολογιστών στο Πανεπιστήμιο του Σαουθάμπτον. Είναι, επίσης, συν-διευθυντής του φορέα «Web Science Research Initiative (WSRI)» (Πρωτοβουλία Επιστημονικής Έρευνας στον Ιστό), που δημιουργήθηκε το 2006.<sup>av.1</sup>

Η επινόηση του Τιμ Μπέρνερς Λι μεταμόρφωσε τον κόσμο, αφού άλλαξε τα στάνταρ όχι μόνο στην ανταλλαγή πληροφοριών, αλλά και σε θέματα της καθημερινότητας, όπως διακίνηση και εμπόριο αγαθών, εκπαίδευση, ταξίδια, ενημέρωση, χρηματοοικονομικές συναλλαγές. Είναι αυτή που πραγματικά άνοιξε το Διαδίκτυο σε πολύ ευρεία μάζα χρηστών σε ολόκληρο τον πλανήτη.

## **1.2 Ηλεκτρονικό εμπόριο - Τουρισμός και Παγκόσμιος Ιστός**

«Το ηλεκτρονικό εμπόριο (HE) περιγράφει την διαδικασία αγοράς, πώλησης, μεταφοράς ή ανταλλαγής προϊόντων, υπηρεσιών και/ή πληροφοριών μέσω δικτύων υπολογιστών, περιλαμβανομένου και του Παγκόσμιου Ιστού.»<sup>av.2</sup> Το (HE) μπορεί να οριστεί από τις παρακάτω σκοπιές:

**Επιχειρησιακή διεργασία.** Από την σκοπιά των επιχειρησιακών διεργασιών, το HE αφορά στην εκτέλεση των εργασιών με ηλεκτρονικό τρόπο, ολοκληρώνοντας επιχειρησιακές διεργασίες μέσω ηλεκτρονικών δικτύων, και έτσι αντικαθιστώντας με πληροφορίες για φυσικές επιχειρησιακές διεργασίες.

**Εξυπηρέτηση.** Από την σκοπιά των υπηρεσιών, το HE είναι ένα εργαλείο που απευθύνεται στην επιθυμία των κυβερνήσεων, των εταιρειών, των πελατών και της διοίκησης να περικόψουν το κόστος των υπηρεσιών και ταυτόχρονα να βελτιώσουν την ποιότητα της εξυπηρέτησης πελατών και να αυξήσουν την ταχύτητα της εξυπηρέτησης.

**Εκπαίδευση.** Από την σκοπιά της εκπαίδευσης, το HE παρέχει την δυνατότητα εκπαίδευσης και επιμόρφωσης μέσω του παγκόσμιου ιστού σε σχολεία, πανεπιστήμια και σε άλλους οργανισμούς, περιλαμβανομένων και των επιχειρήσεων.

**Συνεργατική.** Από την σκοπιά της συνεργασίας, το HE είναι το πλαίσιο για διεπιχειρησιακή και ενδοεπιχειρησιακή συνεργασία.

**Κοινωνική.** Από την κοινωνική σκοπιά, το HE παρέχει μία θέση συγκέντρωσης μελών της κοινωνίας για εκμάθηση, συνδιαλλαγή και συνεργασία. Ο δημοφιλέστερος τύπος κοινότητας είναι τα κοινωνικά δίκτυα, όπως το Myspace.

## **Το ηλεκτρονικό επιχειρείν**

Στο σημείο αυτό καλό είναι να διευκρινιστεί ότι το ηλεκτρονικό επιχειρείν<sup>av.3</sup> δεν αναφέρεται μόνο στην πράξη της αγοραπωλησίας, η οποία είναι το τελικό στάδιο της διαδικασίας του επιχειρείν. Αναφέρεται σε οποιαδήποτε οικονομική δραστηριότητα που υποστηρίζει τη διαδικασία αυτή. Οι εφαρμογές του ηλεκτρονικού επιχειρείν παρέχουν τη δυνατότητα ανάκτησης και εύρεσης πληροφοριών καθώς επίσης και της συναλλαγής αγαθών.

Τα προϊόντα που διακινούνται μέσω του διαδικτύου είναι αγαθά και υπηρεσίες. Η τουριστική βιομηχανία είναι ένας από τους σημαντικότερους κλάδους στο Ηλεκτρονικό Επιχειρείν και απαιτεί άμεσα την εφαρμογή σύγχρονων τεχνολογιών πληροφόρησης και επικοινωνίας.

Υπάρχουν διάφοροι λόγοι που μπορούν να υποστηρίξουν αυτή την διαπίστωση. Γενικά πάντως θα μπορούσαμε να πούμε ότι η ομαλή και αποδοτική λειτουργία της τουριστικής βιομηχανίας είναι αφενός η έγκαιρη ενημέρωση σχετικά με την ύπαρξη και τη διαθεσιμότητα τουριστικών προϊόντων, αφετέρου δε ο συντονισμός μεταξύ προσφοράς και ζήτησης. Το ηλεκτρονικό επιχειρείν στην συγκεκριμένη περίπτωση, δεν είναι παρά εκείνος ο μηχανισμός ανταλλαγής πληροφοριών και οικονομικών μέσων που έρχεται να υποστηρίξει τη λειτουργία της τουριστικής οικονομίας. Η αλυσίδα διανομής στον τουρισμό αποτελείται από τους τελικούς καταναλωτές, τους μεσάζοντες εθνικοί οργανισμοί τουρισμού, ταξιδιωτικά γραφεία, και τους παροχής υπηρεσιών όπως αεροπορικές και ναυτιλιακές εταιρείες, ξενοδοχειακές επιχειρήσεις κ.α. Το παραδοσιακό αυτό μοντέλο διανομής τουριστικών προϊόντων και υπηρεσιών μπορεί να μεταφερθεί αυτούσιο στον παγκόσμιο ιστό. Η ευρωπαϊκή αγορά κρατήσεων μέσω του παγκόσμιου ιστού, είναι δυναμικά αναπτυσσόμενη αλλά παρόλο που είναι ακόμα νέα, ο ανταγωνισμός είναι ήδη έντονος.

Υπάρχουν μεγάλες ευκαιρίες για τον κάθε νεοεισερχόμενο στην εν λόγω αγορά, υπάρχουν όμως και πολλοί κίνδυνοι οι οποίοι προέρχονται τόσο από το περιβάλλον των ίδιων των τουριστικών εταιριών που δραστηριοποιούνται στο παγκόσμιο ιστό, όσο και από το ίδιο το περιβάλλον του κλάδου του Ηλεκτρονικού Εμπορίου. Ο τουρισμός είναι μια βιομηχανία που από τη φύση της έχει σαν στόχο να διευκολύνει τους δυνητικούς πελάτες. Η εφαρμογή όμως, των σύγχρονων τεχνολογιών πληροφόρησης και επικοινωνίας μέσω της ανάπτυξης του ηλεκτρονικού εμπορίου χρειάζεται έντονη προσοχή και εξέταση της συμβατότητας με τις υπόλοιπες δραστηριότητες της επιχείρησης. Με άλλα λόγια δεν χρειάζεται κάποιος να μπει στη χρονοβόρα και πολυδάπανη διαδικασία της ανάπτυξης του ηλεκτρονικού επιχειρείν χωρίς να έχει προ- εξετάσει την συμβατότητα του με τους στόχους και τις δραστηριότητες της επιχείρησης. Για αυτό τον λόγο λέμε ότι η ύπαρξη στρατηγικής κατά την διαδικασία ανάπτυξης του Ηλεκτρονικού Επιχειρείν καθίσταται περισσότερο σημαντική και απαραίτητη από ποτέ. Το Ηλεκτρονικό Εμπόριο εντάσσεται πλέον στα στρατηγικά επιχειρηματικά εργαλεία γιατί η στρατηγική διοίκηση είναι εκείνη που οδηγεί στην λήψη των βέλτιστων στρατηγικών αποφάσεων. Πολλές είναι οι εταιρίες εκείνες που χρησιμοποιούν το παγκόσμιο ιστό ως ένα εργαλείο προώθησης, προκειμένου να διαφημίσουν τα προϊόντα ή τις υπηρεσίες τους.

Σήμερα, η διαφήμιση αποτελεί - σε μεγάλο βαθμό - και “δικτυακή υπόθεση”, γεγονός που μπορεί να προσδώσει μεγάλα οφέλη σε κάθε εταιρεία που τη χρησιμοποιεί σωστά.

Είναι πολλοί οι τρόποι με τους οποίους ένας ιστότοπος μπορεί να διαφημιστεί εντός του παγκόσμιου ιστού, για να έχει ένας ιστότοπος, οικονομικό όφελος πρέπει να προσελκύσει επαναλαμβανόμενες επισκέψεις από πελάτες και με κάθε επίσκεψη να προσθέτει ακόμη περισσότερες πληροφορίες στο προφίλ του πελάτη. Οι ιστότοποι τουριστικού περιεχομένου έχει διαπιστωθεί ότι το δίκτυο είναι μόνο μια περιορισμένη φέτα του παγκόσμιου ιστού και έχει προσαρμοστεί στις προωθητικές ανάγκες των περισσότερων εταιριών. Οι τελευταίες τεχνολογίες του παγκόσμιου ιστού εμφανίζουν σημεία επαφής που είναι πιο επίκαιρα και σχετικά.

Η σύγκλιση του παγκόσμιου ιστού με ευρύτερες συνδέσεις καθώς και με την τηλεόραση θα επιτρέψει στους πωλητές να ενοποιήσουν το εμπόριο και την διασκέδαση. Οι καταναλωτές θα συνδέονται στο δίκτυο μέσω ασύρματων τηλεφώνων, προσωπικών ψηφιακών βοηθών, διαδραστικών τηλεοράσεων ή φορητών ηλεκτρολογικών υπολογιστών με ασύρματες συνδέσεις και θα βρίσκονται διαρκώς σε ένα ψηφιακό περιβάλλον.

Έτσι η ιδέα μάρκετινγκ μέσα από το διαδίκτυο θα αλλάξει σημαντικά σε έναν κόσμο που θα είναι συνέχεια σε επαφή μέσω της χρίσης ασύρματων τηλεφώνων, προσωπικών ψηφιακών βοηθών, διαδραστικών τηλεοράσεων ή φορητών ηλεκτρολογικών υπολογιστών με ασύρματες συνδέσεις και θα βρίσκονται διαρκώς σε ένα ψηφιακό περιβάλλον.

### 1.3 Ηλεκτρονικές Υπηρεσίες και Εφαρμογές στον Τουρισμό

Η εύρυθμη λειτουργία της τουριστικής βιομηχανίας στηρίζεται στην παροχή και προώθηση μεγάλου όγκου πληροφοριών (π.χ. πληροφορίες σχετικά με τα χαρακτηριστικά, διαθεσιμότητα και κρατήσεις τουριστικών προϊόντων-υπηρεσιών κλπ.) και συνεπώς, η ψηφιοποίηση των επιχειρησιακών λειτουργιών μέσω των τεχνολογικών εργαλείων και εφαρμογών του διαδικτύου αποτελεί καταλυτικό παράγοντα για την αύξηση της αποτελεσματικότητας αλλά και την ενίσχυση του διεθνούς ανταγωνιστικού πλεονεκτήματος των τουριστικών επιχειρήσεων. Ειδικότερα στην Ελλάδα, όπου οι τουριστικές επιχειρήσεις εξαρτώνται σε μεγάλο βαθμό για την προώθηση και πώληση του προϊόντος τους σε μεγάλους τουριστικούς οργανισμούς, η εκμετάλλευση των δυνατοτήτων του διαδικτύου παρέχει στις επιχειρήσεις τη δυνατότητα να διαφημιστούν και να προσεγγίσουν τους δυνητικούς τους πελάτες σε παγκόσμια κλίμακα όλο το εικοσιτετράωρο με άμεσο τρόπο και χαμηλό κόστος.

Εκτός από την δημιουργία και διάθεση καινοτομικών καναλιών διανομής και προώθησης, άλλες ηλεκτρονικές εφαρμογές όπως η ηλεκτρονική μάθηση (e-learning), ηλεκτρονικές προμήθειες (e-procurement και e-supply chain management), VoIP, παρέχουν στις τουριστικές επιχειρήσεις επιπρόσθετες δυνατότητες επαγγελματικής κατάρτισης, ανάπτυξης διοικητικών ικανοτήτων, εκσυγχρονισμό της αλυσίδας αξίας τους και μείωσης του λειτουργικού τους κόστους. Γενικότερα οι επιχειρηματικές εφαρμογές ηλεκτρονικού τουρισμού (e-tourism<sup>av.4</sup>) περιλαμβάνουν την υποστήριξη αλλά και την καινοτομική παροχή κρίσιμων επιχειρηματικών τουριστικών υπηρεσιών, όπως οι ηλεκτρονικές κρατήσεις (e-

bookings), η ηλεκτρονική διανομή (e-distribution), τα ηλεκτρονικά εισιτήρια (e-ticket), οι ηλεκτρονικές προμήθειες (e-procurement), η ηλεκτρονική διαχείριση επιχειρήσεων (π.χ. e-hotel property management systems), οι ηλεκτρονικές εφαρμογές διαχείρισης προορισμών (e-destination management systems) και άλλες υπηρεσίες παρεχόμενες από B2B ηλεκτρονικές αγορές.

Καθώς, οι επιχειρηματικές εφαρμογές ηλεκτρονικού τουρισμού απαιτούν σημαντικό βαθμό εξειδίκευσης των γενικών μοντέλων και τεχνολογικών λύσεων ηλεκτρονικής επιχειρηματικότητας, νέου είδους επιχειρήσεων, π.χ. ηλεκτρονικοί έμποροι δημιουργήθηκαν, δραστηριοποιούνται και συνεπώς έχουν ενταχθεί στην τουριστική βιομηχανία. Παράλληλα, το τοπίο στην παροχή ηλεκτρονικών υπηρεσιών στον τουρισμό διαμορφώνεται και από την μεγάλη ποικιλία και πλήθος επιχειρήσεων που δραστηριοποιούνται στην τουριστική βιομηχανία (π.χ. καταλύματα, τουριστικοί οργανισμοί, τουριστικά γραφεία, μουσεία, εστιατόρια, εταιρίες μεταφορών, επιχειρήσεις ενοικιάσεων αυτοκινήτων κλπ), την πολυπλοκότητα των λειτουργιών τους καθώς και τον διεπιχειρησιακό συντονισμό των δραστηριοτήτων τους που απαιτείται στην τουριστική αλυσίδα αξίας για την παροχή ολοκληρωμένων (seamless) τουριστικών υπηρεσιών υψηλής ποιότητας. Όμως η πολυμορφία της τουριστικής βιομηχανίας και η πολυπλοκότητα των ηλεκτρονικών υπηρεσιών και εφαρμογών έχει δημιουργήσει επιχειρήσεις δύο ταχυτήτων. Από την μία μεριά, υπάρχουν οι μεγάλες τουριστικές επιχειρήσεις, και συνήθως μέλη άλλων διεθνών ομίλων, που χρησιμοποιούν σύγχρονες μεθόδους διαχείρισης και έχουν υιοθετήσει τις ηλεκτρονικές εφαρμογές στον τουρισμό στο έπακρο, και από την άλλη μεριά υπάρχουν οι μικρές, και συνήθως οικογενειακές τουριστικές επιχειρήσεις, οι οποίες εξακολουθούν να διοικούνται ερασιτεχνικά και να μην διαθέτουν την τεχνογνωσία και ικανότητες να υιοθετήσουν τις νέες ηλεκτρονικές εφαρμογές και υπηρεσίες για τον εκσυγχρονισμό και αύξηση της αποτελεσματικότητας των επιχειρησιακών τους μεθόδων.

Το ψηφιακό χάσμα (digital divide) που επέφερε η επανάσταση του ψηφιακού τουρισμού (e-tourism) όξυνε τις διαφορές και τα μειονεκτήματα των μικρομεσαίων τουριστικών επιχειρήσεων παρέχοντας συγκριτικά ανταγωνιστικά πλεονεκτήματα στις μεγάλες τουριστικές εταιρίες. Ταυτόχρονα το ψηφιακό χάσμα εντοπίζεται επίσης μεταξύ ιδιωτικών και δημόσιων τουριστικών επιχειρήσεων π.χ. μουσεία, τουριστικοί οργανισμοί.

Τέλος το ψηφιακό χάσμα δεν αναφέρεται μόνο στην υιοθέτηση ή μη ηλεκτρονικών εφαρμογών, αλλά χρησιμοποιείται επίσης για τον διαχωρισμό τουριστικών επιχειρήσεων όχι μόνο με βάση τον αριθμό των υιοθετούμενων ηλεκτρονικών εφαρμογών αλλά και με βάση το βαθμό και την καινοτομικότητα της τεχνολογικής αξιοποίησης και εκμετάλλευσης.

Σύμφωνα με διάφορες διεθνής έρευνες, το ψηφιακό χάσμα που υπάρχει στην ελληνική τουριστική βιομηχανία καθώς και ο βαθμός υιοθέτησης των ηλεκτρονικών τουριστικών υπηρεσιών την καθιστούν στις χαμηλότερες κλίμακες ανάπτυξης του ηλεκτρονικού τουρισμού.

## 1.4 Οφέλη μέσω της προβολής μίας τουριστικής μονάδας με ηλεκτρονικό σύστημα κρατήσεων στο διαδίκτυο.

Τα οφέλη της τουριστικής μονάδας μέσω της προβολής στο διαδίκτυο είναι τα εξής:

- Προβολή της επιχείρησης (εγκαταστάσεις, τοποθεσία) σε παγκόσμια κλίμακα.
- Προσέλκυση νέων πελατών.
- Καλύτερη διαχείριση του πελατολογίου με ειδικά CRM συστήματα.
- Πιο γρήγορη εξυπηρέτηση των πελατών της μέσω του ιστότοπου, λόγω πολλών αυτοματοποιημένων συστημάτων.
- Πολύ καλύτερη εποπτεία των κρατήσεων διότι όλες οι ενέργειες (επικύρωση κράτησης, μεταφορά χρημάτων σε τραπεζικό λογαριασμό έπειτα από επικύρωση κράτησης κ.α.) μπορούν να γίνουν είτε αυτοματοποιημένα είτε με λίγα κλικ.<sup>αν.5</sup>

## 1.5 Οι υπηρεσίες του σήμερα και του αύριο στην ηλεκτρονική προβολή τουριστικών καταλυμάτων.

Ο ανταγωνισμός η ευελιξία και η χρηστικότητα είναι σήμερα στόχοι του κάθε σύγχρονου καταλύματος που θέλει να προσφέρει υπηρεσίες ηλεκτρονικού συστήματος κρατήσεων. Παρακάτω θα παρουσιαστούν διάφοροι ιστότοποι μεγάλων ξενοδοχείων οι οποίοι περιέχουν μια γενική εικόνα του επίπεδου υπηρεσιών που υπάρχει σήμερα παγκόσμια αγορά σε ότι αφορά την παρουσίαση, την χρηστικότητα και την προβολή τους μέσω του διαδικτύου .

## 1.6 Ανάλυση των γενικών χαρακτηριστικών που διέπουν μια επιτυχημένη προβολή στο διαδίκτυο.

### 1) Πολύγλωσσοτητα:

Ίσως το σημαντικότερο πλεονέκτημα ενός συστήματος ηλεκτρονικών κρατήσεων, είναι ότι μπορεί να γίνει από οπουδήποτε στον κόσμο, για αυτό ένα χαρακτηριστικό μιας επιτυχημένης προβολής στο διαδίκτυο είναι η δυνατότητα να μπορούν οι χρηστές να είναι σε θέση να διαβάζουν στην γλώσσα τους.

### 2) Αναλυτική παρουσίαση των χώρων και των υπηρεσιών που προσφέρονται:

Να μην παραλείπετε καμία πληροφορία που αφορά τα δωμάτια και τις υπηρεσίες του ξενοδοχείου καθώς σε αντίθετη περίπτωση θα υπάρχει πρόβλημα γιατί για τις υπηρεσίες τις οποίες προσφέρει ένα ξενοδοχείο ο πελάτης πληρώνει και μένει ικανοποιημένος.

### **3) Έκθεση φωτογραφιών (Photo gallery):**

Πρέπει να υπάρχει η δυνατότητα να βλέπει ο χρήστης όλα τα δωμάτια και τους χώρους του ξενοδοχείου που ενδιαφέρεται να επισκεφτεί .

### **4) Στοιχεία για τον καιρό τις περιοχής:**

Πρέπει να υπάρχει η δυνατότητα να ξέρει ο επισκέπτης τον καιρό που κάνει στο μέρος που είναι για να πάει καθώς κατά πάσα πιθανότητα δεν θα έχει καμία σχέση με την περιοχή αυτή.

### **5) Δυνατότητα αποστολής ενημερώσεων και προσφορών (newsletters)**

Αρκετοί πελάτες νέοι κα παλιοί θέλουν να μαθαίνουν τα νέα και τις προσφορές του αγαπημένου τους ξενοδοχείου όποτε πρέπει να υπάρχει οι δυνατότητα αποστολής ενημερωτικών μηνυμάτων ηλεκτρονικού ταχυδρομείου υ ..

### **6) Βίντεο με τους χώρους του ξενοδοχείου:**

Οι μελλοντικοί μας πελάτες δεν θέλουν μόνο να βλέπουν τον χώρο που θα επισκεφτούν αλλά και να τον ζήσουν πριν ακόμα πάνε εκεί, για αυτό το λόγο η προσθήκη βίντεο κρίνεται απαραίτητη.

### **7) Δυνατότητα επικοινωνίας:**

Οι πελάτες πρέπει να έχουν την δυνατότητα να επικοινωνούν με τα διάφορα τμήματα του ξενοδοχείου για τυχόν διευκρινίσεις η ακόμα και παράπονα.

### **8) Κριτικές πελατών:**

Μια πολύ σημαντική ενότητα είναι αυτή με τις κριτικές των πελατών καθώς είναι μια πηγή εμπιστοσύνης προς τους νέους πελάτες.

### **9) Δυνατότητα αναζήτησης διαθεσιμότητας δωματίου:**

Η πιο βασική λειτουργία είναι να μπορεί ο υπονήφιος πελάτης να κάνει κράτηση σε πραγματικό χρόνο αφού αυτός είναι και ο σκοπός που έχει κάθε ξενοδοχειακή μονάδα χωρίς να βάζει σε περεταίρω διαδικασίες τους πελάτες της.

## **1.7 Έρευνα Αγοράς.**

Μπορούμε να πούμε γενικά, ότι βασικοί προβληματισμοί στη συμπεριφορά του τουρίστα αποτελούν η διαδικασία λήψης απόφασης για την αγορά προϊόντων τουρισμού και ο τρόπος με τον οποίο ανταποκρίνεται στα διάφορα ερεθίσματα μάρκετινγκ. Οι παράγοντες που επηρεάζουν τη συμπεριφορά του διακρίνονται σε: πολιτιστικούς, κοινωνικούς, προσωπικούς και ψυχολογικούς. Η ίδια η διαδικασία λήψης της απόφασης χωρίζεται σε πέντε στάδια:

1) αναγνώριση ανάγκης: ο πελάτης εντοπίζει μία διαφορά ανάμεσα στην πραγματική κατάσταση στην οποία βρίσκεται και την επιθυμητή κατάσταση όπου προσδοκά να βρεθεί ωθούμενος από εσωτερικά ή εξωτερικά ερεθίσματα.

2) αναζήτηση και συγκέντρωση πληροφοριών: ο πελάτης γνωρίζει τις υπάρχουσες εναλλακτικές επιλογές και τα χαρακτηριστικά τους. Η έκταση των πληροφοριών που θα αναζητήσει εξαρτάται από την ισχύ του κινήτρου για αγορά, την αρχική πληροφόρηση που αυτός έχει, την ευκολία απόκτησης πρόσθετης πληροφόρησης και το βαθμό ικανοποίησης που έχει μέσα από την αναζήτηση πληροφοριών. Οι πηγές πληροφόρησης σύμφωνα με τον Kotler είναι τρεις. Πρώτον, οι προσωπικές (οικογένεια, φίλοι, γνωστοί), δεύτερον οι εμπορικές (διαφήμιση, πωλητές, έμποροι κλπ) και τρίτον οι δημόσιες (ενώσεις καταναλωτών, άρθρα, κριτικές κλπ).

3) αξιολόγηση επιμέρους εναλλακτικών: ο πελάτης αντιλαμβάνεται κάθε αγαθό ως ένα σύνολο χαρακτηριστικών, δίνοντας μεγαλύτερη βαρύτητα στα χαρακτηριστικά εκείνα που συνδέονται με τις ανάγκες και επιθυμίες του και 69 που του αποδίδουν τα οφέλη που εκείνος αναζητά. Το σύνολο των πεποιθήσεων που ο πελάτης διαμορφώνει για κάθε εναλλακτική επιλογή, ως προς τα επιμέρους χαρακτηριστικά της, αποτελεί τη λεγόμενη εικόνα προϊόντος, το οποίο αποτελεί τη βάση στη διαμόρφωση στάσεων για κάθε εναλλακτική λύση.

4) απόφαση αγοράς: Πρόκειται για το στάδιο που ακολουθεί μετά από την ίδια την πρόθεση αγοράς, όπου είναι και δυνατόν να υπεισέλθουν απρόβλεπτοι παράγοντες που μπορεί να μεταβάλλουν την απόφαση αγοράς. Κατανοούμε ότι ένας σημαντικός παράγοντας για απόφαση αγοράς υπηρεσιών ενός πολυτελούς ξενοδοχείου αποτελεί ο οικονομικός, καθώς ο πελάτης ενός τέτοιου ξενοδοχείου είναι κατά κύριο λόγο υψηλού οικονομικού βιοτικού επιπέδου.

5) συμπεριφορά μετά την αγορά: ο βαθμός ικανοποίησης ενός πελάτη από την αγορά ενός αγαθού προσδιορίζεται από τη σχέση μεταξύ των προσδοκιών του ατόμου και της αντιλαμβανόμενης ποιότητας του αγαθού. Ικανοποίηση επέρχεται στην περίπτωση κατά την οποία η αντιλαμβανόμενη ποιότητα που αισθάνεται ο πελάτης αφού δοκιμάσει τα χαρακτηριστικά του προϊόντος ανταποκρίνεται στην προσδοκώμενη ποιότητα και αποτελεί τη βάση για την πραγματοποίηση επανάληψης της αγοράς. Εάν στα παραπάνω προστεθεί η πολυεθνικότητα του πελάτη, ο οποίος μπορεί να προέρχεται από οποιαδήποτε πόλη του εσωτερικού και του εξωτερικού, καταλαβαίνουμε τη μεγάλη σημασία που καλείται να παίζει στις μέρες η διαδικτυακή διαφήμιση στο μείγμα μάρκετινγκ για τον χώρο των ξενοδοχειακών υπηρεσιών. Η ίδια η ιστοσελίδα της επιχείρησης, διαδικτυακές διαφημίσεις με οποιαδήποτε μορφή, καθώς και η καταχώρηση του ξενοδοχείου σε ιστοσελίδες σχετικές με τον τουρισμό και μάλιστα με επιλογή ανάγνωσης αυτής σε διάφορες γλώσσες, σε συνδυασμό με την παροχή χρήσιμων πληροφοριών - αντιπροσωπευτικών φωτογραφιών, καθώς και τη δυνατότητα για άμεση κράτηση και την υποβολή ερωτήματος για περαιτέρω πληροφορίες έχουν απλουστεύσει τη διαδικασία και μπορεί κάλλιστα να αποτελέσουν το κίνητρο και το ερέθισμα για τον δυνητικό πελάτη να προχωρήσει στη λήψη απόφασης για αγορά της προσφερόμενης υπηρεσίας. Από μία άλλη σκοπιά, μπορούμε να πούμε ότι η επανάσταση των ηλεκτρονικών υπηρεσιών εισάγουν ένα παράδειγμα θεμελιώδους αλλαγής αντίληψης στον τρόπο λειτουργίας και οργάνωσης των ξενοδοχειακών επιχειρήσεων, αλλάζοντας σημαντικά το ανταγωνιστικό πεδίο στις σύγχρονες οικονομίες, αλλά και ειδικότερα στην τουριστική βιομηχανία. Στην κοινωνία της πληροφορίας η αξιοποίηση της τεχνολογίας δεν αποτελεί επιλογή. Τεχνολογικά «υποανάπτυκτες» επιχειρήσεις είναι



καταδικασμένες σε σημαντικά στρατηγικά μειονεκτήματα, αλλά και μείωση της συγκριτικής απόδοσης και αποτελεσματικότητας. Τα οφέλη χρήσης της διαδικτυακής διαφήμισης για τις ξενοδοχειακές μονάδες, στα πλαίσια του ηλεκτρονικού εμπορίου και γενικότερα του ηλεκτρονικού μάρκετινγκ, συνοψίζονται στα εξής:

- 24 ώρες τη μέρα 7 μέρες την εβδομάδα 365 μέρες τον χρόνο παγκόσμιες πωλήσεις και πελάτες.
- Μείωση κόστους κρατήσεων μέσω της αυτοματοποίησης της διαδικασίας που οδηγεί σε μείωση λειτουργικών εξόδων και κατ' επέκταση μείωση του συνολικού κόστους λειτουργίας.
- Αυξημένα έσοδα, νέοι πελάτες, καλύτερες τιμές και άμεση εξόφληση πληρωμών.
- Άμεση σχέση με τους πελάτες.
- Ευκαμψία τιμολογιακής πολιτικής ακόμα και την τελευταία στιγμή.
- Μείωση εξάρτησης από τρίτους.
- Κατανόηση συμπεριφοράς καταναλωτή στο Διαδίκτυο, έρευνα αγοράς και δημιουργία πελατειακής βάσης.
- Προσωποποίηση προϊόντων, υπηρεσιών και ιστοσελίδων.
- Ηλεκτρονικό ταχυδρομείο.
- Ιστοσελίδες με οπτικοακουστικό υλικό και χρήση πολυμέσων.
- Δυνατότητα άμεσης μέτρησης ικανοποίησης των πελατών.
- Διαδικτυακή συνεργασία με άλλες επιχειρήσεις.

Στόχος της παρούσας έρευνας είναι να μελετήσει τη σημασία της διαδικτυακής διαφήμισης στο χώρο των πολυτελών ξενοδοχείων της Αθήνας, δεδομένων α) του σπουδαίου ρόλου της διαφήμισης στην ευρεία έννοια του Μάρκετινγκ β) της ανοδικής τάσης του διαδικτύου και της χρήσης της διαδικτυακής διαφήμισης στην Ελλάδα βάσει των τεχνολογικών εξελίξεων και γ) της μεγάλης σημασίας της διαφήμισης στις ξενοδοχειακές υπηρεσίες. Ακολουθώς παρουσιάζονται συνοπτικά τα πολυτελή ξενοδοχεία που έλαβαν μέρος στην έρευνα. Τα στοιχεία που αναφέρονται αντλήθηκαν τόσο από τις ιστοσελίδες των εν λόγω ξενοδοχείων όσο και από κάποια γενικά στοιχεία που δηλώθηκαν από στελέχη τους στο ερωτηματολόγιο που χρησιμοποιήθηκε στην έρευνα.

#### BLAZER SUITES HOTEL

Έτος ίδρυσης: 1994 Κατηγορία (Αριθμός αστεριών): 4

Χωρητικότητα (κλίνες): 56

Αριθμός εργαζομένων: 15

Τοποθεσία: Βούλα

Ιδιαίτερες Παροχές: corner suites, large suites, typical suites, pool club, πλυντήριο ρούχων, δωρεάν ξένος τύπος, ξεναγήσεις και εκδρομές, υπηρεσίες μασάζ και κομμωτηρίου, baby-sitting με χρέωση.

### ANΘΕΑ ΑΕ – AMARILIA HOTEL

Έτος ίδρυσης: 1981

Κατηγορία (Αριθμός αστεριών): 4

Χωρητικότητα (κλίνες): 185

Αριθμός εργαζομένων: 30

Τοποθεσία: Βουλιαγμένη

Ιδιαίτερες Παροχές: διοργάνωση εκδηλώσεων- ημερίδων- συνεδρίων-γαμήλιων δεξιώσεων, superior rooms, suites, ανταλλαγή συναλλάγματος, 72 διεθνής τύπος, διαθεσιμότητα φαξ-φωτοτυπικού- γραμματειακής υποστήριξης- γιατρού, μεταφορά με λιμουζίνα

### CROWNE PLAZA

Έτος ίδρυσης: 2004

Κατηγορία (Αριθμός αστεριών): 5

Χωρητικότητα (κλίνες): 192

Αριθμός εργαζομένων: 70

Τοποθεσία: Μιχαλακοπούλου, Ιλίσια

Παροχές: superior rooms, club rooms, suites, διαθεσιμότητα 14 υψηλής τεχνολογίας αιθουσών με χωρητικότητα 1000 ατόμων για τη διοργάνωση συνεδρίων & εκδηλώσεων, υπηρεσία συναλλάγματος, δυνατότητα εκτύπωσης εγγράφων, Εξωτερική πισίνα στον 7ο όροφο, πλήρως εξοπλισμένο γυμναστήριο, σάουνα, χώρος για μασάζ, Business Center, υπόγειος χώρος στάθμευσης 150 θέσεων, κατάστημα δώρων, ATM πλύσιμο & στεγνό καθάρισμα ρούχων, φύλαξη παιδιών, έμφαση στην ασφάλεια και ιδιαίτερη μέριμνα για άτομα με ειδικές ανάγκες.

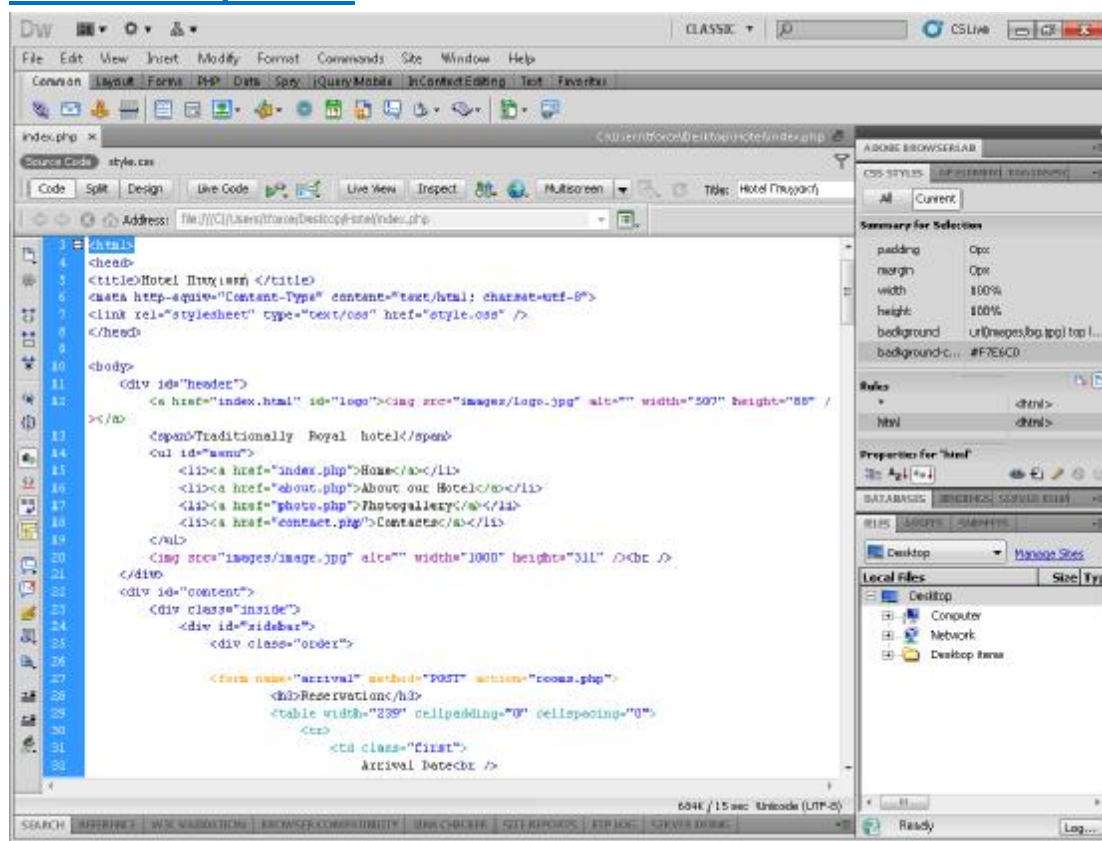
# ΚΕΦΑΛΑΙΟ 2

## ΠΡΟΓΡΑΜΜΑΤΑ ΚΑΙ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ



## ΚΕΦΑΛΑΙΟ 2 - ΠΡΟΓΡΑΜΜΑΤΑ ΚΑΙ ΓΛΩΣΣΕΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΥ

### 2.1 Τι είναι η HTML.



“Η **HTML** (ακρωνύμιο του αγγλικού **HyperText Markup Language**, ελλ. Γλώσσα Σήμανσης Υπερκειμένου) είναι η κύρια γλώσσα σήμανσης για τις ιστοσελίδες, και τα στοιχεία της είναι τα βασικά δομικά στοιχεία των ιστοσελίδων.”<sup>αν.6</sup>

Η HTML γράφεται υπό μορφή στοιχείων HTML τα οποία αποτελούνται από ετικέτες, οι οποίες περικλείονται μέσα σε σύμβολα «μεγαλύτερο από» και «μικρότερο από» (για παράδειγμα <html>), μέσα στο περιεχόμενο της ιστοσελίδας. Οι ετικέτες HTML συνήθως λειτουργούν ανά ζεύγη (για παράδειγμα <h1> και </h1>), με την πρώτη να ονομάζεται ετικέτα έναρξης και τη δεύτερη ετικέτα λήξης (ή σε άλλες περιπτώσεις ετικέτα ανοίγματος και ετικέτα κλεισίματος αντίστοιχα). Ανάμεσα στις ετικέτες, οι σχεδιαστές ιστοσελίδων μπορούν να τοποθετήσουν κείμενο, πίνακες, εικόνες κλπ.

Ο σκοπός ενός φυλλομετρητή ιστοσελίδων (web browser) είναι να διαβάσει τα έγγραφα HTML και τα συνθέτει σε σελίδες που μπορεί κανείς να διαβάσει ή να ακούσει. Ο φυλλομετρητής δεν εμφανίζει τις ετικέτες HTML, αλλά τις χρησιμοποιεί για να ερμηνεύσει το περιεχόμενο της σελίδας. Σύγχρονοι φυλλομετρητές είναι Internet explorer, Mozilla Firefox, Google chrome, Opera browser. Όλοι κάνουν ακριβώς την ίδια δουλειά, ακολουθώντας ελαφρώς διαφορετικές προσεγγίσεις και έχουν εδραιώσει το καθένα τα δικά του κοινά.

Τα στοιχεία της HTML χρησιμοποιούνται για να κτίσουν όλους του ιστότοπους. Η HTML επιτρέπει την ενσωμάτωση εικόνων και άλλων αντικειμένων μέσα στη σελίδα, και μπορεί να χρησιμοποιηθεί για να εμφανίσει διαδραστικές φόρμες. Παρέχει τις μεθόδους δημιουργίας δομημένων εγγράφων (δηλαδή εγγράφων που αποτελούνται από το περιεχόμενο που μεταφέρουν και από τον κώδικα μορφοποίησης του περιεχομένου) καθορίζοντας δομικά σημαντικά στοιχεία για το κείμενο, όπως κεφαλίδες, παραγράφους, λίστες, συνδέσμους, παραθέσεις και άλλα. Μπορούν επίσης να ενσωματώνονται σενάρια εντολών σε γλώσσες όπως η JavaScript, τα οποία επηρεάζουν τη συμπεριφορά των ιστοσελίδων HTML.

Οι φυλλομετρητές μπορούν επίσης να αναφέρονται σε στυλ μορφοποίησης CSS για να ορίζουν την εμφάνιση και τη διάταξη του κειμένου και του υπόλοιπου υλικού. Ο οργανισμός W3C, ο οποίος δημιουργεί και συντηρεί τα πρότυπα για την HTML και τα CSS, ενθαρρύνει τη χρήση των CSS αντί διαφόρων στοιχείων της HTML για σκοπούς παρουσίασης του περιεχομένου.

### 2.1.1 Οι διάφορες εκδόσεις της HTML

**HTML** Το 1990 ο Tim Berners-Lee από το Cern, το εργαστήριο φυσικής της Γενεύης, δημιούργησε ένα νέο πρωτόκολλο με το οποίο θα μπορούσαν να μεταφέρονται κάθε είδος αρχείων και αντικειμένων μέσα από το παγκόσμιο ιστό. Το πρωτόκολλο αυτό ονομάστηκε HTTP (HyperText Transfer Protocol) και σηματοδότησε την αρχή του WWW όπως το ξέρουμε σήμερα. Οι σελίδες που ήταν η βάση του WWW ήταν γραμμένες στην πρώτη έκδοση της γλώσσα HTML.

Το 1994 αναπτύσσεται το πρότυπο HTML 2.0 από ένα διεθνή οργανισμό (Internet Engineering Task Force). Η επόμενη έκδοση η 3.0 δεν έγινε αποδεκτή από τις Microsoft και Netscape οπότε γρήγορα αντικαταστάθηκε από την έκδοση 3.2 (1996). Η τελευταία περιελάμβανε πολλές από τις σημάνσεις (tags) που είχαν εισάγει οι δύο εταιρίες. Η έκδοση 4.0 παρουσιάστηκε τον Ιούνιο του 1997 και ακολουθήθηκε από την 4.01 το επόμενο έτος.



Το 2008 παρουσιάστηκε το HTML 5. Το πρότυπο είναι ακόμη υπό ανάπτυξη, και αναμένεται να παραμείνει έτσι για πολλά χρόνια, παρόλο που μέρη της HTML5 θα τελειώσουν και θα υποστηριχτούν από περιηγητές πριν το όλο πρότυπο φτάσει στη τελική κατάσταση πρότασης “Recommendation”.

## Διαφορές ανάμεσα στην XHTML και την HTML

Μπορούμε να ετοιμαστούμε ψυχολογικά για την XHTML, αρχίζοντας να γράφουμε κώδικα HTML με αυστηρούς κανόνες. Η XHTML αποτελεί τη νέα γενιά της HTML, αλλά θα πάρει φυσικά κάποιο χρόνο πριν οι φυλλομετρητές και τα άλλα προϊόντα λογισμικού είναι έτοιμα γι' αυτήν.

Η XHTML δεν είναι πολύ διαφορετική από την HTML 4.01, έτσι αν ο κώδικάς μας συμμορφωθεί με τα στάνταρτ της 4.01, αυτό θα είναι μια πολύ καλή αρχή. Επιπλέον, πρέπει από ΤΩΡΑ να αρχίσουμε να γράφουμε τον HTML κώδικά μας με πεζά γράμματα και ΠΟΤΕ να μην κάνουμε την κακή συνήθεια να ξεχνάμε τα τελικά tags, όπως είναι το `</p>`.<sup>av.6</sup>

Οι σημαντικότερες διαφορές ανάμεσα στις δύο γλώσσες είναι οι εξής<sup>av.6</sup>:

Τα στοιχεία της XHTML πρέπει να είναι σωστά φωλιασμένα (nested).

Τα έγγραφα της XHTML πρέπει να είναι σωστά μορφοποιημένα (well-formed).

Τα ονόματα των tags πρέπει να γράφονται με πεζά γράμματα.

Όλα τα στοιχεία της XHTML πρέπει να είναι κλειστά.

### 2.1.2 Δημιουργία αρχείων HTML

Η δημιουργία αρχείων HTML είναι πολύ απλή. Αρκεί να “τρέξουμε” έναν οποιοδήποτε διορθωτή κειμένου text, όπως την εφαρμογή Notepad των Windows. Να γράψουμε τον κώδικα HTML που επιθυμούμε και να το αποθηκεύσουμε (σώσουμε) σε ένα αρχείο με κατάληξη **.htm** ή **.html**.

#### **Βασικά χαρακτηριστικά ενός <tag><sup>av.7</sup>**

Η HTML είναι μία γλώσσα σήμανσης κειμένου. Αυτό γίνεται με την βοήθεια "HTML tags". Δεν είναι απαραίτητο μία σελίδα να έχει κείμενο. Μπορεί να περιλαμβάνει μόνο tags. Ένα tag αρχίζει με ένα σύμβολο '<' και τερματίζεται με ένα σύμβολο '>'. Μεταξύ αυτών τοποθετείται κείμενο που καθορίζει ποιο tag χρησιμοποιείται. Παράδειγμα είναι το tag `<html>`, που δηλώνει την αρχή ενός αρχείου html. Το tag αποτελείται από δύο τμήματα, το tag αρχής και το tag τέλους.

Αυτό που βρίσκεται ανάμεσα και μπορεί να επηρεαστεί, επηρεάζεται από το εφέ που έχει το tag. Για παράδειγμα το κείμενο ανάμεσα στα tag μπορεί να γίνει πράσινο αλλά η εικόνα θα μείνει στα ίδια χρώματα για η html δεν αλλάζει τα χρώματα εικόνων. Το tag του τέλους είναι ίδιο με το tag της αρχής εκτός από την ύπαρξη ενός συμβόλου '/', ανάμεσα στο '<' και το πρώτο γράμμα του tag. Για παράδειγμα, έχουμε αναφέρει ότι το tag `<html>` δείχνει την αρχή ενός εγγράφου html, το αντίστοιχο tag τέλους είναι το `</html>`. Ότι βρίσκεται ανάμεσα τους είναι τμήμα του εγγράφου html.

### 2.1.2.1 Tag <head> </head>

```
3 <html>
4 <head>
5 <title>Hotel Πτυχιακή </title>
6 <meta http-equiv="Content-Type" content="text/html; charset=utf-8">
7 <link rel="stylesheet" type="text/css" href="style.css" />
8 <script src="js/prototype.js" type="text/javascript"></script>
9 <script src="js/scriptaculous.js" type="text/javascript"></script>
10 <script src="js/simple-slide-show.js" type="text/javascript"></script>
11 </head>
```

Αυτά τα tags δείχνουν την αρχή και το τέλος της επικεφαλίδας ενός εγγράφου αρχείου HTML. Δεν είναι απαραίτητο στοιχείο ενός εγγράφου HTML, αλλά συνήθως αυξάνει την αποτελεσματικότητά του. Το πιο συνηθισμένο tag του <head> tag είναι το <title>. Παρέχει στην σελίδα με ένα τίτλο που θα εμφανίζεται στη γραμμή τίτλου του φυλλομετρητή. Το <title> tag είναι αρκετά απλό. Παρακάτω δίνεται η εμφάνιση ενός απλού εγγράφου HTML που χρησιμοποιεί του <title> tag.

```
<html>
<head>
<title> Αυτός είναι ο τίτλος μου </title>
</head>
</html>
```

#### <meta>

Τα Meta tags περιλαμβάνονται επίσης στην επικεφαλίδα ενός εγγράφου html. Αυτά τα tags περιλαμβάνουν πληροφορίες για τον ιστότοπο. Δεν είναι απαραίτητα αλλά ορισμένες μηχανές αναζήτησης τα χρησιμοποιούν για να καταλάβουν το περιεχόμενο του ιστότοπου. Κάθε τμήμα πληροφορίας περιέχεται σε ξεχωριστό tag. Τα κυριότερα τμήματα είναι τα keywords (λέξεις κλειδιά) και description (περιγραφή). Επίσης εδώ περιλαμβάνεται και ο χαρακτηρισμός της γλώσσας utf-8, ώστε να υπάρχει συμβατότητα με τους ελληνικούς χαρακτήρες.

```
<meta name=keywords content= "ενδιαφέροντα, προσωπικά στοιχεία">
<meta name=description content="Ελάτε να με γνωρίσετε από κοντά">
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

#### <Script>

Εδώ περιλαμβάνονται η θέση και ο τύπος javascript που χρησιμοποιούνται στην σελίδα.

```
<script src="js/simple-slide-show.js" type="text/javascript"></script>
```

Αυτά τα tags μπορούν να βρίσκονται οπουδήποτε μεταξύ των tags <head> και </head>.

### <link>

Εδώ μπορούμε στον φυλλομετρητή να συνδεθεί με συγκεκριμένο αντικείμενο στον διακομιστή ώστε να έχουμε συγκεκριμένο αποτέλεσμα.

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

Στην συγκεκριμένη περίπτωση λέμε στον φυλλομετρητή να χρησιμοποιήσει την CSS με τον τίτλο ="style.css"

### 2.1.2.2 Tag <body> </body>

Στο προηγούμενο παράδειγμα υπάρχει μία λευκή περιοχή στο μέρος όπου πρέπει να εμφανίζονται τα περιεχόμενα της σελίδας. Τα περιεχόμενα της σελίδας τοποθετούνται μέσα στο tag <body>. Ανάμεσα στα <body> και </body> τοποθετούνται το κείμενο, οι εικόνες και ότι άλλο θέλουμε να περιλαμβάνει η σελίδα.

### **Καθορισμός περιθωρίων**

Στο παράθυρο μπορούμε να ορίσουμε περιθώρια για πάνω και αριστερά. Αυτό σημαίνει ότι κείμενο και εικόνες δεν θα εμφανίζονται μέσα σε αυτό το περιθώριο. Η μέτρηση γίνεται σε pixels και τα tags τοποθετούνται σαν παράμετροι του body tag.

```
13 <body>
14   <div id="header">
15     <a href="index.html" id="logo"></a>
16     <span>Traditionally Royal hotel</span>
17     <ul id="menu">
18       <li><a href="index.php">Home</a></li>
19       <li><a href="about.php">About our Hotel</a></li>
20       <li><a href="messinia.php">Messinia</a></li>
21       <li><a href="photo.php">Photogallery</a></li>
22       <li><a href="contact.php">Contacts</a></li>
23     </ul>
24     <br />
25   </div>
26   <div id="content">
27     <div class="inside">
28       <div id="sidebar">
29         <div class="order">
30           <form name="arrival" method="POST" action="rooms.php">
31             <h3>Reservation</h3>
32             <table width="239" cellpadding="0" cellspacing="0">
33               <tr>
34                 <td class="first">
35                   Arrival Date<br />
36                   <select name="month">
37
```



## Παράδειγμα περιεχομένου <Body>

```
div id="header">
  <a href="index.html" id="logo">
  
  </a><span>Traditionally Royal hotel</span>
  <ul id="menu">
    <li><a href="index.php">Home</a></li>
    <li><a href="about.php">About our Hotel</a></li>
    <li><a href="messenia.php">Messinia</a></li>
    <li><a href="photo.php">Photogallery</a></li>
    <li><a href="contact.php">Contacts</a></li>
  </ul>
  <br />
</div>
```

```
238 | </div>
239 | <div id="footer">
240 |   <ul>
241 |     <li><a href="index.php">Home</a>|</li>
242 |     <li><a href="about.php">About our Hotel</a>|</li>
243 | <li><a href="messenia.php">Messinia</a></li>
244 |     <li><a href="photo.php">Photogallery</a>|</li>
245 |
246 |     <li><a href="contact.php">Contacts</a></li>
247 |   </ul>
248 |   <div id="copyright">
249 |     <br><br>
250 |     <span>Created by Panoyiotis Demis - Galakoutis George - Karagianidis Nick</span>
251 |     <span>©copy; Designed with big effort!</span>
252 |   </div>
253 | </div>
254 | </body>
255 | </html>
256 |
```

## Πολλαπλό διάστημα

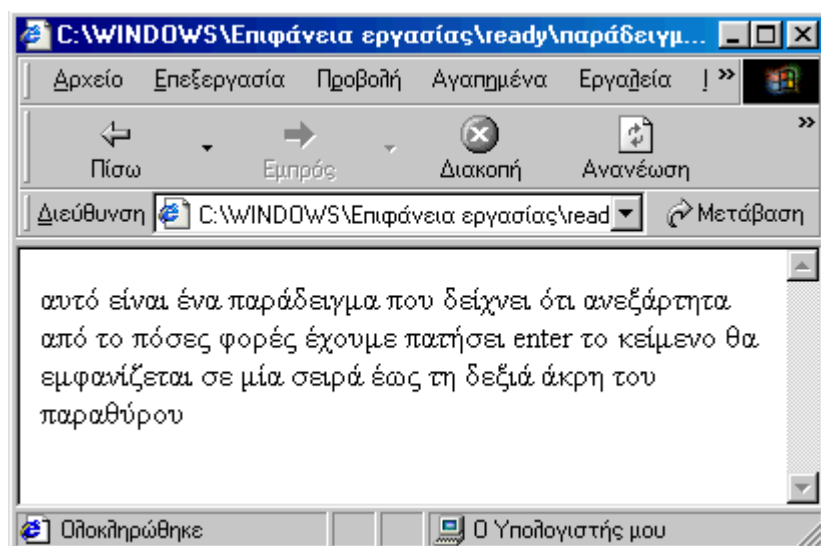
Τα προγράμματα περιήγησης αντιλαμβάνονται τις πολλές κενές σειρές σαν μία κενή σειρά. Αυτό καθιστά δύσκολη την τοποθέτηση κειμένου όπου ακριβώς θέλουμε.

### Παράδειγμα μορφοποίησης κειμένου<sup>αν.7</sup>

Όποιος έχει χρησιμοποιήσει επεξεργαστή κειμένου γνωρίζει τι αλλαγές μπορεί να επιφέρει με απλές εντολές. Πολλές από αυτές τις μορφοποιήσεις μπορούν να υλοποιηθούν και σε ένα έγγραφο html.

### tag <br>

Αν γράφοντας html θελήσουμε να αλλάζουμε συχνά σειρά και δώσουμε την παρακάτω μορφή στο κείμενο:



**<html>**

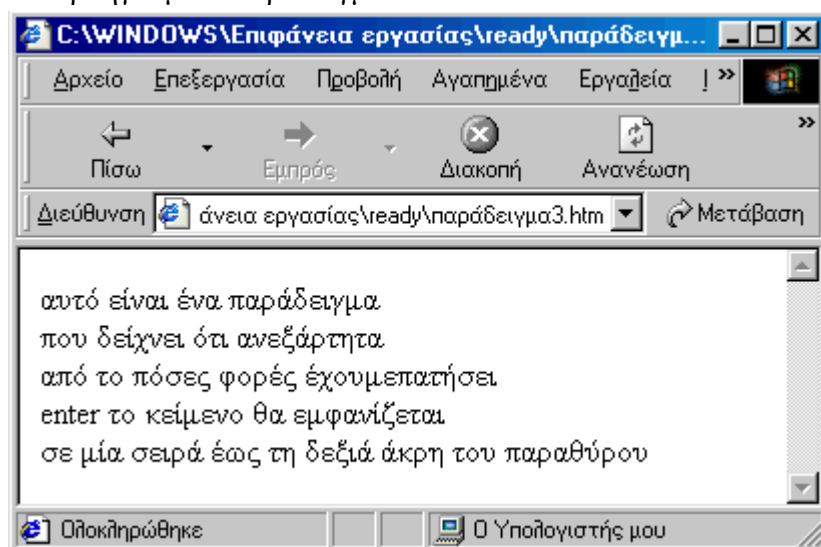
**<body>**

αυτό είναι ένα παράδειγμα που δείχνει ότι ανεξάρτητα από το πόσες φορές έχουμε πατήσει enter το κείμενο θα εμφανίζεται σε μία σειρά έως τη δεξιά άκρη του παραθύρου

**</body>**

**</html>**

Όταν θα δούμε την σελίδα μέσα από το πρόγραμμα περιήγησης θα δούμε την παραπάνω εικόνα. Το κείμενο δεν κατεβαίνει στην επόμενη σειρά. Για να πετύχουμε την αλλαγή σειρά χρησιμοποιούμε του tag `<br>`. Αυτό είναι ένα tag που δεν έχει tag τέλους. Αν το θέσουμε στο μέσο κειμένου, το κείμενο αλλάζει σειρά. Οπότε αν το χρησιμοποιήσουμε στο προηγούμενο παράδειγμα.



```
<html>
<body>
αυτό είναι ένα παράδειγμα <br>
που δείχνει ότι ανεξάρτητα <br>
από το πόσες φορές έχουμε πατήσει <br>
enter το κείμενο θα εμφανίζεται <br>
σε μία σειρά έως τη δεξιά άκρη του παραθύρου <br>
</body>
</html>
```

Παρατηρούμε ότι η χρήση του **<br>** φέρνει τα επιθυμητά αποτελέσματα. Το **<br>** δεν χρησιμοποιείτε μόνο για να για κείμενο. Χρησιμοποιώντας **<br>** μπορούμε να οδηγήσουμε τον φυλλομετρητή ώστε να «σπρώξει» άλλα αντικείμενα σε άλλη θέση. Είναι μια απλούστερη και λιγότερο πολύπλοκη μέθοδος ώστε να αποφύγουμε την όλη διαδικασία του spacing padding

### **Επικεφαλίδα <h1>/<h2>.**

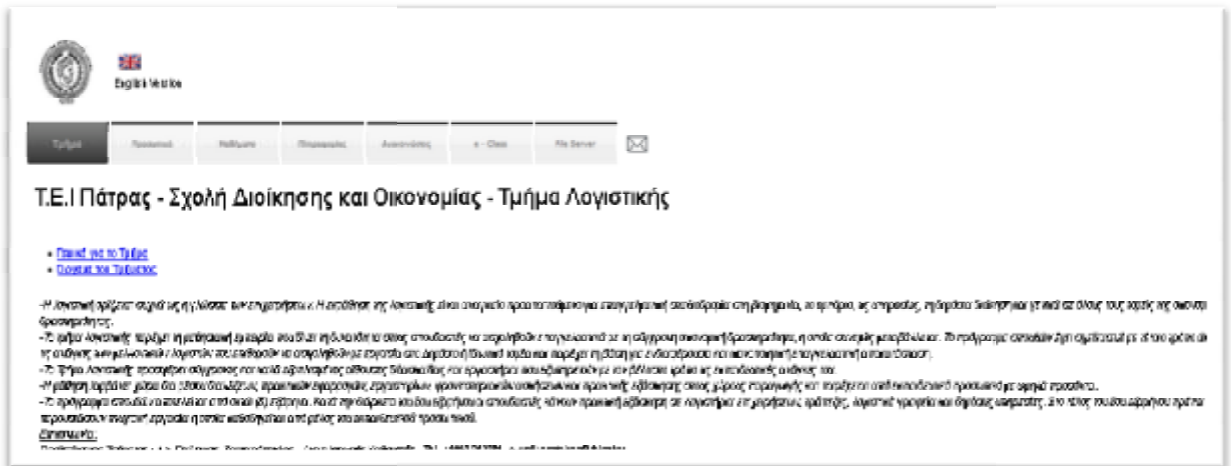
Η γλώσσα HTML παρέχει στον χρήστη με επικεφαλίδες έξι διαφορετικών μεγεθών. Αυτές οι κεφαλίδες ονομάζονται από h1 έως h6, με το h1 να είναι η μεγαλύτερη. Για την εισαγωγή μιας κεφαλίδας αρχικά πληκτρολογούμε **<h2>**, μετά την επικεφαλίδα και τέλος το tag τέλους **</h2>**. Οι επικεφαλίδες μπορούν να στοιχιστούν αριστερά, στο κέντρο, ή δεξιά χρησιμοποιώντας την παράμετρο "align=" στο tag αρχής. Αν χρησιμοποιήσουμε πολλές διαφορετικές επικεφαλίδες το μέγεθος τους πρέπει να μειώνεται.

### **Η tag παραγράφου <p> </p>**

Το tag **<p>** χρησιμοποιείται για την εισαγωγή παραγράφων. Μπορούμε να εισάγουμε ένα **<p>** πριν την αρχή της παραγράφου και ένα **</p>** μετά το τέλος. Μπορούμε να απλοποιήσουμε τη διαδικασία με το να εισάγουμε μόνο **<p>** ανάμεσα στις παραγράφους.

## **2.2 Τι είναι το CSS (Cascading Style Sheets)**

Η CSS (Cascading Style Sheets-Διαδοχικά Φύλλα Στυλ) ή ( αλληλουχία φύλλων στυλ ) είναι μια γλώσσα υπολογιστή που ανήκει στην κατηγορία των γλωσσών φύλλων στυλ που χρησιμοποιείται για τον έλεγχο της εμφάνισης ενός εγγράφου που έχει γραφτεί με μια γλώσσα σήμανσης. Χρησιμοποιείται δηλαδή για τον έλεγχο της εμφάνισης ενός εγγράφου που γράφτηκε στις γλώσσες HTML και XHTML, δηλαδή για τον έλεγχο της εμφάνισης μιας ιστοσελίδας και γενικότερα ενός ιστότοπου. Η CSS είναι μια γλώσσα υπολογιστή προορισμένη να αναπτύσσει στιλιστικά μια ιστοσελίδα δηλαδή να διαμορφώνει περισσότερα χαρακτηριστικά, χρώματα, στοίχιση και δίνει περισσότερες δυνατότητες σε σχέση με την html. Για μια όμορφη και καλοσχεδιασμένη ιστοσελίδα η χρήση της CSS κρίνεται ως απαραίτητη.<sup>3</sup>



Παραπάνω βλέπουμε την CSS στην πράξη. Στην πρώτη φωτογραφία βλέπουμε τον ιστότοπο του τμήματος λογιστικής με την CSS απενεργοποιημένη, ενώ στην δεύτερή φωτογραφία βλέπουμε τον ίδιο ιστότοπο με την CSS ενεργοποιημένη.

### 2.2.1 Η αλληλουχία εφαρμογής των φύλλων στύλ

Για ένα έγγραφο `px` `html` θα υπάρχουν παραπάνω από ένα φύλλα στυλ τα οποία περιέχουν δηλώσεις για την εμφάνιση ενός συγκεκριμένου στοιχείου. Το Φύλλο στύλ που εφαρμόζεται σε ένα έγγραφο μπορεί να προέρχεται από :

- το συγγραφέα μιας ιστοσελίδας
- το χρήστη του πλοηγού
- τον ίδιο τον πλοηγό, αν έχει το δικό του προκαθορισμένο φύλλο στυλ .

Συνεπώς για ένα `html` στοιχείο θα υπάρχουν παραπάνω από μια δηλώσεις που πιθανόν να είναι συγκρουόμενες. Το πρότυπο `css` για να επιλύσει παρόμοιες συγκρούσεις έχει καθορίσει μια αλληλουχία-σειρά στην οποία θα μπουν αυτές οι δηλώσεις και με βάση την οποία θα επιλεγεί `px` η δήλωση που είναι πρώτη στη σειρά.

Ο αλγόριθμος δημιουργίας αυτής της σειράς-αλληλουχίας είναι ο ακόλουθος:

1. Βρες όλες τις δηλώσεις που εφαρμόζονται στο στοιχείο που μας ενδιαφέρει. Οι δηλώσεις εφαρμόζονται στο στοιχείο αν ο επιλογέας του το επιλέξει (ταιριάζει με αυτό).
2. Ταξινόμησε με βάση τη σημασία (κανονική ή σημαντική) και προέλευση (συγγραφέας, χρήστη ή πλοηγός χρήστη). Με αύξουσα σειρά προτεραιότητας:
  1. Δηλώσεις πλοηγού χρήστη
  2. Κανονικές δηλώσεις χρήστη
  3. Κανονικές δηλώσεις συγγραφέα
  4. Σημαντικές δηλώσεις συγγραφέα
  5. Σημαντικές δηλώσεις χρήστη
3. Ταξινόμησε τις δηλώσεις ίδιας σημασίας και προέλευσης με κριτήριο την εξειδίκευση του επιλογέα: οι πιο εξειδικευμένοι επιλογείς υπερισχύουν των πιο γενικών. Τα ψευδό-στοιχεία και οι ψευδο-κλάσεις λογαριάζονται σαν κανονικά στοιχεία και κλάσεις αντίστοιχα.
4. Τέλος ταξινόμησε ανάλογα με τη σειρά καθορισμού: αν δύο δηλώσεις έχουν το ίδιο βάρος, προέλευση και εξειδίκευση, αυτή που προσδιορίστηκε τελευταία επικρατεί. Οι δηλώσεις σε εισαγόμενα φύλλα στυλ θεωρούνται ότι δηλώνονται πριν από τις δηλώσεις στο ίδιο το φύλλο στυλ.

Αφού λοιπόν προκύψει μια σειρά-αλληλουχία κανόνων εμφάνισης που αφορούν το ίδιο στοιχείο θα επιλεγεί προς εφαρμογή (για την αποφυγή συγκρούσεων) η δήλωση που θα είναι τελευταία στην σειρά που αναλύθηκε πιο πάνω.<sup>av.8</sup>

### 2.2.2 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 ιστότοπου 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 στον ιστότοπο. 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 το πρόγραμμα περιήγησης την πρώτη φορά που θα φορτώσει κάποια σελίδα του ιστότοπου μας το αποθηκεύει στην cache, οπότε δεν χρειάζεται να το κατεβάσει ξανά κάθε φορά που κατεβάζει ο χρήστης του κάποια άλλη σελίδα του ιστότοπου μας.

### 2.2.3 Βασικοί κανόνες σύνταξης

#### **Σχόλια**

Αν και η ανάγκη για σχόλια στο CSS δεν είναι τόσο επιτακτική όσο σε γλώσσες προγραμματισμού, μπορούμε να εισάγουμε σχόλια μέσα σε /\*...\*/  
Παράδειγμα: /\* Αυτό είναι ένα σχόλιο \*/

#### **Κανόνες CSS (CSS rules)**

Ένας «κανόνας» CSS αποτελείται από 2 μέρη: Τον επιλογέα (CSS selector) ο οποίος αφορά το ΤΙ θα μορφοποιηθεί και τις ιδιότητες (CSS properties) οι οποίες αφορούν το πώς αυτό θα μορφοποιηθεί. Η σύνταξη είναι η εξής:

```
επιλογέας  
{  
ιδιότητα1: τιμή1; ιδιότητα2: τιμή2; ιδιότητα3: τιμή3;  
...  
}
```

#### **Απλοί επιλογείς CSS (CSS selectors)**

Επιλέγουμε **ποια** στοιχεία θα μορφοποιήσουμε μέσω των επιλογέων CSS (CSS selectors), οι οποίοι ουσιαστικά αποτελούν κάποιους «**κανόνες**» ώστε να καταλάβει το πρόγραμμα περιήγησης **που** θέλουμε να εφαρμοστούν τα όσα γράφουμε κάθε φορά. Υπάρχουν αρκετά περίπλοκοι επιλογείς CSS στο specification του CSS, οι οποίοι προσφέρουν τεράστια ευελιξία, ωστόσο εδώ θα εξετάσουμε τους πιο βασικούς, και για λόγους απλούστευσης, και επειδή αρκετοί από τους πιο περίπλοκους δεν υποστηρίζονται από όλα τα προγράμματα περιήγησης.

\*Όταν ο επιλογέας είναι ένας χαρακτήρας αστερίσκου, τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε στοιχείο της σελίδας μας. Όπως είναι κατανοητό, συνήθως δεν είναι και πολύ χρήσιμος επιλογέας από μόνος του, και χρησιμοποιείται κυρίως σε συνδυασμό με άλλους.

#### **στοιχείο**

Όταν ο επιλογέας αποτελείται απλά από το όνομα ενός html tag, τότε οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS εφαρμόζονται σε κάθε τέτοιο στοιχείο html. Για παράδειγμα, ο επιλογέας p θα εφαρμοστεί σε οτιδήποτε στη σελίδα μας περιλαμβάνεται εντός των tags <p>...</p>, ο επιλογέας table θα εφαρμοστεί σε όλους τους πίνακες στη σελίδα μας, ο επιλογέας img θα αφορά όλες τις εικόνες στη σελίδα κοκ. Προφανώς όταν θέλουμε να εφαρμόσουμε κάποιες ιδιότητες CSS σε ολόκληρη τη σελίδα, χρησιμοποιούμε ως επιλογέα body μιας και όλο το ορατό τμήμα της σελίδας περιέχεται εντός των tags

```
<body>...</body>.
```

#### **όνομα\_κλάσης**

Όταν ο επιλογέας μας περιλαμβάνει μια τελεία (.) στην αρχή του, τότε το πρόγραμμα περιήγησης ψάχνει όσα στοιχεία στη σελίδα μας περιλαμβάνουν την ιδιότητα class και εφαρμόζει τις ιδιότητες που θα γράψουμε στον κανόνα CSS αυτό σε οποιοδήποτε στοιχείο

περιλαμβάνει την κλάση «όνομα\_κλάσης» στην ιδιότητα class του. Φυσικά ως όνομα\_κλάσης μπορούμε να γράψουμε οτιδήποτε αποτελείται από γράμματα, αριθμούς, παύλες και χαρακτήρες underscore (\_) και να ξεκινάει με γράμμα. Αξίζει να σημειωθεί ότι μπορεί το ίδιο στοιχείο να ανήκει σε περισσότερες από μια κλάσεις, διαχωρισμένες με κενά μέσα στην class html attribute του.

Πχ `<p class="emphasis bodytext">...</p>`.

Για παράδειγμα, ο παρακάτω κανόνας CSS:

```
.emphasis
{
color: red;
}
```

θα κάνει κόκκινα τα γράμματα και στο στοιχείο `<p class="emphasis">blah blah</p>`, και στο στοιχείο `<div class="emphasis otherclass">blah blah</div>` αλλά **όχι** στο στοιχείο `<h1 class="otherclass">blah blah</h1>`.

Οι κλάσεις γενικά χρησιμοποιούνται όταν θέλουμε να **ομαδοποιήσουμε** κάποια στοιχεία html για τα οποία δεν μπορούμε να βρούμε κάποιον άλλο επιλογέα που να αφορά **όλα αυτά** και **μόνον αυτά**, οπότε τους προσδίδουμε μια συγκεκριμένη κλάση, ώστε να μπορούμε στο CSS μας να αναφερθούμε **μόνο σε αυτά** και να τα μορφοποιήσουμε.

### στοιχείο.όνομα\_κλάσης

Αποτελεί ουσιαστικά συνδυασμό των δύο παραπάνω επιλογέων. Εφαρμόζεται σε όσα στοιχεία αποτελούνται από το html tag <στοιχείο> και ανήκουν στην κλάση όνομα\_κλάσης. Πχ ο επιλογέας p.emphasis εφαρμόζεται σε ο,τι περιέχεται σε tags της μορφής `<p class="emphasis">...</p>`. Ο επιλογέας αυτός είναι χρήσιμος όταν έχουμε πολλά **διαφορετικού τύπου** στοιχεία **με την ίδια κλάση** και επιθυμούμε να εφαρμόσουμε **διαφορετική μορφοποίηση** ανάλογα με τον **τύπο** του στοιχείου.

### #όνομα\_id

Όταν ο επιλογέας μας περιλαμβάνει ένα χαρακτήρα δέσσης (#) στην αρχή του, τότε ο φυλλομετρητής εφαρμόζει τις ιδιότητες που θα γράψουμε στο στοιχείο το οποίο περιλαμβάνει την ιδιότητα `id="όνομα_id"`. **Δεν πρέπει να υπάρχουν δύο (ή περισσότερα) στοιχεία στη σελίδα μας με το ίδιο id**. Τα ids διέπονται από τους ίδιους κανόνες ονοματολογίας με τις κλάσεις. Ουσιαστικά, ο,τι μπορούμε να κάνουμε με τα ids μπορούμε να το κάνουμε και με τη χρήση κλάσεων, απλά όταν το στοιχείο που θέλουμε να μορφοποιήσουμε είναι **μοναδικό**, είναι γενικά καλύτερο να χρησιμοποιούμε ids.

### στοιχείο[attribute="value"]

Αποτελεί ουσιαστικά μια «επέκταση» του επιλογέα στοιχείο που αναλύθηκε πρώτος. Ο εν λόγω επιλογέας κάνει τον σελιδομετρητή να εφαρμόζει τις ιδιότητες που θα γράψουμε σε αυτόν σε κάθε στοιχείο με tag <στοιχείο> το οποίο επιπροσθέτως έχει την τιμή value στην html ιδιότητα attribute. Παραδείγματος χάριν, ο επιλογέας `input[type="submit"]` αφορά

όλα τα κουμπιά υποβολής φόρμας που υπάρχουν στη σελίδα μας, χωρίς ωστόσο να εφαρμόζεται σε άλλα στοιχεία φορμών όπως τα πεδία κειμένου (στα οποία η ιδιότητα `type` είναι `text`). Άλλο ένα παράδειγμα: Έστω ότι θέλουμε να μορφοποιήσουμε μόνο όσους πίνακες στη σελίδα μας είναι κεντραρισμένοι. Αν χρησιμοποιούσαμε ως επιλογή `table`, τότε οι ιδιότητες που θα γράφαμε σε αυτόν τον επιλογή θα εφαρμόζονταν σε όλους τους πίνακες ανεξαιρέτως. Ενώ αν χρησιμοποιήσουμε τον επιλογή `table[align="center"]` τότε ο κανόνας CSS που θα γράψουμε θα εφαρμοστεί μόνο σε όσους πίνακες έχουν την ιδιότητα `align="center"`.

### **Ψευδό-κλάσεις και ψευδό-στοιχεία**

Ορισμένες φορές χρησιμοποιούμε τις λεγόμενες ψευδό-κλάσεις (`pseudo-classes`) ή ψευδό-στοιχεία (`pseudo-elements`), τα οποία μας επιτρέπουν να επιλέγουμε στοιχεία τα οποία δεν αποτελούν `html elements`, αλλά κομμάτια τους ή συγκεκριμένες καταστάσεις τους. Ουσιαστικά αποτελούν κάποιες λέξεις-κλειδιά που γράφουμε μετά από έναν επιλογή του τύπου στοιχείο και ξεκινούν με `:`. Αν όλα αυτά σας ακούγονται κάπως μπερδεμένα ή δυσνόητα, μην προβληματίζεστε, θα γίνουν πιο κατανοητά όταν εξετάσουμε τις συγκεκριμένες ψευδό-κλάσεις/στοιχεία που χρησιμοποιούνται συνηθέστερα παρακάτω.

#### **a:link**

Χρησιμοποιείται αποκλειστικά για στοιχεία `a` (δηλαδή ως εξής: `a:link`) και αφορά τους συνδέσμους που ο χρήστης δεν έχει ακόμη επισκεφθεί.

#### **a:visited**

Επίσης χρησιμοποιείται αποκλειστικά για στοιχεία `a` (δηλαδή ως εξής: `a:visited`) και αφορά τους συνδέσμους που ο χρήστης έχει επισκεφθεί.

#### **στοιχείο:active**

Αφορά τα στοιχεία τύπου `<στοιχείο>` τη στιγμή που ο χρήστης έχει πατημένο το ποντίκι πάνω σε αυτά. Πχ ο επιλογέας `a:active` εφαρμόζεται σε συνδέσμους την ώρα που ο χρήστης έχει πατημένο το ποντίκι πάνω τους.

#### **στοιχείο:hover**

Από τις πιο συχνά χρησιμοποιούμενες ψευδό-κλάσεις. Αφορά τα στοιχεία τύπου `<στοιχείο>` τη στιγμή που ο χρήστης έχει το δείκτη του ποντικιού πάνω σε κάποιο από αυτά (χωρίς να πατάει κάποιο πλήκτρο). Μπορεί να μας βοηθήσει να δημιουργήσουμε διάφορα όμορφα εφέ, τα οποία παλιότερα ήταν εφικτά μόνο με `javascript`.

#### **στοιχείο:focus**

Χρησιμοποιείται κυρίως για στοιχεία φορμών και εφαρμόζεται στα στοιχεία τύπου `<στοιχείο>` που εκείνη τη στιγμή έχουν «`focus`», παραδείγματος χάριν, ένα πεδίο κειμένου στο οποίο ο χρήστης έκανε κλικ για να εισάγει κείμενο.

#### **στοιχείο:first-letter**

Αφορά το πρώτο γράμμα του κειμένου εντός κάποιου στοιχείου τύπου `<στοιχείο>`. Το ψευδό-στοιχείο αυτό μπορεί να μας βοηθήσει να δημιουργήσουμε αρχιγράμματα. Πχ ο επιλογέας `p:first-letter` αφορά το πρώτο γράμμα κάθε παραγράφου.

#### **στοιχείο:first-line**



Ψευδό-στοιχείο παρόμοιο με το παραπάνω, μόνο που αντί να αφορά μόνο το πρώτο γράμμα του κειμένου μέσα στο στοιχείο τύπου <στοιχείο>, αφορά ολόκληρη την πρώτη γραμμή.

Αν και δεν έχει ιδιαίτερη πρακτική χρησιμότητα, αξίζει να σημειωθεί ότι τα 5 πρώτα θεωρούνται ψευδό-κλάσεις (pseudo-classes) ενώ τα 2 τελευταία ψευδό-στοιχεία (pseudo-elements).

## Σύνθετοι επιλογείς CSS (CSS selectors)

Πολλές φορές μπορούμε να συνδυάσουμε σε έναν επιλογέα περισσότερους από έναν υπό-επιλογείς, βάσει συγκεκριμένων κανόνων σύνταξης, κάτι που μας προσφέρει μεγαλύτερη ευελιξία και εξοικονόμηση χρόνου. Οι πιο βασικοί και συχνά χρησιμοποιούμενοι τρόποι συνδυασμού επιλογέων παρουσιάζονται παρακάτω:

### επιλογέας<sub>1</sub>, επιλογέας<sub>2</sub>, επιλογέας<sub>3</sub>, ...

Οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS, θα εφαρμοστούν σε κάθε στοιχείο που πληροί τις προϋποθέσεις επιλογής είτε για τον επιλογέα<sub>1</sub>, είτε για τον επιλογέα<sub>2</sub>, είτε για τον επιλογέα<sub>3</sub> κοκ. Παραδείγματος χάριν, ο επιλογέας `input[type="text"]`, `textarea` θα εφαρμοστεί σε κάθε πεδίο κειμένου που υπάρχει στη σελίδα μας, είτε είναι για εισαγωγή κειμένου μιας γραμμής (`<input type="text" />`) είτε είναι πολλών γραμμών (`<textarea></textarea>`).

### επιλογέας<sub>1</sub> επιλογέας<sub>2</sub> επιλογέας<sub>3</sub> ... επιλογέας<sub>n</sub>

Οι ιδιότητες που θα γράψουμε σε αυτόν τον κανόνα CSS θα εφαρμοστούν σε στοιχεία που πληρούν τις προϋποθέσεις του επιλογέα και επιπροσθέτως περιέχονται μέσα σε κάποιο στοιχείο που πληροί τις προϋποθέσεις του επιλογέα<sub>n-1</sub> το οποίο βρίσκεται μέσα σε κάποιο στοιχείο που πληροί τις προϋποθέσεις του επιλογέα<sub>n-2</sub> κοκ. Παραδείγματος χάριν ο επιλογέας `p img` θα εφαρμοστεί σε όσες εικόνες περιέχονται μέσα σε tags `<p>...</p>`.

## 2.2.4 Πως εισάγουμε CSS στη σελίδα μας;

Μπορούμε να εισάγουμε CSS στη σελίδα μας με τρεις διαφορετικούς τρόπους, αναλόγως την περίπτωση. Οι τρόποι αυτοί, κατά σειρά φθίνουσα προτεραιότητας, είναι<sup>av.9</sup>:

**Inline CSS:** Αν επιθυμούμε να μορφοποιήσουμε ένα στοιχείο μόνο, και δεν πρόκειται να χρειαστούμε αυτό το είδος μορφοποίησης για τίποτε άλλο στον ιστότοπο, μπορούμε να γράψουμε «χύμα» (δηλ. χωρίς το κομμάτι του CSS selector και χωρίς αγκύλες) CSS properties μέσα στο attribute `style`, το οποίο το διαθέτει σχεδόν κάθε στοιχείο html. Παραδείγματος χάριν, για να κάνουμε μια συγκεκριμένη παράγραφο κόκκινη, μπορούμε να προσθέσουμε `style="color:red;"` στο `<p>` tag (ολοκληρωμένα: `<p style="color:red;">`).

Αξίζει να σημειωθεί, ότι αν κάποιες από τις ιδιότητες που θα γράψουμε εντός της attribute style ενός στοιχείου «συγκρούονται» με κανόνες CSS που έχουν οριστεί για αυτό αλλού και το αφορούν, τότε ο φυλλομετρητής θα επιλέξει να εφαρμόσει αυτά που γράψαμε εντός του attribute style, ως πιο συγκεκριμένα για το στοιχείο αυτό.

**CSS για μια συγκεκριμένη σελίδα:** Πολλές φορές, μπορεί να θέλουμε να εφαρμόσουμε κάποιους κανόνες CSS μόνο για τη συγκεκριμένη σελίδα και όχι για όλο τον ιστότοπο (ή να διαφοροποιήσουμε σε κάποιες ιδιότητες τους υπάρχοντες). 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 μέσα σε <style> tags (με την html attribute type τους σε text/css). Παραδείγματος χάριν, για να κάνουμε το χρώμα γραμμάτων μιας συγκεκριμένης σελίδας γκρι, θα γράφαμε στο head της:

```
<style
type="text/css">
body
{
color:gray;
}
</style>
```

Αξίζει να σημειωθεί ότι συνήθως όταν ξεκινάμε την ανάπτυξη ενός ιστότοπου, είναι πιο βολικό να χρησιμοποιήσουμε αυτό τον τρόπο αρχικά, μιας και πρέπει να επεξεργαστούμε μόνο ένα αρχείο αντί για δύο. Όταν τελειώνουμε την πρώτη σελίδα, συνήθως μεταφέρουμε το CSS που έχουμε γράψει σε εξωτερικό αρχείο (βλ. παρακάτω) ώστε να μπορούμε να χρησιμοποιήσουμε τους ίδιους κανόνες CSS και στις άλλες σελίδες του ιστότοπου μας, χωρίς να πρέπει φυσικά να τους κάνουμε αντιγραφή - επικόλληση σε κάθε σελίδα.<sup>4</sup>

**Εξωτερικό αρχείο 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. 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 ιστότοπου μας γράφουμε <link href="main.css" type="text/css" /> (αν το όνομα του αρχείου CSS είναι main.css, αλλιώς προφανώς γράφουμε το σωστό όνομα αρχείου). Αξίζει να σημειωθεί ότι μπορούμε να έχουμε πολλά αρχεία css στην ίδια σελίδα, και μεγαλύτερη προτεραιότητα έχει πάντα αυτό που έχει γραφτεί τελευταίο.<sup>αν4</sup>

## 2.2.5 Βασικές ιδιότητες CSS (CSS properties)

### Ιδιότητες που αφορούν μορφοποίηση κειμένου

#### **color**

Αφορά το χρώμα του κειμένου, αλλά αν δεν οριστεί χρώμα περιγράμματος (μέσω της ιδιότητας border-color), ο σελιδομετρητής χρησιμοποιεί αυτό που ορίστηκε στην ιδιότητα color. 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. Οι τιμές που δέχεται μπορούν να είναι εκφρασμένες σε ένα μεγάλο πλήθος μονάδων μεγέθους, από τις οποίες οι πιο ευρέως διαδεδομένες είναι τα pixels (πχ font-size: 12px;) και οι 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. Ουσιαστικά μέσω αυτής ορίζουμε τη γραμματοσειρά του κειμένου, απλά μας επιτρέπει να ορίσουμε και εναλλακτικές επιλογές, ώστε αν η γραμματοσειρά που ορίσαμε δεν υπάρχει στον υπολογιστή του χρήστη, να μην μας καταστρέψει ο σελιδομετρητής την εμφάνιση της σελίδας επιλέγοντας όποια αυτός νομίζει, απλά την επόμενη επιλογή μας. Αν καμία από όσες γραμματοσειρές ορίσαμε δεν υπάρχει στον υπολογιστή του χρήστη, τότε ο φυλλομετρητής επιλέγει κάποια που να ανήκει στη γενική οικογένεια γραμματοσειρών (generic font-family) που θα ορίσουμε στο τέλος με τη κατάλληλη λέξη κλειδί (keyword). Τέτοια keywords είναι τα εξής:

serif: Γραμματοσειρές με «πατούρες» όπως η Georgia.

sans-serif: Γραμματοσειρές χωρίς «πατούρες» όπως η Trebuchet MS.

monospace: Γραμματοσειρές όπου το κάθε γράμμα καταλαμβάνει ίσο πλάτος με τα υπόλοιπα, όπως η Courier New.

cursive: Καλλιγραφικές γραμματοσειρές όπως η Monotype Corsiva

fantasy: «Διακοσμητικές» γραμματοσειρές.

Οι τελευταίες 2 γενικές οικογένειες γραμματοσειρών χρησιμοποιούνται πολύ σπάνια.

Παράδειγμα χρήσης της ιδιότητας: font-family: Calibri, Trebuchet MS, Verdana, sans-serif;

## font-style

Σε αντίθεση με αυτό που θα περίμενε κανείς από μια ιδιότητα με αυτό το όνομα, δηλαδή την εφαρμογή πολλών και διαφόρων «εφέ» στο κείμενο, στην πραγματικότητα αφορά μόνο την περίπτωση όπου το κείμενο θα είναι πλάγιο. Για άλλα εφέ, χρησιμοποιούνται άλλες ιδιότητες, οι οποίες θα εξεταστούν παρακάτω. Οι πιθανές τιμές της είναι normal, italic και oblique. Οι δύο τελευταίες συνήθως κάνουν το ίδιο πράγμα (και είναι προτιμότερη η χρήση της italic μιας και υποστηρίζεται και από Internet Explorer 3, ενώ η υποστήριξη για την oblique ξεκινά από Internet Explorer 4), ενώ είναι προφανής η λειτουργία της τιμής normal.

## font-weight

Αφορά το «βάρος» της γραμματοσειράς και στην πράξη χρησιμοποιείται για να ορίσει αν το κείμενο μας θα είναι **έντονο** ή όχι, μιας και οι περισσότερες γραμματοσειρές που χρησιμοποιούνται στο παγκόσμιο ιστό διατίθενται μόνο σε δύο βάρη: Κανονικό και έντονο, σε αντίθεση με πιο εξειδικευμένες γραμματοσειρές που χρησιμοποιούνται από γραφίστες, οι οποίες πολλές φορές διατίθενται σε διάφορα βάρη. Οπότε οι τιμές που συνήθως χρησιμοποιούνται σε αυτή την ιδιότητα είναι οι normal και bold, η λειτουργία των οποίων είναι προφανής.

## 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. Δεν πρέπει να χρησιμοποιείται, παρά μόνο σε εξαιρετικά σπάνιες περιπτώσεις, μιας και η εμπειρία έχει δείξει ότι είναι ξαιρετικά κουραστικό για τον αναγνώστη και αφαιρεί επαγγελματικότητα από το design της ιστοσελίδας.

Οι παραπάνω τιμές μπορούν να χρησιμοποιηθούν και συνδυαστικά, όταν επιθυμούμε να εφαρμόσουμε πάνω από ένα τέτοιο εφέ στο κείμενο μας. Παραδείγματος χάριν text-decoration: underline overline;

## 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 από όσες αναφέρθηκαν παραπάνω, καθώς και κάποιες άλλες που δεν αναφέρθηκαν). Αν κάποια ιδιότητα δεν συμπεριληφθεί, ο φυλλομετρητής θα «υποθέσει» την προεπιλεγμένη τιμή της.

Παραδείγματα:

```
font: 12px Calibri, Trebuchet MS, Verdana, sans-serif;
```

```
font: italic bold 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), παραδείγματος χάριν background-image:url(<http://www.e-steki.gr/logo.gif>);. Γενικά όταν εισάγουμε μια διεύθυνση URL ως τιμή (ή τμήμα τιμής) κάποιας ιδιότητας CSS, πάντα τη γράφουμε εντός των παρενθέσεων του url(). Φυσικά μπορούμε να εισάγουμε και σχετικά URLs, πχ background-image:url([logo.gif](#));.

```
4 }
5 img(border:0px)
6 html{
7 width:100%;
8 height:100%;
9 background:url(images/bg.jpg) top left repeat-x;
10 background-color:#F7E6C0;
11 }
12 body{
13 width:1000px;
14 margin:0 auto;
15 }
16 #header{
17 background:url(images/top_bg.jpg) top left no-repeat;
18 padding:33px 0 0 0;
19 text-align:center;
20 }
```

### **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

Τα παραπάνω μπορούμε να τα χρησιμοποιήσουμε και συνδυαστικά (και συνήθως αυτό γίνεται), όπως στα

παρακάτω παραδείγματα:

background-position: left

bottom; background-

position: top right;

background-position: center

top;

Αντί για τους παραπάνω λέξεις κλειδιά, μπορούμε να ορίσουμε τη θέση της εικόνας φόντου και αριθμητικά, είτε με ποσοστό ή με κάποια μονάδα μεγέθους, όπως στα παρακάτω

παραδείγματα:

background-position: 10% 25%;

background-position: 60px 100px;

αν τέτοιες τιμές χρησιμοποιούνται πολύ πιο σπάνια.

Η ιδιότητα αυτή χρησιμοποιείται συνήθως όταν έχουμε ορίσει να μην επαναλαμβάνεται η εικόνα φόντου (μέσω της ιδιότητας background-repeat). Κάποιοι φυλλομετρητές υποστηρίζουν τη χωριστή δήλωση θέσης της εικόνας στον κάθετο και τον οριζόντιο άξονα, μέσω δύο ιδιοτήτων με όνομα background-position-x και background-position-y, αλλά καλύτερα να αποφεύγεται η χρήση αυτών των ιδιοτήτων, μιας και δεν υποστηρίζονται από όλους τους σελιδομετρητές.

## background

Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να ορίσουμε με μία μόνο ιδιότητα όλες τις παραπάνω

(όπως είναι η ιδιότητα font για τα χαρακτηριστικά κειμένου).

Παραδείγματα χρήσης: background: #CCCCCC

url(background.gif) no-repeat top center; background: #CCCCCC

url(background.gif);

## Ιδιότητες που αφορούν το περίγραμμα

### border-color

Ρυθμίζει το χρώμα περιγράμματος. Το χρώμα μπορεί να γραφεί σε οποιαδήποτε από τις μορφές που περιγράφηκαν για την ιδιότητα color. Αν δεν οριστεί αυτή η ιδιότητα, χρησιμοποιείται το χρώμα που ορίστηκε στην ιδιότητα color.

### border-width

Ρυθμίζει το πάχος του περιγράμματος σε κάποια από τις μονάδες μέτρησης που μπορούν να χρησιμοποιηθούν στο CSS, συνηθέστερα σε pixels.

Παράδειγμα: border-width: 10px;

### border-style

Ορίζει το στυλ του περιγράμματος. Οι τιμές που χρησιμοποιούνται συνήθως για την ιδιότητα αυτή 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. Αξίζει να σημειωθεί ότι τα αποτελέσματα της χρήσης της τιμής αυτής διαφέρουν ελαφρώς ανά τους φυλλομετρητές. Για παράδειγμα ο Internet Explorer εμφανίζει τα dotted περιγράμματα ως αποτελούμενα από μικρούς κύκλους, ενώ ο Mozilla Firefox τα εμφανίζει ως αποτελούμενα από μικρά τετράγωνα.

**double:** Σαν την τιμή solid, μόνο που δημιουργεί δύο περιγράμματα, η απόσταση και το πάχος των οποίων δεν μπορούν να ρυθμιστούν χωριστά αλλά ρυθμίζονται αυτόματα ώστε το πάχος τους να είναι συνολικά όσο η τιμή που ορίσαμε στην ιδιότητα border-width. Συνήθως χρησιμοποιείται σε συνδυασμό με πάχος περιγράμματος 3px, για το οποίο δίνει

ένα αρκετά καλαίσθητο αποτέλεσμα.

Υπάρχουν και οι τιμές **groove**, **ridge**, **inset**, **outset** οι οποίες δημιουργούν διαφόρων ειδών «ψευδό-3D» περιγράμματα, οι οποίες καλύτερα να αποφεύγονται για λόγους αισθητικής.

## border

Η ιδιότητα αυτή αποτελεί ουσιαστικά συντόμευση για να ορίσουμε με μία μόνο ιδιότητα όλες τις παραπάνω

(όπως είναι η ιδιότητα font για τα χαρακτηριστικά κειμένου).

Παραδείγματα χρήσης: border: 10px solid #999999;

border: 2px dashed;

```
1  *{
2  padding: 0px;
3  margin: 0px;
4  }
5  img{border: 0px}
6  html{
7  width: 100%;
8  height: 100%;
9  background: url(images/bg.jpg) top left repeat-x;
10 background-color: #F7E6C0;
11 }
```

**Προσοχή:** Αν δεν ορίσουμε κάποια επιμέρους ιδιότητα όταν χρησιμοποιούμε την συγκεντρωτική ιδιότητα border, ο σελιδομετρητής «υποθέτει» την προεπιλεγμένη τιμή για αυτή, όχι ότι δεν ορίστηκε.

Αυτό σημαίνει ότι αν γράψουμε για παράδειγμα:

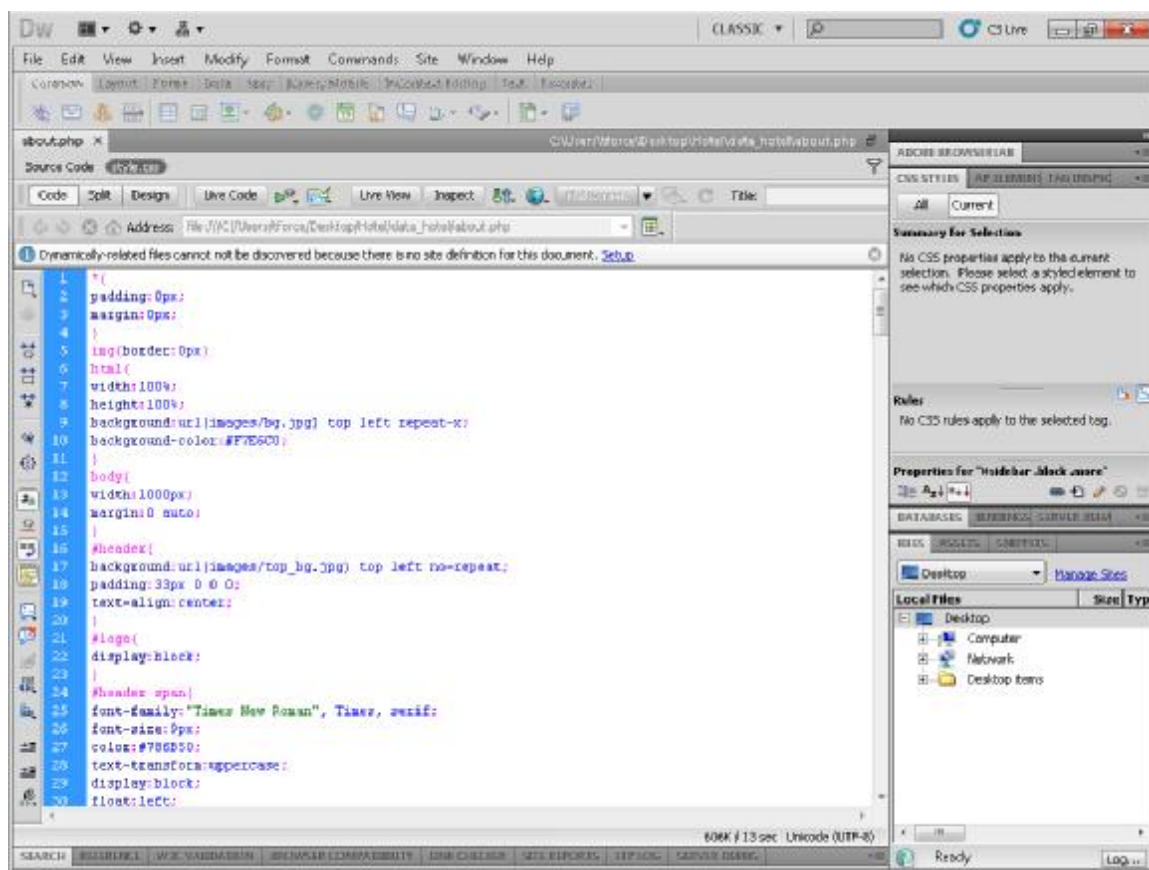
```
border-color: orange; border: 2px dashed;
```

τότε το περίγραμμα μας δεν θα είναι πορτοκαλί, τουλάχιστον όχι σε όλους τους σελιδομετρητές (εκτός αν είναι πορτοκαλί και η ιδιότητα color), μιας και αφού δεν ορίσαμε στη border και την border-color, τότε ο σελιδομετρητής «υποθέτει» την προεπιλεγμένη τιμή για αυτή την ιδιότητα, και μιας και η border βρίσκεται κάτω από την border-color, έχει μεγαλύτερη προτεραιότητα, οπότε επικρατεί για τη border-color η προεπιλεγμένη τιμή.

## Τι κάνουμε αν θέλουμε να ορίσουμε διαφορετικό πάχος/χρώμα/στυλ περιγράμματος για το πάνω/δεξί/κάτω/αριστερό περίγραμμα;

Κάθε μια από τις ιδιότητες που αναφέρθηκαν παραπάνω, αποτελείται από τέσσερις επιμέρους ιδιότητες, οι οποίες μας επιτρέπουν να ορίσουμε διαφορετικές τιμές για την ιδιότητα αυτή όσον αφορά το πάνω/δεξί/κάτω/αριστερό περίγραμμα.





Κάθε ιδιότητα της μορφής border-κάτι αντιστοιχεί σε τέσσερις ιδιότητες της μορφής border-top-κάτι, border-right-κάτι, border-bottom-κάτι, border-left-κάτι. Επίσης, κάθε ιδιότητα από αυτές που αναφέρθηκαν παραπάνω δέχεται και τιμές της μορφής [top] [right] [bottom] [left] (ένας εύκολος μνημονικός κανόνας για τη σειρά των τιμών είναι ότι η σειρά αυτή είναι σύμφωνη με την πορεία των δεικτών του ρολογιού) ή της μορφής [top,bottom] [right,left] αντί για μόνο μια τιμή που θα εφαρμοστεί και για τις τέσσερις πλευρές του περιγράμματος. Για παράδειγμα, για να ορίσουμε ένα γκρι συμπαγές περίγραμμα με πάχος αριστερά 1px, δεξιά 2px, πάνω 3px και κάτω 4px θα μπορούσαμε να γράψουμε είτε:

```
border-left-width: 1px;
border-right-width: 2px;
border-top-width: 3px;
border-bottom-width: 4px;
border-style: gray; border-style: solid;
είτε:
border-width: 3px 2px 4px 1px;
border-style: gray;
border-style: solid;
```

Προφανώς το δεύτερο είναι πιο σύντομο, αλλά το πρώτο είναι πιο εύκολο στην κατανόηση. Η επιλογή είναι δική μας.

## Διάφορες χρήσιμες ιδιότητες

## Padding

```
61 }
62 .inside{
63 background:url(images/main_bg.jpg) top left no-repeat;
64 padding:30px 0 0 0;
65 float:left;
66 }
```

Ορίζει το κενό που θα υπάρχει μεταξύ των ορίων ενός στοιχείου και των περιεχομένων του. Είναι πολύ σημαντικό να ορίζουμε padding σε στοιχεία στα οποία έχουμε ορίσει κάποιο περίγραμμα, ώστε να μην «κολλάνε» τα περιεχόμενα τους με το περίγραμμα, κάτι που φαίνεται ιδιαίτερα άσχημο και ερασιτεχνικό. Επίσης, καλό είναι να μην είστε ιδιαίτερα φειδωλοί με το padding. Περισσότερο padding δίνει πιο επαγγελματική εμφάνιση (μέχρι κάποιων ορίων φυσικά). Για να ορίσουμε διαφορετικό padding ανά πλευρά, μπορούμε ομοίως με το border, είτε να χρησιμοποιήσουμε τις ιδιότητες padding-top, padding-right, padding-bottom, padding-left, είτε να εισάγουμε και τις τέσσερις τιμές στην ιδιότητα padding με τη σειρά [top] [right].

[bottom] [left], είτε [top,bottom]  
[right,left]. Παραδείγματα: padding:  
8px;

padding: 2px  
6px; padding:  
0px 6px 6px  
6px;

## margin

```
157 }
158 #sidebar .block a:visited{text-decoration:underline}
159 #sidebar .block a:hover{text-decoration:none}
160 #sidebar .block .more{
161 margin:0 145px 0 26px;
162 / margin:0 145px 0 13px;
163 }
```

Το αντίθετο ουσιαστικά του padding. Ορίζει τον χώρο μεταξύ των ορίων ενός στοιχείου και όσων το περιβάλλουν. Είναι ιδιαίτερα σημαντικό να ορίζουμε margin σε εικόνες, ώστε να έχουν απόσταση από τα περιεχόμενα τους, μιας και είναι ιδιαίτερα αντιαισθητικό να «κολλάνε» με το κείμενο. Φυσικά και εδώ μπορούμε να χρησιμοποιήσουμε είτε τις ιδιότητες margin-top, margin-right, margin-bottom, margin-left για να ορίσουμε διαφορετικές τιμές margin ανά πλευρά, είτε να εισάγουμε τις τέσσερις

διαφορετικές τιμές

margin με τη σειρά [top] [right] [bottom] [left], ή [top,bottom] [right,left]. Παραδείγματα: margin: 4px;

margin: 2px 4px;  
margin: 0px 8px 2px 2px;

### **width, height**

Όπως είναι προφανές, οι δύο αυτές ιδιότητες ορίζουν το πλάτος και το ύψος ενός στοιχείου, είτε σε ποσοστό

(το οποίο υπολογίζεται βάσει του στοιχείου που το περιέχει), είτε σε κάποια μονάδα μήκους. Παραδείγματα:

width: 100px;  
width: 90%;  
height: 300px;  
height: 100%;

Υπάρχουν και οι ιδιότητες min-width, min-height, max-width, max-height οι οποίες ορίζουν τα όρια στα οποία μπορούν να κινούνται οι διαστάσεις ενός στοιχείου, όταν δεν θέλουμε να ορίσουμε συγκεκριμένες διαστάσεις, αλλά καλό είναι να μην χρησιμοποιούνται ακόμα, όσο βολικές κι αν φαίνονται, μιας και δυστυχώς η υποστήριξη τους από τους υπάρχοντες σελιδομετρητές είναι περιορισμένη.

### **float**

Η ιδιότητα αυτή είναι ανεκτίμητης χρησιμότητας στο όταν χρησιμοποιούμε CSS και για το layout της σελίδας μας, αν και είναι η δυσκολότερη στην κατανόηση από όσες αναφέρθηκαν. 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. Μια συνηθισμένη χρήση της ιδιότητας είναι στις εικόνες που συνοδεύουν ένα άρθρο. Πολλές φορές χρησιμοποιούμε μετά από floated στοιχεία κάποιο στοιχείο (συνήθως ένα άδειο div) με την ιδιότητα clear: both; για να «καθαρίσει» τα floats για το μετέπειτα περιεχόμενο.

## [2.3 Τι Είναι η Php \(εν συντομία\)](#)



Η γλώσσα προγραμματισμού **Php** (Hypertext PreProcessor) έχει το ιδιαίτερο χαρακτηριστικό ότι ο κώδικάς της πρώτα μεταγλωττίζεται στον διακομιστή και μετά

φορτώνεται σαν ένα κανονικό html έγγραφο, χωρίς ο χρήστης να είναι σε θέση να δει τον αρχικό κώδικα. Με την τεχνολογία της Php μπορούμε να δημιουργήσουμε δυναμικές ιστοσελίδες (dynamic web pages).

Ανταγωνιστικές της τεχνολογίας Php είναι η τεχνολογία **ASP** (Active Server Pages) της εταιρείας Microsoft καθώς και η τεχνολογία **CFML** (ColdFusion Markup Language). Θα δούμε ένα πολύ απλό αρχικό παράδειγμα.<sup>av.10</sup>

### 2.3.1 Το Πρώτο Παράδειγμα σε Php

Γράφουμε τον παρακάτω κώδικα της Php σ' ένα αρχείο κειμένου με το Σημειωματάριο (Notepad) των Windows και τον αποθηκεύουμε με το όνομα **Php01.php** :

```
<?php  
echo "<html><body>";  
echo "Hello";  
echo "</body></html>";  
?>
```

Αποθηκεύουμε το παραπάνω αρχείο σ' έναν διακομιστή και αν αυτός υποστηρίζει την τεχνολογία Php, θα δούμε το μήνυμα **Hello** στην οθόνη μας, ενώ αν επιχειρήσουμε να δούμε τον πηγαίο κώδικα (source code) θα υπάρχουν μόνο τα tags της HTML, χωρίς τις σημάνσεις της Php.

Οι ετικέτες (tags) **<?php** και **?>** χρησιμοποιούνται για να δηλώσουμε ένα μπλοκ κώδικα Php. Για να είναι σε θέση ο διακομιστής να επεξεργαστεί τον κώδικα της Php, πρέπει να του πούμε πού αρχίζει και πού τελειώνει ένα μπλοκ κώδικα Php. Μπορούμε να τοποθετήσουμε κώδικα της HTML μέσα στον κώδικα της Php, ο οποίος θα εκτελεσθεί κανονικά.

Το παραπάνω παράδειγμα θα μπορούσε να είχε γραφεί και ως εξής :

```
<html><body>  
<?php  
echo "Hello";  
?>  
</body></html>
```

Η εντολή echo χρησιμοποιείται για να στείλουμε ένα κείμενο (string) στον φυλλομετρητή (browser). Όλες οι εντολές της Php πρέπει να τελειώνουν με τον χαρακτήρα ; και μια εντολή μπορεί να επεκταθεί και σε περισσότερες από μία γραμμές. Όταν ο φυλλομετρητής ενός χρήστη ζητήσει μια σελίδα Php σαν την παραπάνω, ο διακομιστής θα την επεξεργαστεί, θα μετατρέψει τον κώδικα Php σε καθαρή HTML μορφή και έτσι ο χρήστης δεν θα μπορέσει να δει τον αρχικό κώδικα Php.<sup>av.10</sup>



## 2.3.2 Το πρώτο πρόγραμμα σε PHP

### PHP στην γραμμή εντολών



Αν και ο σκοπός του βιβλίου αυτού είναι ο σχεδιασμός δυναμικών ιστοσελίδων με PHP και MySQL, θα δείξουμε στο πρώτο παράδειγμα πως η γλώσσα PHP μπορεί να χρησιμοποιηθεί αυτόνομα για σχεδιασμών σεναρίων τα οποία εκτελούνται τοπικά από τον υπολογιστή σε επίπεδο γραμμής εντολών.

#### Καλημέρα κόσμε

Ένα πρόγραμμα το οποίο καλωσορίζει το χρήστη είναι συνήθως το πρώτο πρόγραμμα που μαθαίνει ένας προγραμματιστής σε μια νέα γλώσσα προγραμματισμού. Στο παρακάτω παράδειγμα τυπώνεται στην προκαθορισμένη έξοδο το μήνυμα "Καλημέρα κόσμε!". Για να εκτελέσουμε το παραπάνω σενάριο το αποθηκεύουμε στο αρχείο `hello.php` και εκτελούμε από την γραμμή εντολών το `php hello.php`.



 <b>Κώδικας:</b>
<pre>&lt;?php echo "Καλημέρα κόσμε!"; ?&gt;</pre>
 <b>Έξοδος:</b>
Καλημέρα κόσμε!

#### Έξοδος με print

Εναλλακτικά μπορεί να χρησιμοποιηθεί η <code>print</code> .
 <b>Κώδικας:</b>
<pre>&lt;?php print "Καλημέρα κόσμε!"; ?&gt;</pre>
 <b>Έξοδος:</b>
Καλημέρα κόσμε!

#### Είσοδος/έξοδος στη γραμμή εντολών

Για να εκτελέσουμε το παραπάνω σενάριο το αποθηκεύουμε στο αρχείο `helloIO.php` και εκτελούμε από την γραμμή εντολών το `php helloIO.php`.

 <b>Κώδικας:</b>
<pre>&lt;?php echo "Καλώς ήρθες στο πρώτο πρόγραμμα στην PHP!\n"; echo "Ποιό είναι το όνομά σου;\n";  \$name = trim(fgets(STDIN));  echo "Γειά σου " . \$name . "!\n"; ?&gt;</pre>
 <b>Έξοδος:</b>
<pre>ggia@ubuntu:~/php\$ php helloIO.php Καλώς ήρθες στο πρώτο πρόγραμμα στην PHP! Ποιό είναι το όνομά σου; Γιώργος Γειά σου Γιώργος!</pre>

Βασικά στοιχεία της PHP από το παραπάνω παράδειγμα:

- Το πρόγραμμα ξεκινά και τελειώνει με `<?php` και `<?>`.
- Η εντολή `echo` τυπώνει ένα μήνυμα στην έξοδο (οθόνη).
- Η `$name` είναι η μεταβλητή `name`. Στην γλώσσα php οι μεταβλητές ξεκινάνε με τον χαρακτήρα `$`.
- Η εντολή `trim(fgets(STDIN))` διαβάσει από την καθορισμένη είσοδο (STDIN) μια συμβολοσειρά (string) με το όνομά μας το οποίο αποθηκεύεται στην μεταβλητή `name`.
- Ο τελεστής `.` ενώνει δύο ή περισσότερες συμβολοσειρές σε μια μεγαλύτερη. Η συμβολοσειρά "Γειά σου " συνενώνεται με την συμβολοσειρά που περιέχει η μεταβλητή `name` μαζί με την συμβολοσειρά `!\n`;
- Το `\n` είναι ο χαρακτήρας αλλαγής γραμμής (newline).
- Το `;` ερωτηματικό δηλώνει τέλος της εντολής.

## Η συνάρτηση fgets()

Η συνάρτηση `fgets()` είναι μια συνάρτηση στην php η οποία διαβάζει μια γραμμή από κάποιο δείκτη (εδώ είναι το STDIN που είναι η καθορισμένη είσοδος - δηλαδή το πληκτρολόγιο). Ο χρήστης εισάγει το όνομα και μόλις πατήσει το `<enter>` εισάγεται ο χαρακτήρας αλλαγής γραμμής και διαβάζεται η συμβολοσειρά μέσα στην μεταβλητή `$name`. Η συνάρτηση `fgets()` είναι αντίστοιχη της `fgets()` στην γλώσσα προγραμματισμού C.

Για όποιον γνωρίζει προγραμματισμό στην γλώσσα C θα μπορέσει να δει ομοιότητες στους παρακάτω δύο πηγαίους κώδικες ο οποίοι διαβάζουν τα περιεχόμενα δύο αρχείων κειμένου γραμμή προς γραμμή (μέσω της `fgets()`) και εμφανίζουν το περιεχόμενο στην οθόνη του υπολογιστή. Η συνάρτηση `fgets()` στην C++ βρίσκεται στην βιβλιοθήκη `<stdio.h>`. Για να εκτελεστεί το πρόγραμμα της C πρέπει πρώτα να γίνει εκτελέσιμο. Ας υποθέσουμε ότι ο παρακάτω κώδικας έχει αποθηκευτεί στο αρχείο `"fgets_cpp.cpp"`. Τότε στο linux με την εντολή `gcc fgets_cpp.cpp -o fgets_cpp` δημιουργείται το εκτελέσιμο αρχείο `fgets_cpp` το οποίο μπορεί να εκτελεστεί με το `./fgets_cpp` από την γραμμή εντολών.

```
#include <stdio.h>

int main() {
    FILE * handle;
    char mystring [100];

    handle = fopen ("myfile.txt" , "r");
    if (handle == NULL)
        perror ("Error opening file myfile.txt");
    else {
        while (fgets(mystring, 100, handle) != NULL)
            // Γράφει τον πίνακα mystring στην καθορισμένη έξοδο stdout (οθόνη)
            fputs (mystring,stdout);
        fclose (handle);
    }
    return 0;
}
```

### ΣΕ PHP:

```
<?php
$handle = @fopen("myfile.txt", "r");
if ($handle) {
    while (($mystring = fgets($handle, 100)) !== false) {
        echo $buffer;
    }
    if (!feof($handle)) {
        echo "Error opening file myfile.txt\n";
    }
    fclose($handle);
}
?>
```

Το αντίστοιχο σενάριο εντολών σε php μπορεί να έχει την παρακάτω μορφή και μπορεί να εκτελεστεί με την εντολή `php fgets_php.php` (θεωρώντας ότι ο παρακάτω κώδικας έχει αποθηκευτεί στο αρχείο `"fgets_php.php"`).

Τα δύο παραπάνω προγράμματα διαβάζουν το αρχείο κειμένου `myfile.txt` (γραμμή προς γραμμή χρησιμοποιώντας ένα βρόχο επανάληψης `while`) και εμφανίζουν τα περιεχόμενα του αρχείου στην οθόνη του υπολογιστή. Ίσως οι παραπάνω κώδικες να μην είναι απόλυτα κατανοητοί αλλά φαίνεται ξεκάθαρα ότι ο προγραμματισμός στην γλώσσα PHP έχει αρκετά στοιχεία - ομοιότητες με την γλώσσα C++.

### Η συνάρτηση printf()

Αρχικοποιούμε την μεταβλητή `$string` με την συμβολοσειρά `"Γειά σου κόσμε!\n"` (το `\n` είναι ο χαρακτήρας αλλαγής γραμμής) και στον κώδικα php εμφανίζουμε την συμβολοσειρά στην προκαθορισμένη έξοδο με 3 τρόπους: με την `echo`, την `print` και την `printf(..)` ή οποία είναι παρόμοια με την αντίστοιχη `printf()` της γλώσσας προγραμματισμού C.

#### Κώδικας:

```
<?php

// Όρισε μια μεταβλητή τύπου αλφαριθμητικού 'string'
// και αρχικοποίησέ την με μια τιμή.
$string = "Γειά σου κόσμε!\n";

// Μπορείς να χρησιμοποιήσεις την echo για να τυπώσεις την
// μεταβλητή στην προκαθορισμένη έξοδο.
echo $string;

// Μπορείς επίσης να χρησιμοποιήσεις την εντολή print.
print $string;

// Ή ακόμη και την συνάρτηση printf αν έχεις εμπειρία από την C.
printf('%s', $string);

?>
```

*\*Συνέχεια στην επόμενη σελίδα.  
\*Συνέχεια από την προηγούμενη σελίδα.*

#### Εξοδος:

```
ggia@ubuntu:~/php$ php php_printf.php
Γειά σου κόσμε!
Γειά σου κόσμε!
Γειά σου κόσμε!
```

Η συνάρτηση `echo` και `print` είναι σχεδόν οι ίδιες αλλά υπάρχουν μερικές διαφορές. Η συνάρτηση `echo` μπορεί να πάρει περισσότερες από μια παραμέτρους και δεν επιστρέφει

καμία τιμή ενώ η print παίρνει μόνο μια παράμετρο. Από πλευράς απόδοσης είναι καλύτερη η echo η οποία μπορεί να πάρει περισσότερες από μια παραμέτρους. Για παράδειγμα ο κώδικας `echo "Καλημέρα" . " " . "κόσμε!";` όπου χρησιμοποιείται ο τελεστής πρόσθεσης . θα μπορούσε να γραφτεί απλούστερα και αποδοτικότερα χρησιμοποιώντας παράμετρο `echo "Καλημέρα", " ", "κόσμε!";`.

### **PHP-GTK παράδειγμα**

Η γλώσσα PHP μπορεί να χρησιμοποιηθεί για να σχεδιαστούν προγράμματα με γραφικό περιβάλλον μέσω του PHP-GTK. Στο παρακάτω σενάριο εμφανίζεται ένα παράθυρο με τίτλο Hello World! και με ένα κουμπί με τίτλο Click Me.

#### **Κώδικας:**

```
<?php
function pressed() {
    echo "Hello again - The button was pressed!\n";
}

$window = new GtkWindow();
$button = new GtkButton('Click Me');

$window->set_title('Hello World!');
$window->connect_simple('destroy', array('Gtk', 'main_quit'));
$button->connect_simple('clicked', 'pressed');

$window->add($button);
$window->show_all();

Gtk::main();
?>
```

#### **Εξοδος:**



## **2.3.3 PHP στον εξυπηρετητή ιστοσελίδων**

Αν έχουμε ακολουθήσει τα βήματα για την εγκατάσταση των PHP και MySQL σωστά, ο κεντρικός κατάλογος όπου τα αρχεία εξυπηρετητή ιστοσελίδων Apache είναι ο `/var/www`. Μπορούμε εναλλακτικά να φτιάξουμε ένα directory στο προσωπικό φάκελό μας π.χ. να τον ονομάσουμε `php` και στην συνέχεια να φτιάξουμε ένα συμβολικό σύνδεσμο στον κατάλογο αυτό μέσα στο `/var/www`. Συγκεκριμένα αφού πάμε στον κατάλογο `cd /var/www`, τρέχουμε με ιδιότητα διαχειριστή `sudo ln -s ~{user name}/php php` (το `{user name}` είναι το όνομα χρήστη στο linux). Κατόπιν ανοίγουμε την ιστοσελίδα <http://localhost/php> η οποία δείχνει τον κατάλογο `php` που έχουμε δημιουργήσει στον προσωπικό μας κατάλογο.



## PHP μέσα σε μια html σελίδα

Ο παρακάτω κώδικας hello.php περιέχει μια βασική ιστοσελίδα html όπου εμπεριέχεται κώδικας στην γλώσσα σεναρίων php. Η επέκταση του αρχείου php, παρόλο που είναι html σελίδα θα δώσει την εντολή να κληθεί ο διερμηνέας php για την δυναμική δημιουργία της σελίδας που θα εμφανιστεί στον φυλλομετρητή.

### PHP κώδικας:

```
<html>
<head>
  <title> Γεια σου κόσμε!</title>
</head>
<body>
  <?php
    print "Γεια σου κόσμε!<br/>";
    print "Αυτό είναι ένα παράδειγμα ενσωματωμένου κώδικα php σε μια σελίδα html.<br/>";
  ?>
</body>
</html>
```

### PHP έξοδος:

```
<html>
<head>
  <title> Γεια σου κόσμε!</title>
</head>
<body>
  Γεια σου κόσμε!<br/>Αυτό είναι ένα παράδειγμα html όπου έχει μέσα php!<br/> </body>
</html>
```

### HTML εμφάνιση:

Γειά σου κόσμε!  
Αυτό είναι ένα παράδειγμα ενσωματωμένου κώδικα php σε μια σελίδα html.

## Κλήση php σελίδας

Θεωρώντας το παρακάτω κώδικα αποθηκευμένο στο php\_printf.php, μέσα στον κατάλογο php μπορούμε να εκτελέσουμε τον κώδικα php μέσα από τον φυλλομετρητή καλώντας την σελίδα php\_printf.php: [http://localhost/php/php\\_printf.php](http://localhost/php/php_printf.php)

### PHP κώδικας:

```
<?php



// Όρισε μια μεταβλητή τύπου αλφαριθμητικού 'string'
// και αρχικοποίησέ την με μια τιμή.
$string = "Γειά σου κόσμε!\n";

// Μπορείς να χρησιμοποιήσεις την echo για να τυπώσεις την
// μεταβλητή στην προκαθορισμένη έξοδο.
echo $string;




// Μπορείς επίσης να χρησιμοποιήσεις την εντολή print.
print $string;

// Ή ακόμη και την συνάρτηση printf αν έχεις εμπειρία από την C.
printf('%s', $string);

?>
```

 <b>PHP έξοδος:</b>
Γειά σου κόσμε! Γειά σου κόσμε! Γειά σου κόσμε!
 <b>HTML εμφάνιση:</b>
Γειά σου κόσμε! Γειά σου κόσμε! Γειά σου κόσμε!
Παρατηρούμε ότι ο χαρακτήρας αλλαγής γραμμής \n μετατράπηκε σε απλό χαρακτήρα κενού χαρακτήρα (whitespace) στην html εμφάνιση.

Για να μπορέσει ο σελιδομετρητής να εμφανίζει το κείμενο με αλλαγές γραμμών στη θέση του \n βάζουμε τον κώδικα html `<br/>` ο οποίος υποδεικνύει στον φυλλομετρητή να εφαρμόζει αλλαγή γραμμής.

 <b>PHP κώδικας:</b>
<pre>&lt;?php  // Όρισε μια μεταβλητή τύπου αλφαριθμητικού 'string' // και αρχικοποίησέ την με μια τιμή. // Ο κώδικας &lt;br/&gt; στην html είναι ο χαρακτήρας αλλαγής γραμμής. \$string = "Γειά σου κόσμε!&lt;br/&gt;";  // Μπορείς να χρησιμοποιήσεις την echo για να τυπώσεις την // μεταβλητή στην προκαθορισμένη έξοδο. echo \$string;  // Μπορείς επίσης να χρησιμοποιήσεις την εντολή print. print \$string;  // Ή ακόμη και την συνάρτηση printf αν έχεις εμπειρία από την C. printf('%s', \$string);  ?&gt;</pre>
 <b>PHP έξοδος:</b>
Γειά σου κόσμε! Γειά σου κόσμε! Γειά σου κόσμε! 
 <b>HTML εμφάνιση:</b>
Γειά σου κόσμε! Γειά σου κόσμε! Γειά σου κόσμε!
Στην ιστοσελίδα παρουσιάζεται η επιθυμητή έξοδος όπου το <code>&lt;br/&gt;</code> δηλώνει την αλλαγή γραμμής.

## Βασική είσοδος/έξοδος με φόρμα και PHP σε μια html σελίδα

Το παρακάτω πρόγραμμα είναι μια φόρμα η οποία χρησιμοποιεί κώδικα html μαζί με κώδικα php για να πάρει ως είσοδο το όνομα και στην συνέχεια να εμφανίσει ένα μήνυμα καλωσορίσματος.

```
<?php
if (isset($_POST['name'])) {
    $name = $_POST['name'];
    if (!empty($name)) {
        $sentence = $name . " γειά σου!";
        echo $sentence;
    }
    else {
        echo 'Παρακαλώ εισάγετε όνομα πριν πατήσετε το κουμπί!';
    }
}
?>

<html>
<meta charset="UTF-8">
<head>

</head>
<body>

<form action="form_html_php.php" method="POST">
    Παρακαλώ βάλτε το όνομά σας: <br><input type="text" name="name"><br>
    <input type="submit" value="Αποστολή">
</form>

</body>
</html>
```

Το πρόγραμμα αποθηκεύεται στο αρχείο form\_html\_php.php. Προσέξτε ότι στον κώδικα html της φόρμας καλείται ο ίδιος κώδικας php.

Ας προσπαθήσουμε να απομυθοποιήσουμε το παραπάνω κομμάτι κώδικα. Η γλώσσα html παρέχει ένα μηχανισμό στον προγραμματιστή για να δημιουργεί φόρμες από τις οποίες ο χρήστης εισάγει δεδομένα και στην συνέχεια να καλείται κάποιο σενάριο επεξεργασίας αυτών. Στο παραπάνω παράδειγμα καλούμε ένα απλό σενάριο επεξεργασίας σε γλώσσα php. Μια μια φόρμα θα μπορούσαμε με τον ίδιο τρόπο να καλέσουμε την εκτέλεση κώδικα perl μέσω cgi σεναρίου ή ASP κλπ. Εδώ η φόρμα καλεί το ίδιο το αρχείο και χρησιμοποιώντας κώδικα php διαμορφώνει ένα μήνυμα καλωσορίσματος το οποίο εμφανίζει στον φυλλομετρητή. Ένα πιο εξελιγμένο σενάριο θα μπορούσε να πάρει την είσοδο και να εκτελέσει λειτουργίες ανάκτησης πληροφοριών από μια βάση δεδομένων η οποία τρέπει στον εξυπηρετητή ιστοσελίδων ή να ανακτήσει δεδομένα από τον σκληρό δίσκο του εξυπηρετητή (π.χ. πολυμεσικά στοιχεία όπως εικόνες, ήχους ή βίντεο).

## 2.4 Javascript

Οι ενδογενείς αδυναμίες της HTML, μιας κατεξοχήν γλώσσας μορφοποίησης υπερκειμένου, σύντομα οδήγησαν στην ανάπτυξη γλωσσών προγραμματισμού στο παγκόσμιο ιστό. Η χρήση των γλωσσών προγραμματισμού κρίθηκε απαραίτητη αφού η εξάπλωση του παγκόσμιου ιστού και η χρήση των σελίδων HTML για αλληλεπίδραση με τους χρήστες (δυναμική αποστολή και λήψη δεδομένων) δεν μπορούσε να πραγματοποιηθεί μέσω της HTML. Αρχικά, αναπτύχθηκε από την εταιρεία Netscape η γλώσσα **JavaScript** η οποία, όπως και η HTML μεταφράζεται από τον φυλλομετρητή κατά την εμφάνιση της σελίδας. Η Microsoft ανέπτυξε απ' την πλευρά της μια δική της έκδοση της γλώσσας JavaScript την οποία ονόμασε **JScript** και μια έκδοση της γλώσσας Basic ειδικά για το παγκόσμιο ιστό που ονόμασε **VBScript**.

Έτσι πλέον με την χρήση των παραπάνω γλωσσών είναι δυνατό να ελεγχθούν και να προγραμματιστούν όλα σχεδόν τα αντικείμενα που μπορεί να περιέχει μία ιστοσελίδα και με τη χρήση είτε εντολών διαδικαστικού χαρακτήρα, είτε εντολών διακλάδωσης, όπως για παράδειγμα η δομή [if ... then... else...]. Η χρήση τέτοιου είδους "προγραμμάτων" ή όπως ονομάζονται "scripts" είναι διαδικασία που έχει μεγάλες απαιτήσεις σε προγραμματισμό και απευθύνεται σε προγραμματιστές και μόνο.

Η Javascript δεν θα πρέπει να συγχέεται με τη Java, που είναι διαφορετική γλώσσα προγραμματισμού και με διαφορετικές εφαρμογές. Η χρήση της λέξης "Java" στο όνομα της γλώσσας έχει περισσότερη σχέση με το προφίλ του προϊόντος που έπρεπε να έχει και λιγότερο με κάποια πιθανή συμβατότητα ή άλλη στενή σχέση με τη Java. Ρόλο σε αυτήν τη σύγχυση έπαιξε και ότι η Java και η Javascript έχουν δεχτεί σημαντικές επιρροές από τη γλώσσα C, ειδικά στο συντακτικό, ενώ είναι και οι δύο αντικειμενοστρεφείς γλώσσες. Τονίζεται ότι ο σωστός τρόπος γραφής της είναι "Javascript" και όχι 'Java script' σαν δύο λέξεις, όπως λανθασμένα γράφεται ορισμένες φορές.

Η αρχική έκδοση της Javascript βασίστηκε στη σύνταξη στη γλώσσα προγραμματισμού C, αν και έχει εξελιχθεί, ενσωματώνοντας πια χαρακτηριστικά από νεότερες γλώσσες. Αρχικά χρησιμοποιήθηκε για προγραμματισμό από την πλευρά του πελάτη (client), που ήταν ο φυλλομετρητής (browser) του χρήστη, και χαρακτηρίστηκε σαν client-side γλώσσα προγραμματισμού. Αυτό σημαίνει ότι η επεξεργασία του κώδικα Javascript και η παραγωγή του τελικού περιεχομένου HTML δεν πραγματοποιείται στο διακομιστή, αλλά στο πρόγραμμα περιήγησης των επισκεπτών, ενώ μπορεί να ενσωματωθεί σε στατικές σελίδες HTML. Αντίθετα, άλλες γλώσσες όπως η PHP εκτελούνται στο διακομιστή (server-side γλώσσες προγραμματισμού).

### Παράδειγμα Javascript.

Παρακάτω παραθέτω ένα παράδειγμα ώστε να γίνει ποιο κατανοητή η χρήση της Javascript στην πράξη.

Στην παρακάτω εικόνα βλέπουμε το Javascript που χρησιμοποιούμε στον ιστότοπο

μας. Αυτό το Javascript με τίτλος simple slideshow είναι ένα απλούστατο javascript, και το χρησιμοποιούμε για να δημιουργήσουμε μια αυτόματη παρουσίαση εικόνων. Οι εικόνες έχουν ορισθεί στην HTML με την μορφή λίστας και παρουσιάζονται με την σειρά που βρίσκονται στη λίστα.

```

1  var delay = 1000;
2  var start_frame = 0;
3
4  function init() {
5      var lis = $('slide-images').getElementsByTagName('li');
6
7      for (i=0; i < lis.length; i++){
8          Effect.FadeOut(lis[i]);
9          lis[i].style.display = 'none';
10         }
11
12         end_frame = lis.length -1;
13
14         start_slideshow(start_frame, end_frame, delay, lis);
15     }
16
17 }
18
19
20
21 function start_slideshow(start_frame, end_frame, delay, lis) {
22     setTimeout(fadeInOut(start_frame, start_frame, end_frame, delay, lis), delay);
23 }
24
25
26 function fadeInOut(frame, start_frame, end_frame, delay, lis) {
27     return (function() {
28         lis = $('slide-images').getElementsByTagName('li');
29         Effect.FadeIn(frame);
30         if (frame == end_frame) { frame = start_frame; } else { frame++; }
31         lisAppear = lis[frame];
32         setTimeout("Effect.Appear(lisAppear)", 0);
33         setTimeout(fadeInOut(frame, start_frame, end_frame, delay), delay + 1000);
34     })();
35 }
36
37
38
39 Event.observe(window, 'load', init, false);

```

Στην παρακάτω εικόνα βλέπουμε το DIV του Javascript και μια λίστα με τις τοποθεσίες των φωτογραφιών. Το Javascript θα καταφύγει στα links των φωτογραφιών και θα τις παρουσιάσει με την σειρά που είναι στη λίστα, προχωρώντας στην επόμενη κάθε 1000 χιλιοστά του δευτερολέπτου (1 δευτερόλεπτο).

```

197 <div id="slide-show">
198
199 <ul id="slide-images">
200 <li></li>
201 <li></li>
202 <li></li>
203 <li></li>
204 <li></li>
205 <li></li>
206 <li></li>
207 <li></li>
208 <li></li>
209 <li></li>
210 <li></li>
211 <li></li>
212 <li></li>
213
214 </ul>
215
216 </div>

```

# ΚΕΦΑΛΑΙΟ 3

## Σχεδιασμός και λειτουργία



## **ΚΕΦΑΛΑΙΟ 3 - Σχεδιασμός και λειτουργία**

### **3.1 Περιγραφή σελίδας.**

Ο ιστότοπος που εξελίξαμε έχει ως σκεπτικό την απλότητα και την ευχρηστία. Αποφύγαμε την περιττή πολυπλοκότητα και βαριές εφαρμογές. Προσπαθούμε να διατηρήσουμε ψιλά standard ποιότητας και εμφάνισης, μέσω απλών και έξυπνων μεθόδων.

Προσπαθούμε να ``παίζουμε`` με τα χρώματα και με γεωμετρικά σχήματα παρά να προσθέσουμε apps κινούμενες εικόνες, flash και εκνευριστικούς ήχους. Δεν χρησιμοποιούμε κάποιο δαιδαλώδες σύστημα διαχείρισης ξενοδοχειακών μονάδων τύπου CMS (Σύστημα Διαχείρισης Περιεχομένου), προφανώς διότι δεν είναι open source και φυσικά διότι 97% οι δυνατότητες που προσφέρουν αφήνουν παγερά αδιάφορους τους επιχειρηματίες και το προσωπικό (κυρίως λόγω πολυπλοκότητας και δυσχρηστίας).

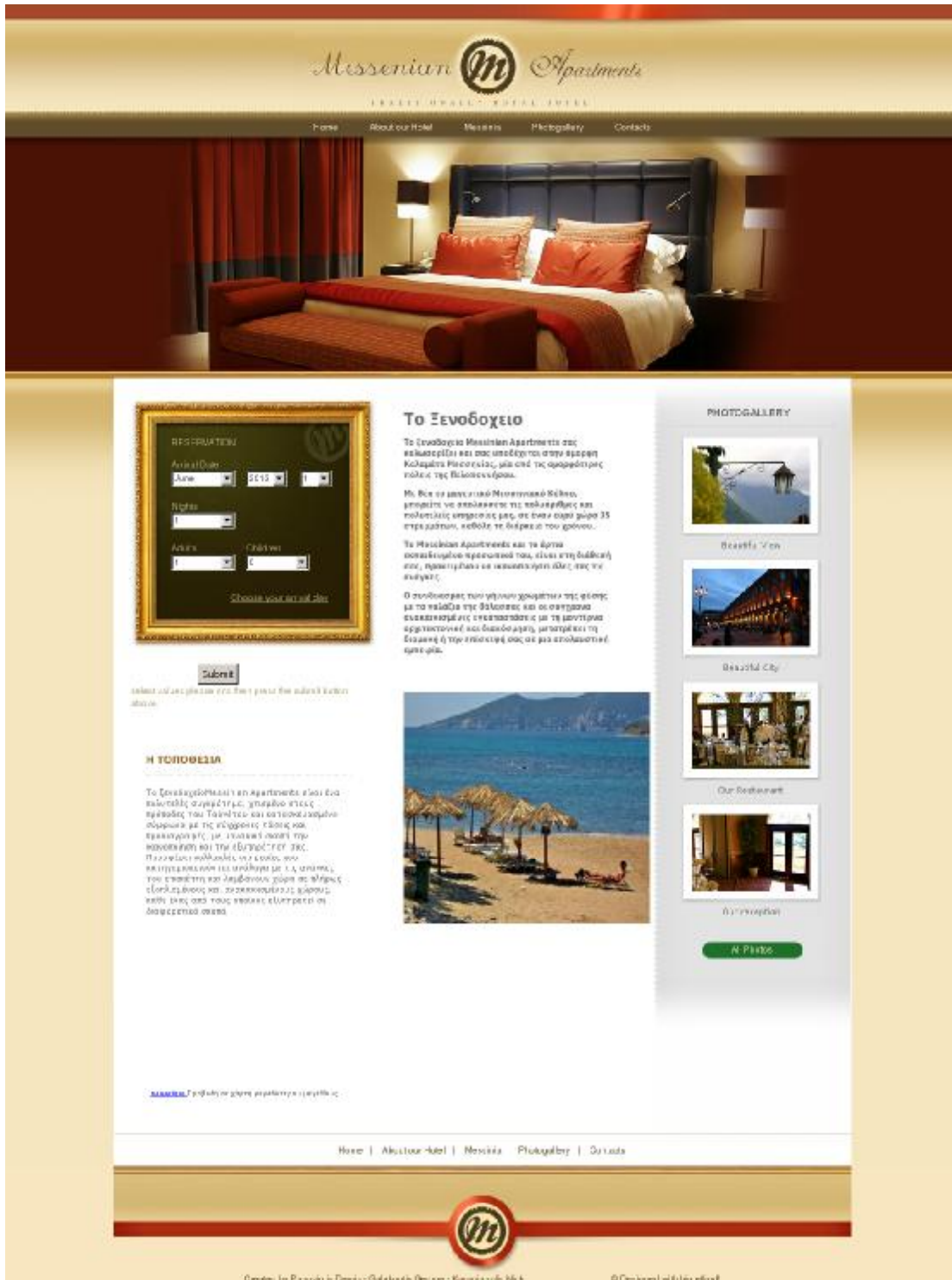
Ο ιστότοπος είναι σχεδιασμένος ώστε να απευθύνεται σε αλλοδαπούς πελάτες. Για το γιατί, παρακαλώ απευθυνθείτε στον ιστότοπο υπουργείο Τουρισμού για τα ποιοτικά στατιστικά του τουρισμού 11-12. Δεν θεωρήσαμε χρηστικό το να επεκταθούμε στην δημιουργία αντιστοίχου ιστότοπου εξολοκλήρου στα ελληνικά. Όμως για λόγους παρουσίασης και με το σκεπτικό ότι η πτυχιακή εργασία αυτή είναι ελληνικού τμήματος, τα κείμενα είναι γραμμένα εξολοκλήρου στα ελληνικά.

Ο πίνακας με την φόρμα εισαγωγής στοιχείων δεν χάνεται ποτέ από το οπτικό πεδίο του επισκέπτη (τουλάχιστον για του Η/Υ), δίνοντας την δυνατότητα σε αυτόν, ανά πάσα στιγμή να ξεκινήσει την διαδικασία της κράτησης ενός δωματίου. Η χρήση του πίνακα με την φόρμα αναλύεται βήμα-βήμα και αυτή αναλυτικότερα στο τέλος αυτού του κεφαλαίου.

Στην συνέχεια αυτού του κεφαλαίου θα ανακαλύψουμε την δομή της ιστοσελίδας μας, αλλά και τον τρόπο χρήσης. Το σύστημα κρατήσεων όπως θα δείτε είναι εύκολο να κατανοηθεί σχεδιασμένο με απλή λογική και απαιτεί την ελάχιστη δυνατή εισαγωγή στοιχείων από τον επισκέπτη. Σχεδόν δεν χρησιμοποιείται καθόλου το πληκτρολόγιο, παρά μόνο για το τελευταίο στάδιο, πριν την ολοκλήρωση της κράτησης.

### 3.1.1 Γενικά χαρακτηριστικά και δομή.

Η σελίδα έχει φτιαχτεί με γνώμονα ότι θα χρησιμοποιηθεί από φυλλομετρητή σε ηλεκτρονικούς υπολογιστές με ανάλυση οθόνης ίσης ή μεγαλύτερης των 1024X768. Η σελίδα χρησιμοποιεί ένα σταθερό template και αλλάζει μόνο το περιεχόμενο της κεντρικής στήλης.





### 3.1.2 Τομείς ιστότοπου

Η σελίδα όπως και κάθε σελίδα στο διαδίκτυο χωρίζεται σε κάποια τμήματα «divisions». Δεν αναφέρομαι στα HTML DIVS τα οποία μπορείτε να βρείτε στο συνοδευτικό οπτικό δίσκο ανοίγοντας τα με το σημειωματάριο (WordPad), αλλά στη κατάτμηση του ιστότοπου. Παρακάτω θα βρείτε μια εικόνα που περιγράφει τα διάφορα τμήματα του ιστότοπου.

The screenshot shows the website layout for Messinian Apartments. The header (A) contains the logo and navigation links. Below it is a large banner image (B) of a hotel room. The main content area is divided into three columns: a reservation form (C), a list of content areas (D), and a photo gallery (E). The reservation form includes fields for arrival date, departure date, and number of guests, along with a submit button. The list of content areas (D) consists of several 'Περιοχή κειμένου' (Text area) labels. The photo gallery (E) displays several images of the property. The footer (H) contains the navigation menu, the logo, and contact information.

A) Header-Λογότυπο

Δ) 1<sup>η</sup> στήλη περιεχομένου.

περιεχομένου. Έκθεση

B) Άνω και κάτω μπάρες πλοήγησης

E) 2<sup>η</sup> στήλη περιεχομένου.

Z) Footer

Γ) Banner

ΣΤ) 3<sup>η</sup> στήλη

H) Φόντο

### 3.1.3 Μπάρες πλοήγησής (Navigation Bars)

Υπάρχουν 2 μπάρες πλοήγησης μια στο επάνω μέρος της σελίδας και μια στο κάτω μέρος της σελίδας. Αυτές οι μπάρες είναι ο μόνος τρόπος πλοήγησης μέσα στη σελίδα.

Και οι 2 μπάρες έχουν ακριβώς τις ίδιες λειτουργίες και παραπέμπουν ακριβώς στα ίδια url. Στην πάνω μπάρα χρησιμοποιείται background ώστε να αποδώσουμε ανώτερο εικαστικό αποτέλεσμα, ενώ στην κάτω μπάρα πλοήγησης χρησιμοποιείται απλό κείμενο, για την περίπτωση που κάποια συσκευή πλοήγησής δεν μπορεί να αποδώσει καθαρά την πάνω μπάρα.

#### Επάνω μπάρα πλοήγησης.



Κουμπί	Home	About our Hotel	Messinia	Photogallery	Contacts	Κουμπί
Παραπέμπει στην σελίδα	Αρχική	Πληροφορίες	Μεσσηνία	Έκθεση	Επαφή	Παραπέμπει στην σελίδα
*.PHP	index	about	messenia	photo	contact	*.PHP
Παράγραφος	3.1.3.1	3.1.3.2	3.1.3.3	3.1.3.4	3.1.3.5	Παράγραφος

Home | About our Hotel | Messinia | Photogallery | Contacts

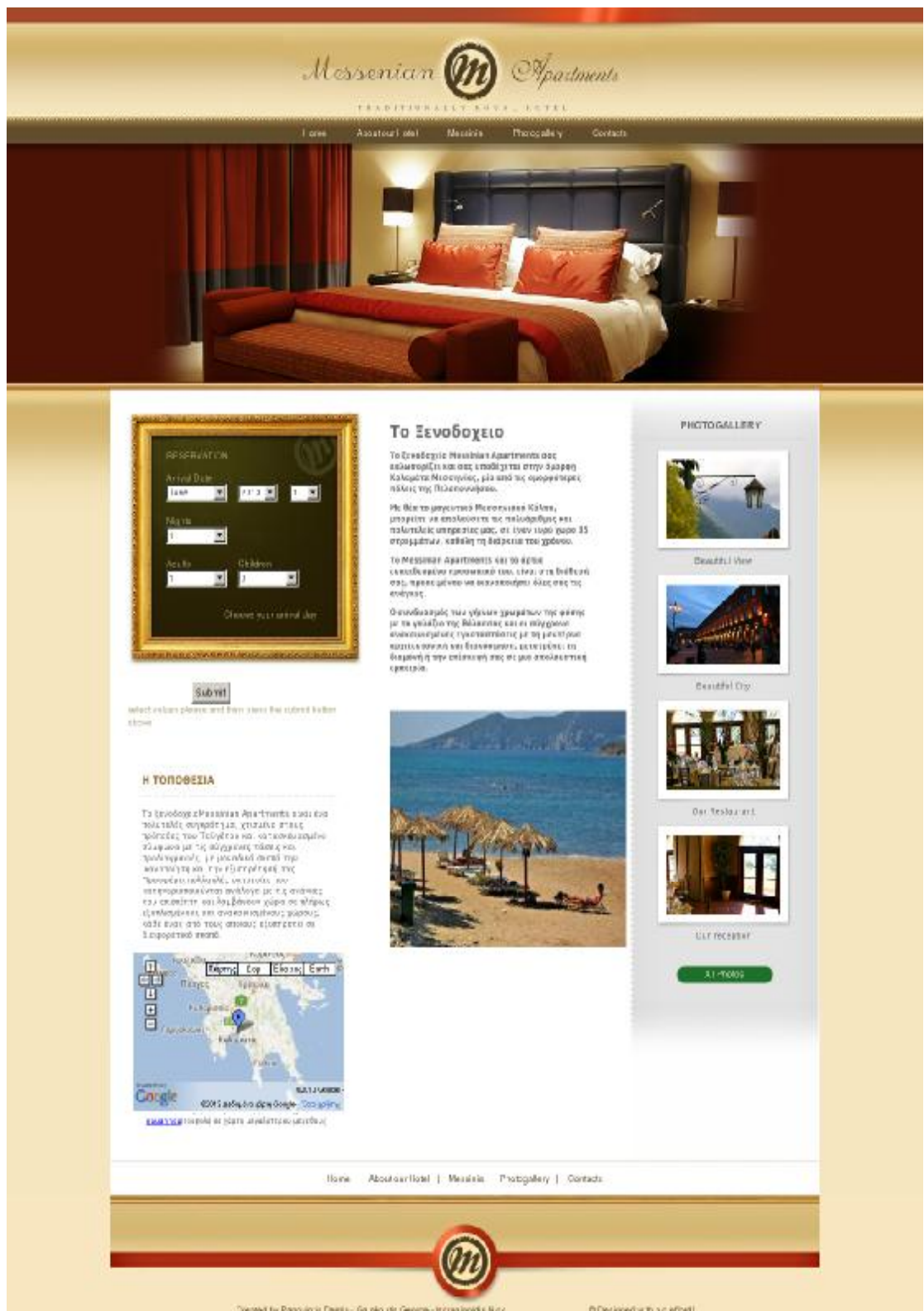


#### Κάτω μπάρα πλοήγησης

### 3.1.3.1 Κύρια σελίδα (Homepage).

Home

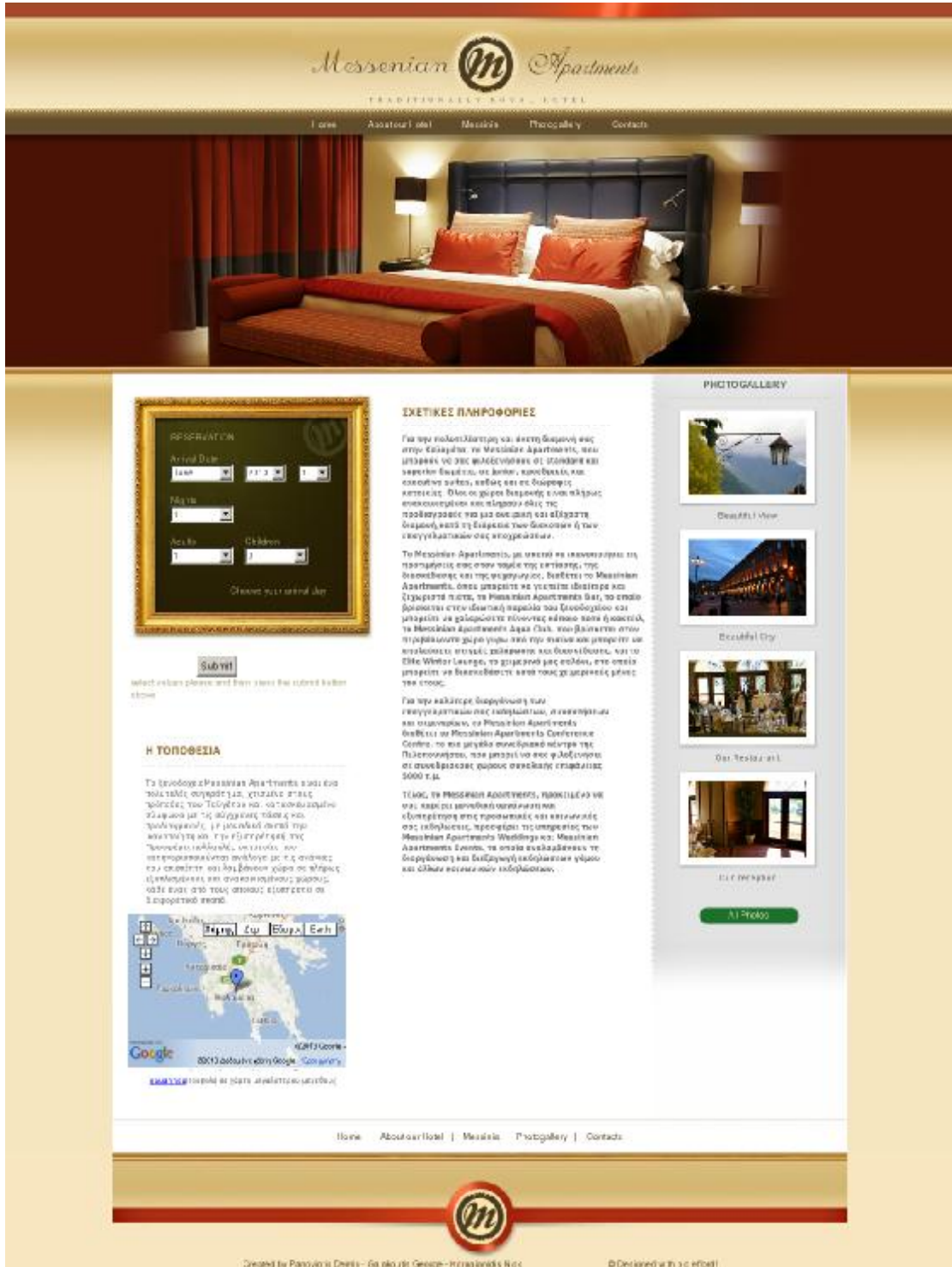
Αυτή είναι και η κύρια (Αρχική) σελίδα. Είναι η πρώτη σελίδα που εμφανίζεται όταν επισκέπτεται τον ιστότοπο ένας πελάτης. Η σελίδα καλωσορίζει τον επισκέπτη και προσπαθεί να τον παροτρύνει να περιηγηθεί στο σύνολο του ιστότοπου.



### 3.1.3.2 Σχετικά με το ξενοδοχείο μας (About our hotel).

About our Hotel

Εδώ ένας επισκέπτης μπορεί να βρει περισσότερες πληροφορίες σχετικά με το ξενοδοχείο τις εγκαταστάσεις τις υπηρεσίες που θα μπορέσει να απολαύσει κατά την διαμονή του.



### 3.1.3.3 Μεσσηνία (Messinia)

Messinia

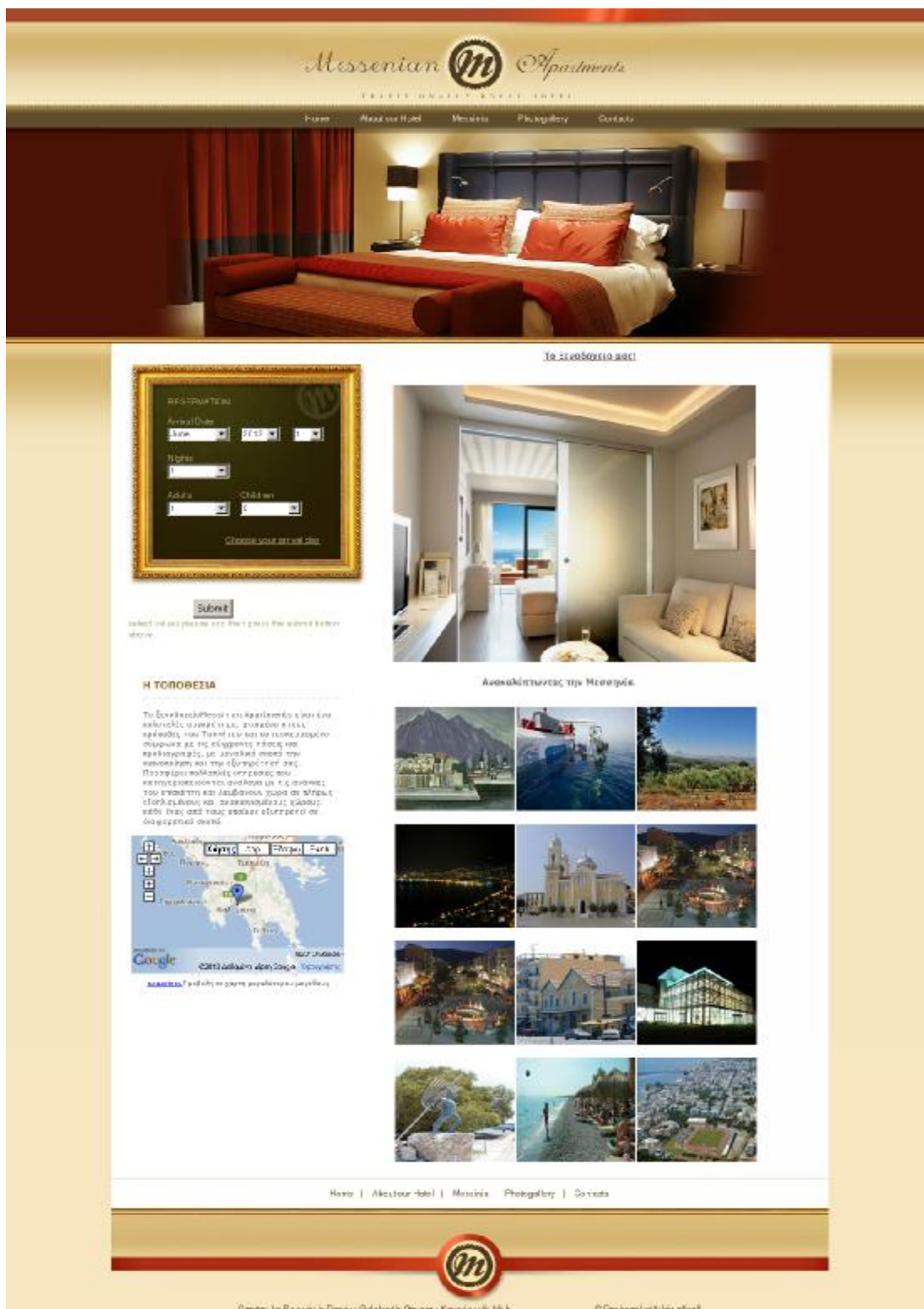
Εδώ ο επισκέπτης μπορεί να αντλήσει πληροφορίες και να ενημερωθεί για την περιοχή διαμονής του. Ένα slideshow αναπαράγει φωτογραφίες της Μεσσηνίας.

The screenshot displays the website for Messinian Apartments. At the top, the logo 'Messinian Apartments' is centered, with the tagline 'TRADITIONALLY BORN - ESTABLISHED' below it. A navigation menu includes 'Home', 'About our Hotel', 'Messinia', 'Photogallery', and 'Contacts'. The main visual is a large, warm-toned photograph of a hotel room with a bed and a sofa. Below the hero image, the page is divided into three columns. The left column contains a 'RESERVATION' form with fields for 'Arrival Date', 'Night', and 'Check out', along with a 'Submit' button. The middle column is titled 'MESSINIA' and contains a paragraph of text, a small image of a beach with umbrellas, and another paragraph. The right column is titled 'PHOTOGALLERY' and features a vertical list of five thumbnail images with captions: 'Beach & View', 'Beautiful City', 'Our Restaurant', and 'Our terrace'. At the bottom of the page, there is a navigation bar with the same menu items as the top, and a footer with the text 'Created by Panayiotis Demiris - Gd. sk. 011, George - Pitaragakis Road' and '© Designed with x3g efordi'.

### 3.1.3.4 Έκθεσή (Photogallery).

#### Photogallery

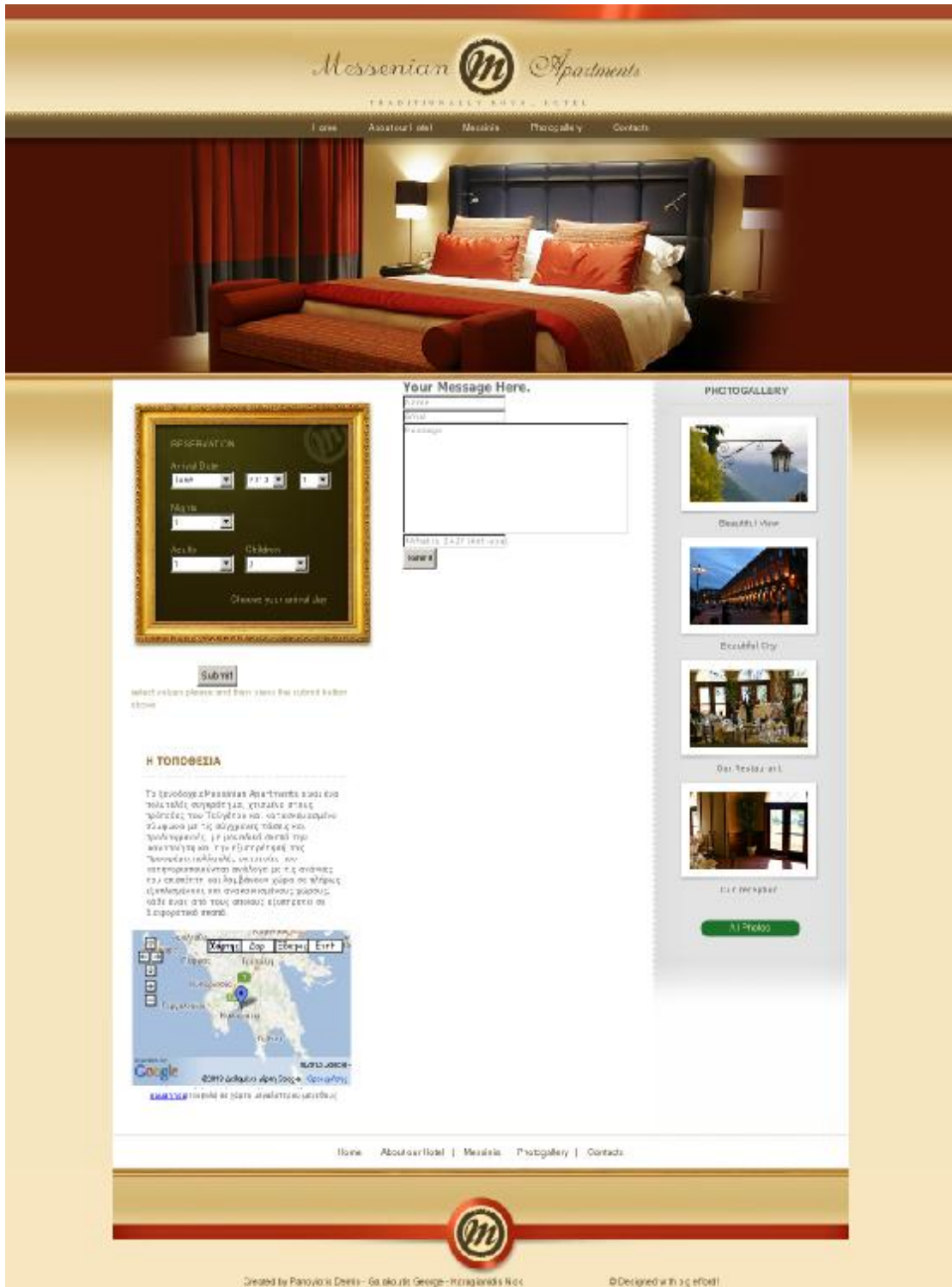
Εδώ παρουσιάζονται φωτογραφίες τόσο από την Μεσσηνία αλλά και την Καλαμάτα. Εδώ μπορεί ο επισκέπτης να λάβει μια άποψη για τον ξενοδοχείο που αποδημεί να διαμείνει, καθώς και για την περιοχή στη οποία βρίσκεται το ξενοδοχείο. Η σελίδα περιέχει ένα slideshow το οποίο αναπαράγει φωτογραφίες από Lichnos Beach hotel<sup>av.11</sup>.



### 3.1.3.5 Επικοινωνία (Contacts).

Contacts

Εδώ ένας επισκέπτης μπορεί να επικοινωνήσει με την διαχείριση του ξενοδοχείου. Για να αποφευχθούν μηνύματα κακόβουλου περιεχομένου έχει περιληφθεί και ένα πεδίο ερωτήματος. Ο επισκέπτης θα πρέπει να απαντήσει πόσο είναι το άθροισμα 2+2 για να μπορέσει να στείλει μήνυμα ηλεκτρονικού ταχυδρομείου στον διαχειριστή.



## **3.2 Λειτουργία συστήματος κρατήσεων και βάσης δεδομένων.**

Ο ιστότοπος είναι σχεδιασμένος με απλότητα και ευχρηστία ως κύριο μέλημα. Το τελευταίο που θέλει κάποιος υποψήφιος πελάτης, ο οποίος ψάχνει για ένα ξενοδοχείο να ξεκουραστεί, είναι μια πολύπλοκη, “βαριά” αργή και δύσχρηστη ιστοσελίδα. Για χάρη ευκολίας θα θεωρήσουμε ότι ιστότοπος αυτός, χωρίζεται σε 2 κύρια μέρη. Πρώτον στο περιβάλλον ορατού ιστότοπου στο πελάτη και δεύτερον στο σύστημα διαχείρισης και κρατήσεων το οποίο δεν είναι άμεσα ορατό στον πελάτη.

### **3.2.1 Η προσέγγιση μας.**

Φυσικά αυτή η περιγραφή δεν έχει καμία απολύτως επιστημονική βάση, και υπάρχουν διαφορές διαφορετικές προσεγγίσεις. Εμείς διαλέξαμε την πιο βασική από όλες, Βασιζόμενοι σε προσωπικές εμπειρίες και κοινή λογική δημιουργήσαμε ένα σενάριο με το οποίο προσπαθήσαμε να μεταφέρουμε την καθημερινότητα σε ένα πρακτικό και εύχρηστο και εύκολα κατανοητό ηλεκτρονικό-ψηφιακό σύστημα

### **4.2.2 Παρουσίαση του ιστότοπου στην πράξη.**

**Σενάριο:** Ας υποθέσουμε πως ένας πραγματικός πελάτης, εισέρχεται από την είσοδο του ξενοδοχείου, πλησιάζει την ρεσεψιόν και ρωτάει για δωμάτιο. Η ρεσεψιονίστ αφού τον καλωσορίσει, του ζητάει μερικές πληροφορίες για το τι ενδιαφέρει (διανυκτερεύσεις, άτομα, παιδιά, τύπος δωματίου) και αφού επεξεργαστεί όλες τις πληροφορίες προτείνει στον πελάτη δωμάτια και τιμές. Εκείνος λοιπόν αποφασίζει και κλίνει το ανάλογο δωμάτιο.

#### **Ανάλυση σεναρίου και αντιμετώπιση του ηλεκτρονικά.**

«Ας υποθέσουμε πως ένας πραγματικός πελάτης, εισέρχεται από την είσοδο του ξενοδοχείου, πλησιάζει την ρεσεψιόν και ρωτάει για δωμάτιο.»

Ο ιστότοπος είναι ο φυσικός χώρος του ηλεκτρονικού ξενοδοχείου. Ο υποψήφιος πελάτης αναζητά και επισκέπτεται το ξενοδοχείο με οποιαδήποτε συμβατή συσκευή η οποία έχει πρόσβαση στο παγκόσμιο ιστό. Η ρεσεψιόν είναι το κάδρο με τα πεδία της φόρμας.

«Η ρεσεψιονίστ τον καλωσορίζει, του ζητάει μερικές πληροφορίες για το τι ενδιαφέρει (διανυκτερεύσεις, άτομα, παιδιά, τύπος δωματίου)...»

Η αρχική σελίδα καλωσορίζει τον επισκέπτη και του προσφέρει μια πρώτη εντύπωση για το ξενοδοχείο. Στην αρχική σελίδα ο επισκέπτης-πελάτης μπορεί να αναζητήσει το σύνολο των πληροφοριών που χρειάζεται.

Το κάδρο περιλαμβάνει την φόρμα η οποία χρησιμοποιείται για να αλληλεπιδράσει ο χρήστης με την βάση δεδομένων. Η φόρμα περιλαμβάνει όλα τα βασικά στοιχεία που χρειάζεται το σύστημα για να υποβάλει ένα ερώτημα στη βάση δεδομένων.





RESERVATION

Arrival Date  
June 2013 1

Nights  
1

Adults Children  
1 0

[Choose your arrival day](#)

Η φόρμα ζητεί την ημερομηνία άφιξης, τον αριθμό των διανυκτερεύσεων και τον αριθμό-τύπο των επισκεπτών. Αφού λοιπόν ο πελάτης-επισκέπτης συμπληρώσει υποχρεωτικά όλα τα πεδία, απλώς με τη χρήση του ποντικιού ή με taps σε συσκευές με οθόνη αφή, είναι έτοιμος να υποβάλει το ερώτημά του.

«και αφού επεξεργαστεί όλες τις πληροφορίες προτείνει στον πελάτη δωμάτια και τιμές.»

Η επεξεργασία των πληροφοριών που έχουν επιλεγεί στην φόρμα γίνεται με την επιλογή του κουμπιού  ακριβώς κάτω από τον πίνακα της φόρμας.



RESERVATION

Arrival Date  
August 2013 28

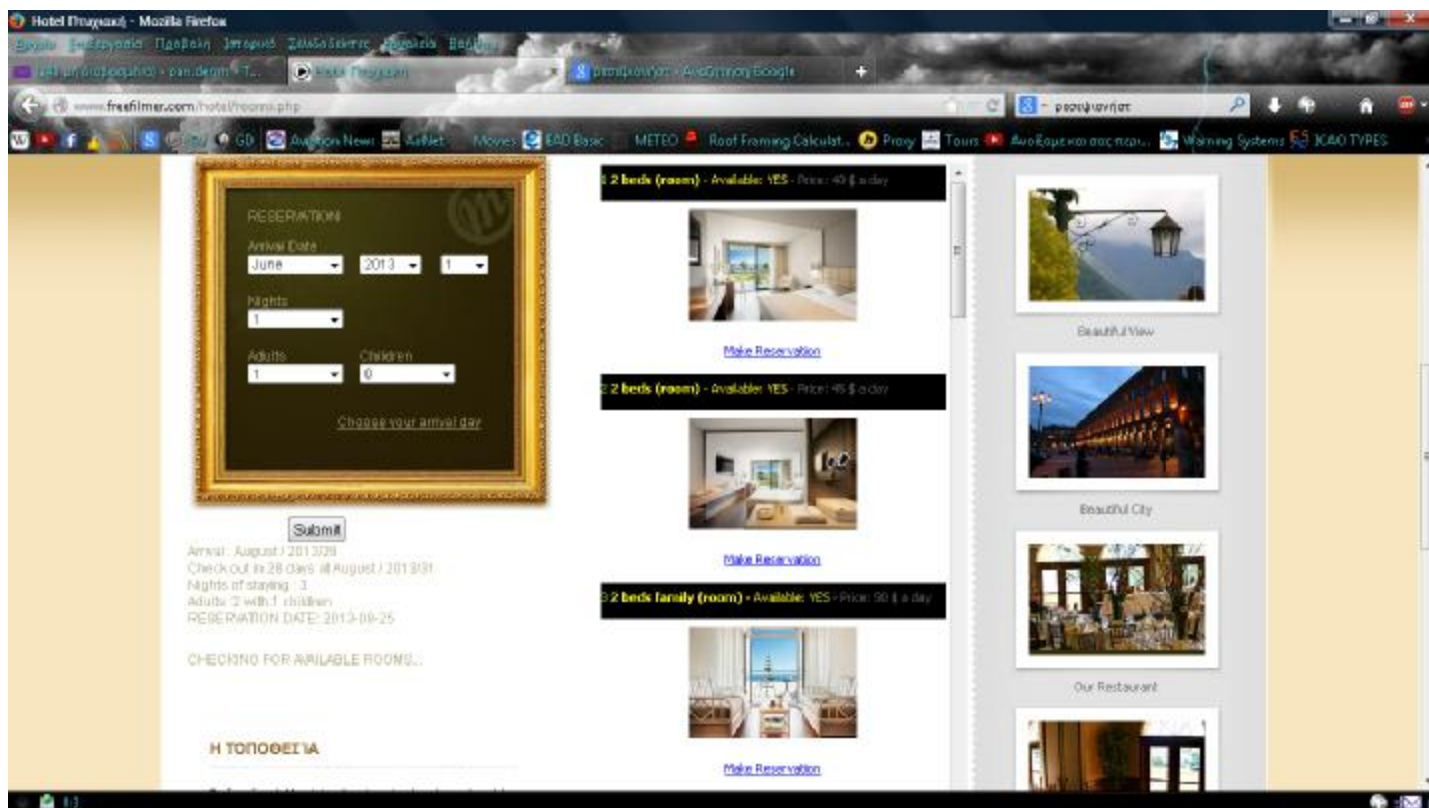
Nights  
3

Adults Children  
2 1

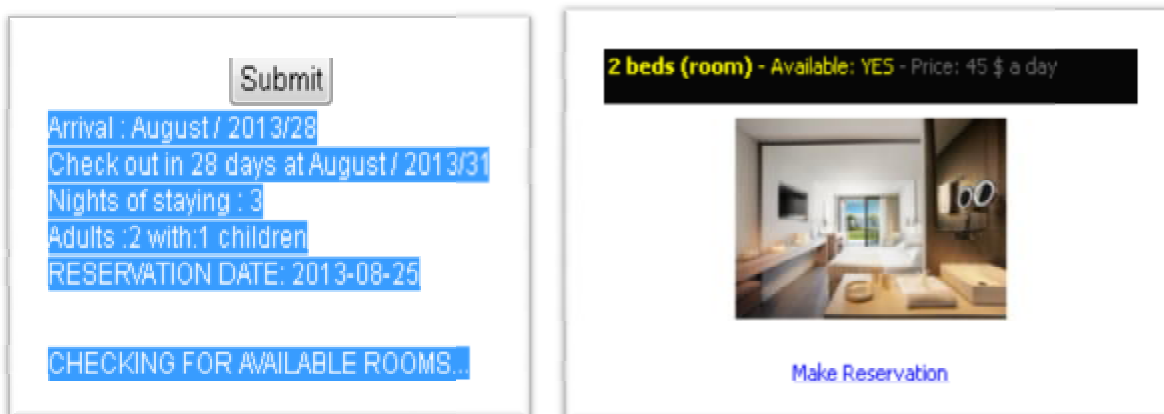
[Choose your arrival day](#)

select values please and then press the submit button above.

Πατώντας λοιπόν το κουμπί αυτό η φόρμα υποβάλει ερώτημα στην βάση δεδομένων και η βάση δεδομένων μας δίνει αποτελέσματα τα οποία παρουσιάζονται στον χρήστη.



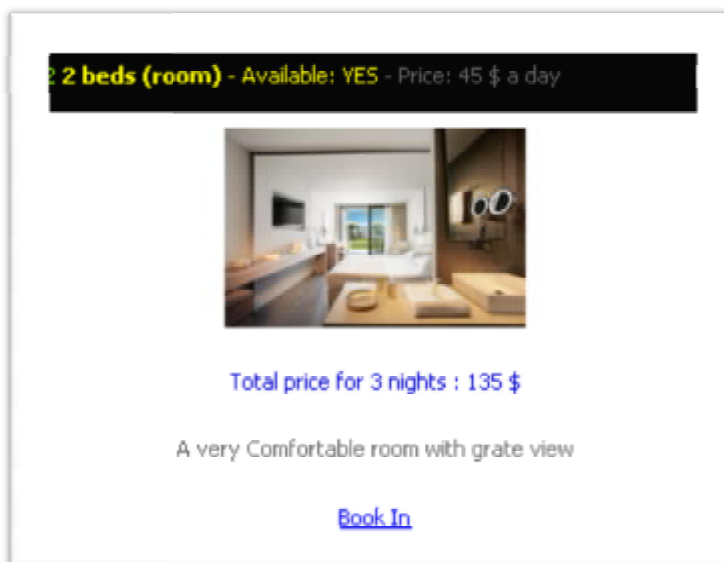
Κάτω από το κουμπί **Submit** πλέον εμφανίζονται οι πληροφορίες που έχει επιλέξει ο χρήστης και στο κέντρο του ιστότοπου εμφανίζεται η κατάσταση των δωματίων και η διαθεσιμότητα τους μαζί με την τιμή τους, για την περίοδο του ερωτήματος.



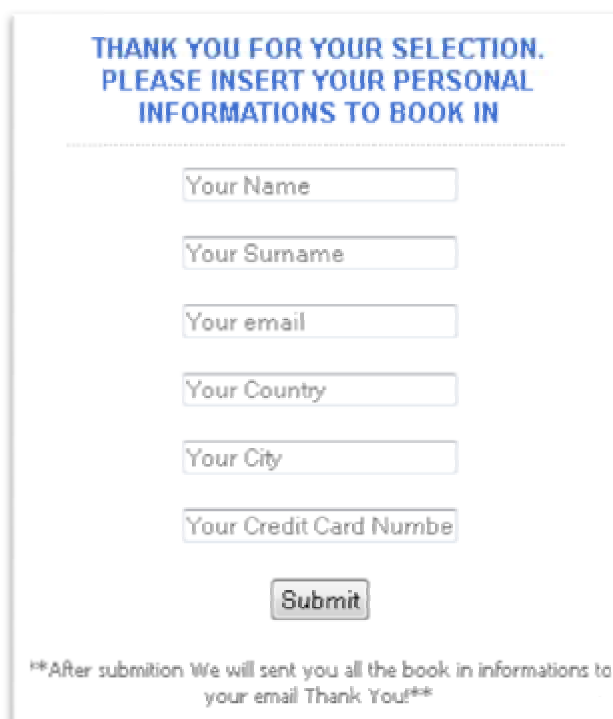
Από αυτό το σημείο ο επισκέπτης πελάτης μπορεί να προχωρήσει στην κράτηση του δωματίου που τον διαφέρει, απλώς πατώντας την επιλογή **Make Reservation**.

«Εκείνος λοιπόν αποφασίζει και κλίνει το ανάλογο δωμάτιο.»

Αφού λοιπόν ο πελάτης περιηγηθεί ανάμεσα στα δωμάτια και καταλήξει στο ποίο θέλει να επιλέξει πατάει την επιλογή **Make Reservation** ώστε να προχωρήσει στην διαδικασία της κράτησης.



Αμέσως μετά του εμφανίζεται ΜΟΝΟ το δωμάτιο που επέλεξε, με κάποιες επιπλέον πληροφορίες σχετικά με το δωμάτιο και την συνολική τιμή για όλη τη διάρκεια της διαμονής του. Επιλέγοντας λοιπόν το **Book In** ξεκινάει η διαδικασία της κράτησης.

A screenshot of a booking confirmation form. The title reads 'THANK YOU FOR YOUR SELECTION. PLEASE INSERT YOUR PERSONAL INFORMATIONS TO BOOK IN'. Below the title are several input fields: 'Your Name', 'Your Surname', 'Your email', 'Your Country', 'Your City', and 'Your Credit Card Numbe'. A 'Submit' button is located below the fields. At the bottom, there is a note: '\*\*After submission We will sent you all the book in informations to your email Thank You!\*\*'

Για να πραγματοποιηθεί επιτυχώς η κράτηση ο πελάτης θα πρέπει να συμπληρώσει τα υποχρεωτικά πεδία και να πατήσει το κουμπί .

THANK YOU FOR YOUR SELECTION.  
PLEASE INSERT YOUR PERSONAL  
INFORMATIONS TO BOOK IN

Demis

Panayiotis

.....@yahoo.gr

Greece

Argos

.....

Submit

\*\*After submission We will sent you all the book in informations to your email Thank You!\*\*

\* Σε αυτό το σημείο, αξίζει να σημειωθεί πως αυτό είναι και το μόνο στάδιο στο οποίο ο επισκέπτης πελάτης πρέπει να κάνει χρήση πληκτρολογίου για να εισάγει δεδομένα.

Αμέσως μετά παραπέμπεται στη σελίδα επιβεβαίωσης η οποία τον ενημερώνει για την επιτυχή κράτηση και του δίνει τοις πληροφορίες της κράτησής του, τις οποίες και του έχει σταλεί @mail μαζί με άλλα στοιχεία.

Success reservation .Please Check Your email box , we have sent you all your book in Informations

Arrival : August / 2013 / 28

Check out in 28 days at August / 2013 /31

Nights of staying : 3

Adults : 2 with: 1 children

**Reservation Date: 2013-08-25**

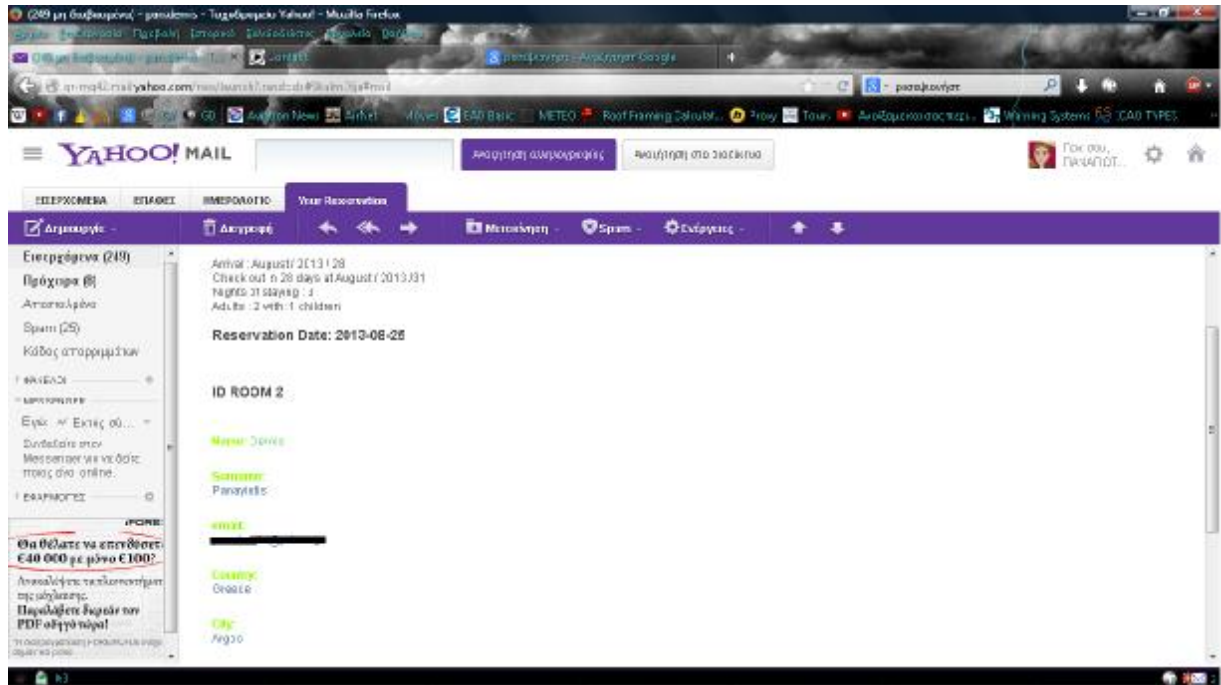
### **ID ROOM 2**

Success reservation. Please Check Your email box , we have sent you all your book in Informations Success reservation

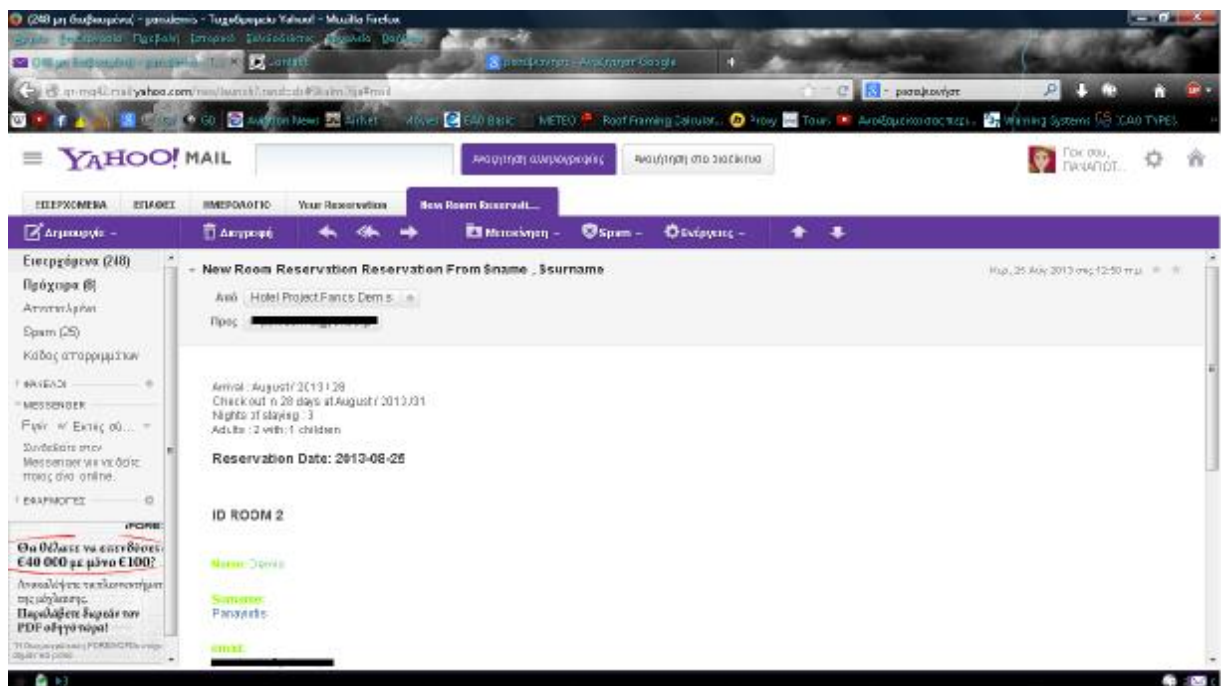
Σε κάθε επιτυχής κράτηση αποστέλλονται 2 @mail.

1 στον πελάτη και 1 στον διαχειριστή.

@mail πελάτη:



@mail διαχειριστή:



Η κράτηση πλέον έχει ολοκληρωθεί, η πληρωμή έχει γίνει και ο πελάτης μπορεί να εκτυπώσει το απόκομμα του είτε από την επιβεβαίωση της κράτησης είτε από το @mail του και να παρουσιαστεί στο ξενοδοχείο την αναγραφόμενη μέρα ώρα για να απολαύσει τις διακοπές του.

# ΚΕΦΑΛΑΙΟ 4

## Εφαρμογές και ανάληψη βάσης δεδομένων



## 4.1 Βάση δεδομένων

Μια **βάση δεδομένων** (database) αποτελείται από έναν ή περισσότερους **πίνακες** (tables), ο καθένας από τους οποίους περιέχει μια λίστα από κάποια πράγματα. Για μια βάση δεδομένων πελατών (clients), είναι φυσικό να ξεκινήσουμε μ' έναν πίνακα με όνομα clients που θα περιέχει μια λίστα από στοιχεία πελατών.

Ο κάθε πίνακας σε μια βάση δεδομένων περιέχει μια ή περισσότερες **στήλες** (columns) ή **πεδία** (fields), όπου η κάθε στήλη περιέχει μια συγκεκριμένη πληροφορία για τον κάθε πελάτη που υπάρχει στην βάση δεδομένων (database).<sup>av.12</sup>

Ο πίνακας clients μπορεί να περιέχει στήλες για τον κωδικό ενός πελάτη (ID), για το όνομά του (Name) καθώς και για την ημερομηνία γέννησής του (Date). Ο κάθε πελάτης που αποθηκεύουμε σ' αυτόν τον πίνακα λέμε ότι αποτελεί μια **γραμμή** (row) ή μια **εγγραφή** (record) του πίνακα. Για παράδειγμα, ας δούμε τον παρακάτω πίνακα :

<b>ID</b>	<b>Name</b>	<b>Date</b>
1	Αντωνιάδης	1970-04-01
2	Παπαδόπουλος	1968-02-22

Εκτός από τις στήλες για το όνομα του πελάτη (Name) και την ημερομηνία γέννησής του (Date), υπάρχει και μια στήλη με όνομα ID, ο σκοπός της οποίας είναι να εκχωρήσει έναν μοναδικό αριθμό στον κάθε πελάτη έτσι ώστε να έχουμε έναν εύκολο τρόπο αναφοράς σ' αυτόν και να μπορούμε να τον ξεχωρίσουμε από τους άλλους πελάτες.

Σαν επισκόπηση, το παραπάνω είναι ένας πίνακας τριών στηλών που περιέχει δύο γραμμές ή καταχωρήσεις. Η κάθε γραμμή του πίνακα περιέχει έναν κωδικό (ID) αναγνώρισης του πελάτη, το όνομά του (text) καθώς και την ημερομηνία γέννησής του (date). Με βάση αυτήν την βασική ορολογία, είμαστε έτοιμοι να αρχίσουμε να χρησιμοποιούμε την MySQL.

### 4.1.1 MY SQL εν συντομία



Η MySQL είναι ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων που μετρά περισσότερες από 11 εκατομμύρια εγκαταστάσεις. Έλαβε το όνομά της από την κόρη του Μόντυ Βιντένιους, τη Μάι. Το πρόγραμμα τρέχει έναν εξυπηρετητή (server) παρέχοντας πρόσβαση πολλών χρηστών σε ένα σύνολο βάσεων δεδομένων.

Ο κωδικός του εγχειρήματος είναι διαθέσιμος μέσω της GNU General Public License, καθώς και μέσω ορισμένων ιδιόκτητων συμφωνιών. Ανήκει και χρηματοδοτείται από μία και μοναδική κερδοσκοπική εταιρία, τη σουηδική MySQL AB, η οποία σήμερα ανήκει στην Oracle.

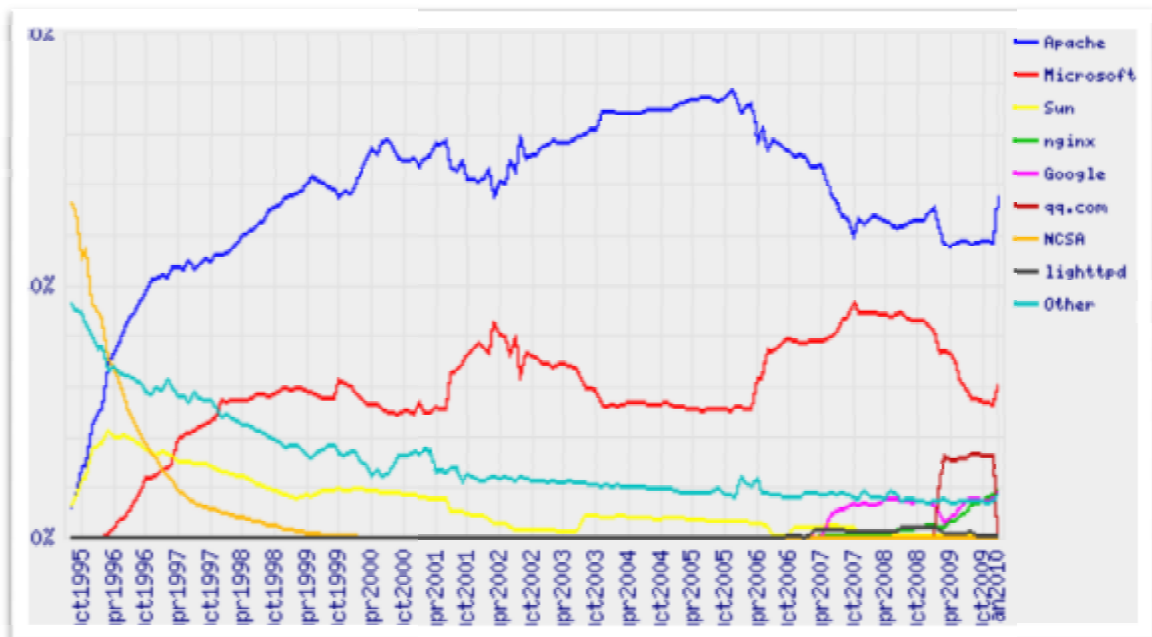
Η MySQL είναι δημοφιλής βάση δεδομένων για διαδικτυακά προγράμματα και ιστοσελίδες. Χρησιμοποιείται σε κάποιες από τις πιο διαδεδομένες διαδικτυακές υπηρεσίες, όπως το Flickr, το YouTube, η Wikipedia, το Google, το Facebook και το Twitter.

#### 4.1.2 Τι είναι ο Apache HTTP.



Ο **Apache HTTP** γνωστός και απλά σαν Apache είναι ένας εξυπηρετητής του παγκόσμιου ιστού (web). Όποτε ένας χρήστης επισκέπτεται ένα ιστότοπο το πρόγραμμα πλοήγησης (browser) επικοινωνεί με έναν διακομιστή (server) μέσω του πρωτοκόλλου HTTP, ο οποίος παράγει τις ιστοσελίδες και τις αποστέλλει στο πρόγραμμα πλοήγησης. Ο Apache είναι ένας από τους δημοφιλέστερους εξυπηρετητές ιστού, εν μέρει γιατί λειτουργεί σε διάφορες πλατφόρμες όπως τα Windows, το Linux, το Unix και το Mac OS X. Κυκλοφόρησε υπό την άδεια λογισμικού Apache και είναι λογισμικό ανοιχτού κώδικα. Συντηρείται από μια κοινότητα ανοικτού κώδικα με επιτήρηση από το Ίδρυμα Λογισμικού Apache (Apache Software Foundation).<sup>av.13</sup>

Ο Apache χρησιμοποιείται και σε τοπικά δίκτυα σαν διακομιστής συνεργαζόμενος με συστήματα διαχείρισης Βάσης Δεδομένων π.χ. Oracle, MySQL.

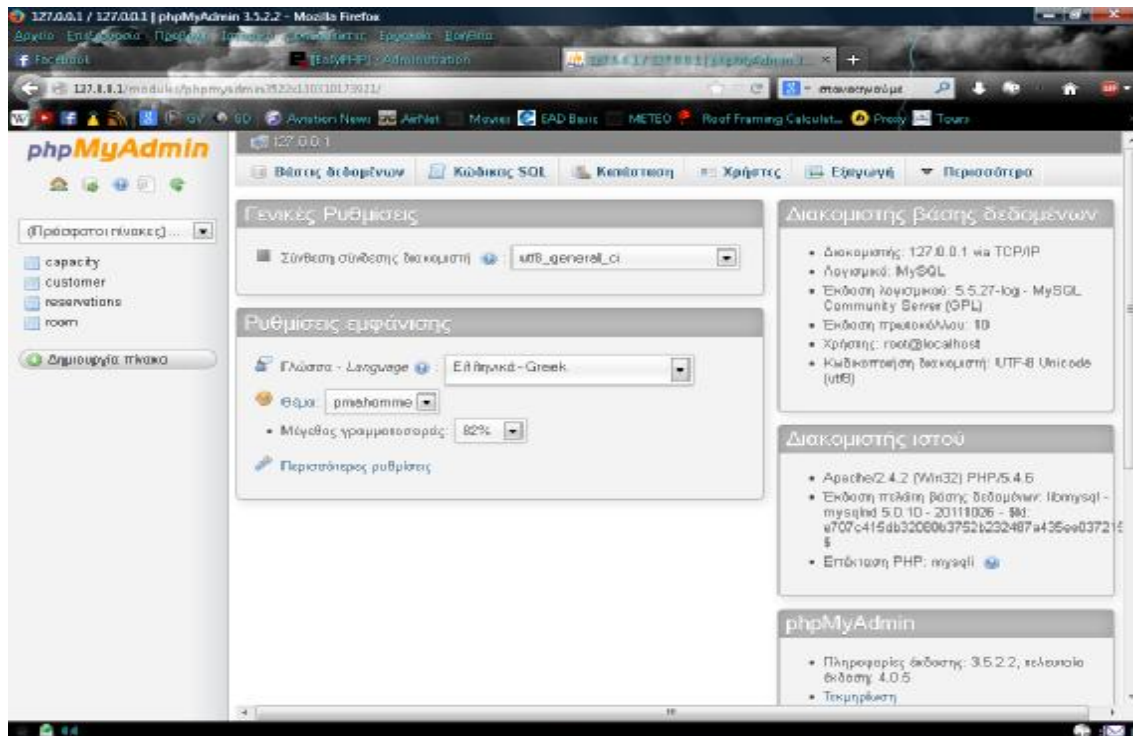




Στο παραπάνω διάγραμμα βλέπουμε το μερίδιο που διατηρεί ο Apache απέναντι στον ανταγωνισμό μέχρι και το 2010

### 4.1.3 Τι είναι το phpMyAdmin

Το **phpMyAdmin** είναι μια εφαρμογή LAMP γραμμένη ειδικά για τη διαχείριση εξυπηρετητών **MySQL**. Γραμμένο σε **PHP** και προσβάσιμο μέσω ενός περιηγητή ιστοσελίδων, το phpMyAdmin προσφέρει ένα γραφικό περιβάλλον για εργασίες διαχείρισης βάσεων δεδομένων.<sup>αν.14</sup>



### Εγκατάσταση

Πριν εγκαταστήσετε το **phpMyAdmin** θα χρειαστείτε πρόσβαση σε μια βάση δεδομένων **MySQL** είτε στον ίδιο υπολογιστή που είναι εγκατεστημένο το phpMyAdmin, ή σε έναν υπολογιστή προσβάσιμο μέσω δικτύου. Σε ένα τερματικό πληκτρολογήστε:

```
sudo apt-get install phpmyadmin
```

Στο τερματικό επιλέξτε ποιος εξυπηρετητής ιστού θα ρυθμιστεί για το **phpMyAdmin**. Το υπόλοιπο αυτής της ενότητας θα χρησιμοποιεί το **Apache2** για εξυπηρετητή ιστού.

Σε έναν περιηγητή πηγαίνετε στο `http://όνομα_εξυπηρετητή/phpmyadmin`, αντικαθιστώντας το `όνομα_εξυπηρετητή` με το πραγματικό όνομα του εξυπηρετητή. Στη σελίδα εισόδου, πληκτρολογήστε `root` για όνομα χρήστη, ή κάποιον άλλο χρήστη **MySQL**, αν έχετε κάνει κάποια ρύθμιση, και πληκτρολογήστε τον κωδικό πρόσβασης **MySQL** του χρήστη.

Μόλις συνδεθείτε, μπορείτε να επαναφέρετε τον κωδικό του root, αν χρειάζεται, να δημιουργήσετε χρήστες, να δημιουργήσετε/διαγράψετε βάσεις δεδομένων και πίνακες, κτλ.

## Ρυθμίσεις

Τα αρχεία ρύθμισης του **phpMyAdmin** βρίσκονται στο `/etc/phpmyadmin`. Το κύριο αρχείο ρυθμίσεων είναι το `/etc/phpmyadmin/config.inc.php`. Αυτό το αρχείο περιέχει ρυθμίσεις που ισχύουν για ολόκληρο το **phpMyAdmin**.

Για να χρησιμοποιήσετε το **phpMyAdmin** για να διαχειριστείτε μια βάση δεδομένων MySQL που φιλοξενείται σε έναν άλλον εξυπηρετητή, ρυθμίστε τα ακόλουθα στο `/etc/phpmyadmin/config.inc.php`:

```
$cfg['Servers'][$i]['host'] = 'db_server';
```



Αντικαταστήστε το `db_server` με το πραγματικό όνομα ή τη διεύθυνση IP του απομακρυσμένου εξυπηρετητή βάσεων δεδομένων. Επίσης, σιγουρευτείτε πως ο υπολογιστής που είναι εγκατεστημένο το **phpMyAdmin** έχει δικαιώματα πρόσβασης στην απομακρυσμένη βάση δεδομένων.

Μόλις το ρυθμίσετε, αποσυνδεθείτε από το **phpMyAdmin** και συνδεθείτε ξανά, και θα πρέπει να έχετε πρόσβαση στο νέο εξυπηρετητή.

Τα αρχεία `config.header.inc.php` και `config.footer.inc.php` χρησιμοποιούνται για την προσθήκη κεφαλίδας και υποσέλιδου HTML στο **phpMyAdmin**.

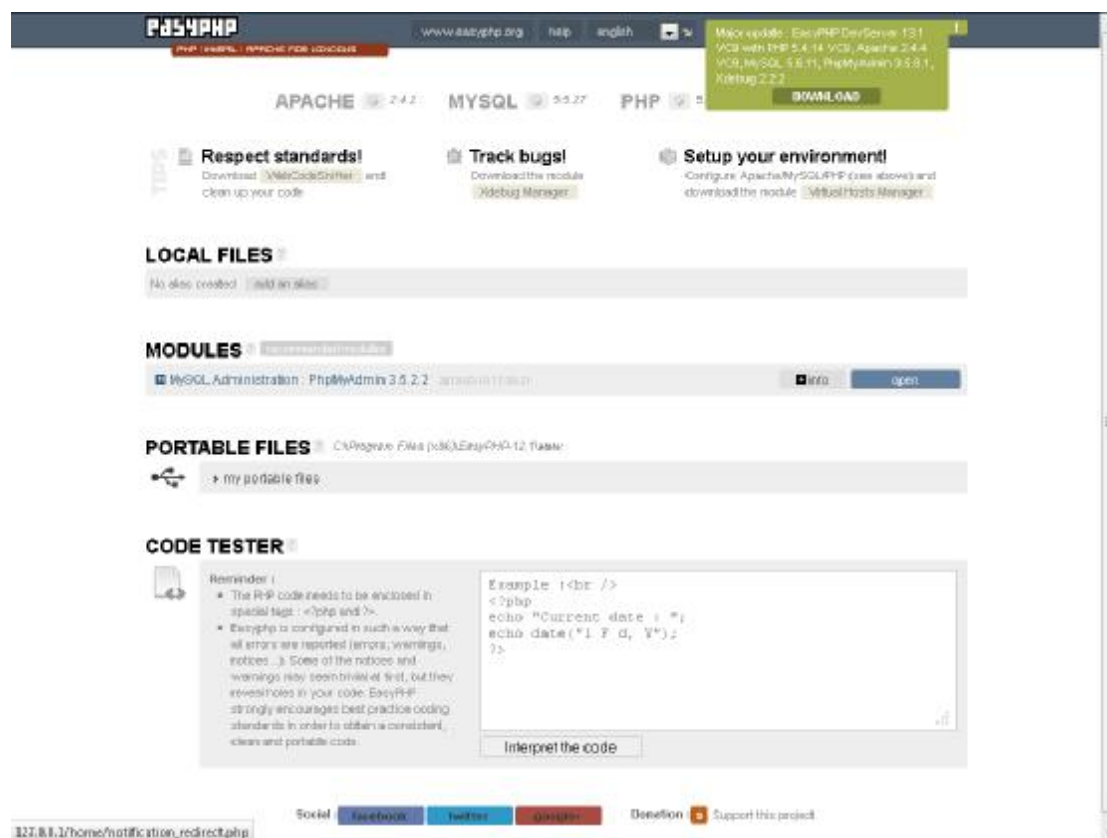
Ένα άλλο σημαντικό αρχείο ρυθμίσεων είναι το `/etc/phpmyadmin/apache.conf`, αυτό το αρχείο είναι συμβολικός σύνδεσμος στο `/etc/apache2/conf.d/phpmyadmin.conf` και χρησιμοποιείται για τη ρύθμιση του **Apache2** ώστε να παρέχει τον ιστότοπο του **phpMyAdmin**. Το αρχείο περιέχει οδηγίες για την φόρτωση της **PHP**, για δικαιώματα καταλόγων, κτλ.<sup>8</sup>



Περιβάλλον εργασίας του `phpmyadmin`. Στο κέντρο βλέπουμε τις διαθέσιμες βάσεις δεδομένων στον διακομιστή και αριστερά είναι ορατοί οι διάφοροι διαθέσιμοι πίνακες βάσεων δεδομένων.

## 4.1.4 Εμείς χρησιμοποιούμε EASYPHP

Εμείς λοιπόν χρησιμοποιούμε ο πακέτο EASYPHP PERSONAL WEB HOSTING 13.1. Είναι ίσως ο ποίο απλός και ποίο εύκολος τρόπος για να έχουμε ένα πλήρες σύστημα βάσης δεδομένων, όλα από ένα setup.



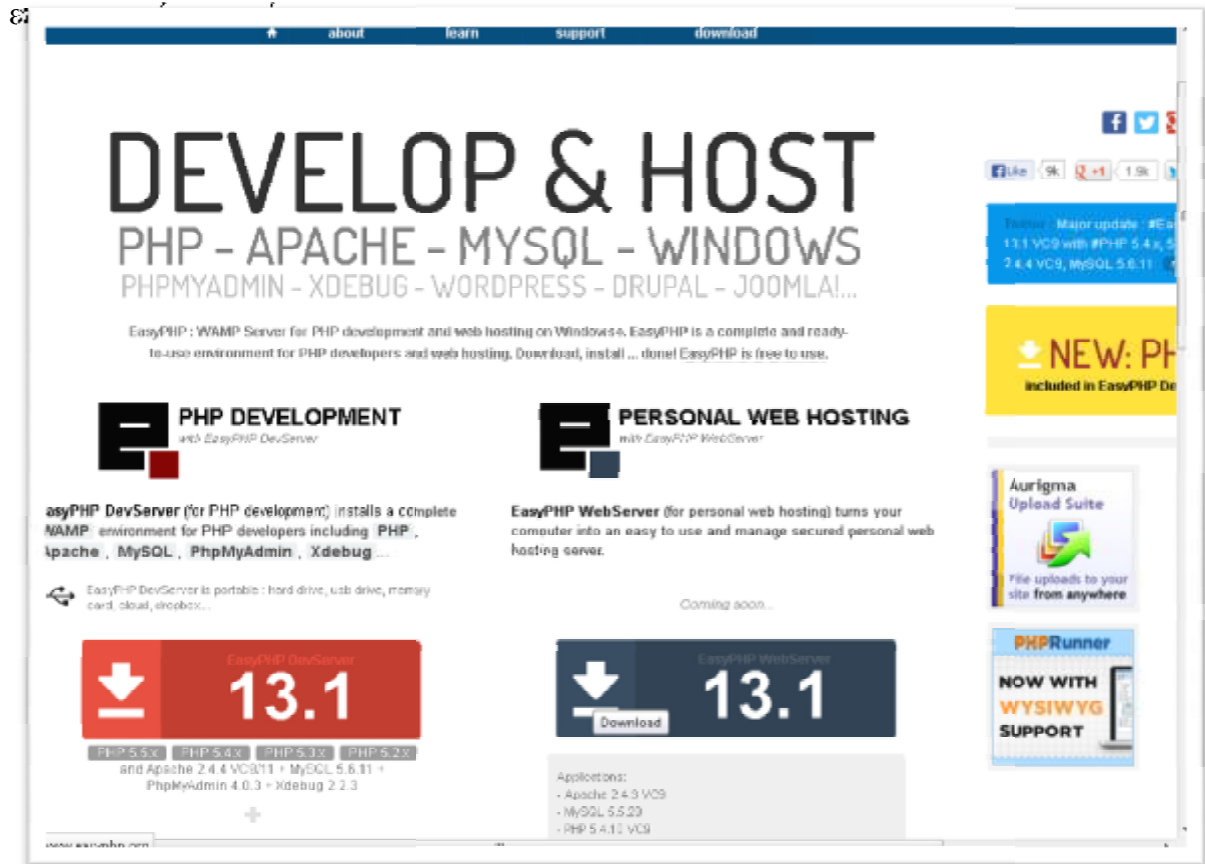
### Η έκδοση 13.1 περιέχει:

- Apache 2.4.3 VC9
- MySQL 5.5.29
- PHP 5.4.10 VC9
- PhpMyAdmin 3.5.5
- hMailServer
- FileZilla
- Webalizer
- RoundCube
- phpSysInfo
- ... και άλλα

### Η έκδοση 13.1 υποστηρίζει:

- Server Admin (backup, bandwidth Throttle, DNS Admin, permission Manager, PHPInfo, phpSysInfo, user shadowing, updates, cron Manager, password protected directories, metatag generator, service Status, webalizer stats, ...)
- Database management (MySQL Users, MySQL Databases, phpMyAdmin, ...)
- Domain management (domains, parked domains, sub-domains, DNS Manager, ...)
- Mail (aliases, distribution Lists, forwarders, mailboxes, webmail,...)
- FTP accounts
- Encrypted Accounts
- FTP Jailing
- Suhosin
- Open Base Directory
- ... και άλλα

Η εγκατάστασή δεν θα μπορούσε να είναι ποίο απλή. Απλά πηγαίνουμε στην ιστοσελίδα, κατεβάζουμε την νεότερη έκδοσή που μας αφορά, εγκαθιστούμε και

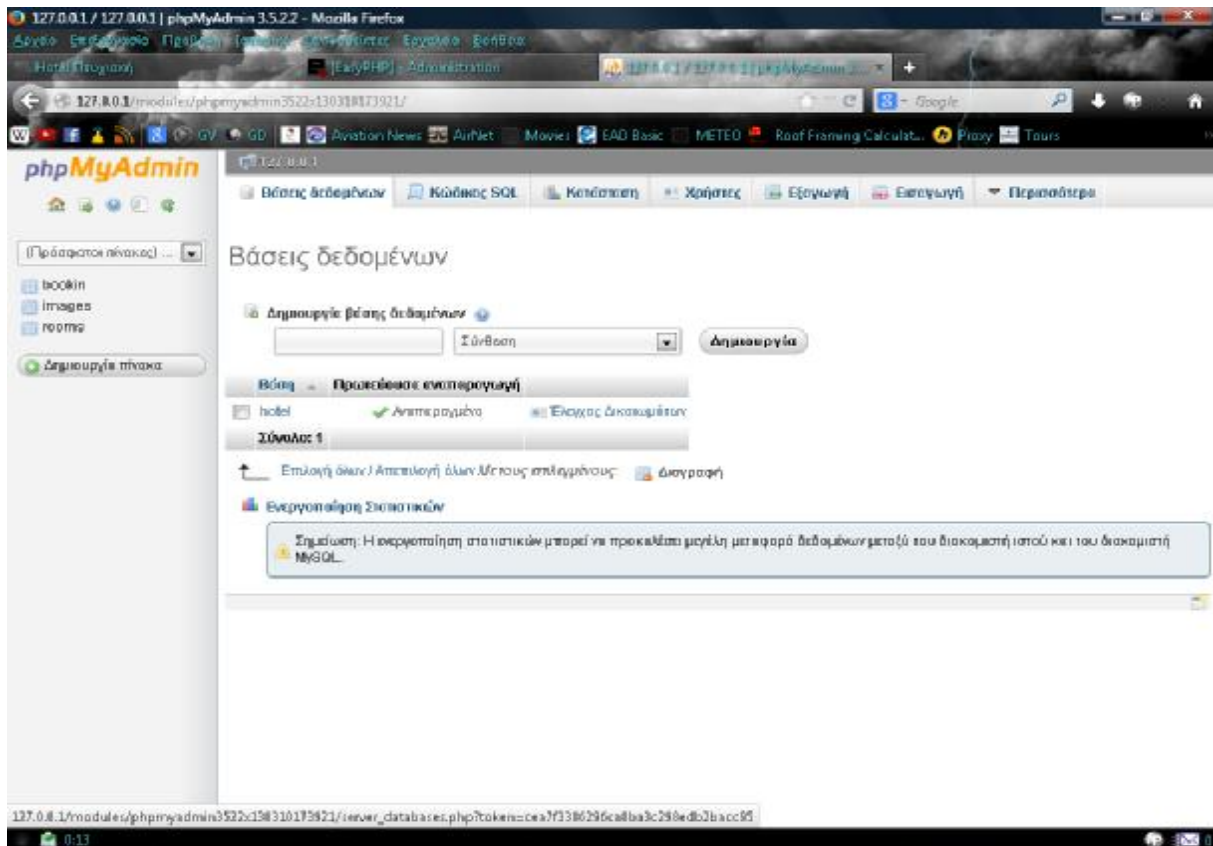


## 4.2 Βάση δεδομένων.

Ο ιστότοπος μας έχει φτιαχτεί ώστε να χρησιμοποιεί μια βάση δεδομένων. Αυτή η βάση δεδομένων με το ομολογουμένως ευφάνταστο όνομα hotel περιλαμβάνει 3 πίνακες.

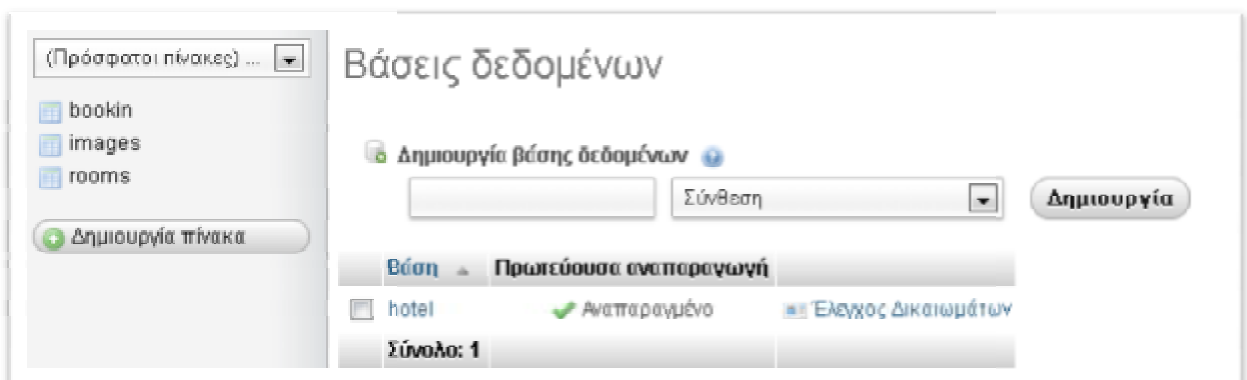
- **Bookin** - Ο πίνακας Bookin περιλαμβάνει όλα τα στοιχεία που χρειάζονται για να πραγματοποιηθεί μια κράτηση ενός δωματίου, στο ξενοδοχείο
- **Images** - Ο πίνακας Images περιλαμβάνει συνδέσμους για τις εικόνες που χρειάζεται η CSS προκειμένου να απεικονιστούν στον ιστότοπο.
- **Rooms** - Ο πίνακας Rooms περιλαμβάνει όλα τα στοιχεία του συνόλου των δωματίων, τα οποία χρειάζονται για να πραγματοποιηθεί μια κράτηση ενός δωματίου, στο ξενοδοχείο

Στην ιστοσελίδα έχουν πρόσβαση όλοι οι χρήστες – επισκέπτες της ιστοσελίδας, ενώ ο διαχειριστής διαθέτει και δικαιώματα επεξεργασίας των πληροφοριών. Οι χρήστες έχουν το δικαίωμα να καταχωρούν μόνο, ενώ ο διαχειριστής έχει το δικαίωμα να καταχωρεί, να ενημερώνει, να διαγράφει και να μετακινεί τις απεσταλμένες πληροφορίες ύστερα από κάποιους ελέγχους. Τα δεδομένα πριν αποθηκεύονται στις βάσεις δεδομένων επαληθεύονται για την συν-τακτική ορθότητα τους, για προστασία από εσφαλμένες αποθηκεύσεις. Μια εσφαλμένη αποθήκευση μπορεί να καταστρέψει τη λογική ιεραρχία των πληροφοριών στη βάση και συνεπώς, να μην είναι προσβάσιμες από το σύστημα.



Στην παραπάνω φωτογραφία βλέπουμε την βάση δεδομένων “**hotel**” στο κέντρο της σελίδας. Αυτή τη στιγμή η βάση δεδομένων τρέχει σε τοπικό επίπεδο με τη χρήση του πακέτου Easyphp 12 που αναφέραμε στο δεύτερο κεφάλαιο. Μέσω της εφαρμογής phpmyadmin μπορούμε να διαχειριστούμε την βάση δεδομένων τροποποιώντας οπουδήποτε περιεχόμενο στους πίνακες του. Επίσης από αυτό το σημείο μπορούμε πολύ εύκολα να προσθέσουμε νέους πίνακες, να μεταφέρουμε/συγχωνεύσουμε πίνακες από άλλες βάσεις δεδομένων ή ακόμα και να τις διαγράψουμε.

Στο αριστερό μέρος της φωτογραφίας φαίνονται οι πίνακες που αναφέραμε παραπάνω και αποτελούν το σύστημα διαχείρισης του ξενοδοχείου.



Από αυτούς του πίνακες της βάσης δεδομένων, οι διαχειριστές και οι επισκέπτες μπορούν να υποβάλουν οποιαδήποτε ερωτήματα τα οποία θέλουν. Χαρακτηριστικά, ένα πελάτης μπορεί μέσω της φόρμας στην κεντρική σελίδα να αντλήσει πληροφορίες

διαθεσιμότητα, τις τιμές, το ποσοστό πληρότητας και τις τιμές των δωματίων, για μια συγκεκριμένη ημερομηνία.

Από την άλλη ο διαχειριστής, ανάλογα με τον βαθμό του μπορεί με μια ανεξάρτητη εφαρμογή αν αντλεί ανά πάσας στιγμή πληροφορίες για την πληρότητα, να μεταβάλλει τις τιμές, να πληροφορείτε για τυχόν νέες κρατήσεις ακυρώσεις, να βρίσκει στοιχεία πελατών και πληρωμών καθώς και να ακυρώνει ο ίδιος κρατήσεις. Αξίζει να σημειωθεί πως η ανάπτυξη μιας τέτοιας εφαρμογής, δεν πραγματοποιήθηκε από εμάς διότι θεωρήσαμε πως είναι εκτός του θέματος της πτυχιακής.

### 4.2.1 Συνοπτική περιγραφή πίνακα Bookin

Ο πίνακας Bookin περιλαμβάνει όλα τα στοιχεία μιας κράτησης. Εδώ συμπεριλαμβάνονται οτιδήποτε χρειάζεται ένας διαχειριστής για να διαβάσει, να επεξεργαστεί ή να τροποποιήσει καποιος διαχειριστής για την συνήθη λειτουργία του ξενοδοχείου. Επίσης από εδώ ένας διαχειριστή μπορεί να αντλήσει στοιχεία επικοινωνίας με τον πελάτη καθώς και στατιστικά στοιχεία για αυτόν.

#### Ο πίνακας Bookin αποτελείτε από 9 πεδία:

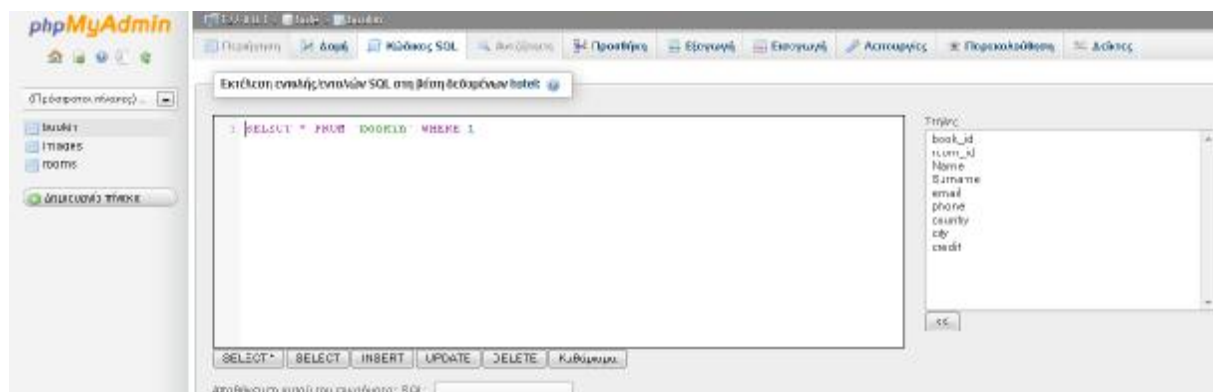
1. Book\_id – Είναι ο σειριακός κωδικός που λαμβάνει κάθε κράτηση, επίσης στον πίνακα αυτόν έχει τον ρόλο του μοναδικού πρωτεύων κλειδιού.
2. Room\_id – Είναι ο κωδικός του δωματίου και χρησιμοποιείται ως πρωτεύων κλειδί στον πίνακα Rooms.
3. Name – Σε αυτό το πεδίο αποθηκεύεται το μικρό όνομα του πελάτη ο οποίος πραγματοποιεί την κράτηση.
4. Surname- Σε αυτό το πεδίο αποθηκεύεται το Επώνυμο του πελάτη ο οποίος πραγματοποιεί την κράτηση.

#	Όνομα	Τύπος	Σύμπτωση	Χαρακτηριστικό Κενό	Προεπιλογή	Πρόσθετα	Ενέργεια
1	book_id	int(11)		Όχι	Καμία		Αλλαγή Διαγραφή Περισσότερα
2	room_id	int(11)		Όχι	Καμία		Αλλαγή Διαγραφή Περισσότερα
3	Name	char(50)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
4	Surname	char(50)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
5	email	char(250)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
6	phone	char(30)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
7	country	char(250)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
8	city	char(250)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα
9	credit	char(250)	utf8_general_ci	Ναι	NULL		Αλλαγή Διαγραφή Περισσότερα

5. @mail - Σε αυτό το πεδίο αποθηκεύεται διεύθυνση ηλεκτρονικού ταχυδρομείου του πελάτη. Χρησιμοποιείτε αυτόματα από το σύστημα για την επιβεβαίωση της κράτησης.
6. Phone - Σε αυτό το πεδίο αποθηκεύεται το τηλέφωνο επικοινωνίας που έχει δηλώσει ο πελάτης κατά την πραγματοποίηση της κράτησης του δωματίου.
7. Country - Σε αυτό το πεδίο αποθηκεύεται η χώρα προέλευσης του πελάτη
8. City - Σε αυτό το πεδίο αποθηκεύεται η διεύθυνση του πελάτη.
9. Credit - Σε αυτό το πεδίο αποθηκεύεται ο αριθμός και ο τύπος πιστωτικής κάρτας που χρησιμοποιείται για την πραγματοποίηση της κράτησης.

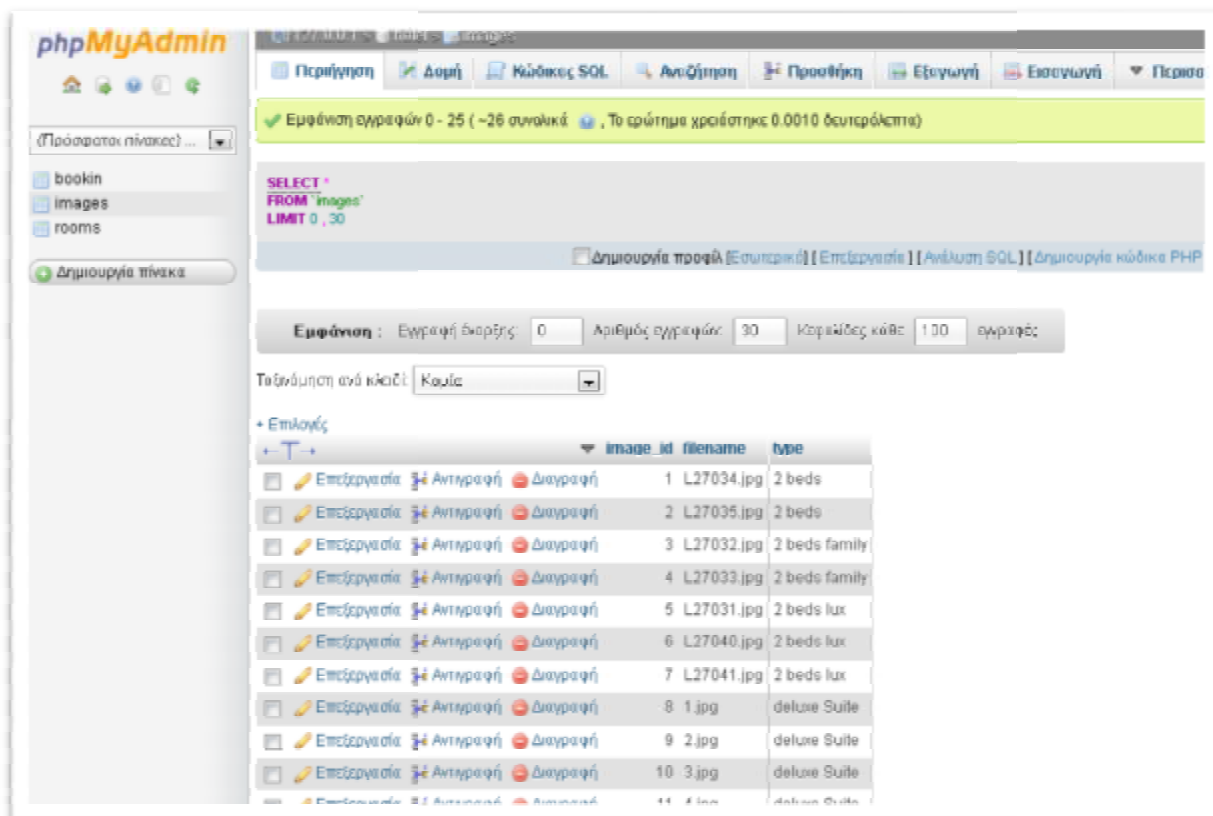
### Λειτουργία του πίνακα Bookin

Ο επισκέπτης καταφεύγοντας στην φόρμα η οποία βρίσκεται στην αρχική σελίδα του ιστότοπου επιλέγει τα δεδομένα που χρειάζονται για να πραγματοποιηθεί μια κράτηση ακολουθώντας την διαδικασία όπως αυτή παρουσιάστηκε στο κεφάλαιο 3. Αφού γίνει λοιπόν επιτυχώς η κράτηση του, συμπληρώνεται ο πίνακας αυτός με τα στοιχεία του πελάτη και τα στοιχεία της συναλλαγής και αποστέλλεται απαντητικό μήνυμα ηλεκτρονικού ταχυδρομείου στον πελάτη και στον διαχειριστή. Τα στοιχεία αυτά παραμένουν στην βάση δεδομένων ως ιστορικό σημείο αναφοράς. Εάν κάποια κράτηση διαγραφεί ή ακυρωθεί τότε η επόμενη κράτηση θα καταλάβει των αμέσως επόμενο διαθέσιμο αύξοντα αριθμό.



### 4.2.2 Συνοπτική περιγραφή πίνακα images

Ο πίνακας images αποτελεί την δεξαμενή της CSS για όλες τις εικόνες και τα icons που χρησιμοποιούνται για το εικαστικό αποτέλεσμα της σελίδας κατά τη διενέργεια της επιλογής και της κράτησης ενός δωματίου. Κάθε πεδίο περιλαμβάνει και ένα σύνδεσμο στον φάκελο images στον διακομιστή.



### Ο πίνακας περιλαμβάνει 3 πεδία:

Image\_id – Είναι ο αύξωντος αριθμός που αποδίδεται σε κάθε εγγραφή στον πίνακα. Χρησιμεύει ως πρωτεύων κλειδί στον πίνακα αυτόν και τον συναντάμε επίσης στον πίνακα Rooms να συνδέει τους 2 πίνακες



1. Filename – Είναι η ονομασία του αρχείου μέσα στον φάκελο images.
2. Type –Αναγράφεται η κατηγορία του δωματίου που απεικονίζεται στην αντίστοιχη φωτογραφία.

### Που χρησιμοποιούνται αυτές οι φωτογραφίες:

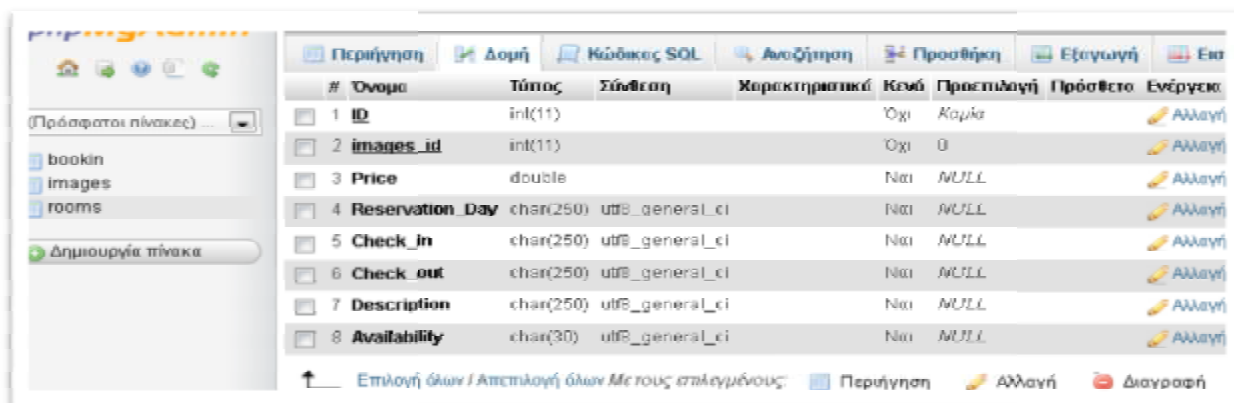
Οι φωτογραφίες αυτές εμφανίζονται στο μενού επιλογής δωματίων που παρουσιάζεται στον επισκέπτη. Με την βοήθεια αυτών των φωτογραφιών προσπαθούμε να διευκολύνουμε τον επισκέπτη εντοπίσει και να διαλέξει το κατάλληλο δωμάτιο για τον ίδιο και για την οικογένεια του. Κάθε δωμάτιο έχει και την δικιά του ξεχωριστή φωτογραφία



(αν και σε μερικά δωμάτια κάποιες φωτογραφίες είναι οι ίδιες διευθύνονται στην ίδια κατηγορία.)

### 4.2.3 Συνοπτική περιγραφή πίνακα Rooms

Εδώ περιλαμβάνονται όλα τα δεδομένα για τα δωμάτια του ξενοδοχείου καθώς στοιχεία για τον προγραμματισμό και την κατάσταση της διαθεσιμότητας τους.



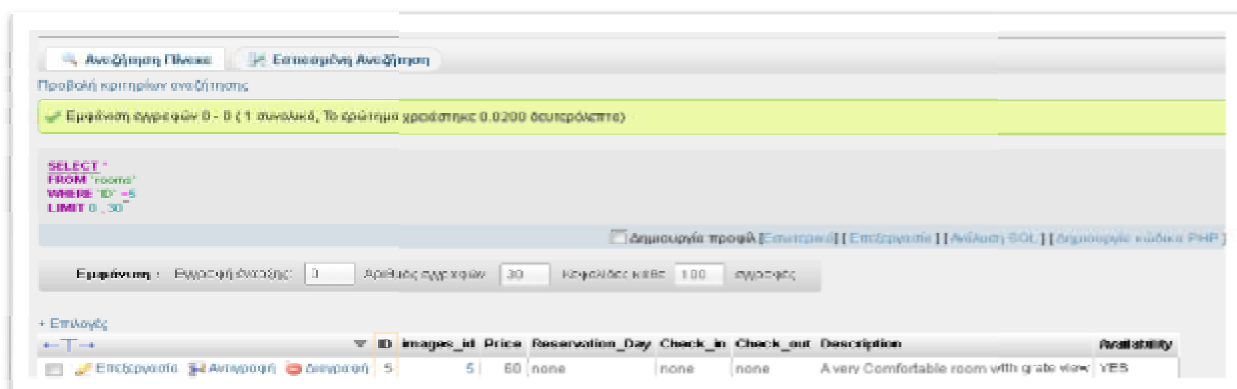
#	Όνομα	Τύπος	Σύνθεση	Χαρακτηριστικά	Κενό	Προεπιλογή	Πρόσθετο	Ενέργεια
<input type="checkbox"/>	1 ID	int(11)			Όχι	Καμία		Αλλαγή
<input type="checkbox"/>	2 images_id	int(11)			Όχι	0		Αλλαγή
<input type="checkbox"/>	3 Price	double			Ναι	NULL		Αλλαγή
<input type="checkbox"/>	4 Reservation_Day	char(250)	utf8_general_ci		Ναι	NULL		Αλλαγή
<input type="checkbox"/>	5 Check_in	char(250)	utf8_general_ci		Ναι	NULL		Αλλαγή
<input type="checkbox"/>	6 Check_out	char(250)	utf8_general_ci		Ναι	NULL		Αλλαγή
<input type="checkbox"/>	7 Description	char(250)	utf8_general_ci		Ναι	NULL		Αλλαγή
<input type="checkbox"/>	8 Availability	char(30)	utf8_general_ci		Ναι	NULL		Αλλαγή

Ο πίνακας αποτελείται από τα εξής 8 πεδία:

1. ID – Αναγράφεται ο αριθμός του δωματίου
2. Image\_ID – Αναγράφεται ο κωδικός της φωτογραφίας από τον πίνακα images.
3. Price – Αναγράφεται η τρέχουσα τιμή διανυκτέρευσης αυτού του δωματίου, για άμεση κράτηση.
4. Reservation\_day – Αναγράφεται η ημερομηνία στην οποία έγινε η τελευταία κράτηση
5. Check\_in - Αναγράφεται η ημερομηνία έναρξης της διαμονής
6. Check\_out - Αναγράφεται η ημερομηνία τερματισμού της διαμονής
7. Description – Περιλαμβάνει μια σύντομη περιγραφή του κάθε δωματίου.
8. Availability – Αναγράφει την κατάσταση άμεσης διαθεσιμότητας του κάθε δωματίου.

#### Σε τι χρεισιμεύει αυτός ο πίνακας.

Αυτός ο πίνακας είναι ο κύριος διαχειριστικός πίνακας του ξενοδοχείου. Από εδώ ο διαχειριστής μπορεί να εισάγει δωμάτια, να βγάλει δωμάτια ως μη διαθέσιμα και να επεξεργαστεί τιμές και επιλογές για μελλοντικές κρατήσεις. Επίσης από αυτόν τον πίνακα ο διαχειριστής έχει μια πλήρη εικόνα για την κατάσταση του ξενοδοχείου την πληρότητα, την κερδοφορία και δύναται να πραγματοποιήσει στοχευόμενες αναζητήσεις όπως στην φωτογραφία που ακολουθεί.



Αποτέλεσμα διαθεσιμότητας για το δωμάτιο 5

## 4.3 Διάγραμμα και περιγραφή Δραστηριοτήτων Κράτησης

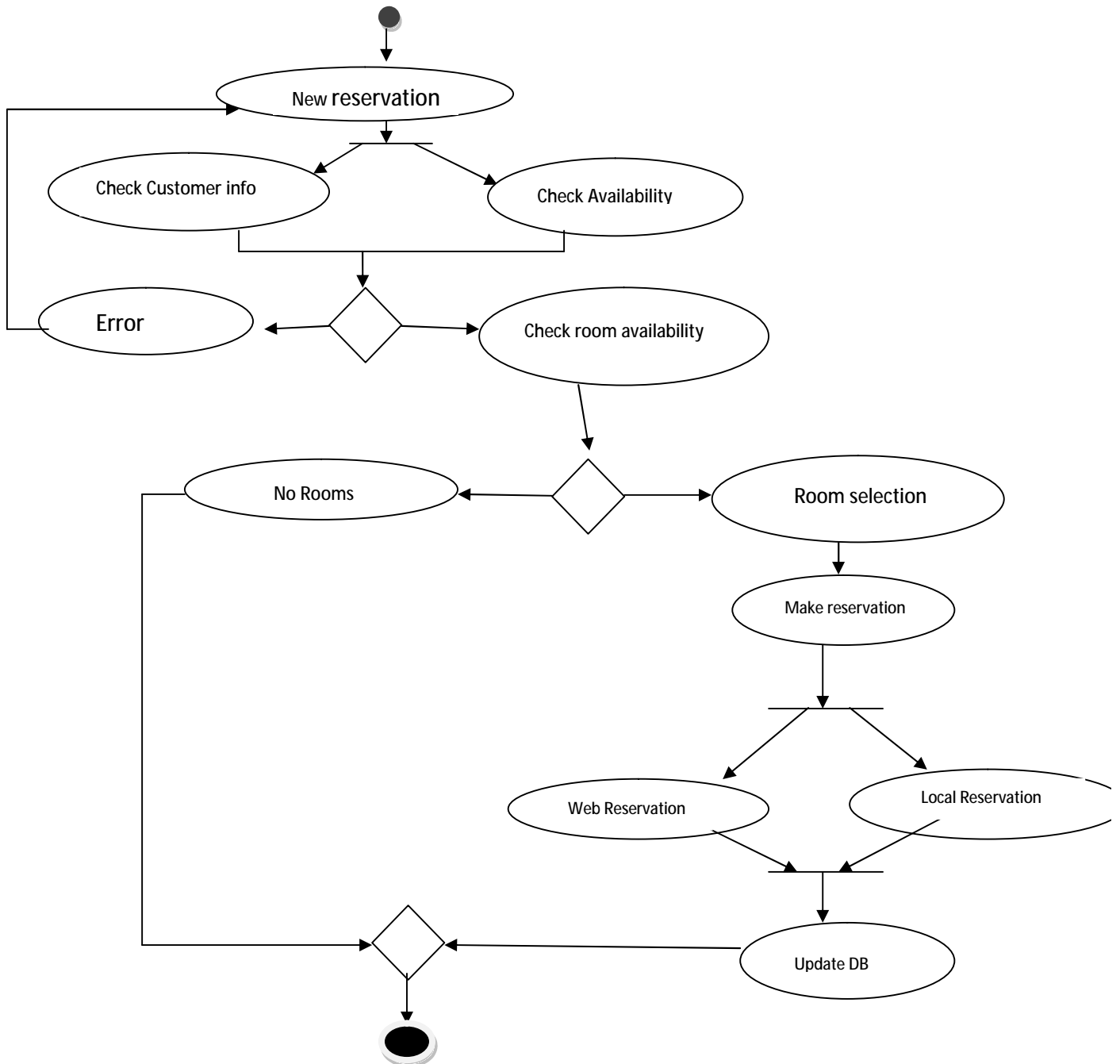
### 4.3.1 περιγραφή δραστηριότητας κράτησης

Αρχικά σύστημα επεξεργάζεται τα στοιχεία που έχει δώσει ο χρήστης για μια νέα κράτηση και τα συγκρίνει με τα δεδομένα του δωματίων. Εάν υπάρξει πρόβλημα τότε η διαδικασία σταματά και ο επισκέπτης καλείτε να επανεισάγει τα στοιχεία για την κράτηση. Εάν η επεξεργασία των στοιχείων είναι επιτυχής τότε ελέγχεται η διαθεσιμότητα δωματίων. Εάν δεν υπάρχουν διαθέσιμα δωμάτια τότε η διαδικασία ολοκληρώνεται χωρίς ο επισκέπτης να μπορεί να ολοκληρώσει κάποια κράτηση. Εάν υπάρχει διαθεσιμότητα τότε δύναται να προχωρήσει στην επιλογή του δωματίου της αρεσκείας του. Μετά την επιλογή του δωματίου γίνεται και η κράτηση.

Η κράτηση μπορεί να ολοκληρωθεί με 2 τρόπους. Είτε ηλεκτρονικά μέσω της ιστοσελίδας, είτε από τον διαχειριστή χειροκίνητα ύστερα από επικοινωνία του πελάτη με τον διαχειριστή. Τέλος ενημερώνεται η Βάση δεδομένων με τα στοιχεία της κράτησης και ολοκληρώνετε η διαδικασία.

### 4.3.2 Σχήμα διαγραμματικής απεικόνισης διαδικασία μιας κράτησης.

ΑΡΧΗ Εισαγωγή αιτήματος επισκέπτη



### 4.3.3 Κανονικοποίηση

Για την επιτυχή κανονικοποίηση των δεδομένων της εφαρμογής πρέπει να ακολουθηθούν οι εξής φάσεις:

A) Εύρεση των πεδίων που χρειάζονται για την συγκεκριμένη εφαρμογή και την επεξήγησή τους, ώστε να δημιουργηθεί μια καλύτερη εικόνα για την εφαρμογή.

B) Οργάνωση των πεδίων σε μη κανονική μορφή (MKM). Εύρεση επαναλαμβανόμενων ομάδων δεδομένων.

Γ) Μετατροπή της MKM σε 1η Κανονική Μορφή (1NF). Απομάκρυνση όλων των επαναλαμβανόμενων ομάδων δεδομένων. Επιλογή μοναδικών κλειδιών για κάθε επαναλαμβανόμενη ομάδα και για ολόκληρη τη δομή δεδομένων.

Δ) Μετατροπή 1NF σε 2η Κανονική Μορφή (2NF). Εφόσον υπάρχουν σύνθετα κλειδιά θα πρέπει κάθε χαρακτηριστικό της δομής εκτός κλειδιού να εξαρτάται από ολόκληρο το κλειδί της δομής και όχι από τμήμα του κλειδιού της δομής.

Ε) Μετατροπή 2NF σε 3<sup>η</sup> Κανονική Μορφή (3NF). Εύρεση εξαρτήσεων μεταξύ των χαρακτηριστικών εκτός κλειδιού της δομής ή μεταξύ των χαρακτηριστικών τα οποία συμμετέχουν στο σχηματισμό του κλειδιού της δομής.

A) Επεξήγηση κάθε πεδίου, το οποίο χρησιμοποιείται στη Βάση Δεδομένων για την εφαρμογή.

Όνομα Πεδίου	Επεξήγηση
Customer ID	Ο κωδικός του πελάτη προσδιορίζει μονοσήμαντα τον κάθε πελάτη. Δεν υπάρχουν πελάτες με τον ίδιο κωδικό.
First Name	Αναφέρεται στο όνομα του πελάτη. Υπάρχει η περίπτωση περισσότεροι του ενός πελάτες να έχουν το ίδιο όνομα.
Last Name	Αναφέρεται στο επίθετο του πελάτη. Υπάρχει η περίπτωση περισσότεροι του ενός πελάτες να έχουν το ίδιο όνομα.
Email	Αναφέρεται στην ηλεκτρονική διεύθυνση του πελάτη.
Country	Αναφέρεται στην χώρα καταγωγής του πελάτη.
Address	Αναφέρεται στην ταχυδρομική διεύθυνση του πελάτη.

City	Αναφέρεται στην πόλη όπου ο πελάτης έχει μόνιμη κατοικία.
Postal code	Αναφέρεται στον ταχυδρομικό κώδικα του πελάτη.
Phone number	Αναφέρεται στο τηλεφωνικό νούμερο του πελάτη.
Name on card	Αναφέρεται στο όνομα πάνω στη κάρτα
Card number	Αναφέρεται στο κωδικό της κάρτας.
Card number key side	Αναφέρεται στο κώδικα ασφαλείας της κάρτας
Room ID	Ο κωδικός του δωματίου προσδιορίζει μονοσήμαντα το κάθε δωμάτιο. Δεν υπάρχουν δωμάτια με τον ίδιο κωδικό.
Select rate	Επιλογή τύπου χρέωσης
Room type	Καθορίζει τον τύπο του δωματίου
Type of rate	Καθορίζει τον τύπο της χρέωσης
Value of rate	Καθορίζει την αξία της χρέωσης
Number of beds	Περιγράφει τον αριθμό των κλινών(1 διπλό κρεβάτι μετράτε ως 2 κλίνες)
Capacity ID	Περιγράφει μονοσήμαντα τον κωδικό της χωρητικότητας
Type of room	Αναφέρεται στον τύπο του δωματίου
Capacity	Περιγράφει την χωρητικότητα
Number of Room	Αναφέρει τον αριθμό του δωματίου
Customer ID	Περιγράφει μονοσήμαντα τον Κωδικό του πελάτη
Room ID	Περιγράφει μονοσήμαντα τον κωδικό του δωματίου
Capacity ID	Περιγράφει μονοσήμαντα τον κωδικό της χωρητικότητας
Dates of reserve	Αναφέρεται στην ημέρα της κράτησης
Reservation ID	Περιγράφει μονοσήμαντα την κράτηση του

Μη Κανονική Μορφή Δομής.

Customer ID
First Name
Last Name
Email
Country
Address
City
Postal code
Phone number
Name on card
Card number
Card number key side
Room ID
Select rate
Room type
Type of rate
Value of rate
Number of beds
Capacity ID
Type of room
Capacity
Number of Room
Dates of reserve

Γ) Μετατροπή της MKM σε 1η Κανονική Μορφή (1NF).

Customer ID
First Name
Last Name
Email
Country
Address
City
Postal code
Phone number
Name on card
Card number
Card number key side
Room ID
Select rate
Room type
Type of rate
Value of rate
Number of beds
Capacity ID
Type of room
Capacity
Number of Room
Reservation ID
Dates of reserve

Παρατηρούμε ότι το δεδομένο Dates of reserve για να χρησιμοποιηθεί και να περιγράψει την ημέρα κράτησης ενός δωματίου κάθε ξεχωριστού πελάτη θα πρέπει να καταχωρηθεί δε διαφορετικό πίνακα Reservation με κύριο κλειδί Reservation ID.

Δ) Μετατροπή της 1NF σε 2NF.

<b>Customer ID</b>
First Name
Last Name
Email
Country
Address
City
Postal code
Phone number
Name on card
Card number
Card number key side
<b>Room ID</b>
Select rate
Room type
Type of rate
Value of rate
Number of beds
<b>Capacity ID</b>
Type of room
Capacity
Number of Room
<b>Reservation ID</b>
<b>Capacity ID</b>
<b>Room ID</b>
<b>Customer ID</b>
Dates of reserve

Παρατηρούμε ότι το δεδομένο Dates of reserve για να χρησιμοποιηθεί και να περιγράψει την ημέρα κράτησης ενός δωματίου κάθε ξεχωριστού πελάτη θα πρέπει να καταχωρηθεί δε διαφορετικό πίνακα Reservation με κύριο κλειδί Reservation ID, με ξένα κλειδιά Capacity ID, Room ID, Customer ID, τα οποία προσδιορίζουν μονοσήμαντα τα



δεδομένα στους πίνακες που περιγράφουν και σε συνδυασμό με το κύριο κλειδί Reservation ID αποτελούν ένα σύνθετο κλειδί του πίνακα Reservation

Ε) Εφόσον δεν υπάρχουν εξαρτήσεις μεταξύ των χαρακτηριστικών εκτός κλειδιού της δομής η 2<sup>η</sup> κανονική μορφή είναι αυτόματα και σε 3<sup>η</sup> κανονικά μορφή (3NF).

#### 4.3.4 Περιγραφή Συστήματος

Ένα σύστημα διαχείρισης δωματίων έχει σε γενικές γραμμές τις ακόλουθες απαιτήσεις:

1) **Καταχώρηση Στοιχείων:** Με την εγκατάσταση του συστήματος καταχωρούνται αρχικά όλα τα δωμάτια τα οποία διατίθενται προς ενοικίαση ή μακροχρόνια μίσθωση. Κάθε δωμάτιο χαρακτηρίζεται από τον κωδικό του, τις ενδεικτικές τιμές, τον τύπο δωματίου, τον τύπο της χρέωσης, την αξία της χρέωσης και τον αριθμό κλινών. Για κάθε πελάτη το σύστημα θα πρέπει να θυμάται το ονοματεπώνυμο του, την διεύθυνσή του, τον αριθμό τηλεφώνου του και τους κωδικούς της πιστωτικής Κάρτας καθώς και το Email.

2) **Ενοικίαση:** Ο πελάτης, έχει την δυνατότητα να ενοικιάζει ένα ή περισσότερα δωμάτια με δυο τρόπους.

A) Ζητώντας επί τόπου από τον υπάλληλο της ρεσεψιόν ή τηλεφωνώντας.

B) Αναζητώντας μέσω διαδικτύου από το σπίτι και να καταχωρήσει τη κράτηση.

3) **Αναχώρηση:** Αναχώρηση γίνεται αποκλειστικά με φυσική παρουσία στη ρεσεψιόν, όπου εξετάζονται οι ακόλουθες περιπτώσεις:

- Υπολογίζεται η αξία της διαμονής του πελάτη στο ξενοδοχείο και εάν κάνει check-out πριν τις 12:00 το μεσημέρι σύμφωνα με τους όρους της συμφωνίας κρατήσεως.
- Εξετάζεται η χρήση εξτρά υπηρεσιών του ξενοδοχείου (Μινι μπαρ, τηλέφωνο, πρωινό)
- Φθορά που μπορεί να προέκυψε στο δωμάτιο ή στον εξοπλισμό του, οι οποίες βαρύνουν τον πελάτη.

Η εξόφληση γίνεται μετρητοίς, με πιστωτική κάρτα αλλά και με πίστωση του λογαριασμού του πελάτη, για εξόφληση κατά τον επόμενο δανεισμό του.

4) **Απόσυρση δωματίου / κλινών:** Το σύστημα εξετάζει κατά χρονικά διαστήματα τις κινήσεις των δωματίων και των κλινών. Έτσι αν κάποια στιγμή κάποιο από το δωμάτιο ή τα κρεβάτια κριθούν ακατάλληλα ή χρίζουν επισκευής / συντήρησης τότε ενημερώνουν τους δίφορους διαχειριστές για τη μετατροπή.

5) **Διόρθωση:** υπάλληλος έχει την δυνατότητα να διορθώνει στοιχεία πελατών ή των δωματίων σε περίπτωση που έχουν εισαχθεί λανθασμένα, δεν είναι έγκυρα ή επήλθε μεταβολή τους.

6) **Μηνιαίος Έλεγχος:** Ο υπάλληλος του καταστήματος έχει την δυνατότητα να ελέγξει τις κινήσεις που πραγματοποιήθηκαν κάθε μήνα. Ανακτά πληροφορίες όπως ποιοι πραγματοποίησαν κινήσεις ένα συγκεκριμένο μήνα, ποιες η απολαβές του ξενοδοχείου από τους πελάτες, τι υλικό υπάρχει σε κάθε δωμάτιο.

7) **Έλεγχος μίνι μπαρ και έξτρα υπηρεσιών:** Το σύστημα δίνει την δυνατότητα στον υπάλληλο να ελέγχει τα αποθέματα του μινι-μπαρ. Οι πληροφορίες που διατίθενται είναι το σύνολο των προϊόντων που έχει το μινι μπαρ. Εξτρά υπηρεσίες μπορεί να περιλαμβάνουν τηλεφωνήματα, μεταφορά, εξυπηρέτηση δωματίου, κατανάλωσή στο μπάρ με πίστωση, πρωινό κλπ

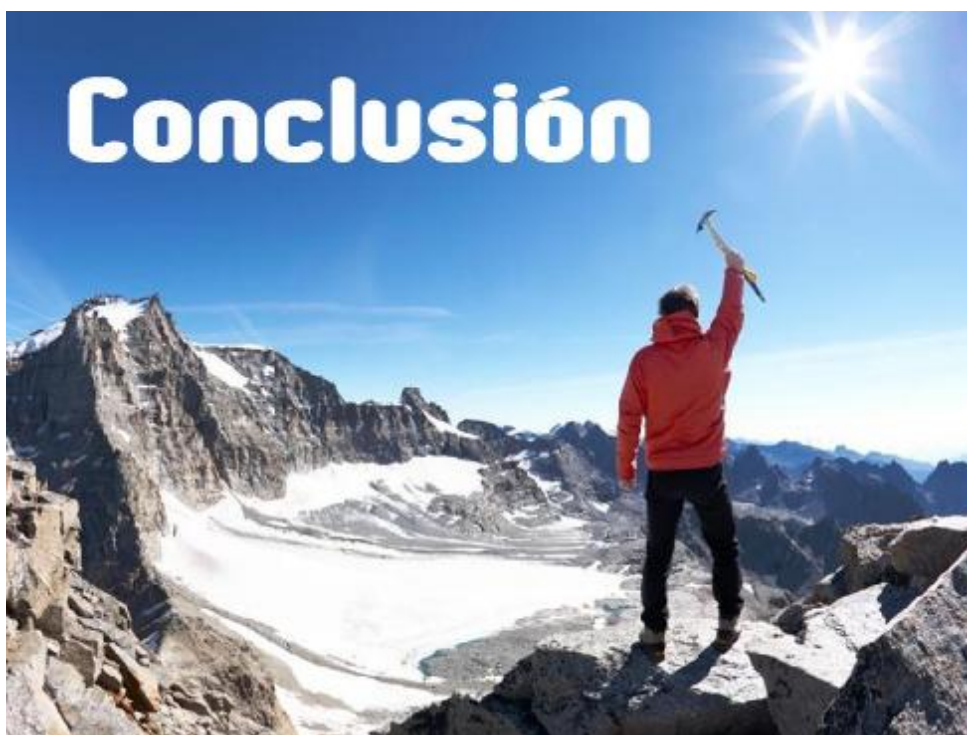
8) **Έλεγχος Κέρδους:** Το σύστημα δίνει την δυνατότητα στον υπάλληλο να πληροφορηθεί για τα κέρδη του ξενοδοχείου (συνολικά κέρδη, κέρδη από δανεισμούς, κέρδη από αγορές).

.....

- A) Καταχώρηση Στοιχείων
- B) Κράτηση
- Γ) Ακύρωση κράτησης
- Δ) Απόσυρση διαθεσιμότητας δωματίου
- E) Έλεγχος κέρδους
- ΣΤ) Εύρεση κράτησης
- Z) Διόρθωση κράτησης

# ΚΕΦΑΛΑΙΟ 5

## Συμπεράσματα



## ΚΑΙΦΑΛΛΙΟ 5 - ΣΥΜΠΕΡΑΣΜΑΤΑ

Η παρούσα εργασία αποτελεί μια πρόταση για τη δημιουργία δικτυακού τόπου με την χρήση διαδικτυακών εφαρμογών που επιτρέπουν την online τροποποίηση του περιεχομένου ενός δικτυακού τόπου, εννοείτε λογισμικό του οποίου ο πηγαίος κώδικας διατίθεται με κάποιο τρόπο ελεύθερα σε όσους ζητούν να τον εξετάσουν, ακόμα και να τον τροποποιήσουν ή αξιοποιήσουν σε άλλες εφαρμογές. Το λογισμικό ανοιχτού κώδικα δεν σημαίνει απαραίτητως δωρεάν λογισμικό, αλλά αναφέρεται στο γεγονός πως επιτρέπεται σε κάθε χρήστη να εξετάσει και να χρησιμοποιήσει τη γνώση και τις δυνατότητες που του προσφέρει ο παρεχόμενος πηγαίος κώδικας.

Τα τελευταία χρόνια οι εξελίξεις στο χώρο του διαδικτύου καθώς και η οικονομική ύφεση καταστούν ακόμα πιο χρήσιμη την εξάπλωση μιας επιχείρησης στο χώρο του διαδικτύου. Η εικόνα της επιχείρησης θα πρέπει να φτάνει στο ευρύ κοινό με διάφορους τρόπους, αν θέλει η επιχείρηση να παραμείνει ανταγωνιστική, στα πλαίσια ενός λειτουργικού μάρκετινγκ. Οι επιχειρήσεις καλούνται να αντιμετωπίσουν ποικίλες και σημαντικές προκλήσεις οι οποίες προκύπτουν από τα ανωτέρω καθώς και την παγκοσμιοποίηση της αγοράς, την εντατικοποίηση της ανταγωνιστικότητας μεταξύ των επιχειρήσεων και την συνεχή αλλαγή του επιχειρηματικού περιβάλλοντος. Γίνεται κατανοητό πως για την επιτυχή αντιμετώπιση των προκλήσεων αυτών οι επιχειρήσεις πρέπει να στραφούν στο “διαδικτυακό δρόμο” ώστε να εγγυηθεί η απρόσκοπτη επαφή με τους πελάτες και τους εκάστοτε προμηθευτές.



Θα ήταν ακόμα άξιο αναφοράς το γεγονός πως το χαμηλό κόστος για ένα τόσο χρήσιμο εργαλείο για μια επιχείρηση κάνει ακόμα πιο προσιτό το συγκεκριμένο εργαλείο. Από την άλλη μεριά, πολλές επιχειρήσεις έχουν ήδη συνειδητοποιήσει τη δύναμη και την αξία που έχουν οι εφαρμογές και οι τεχνολογίες του ενός δικτυακού τόπου και προσπαθούν ολοένα και περισσότερο να επωφεληθούν από αυτήν την εξέλιξη. Η αξιοποίηση του διαδικτύου στις διάφορες λειτουργίες του μάρκετινγκ ( όπως δημόσιες σχέσεις, διαφήμιση, προώθηση, κ.τ.λ.) αποτελεί σήμερα την πλέον γνωστή και ευρέως υιοθετημένη εφαρμογή από τις επιχειρήσεις.

Επίσης η συγκεκριμένη άσκηση μας κάνει εμφανή αρκετά πλεονεκτήματα της χρήσης λογισμικού ανοιχτού κώδικα σε σχέση με το ιδιόκτητο λογισμικό. Παρότι μπορεί να το βρούμε να το βρούμε τα ράφια των καταστημάτων και να το πληρώσουμε για να το αγοράσουμε θα το βρούμε επίσης δωρεάν και διαθέσιμο στον παγκόσμιο ιστό. Μετά θα το εγκαταστήσουμε σε όσους Η/Υ θέλουμε, θα το εκτελέσουμε για όποιους σκοπούς θέλουμε για όσο καιρό θέλουμε.

Τελειώνοντας, είναι γεγονός ότι βρισκόμαστε σε μια νέα φάση εξέλιξης όπου το διαδίκτυο παρέχει στους πάντες κάθε είδους πληροφορία και όπου οι καταναλωτές έχουν ακόμη και τη δυνατότητα διαμόρφωσης πραγμάτων και καταστάσεων. Το πως αυτό θα επηρεάσει τις επιχειρήσεις και τι επιπτώσεις θα έχει γενικότερα είναι κάτι που το μάθουμε στο μέλλον.

## ΑΝΑΦΟΡΕΣ



## ΑΝΑΦΟΡΕΣ

- 1) taklin.gr, 2013
- 2) Efraim Turban, 2008
- 3) el.wikipedia.org/wiki/Ηλεκτρονικό\_επιχειρείν, 2013
- 4) el.wikipedia.org/wiki/Ηλεκτρονικός\_τουρισμός, 2013
- 5) mnec.gr, Υπουργείο οικονομικών, 2010
- 6) dide.flo.sch.gr, 2010
- 7) pacific.jour.auth.gr, 2013
- 8) [www.pointer.gr/el/blog/post/css](http://www.pointer.gr/el/blog/post/css) 2013
- 9) pages.cs.aueb.gr/ 2013
- 10) dide.flo.sch.gr/Plinet/Tutorials/Tutorials-Php.html 2008
- 11) [www.lichnosbeach.gr](http://www.lichnosbeach.gr), 2013
- 12) dide.flo.sch.gr/Plinet/Tutorials/Tutorials-MySQL.html 2009
- 13) [http://en.wikipedia.org/wiki/Apache\\_HTTP\\_Server](http://en.wikipedia.org/wiki/Apache_HTTP_Server) 2011
- 14) <http://help.ubuntu-gr.org/>, 2012

Σημείωση:

Στην διπλανή σελίδα θα βρείτε μια θήκη με το συνοδευτικό DVD της πτυχιακής εργασίας μας το οποίο περιλαμβάνει όλα τα αρχεία του ιστότοπου καθώς και τις απαραίτητες εφαρμογές για να υποστηριχθούν και να λειτουργήσουν σε τοπικό επίπεδο



