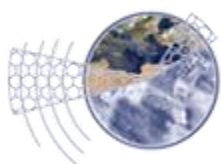




**Τεχνολογικό Εκπαιδευτικό Ίδρυμα Κρήτης**



**ΤΜΗΜΑ ΕΦΑΡΜΟΣΜΕΝΗΣ  
ΠΛΗΡΟΦΟΡΙΚΗΣ & ΠΟΛΥΜΕΣΩΝ**

**Σχολή Τεχνολογικών Εφαρμογών  
Τμήμα Μηχανικών Πληροφορικής – ΤΕΙ Κρήτης**

**Πτυχιακή Εργασία**

**Τίτλος: Μηχανοργάνωση Ενός Ιδιωτικού Εκπαιδευτηρίου**

Αντρέας Κτωρίδης (ΑΜ:2685)  
Καρκανάκης Αποστόλης (ΑΜ:3046)

Επιβλέπων Καθηγητής: Παπαδάκης Νικόλαος

**Ηράκλειο 2015**

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

## Ευχαριστίες

Θα θέλαμε να ευχαριστήσουμε τις οικογένειες μας που μας συμπαράσθηκαν όλα αυτά τα χρόνια στην προσπάθεια εκπλήρωσης των σπουδών μας στο Τμήμα Μηχανικών Πληροφορικής στο Τει Κρήτης .

Ευχαριστούμε τον Καθηγητή μας Κ. Νίκο Παπαδάκη για την υπόδειξη του θέματος της πτυχιακής μας εργασίας καθώς και για την πολύτιμη βοήθειά που μας πρόσφερε καθ' όλη τη διάρκεια εκπόνησης της .



## Κεφάλαιο 1<sup>ο</sup>

# Abstract

In our thesis we created a platform, a useful tool for teachers of high school and they have the opportunity to use it easily and quickly to save the data of the new students in the data base system and there is direct access to the data of each student.

There are many applications and other similar systems in our country education system that helps the daily needs of all the students and teachers. In our system is not allowed from students to use it only from their teachers, who can with a very simple way to import data into tables to be able to have the student's data organized in electronic system. The tables contain the data of student their absences and their degrees while we set up tables and the possibility calculation of students average score for each quarter separately.

In the following report we analyze on four chapters the overall creation of our working methods and languages used to create our platform and why our application can be a useful tool for all teachers of secondary education.

At the same time we developed a part of our work in which we give precise instructions of how to use of the application without the need of specific IT knowledge and all teachers are able to use it. There are detailed descriptions on every step, and a detailed user manual with details to import the students, absences and grades in tables in the system.

Finally we make a general reference to the platform that can evolve in the future and what functions can be integrated into our existing system.

## Σύνοψη

Στην πτυχιακή μας εργασία δημιουργήσαμε μια πλατφόρμα ένα χρήσιμο εργαλείο για τους εκπαιδευτικούς της βασικής εκπαίδευσης και έχουν την δυνατότητα να την χρησιμοποιούν για να μπορούν εύκολα και γρήγορα να αποθηκεύουν νέους μαθητές στο σύστημα και να υπάρχει άμεση πρόσβαση στα δεδομένα του κάθε μαθητή.

Υπάρχουν πολλές εφαρμογές και άλλα παρόμοια συστήματα στη σύγχρονη εκπαίδευση της χώρας μας που βοηθούν στις καθημερινές ανάγκες όλους τους μαθητές και τους καθηγητές. Στο δικό μας σύστημα δεν υπάρχει η δυνατότητα χρήσης απο μαθητές, παρά μόνο απο τους εκπαιδευτικούς τους, οι οποίοι μπορούν με ένα πολύ απλό τρόπο να εισάγουν τα δεδομένα των μαθητών σε πίνακες για να μπορούν να τα έχουν οργανωμένα σε ηλεκτρονική μορφή. Οι πίνακες περιέχουν τα στοιχεία των μαθητών, τις απουσίες και τους βαθμούς τους ενώ έχουμε δημιουργήσει και πίνακες για την δυνατότητα υπολογισμού του μέσου όρου της βαθμολογίας των μαθητών για κάθε τετράμηνο ξεχωριστά.

Στην παρακάτω αναφορά μας αναλύουμε σε τέσσερα κεφάλαια την γενική δημιουργία της εργασίας μας τους τρόπους και τις γλώσσες που χρησιμοποιήθηκαν για να δημιουργηθεί η πλατφόρμα μας και για ποιούς λόγους η εφαρμογή μας μπορεί να είναι ένα χρήσιμο εργαλείο για όλους τους εκπαιδευτικούς της μέσης εκπαίδευσης.

Παράλληλα αναπτύξαμε ένα μέρος απο την εργασία μας με το οποίο δίνουμε ακριβείς οδηγίες για το πώς μπορεί να γίνει η χρήση της εφαρμογής χωρίς να χρειάζονται απαραίτητα ιδιότερες γνώσεις πληροφορικής και όλοι οι καθηγητές είναι σε θέση να την χρησιμοποιήσουν. Υπάρχει αναλυτική περιγραφή σε κάθε βήμα και ένα αναλυτικό εγχειρίδιο χρήσης με λεπτομέριες για την εισαγωγή των στοιχείων των μαθητών, των απουσιών και των βαθμών σε πίνακες μέσα στο σύστημα.

Τέλος έχουμε κάνει μια γενική αναφορά στο πως μπορεί η πλατφόρμα να εξελιχθεί μελλοντικά και ποιές λειτουργίες μπορούν να ενταχθούν στο υπάρχων σύστημά μας.

## Πίνακας Περιεχομένων

Ευχαριστίες.....	σελίδα 2
<b>ΚΕΦΑΛΑΙΟ 1ο</b>	
Abstract.....	σελίδα 3
Σύνοψη.....	σελίδα 4
Λίστα Πινάκων.....	σελίδα 6
Λίστα εικόνων.....	σελίδα 7
<b>ΚΕΦΑΛΑΙΟ 2ο</b>	
SQL.....	σελίδα 9
MySQL.....	σελίδα 9
Βάση Δεδομένων.....	σελίδα 12
PHP.....	σελίδα 12
CSS.....	σελίδα 17
HTML.....	σελίδα 20
Visual Studio.....	σελίδα 25
Microsoft Access.....	σελίδα 26
<b>ΚΕΦΑΛΑΙΟ 3ο</b>	
ASP.....	σελίδα 28
C Sharp C#.....	σελίδα 30
<b>ΚΕΦΑΛΑΙΟ 4ο</b>	
Μελλοντικές Προεκτάσεις.....	σελίδα 50
Βιβλιογραφία.....	σελίδα 51

## **Λίστα πινάκων**

ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ .....	σελίδα 36
ΒΟΗΘΗΤΙΚΟΙ ΠΙΝΑΚΕΣ .....	σελίδα 37
ΑΠΟΥΣΙΕΣ .....	σελίδα 41
ΡΥΘΜΙΣΕΙΣ ΜΑΘΗΜΑΤΩΝ.....	σελίδα 42
ΔΕΛΤΙΟ ΠΡΟΟΔΟΥ ΜΑΘΗΤΗ .....	σελίδα 46
ΛΙΣΤΑ ΚΑΛΥΤΕΡΩΝ ΑΝΑ ΤΜΗΜΑ .....	σελίδα 47

## Λίστα Εικόνων

ΒΑΘΜΟΛΟΓΙΑ Κατά Τμήμα.....	σελίδα 38
ΒΑΘΜΟΛΟΓΙΑ Κατά Μαθητή.....	σελίδα 40
ΛΙΣΤΑ ΜΕ ΑΠΟΥΣΙΕΣ .....	σελίδα 42
ΑΠΟΥΣΙΕΣ .....	σελίδα 40
ΡΥΘΜΙΣΕΙΣ ΜΑΘΗΜΑΤΩΝ.....	σελίδα 42
ΚΑΤΑΣΤΑΣΕΙΣ(ΕΚΘΕΣΕΙΣ) .....	σελίδα 44
ΔΕΛΤΙΟ ΠΡΟΟΔΟΥ ΜΑΘΗΤΗ .....	σελίδα 46
ΛΙΣΤΑ ΚΑΛΥΤΕΡΩΝ ΑΝΑ ΤΜΗΜΑ.....	σελίδα 47
ΛΙΣΤΑ ΚΑΛΥΤΕΡΩΝ ΑΝΑ ΤΑΞΗ (Α,Β,Γ).....	σελίδα 48

## Εισαγωγή

Η πτυχιακή μας εργασία έχει στόχο να διευκολύνει τους καθηγητές της δευτεροβάθμιας εκπαίδευσης έτσι ώστε να μπορέσουν να διαχειρίζονται καλύτερα την λίστα των μαθητών τους. Ζούμε σε μια ψηφιακή εποχή και όλα στις μέρες μας γίνονται ηλεκτρονικά πράγμα που μας διευκολύνει στην καθημερινότητά μας. Το σύστημα που έχουμε δημιουργήσει προσφέρει στους χρήστες του την κατάλληλη διαχείριση της σύγχρονης τεχνολογίας και μπορούν όλοι να ενταχθούν ομαλά σε αυτήν.

Η παρακάτω αναφορά περιέχει όλες τις απαραίτητες πληροφορίες έτσι ώστε όλοι οι καθηγητές με το απλό εγχειρίδιο χρήσης να καταλάβουν την απλή λειτουργία του συστήματός μας και να μπορέσουν να το χρησιμοποιήσουν χωρίς πρόβλημα. Παραθέτουμε ορισμένες φωτογραφίες και κομμάτια απο τον κώδικα που έχει χρησιμοποιηθεί για να είναι πιο κατανοητό και γίνεται μια γενική αναφορά σε ορισμένες γλώσσες προγραμματισμού οι οποίες είναι απαραίτητες για την δημιουργία προγραμμάτων και εφαρμογών όπως η δική μας.

Οι γλώσσες που έχουμε χρησιμοποιήσει μας έχουν γίνει γνωστές απο όλα τα χρόνια σπουδών στο Τμήμα Μηχανικών πληροφορικής. Έχουμε χρησιμοποιήσει την SQL, MySQL, PHP, ASP, HTML, C#, ACCESS, CSS. Ορισμένες απο αυτές χρησιμοποιήθηκαν για την δημιουργία του προγράμματος και σε άλλες έγινε μια γενική αναφορά που παρουσιάζονται ορισμένα βασικά χαρακτηριστικά της κάθε γλώσσας.

Με διάφορους μηχανισμούς οι γλώσσες αυτές μπορούν να συνδιαστούν και να έχουν ένα φανταστικό αποτέλεσμα.

Έτσι λοιπόν αναπτύξαμε κι εμείς ένα σύστημα διαχείρισης της λίστας των μαθητών για να μπορούμε να έχουμε ένα πλήρη έλεγχο.



## Κεφάλαιο 2<sup>ο</sup>

### Γλώσσες προγραμματισμού που χρησιμοποιήσαμε

#### SQL

Η Structured Query Language(SQL) είναι μια γλώσσα προγραμματισμού που αναπτύχθηκε για να υπάρχει η δυνατότητα επεξεργασίας των δεδομένων και των στοιχείων στο RDBMS που ήταν ένα πρόγραμμα που χρησιμοποιούσαν στην IBM. Με την SQL κατάφεραν να επεξεργάζονται δεδομένα να κάνουν ανάκτηση των αποθηκευμένων πληροφοριών που υπήρχαν μέσα στο σύστημα. Η ομάδα που δημιούργησε αυτή τη γλώσσα για την διαχείριση της βάσης δεδομένων αποτελούνταν από τους Raymond Boyce τον Donald Messerly και τον Andrew Richardson και πραγματοποιήθηκε το 1970.

Τα βασικά χαρακτηριστικά και οι δυνατότητες της SQL είναι:

- Έχουμε πρόσβαση στις βάσεις δεδομένων
- Είναι μια βασική γλώσσα για πρόσβαση στο ANSI.
- Έχει τη δυνατότητα για ανάκτηση δεδομένων στη βάση δεδομένων.
- Έχει τη δυνατότητα για εισαγωγή εγγραφών.
- Έχει τη δυνατότητα ενημέρωσης εγγραφών.
- Έχει τη δυνατότητα για διαγραφή εγγραφών.

Με την SQL μπορούμε να έχουμε πρόσβαση και να επεξεργαστούμε δεδομένα σε βάσεις όπως την MySQL, SQL Server, Oracle, Access, DB2. Στη συνέχεια θα αναλύσουμε την MySQL η οποία θεωρείται ένα από τα πιο βασικά συστήματα διαχείρισης των Βάσεων Δεδομένων. [Πληροφορίες από link \(6,7,8\)](#)

#### MySQL

Η MySQL είναι μια βάση δεδομένων η οποία ως κύρια λειτουργία έχει την υποστήριξη ιστοσελίδων και άλλα online προγράμματα. Μεγάλοι και ευρέως γνωστοί διαδικτυακοί ιστότοποι χρησιμοποιούν την MySQL όπως το Facebook το Twitter το Google και το YouTube.

Ο δημιουργός της MySQL, Monty Widenius έδωσε το όνομα της κόρης του (Μάι) και πλέον ο κωδικός του είναι διαθέσιμος από την GNU. Η εταιρία της Oracle έχει αγοράσει τα πλήρη δικαιώματα της MySQL AB και πραγματοποιεί όλες τις απαραίτητες έρευνες.

Η MySQL έχει πολλά θετικά στοιχεία σε σχέση με τα υπόλοιπα προγράμματα διότι είναι γρήγορο και ευέλικτο και πολλοί χρήστες το προτιμούν. Ο κάθε χρήστης που επιθυμεί να χρησιμοποιήσει το πρόγραμμα έχει τη δυνατότητα να εγκαταστήσει το λογισμικό στον υπολογιστή του μέσω της [www.mysql.com](http://www.mysql.com) και να συνδεθεί μέσω του Server.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

Μερικά βασικά βήματα για να δημιουργίσουμε μια βάση είναι τα εξής:  
Αρχικά δημιουργούμε την βάση μας στον Server με την παρακάτω εντολή.  
CREATE DATABASE databaseName; [Πληροφορίες απο link\(9,10\)](#)

Στη συνέχεια επιλέγουμε για χρήση τη Βάση που δημιουργήσαμε.

USE databaseName;

Έπειτα δημιουργούμε ένα νέο πίνακα με τα στοιχεία του μαθητή (Κωδικός,Επίθετο,Όνομα,Τμήμα).

CREATE TABLE TableName (

Code int not null,

Last Name varchar(20),

First Name varchar(20),

Class varchar(5)

PRIMARY KEY (Code)

);

Ο τύπος δεδομένων που χρησιμοποιούμε κάθε φορά μπορεί να διαφέρει ανάλογα με τα δεδομένα.

Μπορεί να είναι Αριθμητικοί, Αλφαριθμητικοί, Ημερομηνίες και Ωρες:

#### **Αριθμητικοί:**

- **INT** : ακέραιος με μήκος έως 11 ψηφία.
- **FLOAT**: κινητής υποδιαστολής έως 24 ψηφία.
- **DOUBLE**: διπλής ακρίβειας έως 53 ψηφία.
- **BIGINT**: ακέραιος με μήκος έως 20 ψηφία.
- **TINYINT**: ακέραιος με μήκος έως 4 ψηφία.
- **SMALLINT**: ακέραιος με μήκος έως 5 ψηφία.
- **MEDIUMINT**: ακέραιος με μήκος έως 9 ψηφία.

#### **Αλφαριθμητικοί:**

- **CHAR**: Έχει σταθερό μέγεθος απο 1-255 χαρακτήρες.
- **VARCHAR**: Έχει μεταβλητό μέγεθος απο 1-255 χαρακτήρες.
- **TEXT**: Έχει μέγιστο μήκος 65535.
- **TINYTEXT**: Έχει μέγιστο μήκος 255.
- **MEDIUMTEXT**: Έχει μέγιστο μήκος 16777215.
- **LONGTEXT**: Έχει μέγιστο μήκος 4294967295.

#### **Ημερομηνίες και Ωρες:**

- **DATE**: Ημερομηνία στη μορφή ΕΕΕΕ-ΜΜ-ΗΗ
- **TIME**: Ωρα στη μορφή ΩΩ:ΛΛ:ΔΔ
- **YEAR**: Έτος στη διψήφια μορφή απο 1970-2069.  
Έτος στην τετραψήφια μορφή 1901-2155.
- **DATETIME**: Ημερομηνία και ώρα στη μορφή ΕΕΕΕ-ΜΜ-ΗΗ ΩΩ:ΛΛ:ΔΔ

Όσο αφορά τις εντολές που χρησιμοποιούμε στην SQL για την εισαγωγή δεδομένων, την διαγραφή, την ανανέωση και την αναζήτηση στοιχείων απο τα ήδη υπάρχοντα μέσα στη βάση θα δούμε μερικούς τρόπους για την κάθε μια περίπτωση.

Ξεκινώντας για να εισάγουμε ορισμένα δεδομένα στον πίνακα που έχουμε δημιουργήσει παραπάνω πρέπει να χρησιμοποιήσουμε τις παρακάτω εντολές.

```
INSERT INTO TableName VALUES  
(1,'Papadopoulos','Nikos','A1'),  
(2,'karkanakis','Apostolis','A2'),  
(3,'Ktoridis','Antreas','A2');
```

Με αυτό τον τρόπο προσθέσαμε στην βάση μας 3 νέους μαθητές με τον κωδικό, επίθετο, όνομα και τμήμα και μπορούμε πλέον να τους κάνουμε αναζήτηση μέσα στη βάση.

Στην συνέχεια και αφού έχουμε ολοκληρώσει την καταχώριση των δεδομένων μέσα στο σύστημα μας μπορούμε να κάνουμε αναζήτηση με τις εντολές που μπορεί να διαβάσει το πρόγραμμάμας και στη γλώσσα που έχουμε επιλέξει.

```
SELECT * FROM TableName;
```

Με αυτή την εντολή θα εμφανιστούν όλα τα στοιχεία του πίνακα TableName που έχουμε καταχωρίσει. Υπάρχουν πολλές εντολές που μπορούμε να αναζητήσουμε στοιχεία και να κάνουμε πιο συγκεκριμένη την αναζήτησή μας όπως για παράδειγμα :

```
SELECT LastName , class FROM TableName;
```

Η συγκεκριμένη εντολή θα μας εμφανίσει όλα τα στοιχεία του πίνακα TableName τα οποία βρίσκονται στη λίστα με το επίθετο και το τμήμα.

Εάν θελήσουμε να αλλάξουμε την αίθουσα σε κάποιο μαθητή μπορούμε να το κάνουμε με την ακόλουθη εντολή.

```
UPDATE TableName SET class="A1" WHERE CODE=1;
```

Επίσης για να διαγράψουμε κάποια εγγραφή ενός μαθητή του οποίου το επίθετο περιλαμβάνει το κείμενο "akis" απο τη λίστα με εντολή στην SQL μπορούμε να χρησιμοποιήσουμε την ακόλουθη εντολή:

```
DELETE FROM TableName WHERE LastName like "%akis";
```

Ενώ αυτή η εντολή αδειάζει τον πίνακα εντελώς:

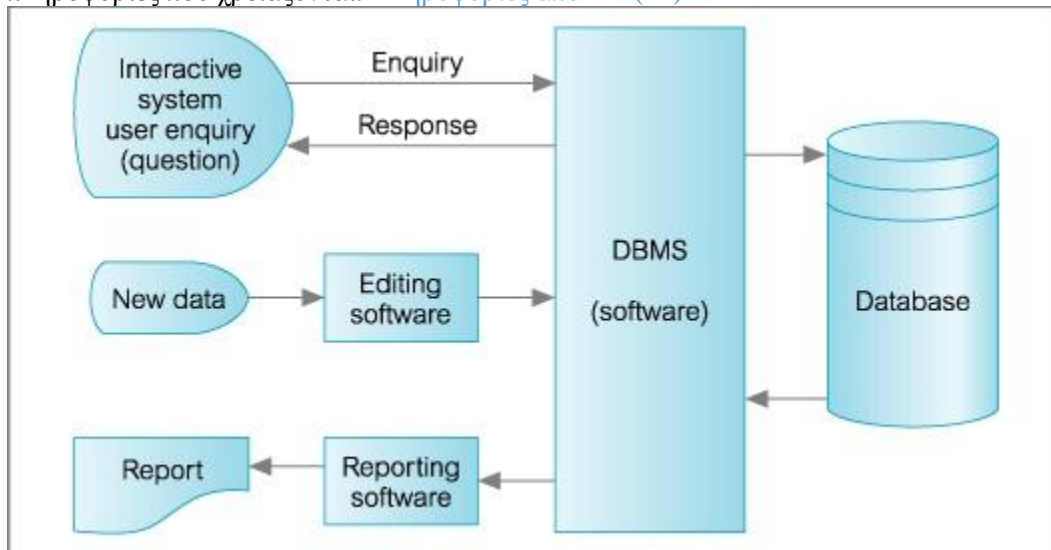
```
DELETE FROM TableName;
```

## Βάση Δεδομένων

Βάση δεδομένων είναι ένα μέσο για να μπορέσουμε να αποθηκεύουμε δεδομένα και πληροφορίες και μπορούμε να τα ανακτήσουμε με διάφορες εντολές. Ανάλογα την εντολή που δίνει ο χρήστης μπορεί να αναζητήσει τις αποθηκευμένες πληροφορίες στο σύστημα. Τα δεδομένα αποθηκεύονται μέσα στην βάση με συγκεκριμένη δομή για να μπορούμε να τα ενοπίσουμε ευκολότερα και για να υπάρχει μια οργάνωση στον τρόπο σκέψης της καταγραφής των στοιχείων.

Για να υπάρχει μια συγκεκριμένη δομή στο συστήμα μας θα πρέπει να χρησιμοποιήσουμε πίνακες που είναι η κύρια δομή της βάσης δεδομένων. Μέσω της γλώσσας SQL ο οποιοσδήποτε χρήστης μπορεί να κάνει εισαγωγή, διαγραφή, αναζήτηση και ανάκτηση στα δεδομένα που υπάρχουν η και να δημιουργήσει νέα.

Υπάρχουν διάφορα προγράμματα που διαχειρίζονται βάσεις δεδομένων κι ένα απο αυτά είναι το DBMS(Σύστημα Διαχείρισης Βάσης Δεδομένων). Σε αυτό το σύστημα ο χειρισμός των δεδομένων μπορεί να γίνει και χωρίς την βοήθεια του προγραμματιστή, διότι γίνεται πολύ απλός με την χρήση των κατάλληλων εργαλείων κι έτσι όλοι οι χρήστες διατυπώνουν ερωτήματα για να αναζητήσουν τις πληροφορίες που χρειάζονται. [Πληροφορίες απο link\(11\)](#)



### PHP (Hypertext PreProcessor)

Η PHP γλώσσα προγραμματισμού ξεκίνησε το 1994 από ένα φοιτητή για τον έλεγχο των χρηστών που έμπεναν στην ιστοσελίδα του για το λόγο αυτό είχε αρχικά το όνομα Personal Home Page. Στη συνέχεια μετονομάστηκε σε Hypertext PreProcessor και μπόρεσε να ενσωματωθεί στην HTML για την δημιουργία ιστοσελίδων κι επίσης την κατασκευή Web Applications.

Η PHP έχει πάρει ένα μεγάλο μέρος από τις γλώσσες προγραμματισμού Perl και C και για αυτό το λόγο η δημιουργία των ιστοσελίδων είναι δυναμική δηλαδή μπορεί ο σχεδιαστής (web developer) να πραγματοποιήσει αλλαγές ανάλογα τις επιθυμίες του κάθε επισκέπτη. Μια βασική λεπτομέρεια της PHP είναι ότι ο εκτελέσιμος κώδικας βρίσκεται σε ένα server και δεν μπορεί ο κάθε χρήστης να έχει πρόσβαση και να κάνει αλλαγές, το μόνο που μπορεί να δει και να αξιολογήσει είναι το αποτέλεσμα.

Μπορούμε να εγκαταστήσουμε την γλώσσα PHP σε όλα σχεδόν τα λειτουργικά συστήματα που κυκλοφορούν στην αγορά (Mac OS X, Windows, Linux) διότι είναι ένα ελεύθερο λογισμικό και οι πιο πολλοί Servers το υποστηρίζουν.

### **Συνταξη της PHP**

#### **Η γλώσσα αυτή συντάσσεται ως εξής:**

Για να εμφανιστεί ένα επιθυμητό μήνυμα όπως για παράδειγμα το «Καλημέρα» στον χρήστη πρέπει να δημιουργήσουμε ένα νέο αρχείο με μορφή kalimera.php . Ο κώδικάς μας πρέπει να αρχίζει με <?php και πάντα να κλείνει με ?> για να μπορέσει το σύστημά μας να το αναγνωρίσει. Στο ενδιάμεσο κομμάτι καταγράφουμε το «echo» το οποίο τυπώνει το μήνυμα που βρίσκεται σε εισαγωγικά "...» στην οθόνη. Μετά από κάθε εντολή πρέπει να ακολουθεί το semicolon(;) για να είναι έγκυρη η εντολή και οι εντολές θα πρέπει να συντάσσονται σε μια σειρά.

```
<?php
```

```
Echo "Καλημέρα";
```

```
?>
```

Έτσι στην οθόνη μας θα εμφανιστεί το μήνυμα «Καλημέρα».

Για να υπάρχει και η ενσωμάτωση με την HTML θα πρέπει αυτό το κομμάτι του κώδικα να περικλείεται από τις ακόλουθες εντολές και να έχει αυτή τη μορφή:

```
<html>
```

```
    <body>
```

```
        <?php
```

```
            Echo "Καλημέρα";
```

```
        ?>
```

```
    </body>
```

```
</html>
```

#### **Σχολια(Comments):**

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

Πρέπει να εισάγουμε σχόλια σε ένα μεγάλο μέρος του κώδικά μας έτσι ώστε να είναι πιο κατανοητός από χρήστες που δεν έχουν ιδιαίτερη γνώση σε θέματα προγραμματισμού.

Μερικά παραδείγματα σχολίων είναι τα εξής:

- Η δίαση (hashtag) # μπορεί να χρησιμοποιηθεί για να σχολιάσουμε μια γραμμή κώδικα.
- Οι δύο κάθετες γραμμές // που επίσης μπορούν να χρησιμοποιηθούν για να σχολιάσουμε μια γραμμή κώδικα.
- /\* \_\_\_ \*/ το βάζουμε κυρίως στην περίπτωση που θέλουμε να κάνουμε σχολιασμό σε περισσότερες από μια γραμμές.

Μεταβλητές(Variables):

Σε όλες τις γλώσσες σχεδόν τις γλώσσες προγραμματισμού δηλώνουμε τις μεταβλητές στην αρχή του προγράμματος και μπορούμε να γνωρίζουμε που τις έχουμε τοποθετήσει. Στην περίπτωση της PHP μπορούμε να δηλώσουμε κατά την ώρα εκτέλεσης του προγράμματος, μπορούμε να αναγνωρίσουμε τις μεταβλητές από τον χαρακτήρα \$ ο οποίος μπαίνει στην αρχή μιας μεταβλητής. Οι πιο βασικές μεταβλητές είναι οι εξής:

Strings (Συμβολοσειρές): Περιέχει χαρακτήρες και δεν υπάρχει κανένας περιορισμός στον αριθμό που μπορούμε να χρησιμοποιήσουμε.

Integers (Ακέραιοι): Οι γνωστοί ακέραιοι από τις μαθηματικές πράξεις με αριθμούς. Εδώ υπάρχει ένα όριο στο πλήθος των αριθμών που στη συνέχεια αλλάζουν σε Floats.

Floats (Πραγματικοί Αριθμοί): Δεν υπάρχει κάποιο όριο χρήσης των Floats οι οποίοι περιέχουν τους δεκαδικούς αριθμούς.

Booleans (Λογικές τιμές): Στην συγκεκριμένη μεταβλητή υπάρχουν οι τιμές αληθές ή ψευδές δηλαδή ακέραιες τιμές με το 0 σαν ψευδές και το 1 σαν αληθές.

Resources (Πηγές): Παίρνει δεδομένα εκτός της php.

Arrays (Πίνακες): Μπορούν να έχουν το ίδιο όνομα μιας μεταβλητής περισσότερες από μια τιμές.

Control Structures:Όπως και στην C έτσι και η PHP έχει κάποιες όμοιες εντολές που μπορούν να εκτελεστούν ώστε να γίνει κλήση της συνάρτησης, μια επανάληψη ή ένας έλεγχος. Κάποιες από αυτές τις εντολές ακολουθούν στη συνέχεια:

For:

Αρχικά παρουσιάζεται η for είναι μια δομή επανάληψης η οποία αποτελεί ένα βρόχο από τους δυσκολότερους στην PHP. Η χρήση της είναι η ίδια με την γλώσσα C δηλαδή υπάρχει ένας μετρητής που κρατάει τον αριθμό επανάληψης και συντάσσεται έτσι:

```
<?php
    for($i=1;$i<=10;$i++ ) {
        echo $i; }
?>
```

Με αυτό τον τρόπο εμφανίζονται στην οθόνη οι αριθμοί από το 1 έως το 10. Επίσης ένα ακόμα παράδειγμα είναι:

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

```
<?php
    for($i=1 ; ; $i++) { if ($i > 10){
        break;}

    echo $i; }

?>
```

**While:**

Ένας πιο απλός τρόπος για να γίνει μια επανάληψη είναι με την χρήση της εντολής While η οποία είναι επίσης ίδια με την χρήση της στην γλώσσα C όπου εκτελούνται εντολές για όσο η συνθήκη μας είναι αληθής. Σε περίπτωση που η συνθήκη είναι ψευδής δεν θα εκτελεστούν εντολές.

```
<?php
$i=1;
while ($i<+10){
    echo $i++; }

?>
```

Εδώ επίσης εμφανίζεται στην οθόνη ο αριθμός απο το 1 ως το 10.Ακολουθεί ένα ακόμα παράδειγμα με την χρήση της Endwhile.

```
<?php

    $i=1;

    Whie ($i<=10):

    Echo $i;

    $i++;

    Endwhile;

?>
```

**IF:**

Η if στην σύνταξη της γίνεται όπως και στην C δηλαδή έχουμε την έκφραση και την εντολή. Αν η έκφρασή μας είναι αληθής τότε θα γίνει η εκτέλεση της εντολής και το πρόγραμμα θα προχωρήσει παρακάτω.

```
<?php
If ($a > $b)
    echo “το a είναι μεγαλύτερο του β ”;

?>
```

**Else:**

Η εντολή else κάνει το ίδιο ακριβώς με την if αλλά σε περίπτωση που είναι ψευδής η συνθήκη εαν τότε δεν θα γίνει καμία εκτέλεση εντολής και το πρόγραμμα θα προχωρήσει παρακάτω:

```
<?php
  If ($a > $b){
    echo “ α μεγαλύτερο του β ”;
  }else{
    echo “ α μικρότερο του β”
  }
?>
```

#### **Elseif:**

Στην περίπτωση αυτή υπάρχει ένας συνδιασμός της if με την else. Δηλαδή μπορεί να εκτελεσθεί μια εντολή και το αποτέλεσμα της if να είναι ψευδής τότε θα προχωρήσει στην επόμενη εντολή μέχρι να είναι αληθής. Για παράδειγμα:

```
<?php
If ($a>$b) {
echo ” α μεγαλύτερο του β”;
```

```
}elseif ($a==$b) {
echo ”α ίσο με β”;
```

```
}else{
echo ”α μικρότερο του β”;
```

```
}
?>
```

#### Μεταβλητές GET και POST

Υπάρχουν δυο τρόποι για να μπορέσει να στείλει πληροφορίες ο Client στο Server. Η μέθοδος GET και η μέθοδος POST.

#### Μέθοδος GET:

- Έχει περιορισμό μέχρι 1024 χαρακτήρες.
- Παράγει ένα string που εμφανίζεται στον server μέσα στο Box.

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---



- Η μέθοδος GET δεν είναι ασφαλής σε περίπτωση που χρησιμοποιήτε κωδικούς ασφαλείας.
- Η GET είναι αδύνατο να χρησιμοποιηθεί για αποστολή εγγράφων.

```
<?php
```

```
If($_POST["name"] || $_POST["age"])
{ echo "Καλώς Ήρθατε".$_POST['name']. "<br/>";
{ echo "Yo".$_POST['name']. "<
```

Μέθοδος POST:

Δεν υπάρχει περιορισμός στο μέγεθος των δεδομένων που μπορούν να σταλούν. Χρησιμοποιείτε για την αποστολή ASCII και δυκτιακών δεδομένων. Η ασφάλεια εξαρτάτε απο το πρωτόκολλο HTTP γιατί τα δεδομένα στέλνονται μέσω του HTTP Header. Η PHP διαθέτει το \$\_POST associate που είναι ένας πίνακας και έχει μέσα όποιες πληροφορίες έχουμε στείλει με την μέθοδο POST. [Πληροφορίες απο link \(2,3\)](#)

## CSS (Cascading Style Sheets)

Το css δηλαδή Cascading Style Sheets που σημαίνει Διαδοχικά Σχεδιαστικά Φύλλα. Το CSS καθορίζει πως η HTML θα εμφανίζεται στην οθόνη του υπολογιστή με ποιά μορφή και τι χαρακτηριστικά θα έχει μια ιστοσελίδα. Μέσω του CSS μπορούμε να καθορίσουμε το πόσο ωραία θα εμφανίζεται η ιστοσελίδα μας στο διαδίκτυο χρησιμοποιώντας περισσότερα χρωματικά στοιχεία και στοίχιση.

Το καλό με την γλώσσα αυτή είναι ότι δεν χρειάζετε να αλλάζουμε μια προς μια όλες τις σελίδες που υπάρχουν σε ένα site αλλά κάποιες λεπτομέριες σε ένα αρχείο με κατάληξη .CSS. Πολλές απο αυτές τις μικρές λεπτομέριες προστέθηκαν για να λύσουν κάποια μικρά προβλήματα στην HTML 4 και υπάρχουν 3 έτοιμες μέθοδοι που μας βοηθάνε στο να εξοικονομίσουμε περισσότερο χρόνο οι οποίες είναι οι **External Style Sheet, Internal Style, Sheet Inline Style**. [Πληροφορίες απο link\(12,13\)](#)

**ESS:**

Με αυτή τη μέθοδο μπορούμε να εμφανίζουμε το ίδιο Style σε όλες τις σελίδες του Web site μας με ένα και μόνο αρχείο. Μέσα στο τμήμα head υπάρχει το tag <link> και εδώ μπορούμε να εισάγουμε το δικό μας link με τον εξής τρόπο.

```
<head>
```

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

```
</head>
```

---

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

**ISS:**

Το χρησιμοποιούμε για να έχουμε ένα και μόνο Style στο έγγραφο μας. Τοποθετώντας λοιπόν μέσα στο αρχείο μας τον κωδικά CSS μπορούμε να το εφαρμόσουμε ως εξής.

```
<head>

<Style type="text/css">

hr{color: red;

text-align: center;}

p{margin-left:20 px}

body {background-color: white}

</head>
```

**Inline Style:**

Εάν χρειαστεί να αλλάξουμε τον κώδικα CSS μέσα στην HTML για ένα μόνο στοιχείο θα χρησιμοποιήσουμε αυτή τη μέθοδο. Για παράδειγμα:

```
<p style="color: grey; margin-left: 20px">
Μια παράγραφος
</p>
```

**Σχόλια στον κώδικα της CSS:**

Στις περιπτώσεις που επιθυμούμε να εισάγουμε κάποιο σχόλιο στον κωδικά μας έτσι ώστε να μπορούμε να αλλάξουμε τον πηγαίο κώδικα στην CSS πρέπει να χρησιμοποιήσουμε ορισμένους χαρακτήρες όπως στο ακόλουθο παράδειγμα:

```
/*.....Comment.....*/
```

Πιο συγκεκριμένα:

```
P {

Text-align: center;

/*.....Comment.....*/ (Εδώ εισάγουμε τα σχολιά μας)

color: black;

font-family: arial

}
```

**Class Attribute**

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

Με αυτό το χαρακτηριστικό μπορούμε στο ίδιο αρχείο να προσαρμόσουμε ένα διαφορετικό style όπως για παράδειγμα μια παράγραφο στοιχισμένη στο κέντρο και μια άλλη στα αριστερά.

```
p.center {text-align:center}
```

Η παράγραφος θα στοιχηθεί στο κέντρο

```
p.left {text-align:left}
```

Η παράγραφος θα στοιχηθεί στα αριστερά

### **ID Attribute**

Το συγκεκριμένο χαρακτηριστικό αφορά την περίπτωση που θέλουμε να βάλουμε το ίδιο style σε πολλά στοιχεία μαζί. Έτσι φαίνεται στον παρακάτω κώδικα πώς κάνουμε τις αλλαγές στην επικεφαλίδα.

```
<p id="right"> παράγραφος
```

Το κείμενο έχει στοιχηθεί στα δεξιά. </p>

```
<h3 id="right"> επικεφαλίδα
```

Η επικεφαλίδα έχει στοιχηθεί στα δεξιά. </h3>

### **Ιδιότητες φόντου**

Στη συνέχεια θα παραθέσουμε ορισμένες ιδιότητες οι οποίες ομορφαίνουν το φόντο με διάφορα στοιχεία. Αρχικά μπορούμε να δηλώσουμε το χρώμα του φόντου (background-color), επίσης για να δηλώσουμε στο φόντο μια εικόνα μπορούμε να το κάνουμε με την εντολή(background-image),ακόμα μπορούμε να επαναλάβουμε την ίδια εικόνα στο φόντο πολλές φορές(background-repeat).

#### **Background-color:**

```
<html> <head> <style>
```

```
body {background-color: blue}
```

```
h1 {background-color: #000000} /*μαυρο*/
```

```
h2 {background-color: transparent}
```

```
p {background-color: rgb(250, 0, 255)}
```

```
</style> </head><body>
    <h1> Κεφαλίδα 1 </h1>
    <h2> Κεφαλίδα 2 </h2>
    <p> παράγραφος </p>
</body> </html>
```

### Background-image:

```
<style> <body>
{background-image: url("../images/bgdesert.jpg")}
</body> </style>
```

### Background-repeat:

```
<style> <body>
    {background-image: url("../images/bgdesert.jpg");
      background-repeat: repeat-y}
</body></style>
```

## HTML

Η HTML δημιουργήθηκε από τον φυσικό Tim Berners και από τον Robert Caillan ο οποίος ήταν μηχανικός συστημάτων πληροφορικής. Δούλευαν και οι δύο για το CERN (Conseil Européenne in Recherche Nucleaire). Το 1990 δημιουργήθηκε η προδιαγραφή για την γλώσσα HTML.

Η πρώτη διαθέσιμη HTML ήταν ένα έγγραφο που είχε το όνομα ετικέτες HTML το οποίο αναφερόταν στο διαδίκτυο και δημιουργήθηκε στα τέλη του 1991 από τον Berners. Το έγγραφο αυτό περιείχε τα 20 στοιχεία τα οποία αποτελούσαν τον αρχικό σχεδιασμό της HTML. Όλες οι ετικέτες εκτός από την ετικέτα υπερσυνδέσμου είναι επηρεασμένες από την SGML guid μια μορφή δημιουργίας τεκμηρίωσης η οποία είχε φτιαχτεί στο CERN και ήταν βασισμένη στην SGML . Δεκατρία από τα 20 αρχικά στοιχεία χρησιμοποιούνται έως σήμερα στην έκδοση της HTML 4

Η γλώσσα HTML χρησιμοποιήθηκε για την κατασκευή ιστοσελίδων. Τα αρχικά σημαίνουν Hyper Text Markup Language δηλαδή γλώσσα χαρακτηρισμού υποκειμένου και είναι η βασική γλώσσα για την κατασκευή ιστοσελίδων. Τα στοιχεία της αποτελούν τη βάση και τη δομή στην δημιουργία μιας Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

ιστοσελίδας. Στην HTML πρώτα γράφεται το κείμενο και στη συνέχεια προστίθενται τα σύμβολα, στην αρχή και στο τέλος των λέξεων, ή των προτάσεων που βρίσκονται στο κείμενο. Τα ειδικά σύμβολα που χρησιμοποιούμε στην HTML ονομάζονται ετικέτες (Tags).

Οι ετικέτες είναι αυτές που δημιουργούν την δομή και τη μορφή ενός υπερκειμένου σε μια ιστοσελίδα. Οι ετικέτες χρησιμοποιούν το σύμβολο “<” στην αρχή μιας λέξης και στο τέλος το σύμβολο “>”. Συνήθως χωρίζονται σε ένα ζεύγος η πρώτη είναι στην αρχή ενός κειμένου και η άλλη στο τέλος π.χ. (<h1>.....</h1>). Ουσιαστικά μας καθορίζει που αρχίζει και που τελειώνει ένα κείμενο.

Το πρότυπο SGML διαχωρίζει την δομή από το περιεχόμενο, στην ίδια κατεύθυνση κινήθηκε αργότερα και η HTML χρησιμοποιώντας τα CSS. Αρκετά στοιχεία κειμένου προέρχονται από την τεχνική αναφορά ISO 9537 (τεχνικές χρήσης της SGML) η οποία έχει χαρακτηριστικά πρώιμων γλωσσών μορφοποίησης κειμένου τα οποία είχαν χρησιμοποιηθεί από το TYPSET και RUNOFF και είχαν δημιουργηθεί για το λειτουργικό CTSS στις αρχές του 1960. Στο ξεκίνημα της δημιουργίας της HTML είχαν δημιουργηθεί δυο πρόχειρα το HTML και το HTML + με το δεύτερο το οποίο είχε προτείνει την προτυποποίηση κάποιων δυνατοτήτων ήδη υλοποιημένων όπως φόρμες και πίνακες.

Κάποιες εκδόσεις της HTML μέχρι και σήμερα είναι οι εξής:

- Η HTML 2.0 είχε δημοσιευτεί ως IETE RFC 1866
- 1995 RFC 1867(ανέβασμα αρχείου από φόρμα)
- 1996 RFC 1942(Δημιουργία πινάκων)
- 1996 RFC 1980(πελατοκεντρικοί χάρτες εικόνων)
- 1997 EFC 2070(διεθνοποίηση και τακτοποίηση)

Το 1997 η HTML 3.2 δημοσιεύτηκε σαν σύσταση από το W3C. Η έκδοση αυτή είχε αναπτυχθεί και είχε τροποποιηθεί από το W3C. Στην έκδοση HTML 3.2 αφαιρέθηκαν οι μαθηματικοί τύποι και οι υπερκαλύψεις που προέρχονταν από επεκτάσεις του προτύπου από άλλους φορείς. Τοποθετήθηκαν περισσότερες ετικέτες οπτικής διαμόρφωσης του Netscape και Marquee της Microsoft μετά από κοινή συναίνεση των 2 εταιρειών. Στα τέλη του 1997 δημοσιεύτηκε η έκδοση της HTML 4.0 από το W3C η οποία είχε 3 εκδοχές:

1. Strict απαγορεύεται η χρήση ξεπερασμένων στοιχείων.
2. Transitional επιτρέπονται τα ξεπερασμένα στοιχεία.
3. Frameset επιτρέπονται μόνο στοιχεία frames.

Η HTML 4 είναι μια εφαρμογή του SGML η οποία βασίζεται στο πρότυπο ISO 8879-SGML. Το 1998 γίνεται επανέκδοση της HTML 4.0 με μερικές αλλαγές. Το 1999 δημοσιεύεται η έκδοση HTML 4.01 από το W3C η οποία είχε τις ίδιες 3 εκδοχές με την προηγούμενη. Το 2000 με το ISO/IEC 15445:2000. Προτυποποιείται η Iso HTML, βασισμένη στην έκδοση HTML 4.01 strict ως διεθνές πρότυπο ISO/IEC. Το 2008 η HTML 4.01 και το ISO//IEC 15445-2000 είναι οι πιο πρόσφατες εκδόσεις HTML. Ακολούθως αναπτύχθηκε η XHTML η οποία είχε βάση τη XML.

Ένα αρχείο HTML περιέχεται **μέσα** στην ετικέτα <HTML>...</HTML> και αποτελείται από δύο ξεχωριστά μέρη : το <HEAD> ...</HEAD> στο οποίο περιέχονται πληροφορίες που αφορούν το ίδιο το αρχείο και όχι το τι θα εμφανιστεί στην οθόνη του browser, και το <BODY>...</BODY> στο οποίο περιέχεται ότι θα εμφανιστεί στην σελίδα . Πριν δείξουμε τα παραπάνω με ένα παράδειγμα , θα αναφερθούμε σε μία άλλη ετικέτα , την <TITLE> Τίτλος </TITLE> . Η ετικέτα αυτή εμφανίζει τον Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

Τίτλο στην μπάρα τίτλου του browser. Η ετικέτα αυτή δεν εμφανίζει κάποιο κείμενο στην οθόνη και βρίσκεται μέσα στην <HEAD> ...<HEAD> . Όπως ανάφερα με και παραπάνω ένα αρχείο HTML είναι ένα αρχείο κειμένου (ASCII). Επομένως , για να δημιουργήσουμε το πρώτο μας παράδειγμα HTML αρκεί να ανοίξουμε έναν επεξεργαστή κειμένου , να πληκτρολογήσουμε το κείμενό μας , να το αποθηκεύσουμε σε ένα αρχείο με κατάληξη .htm ή .html και στη συνέχεια να το εμφανίσουμε κανοντας χρήση του browser. Παρακάτω δίνεται ο κώδικας σε HTML της πρώτης μας σελίδας . Στο παράδειγμα αυτό εμφανίζεται το κείμενο « Μέσα στην ετικέτα body γράφουμε το κείμενο που θέλουμε να εμφανίζεται στη σελίδα .» στον browser. Επίσης , στη μπάρα τίτλου εμφανίζεται το κείμενο « Τίτλος ... Εμφανίζομαι μόνο στη μπάρα τίτλου ». Στην αρχή γράφουμε σε έναν επεξεργαστή κειμένου το παρακάτω κείμενο.

### **Δομή και σύνταξη της HTML:**

```
<HTML>
<HEAD>
<!--This is a comment. The browser ignores this text-->
<TITLE>Simple Web Page</TITLE>
</HEAD>
<BODY>
Welcome to my webpage!
</BODY>
</HTML>
```

- HTML εντολές ή tag έχουν μια σταθερή δομή <name>
- Τα περισσότερα HTML tags τα συναντούμε ως ζεύγη τα οποία ενεργούν στα περιεχόμενα μεταξύ των δύο του ζεύγους. Το tag για κλείσιμο έχει ένα slash ("/") πριν το όνομα, για παράδειγμα, </name>
- Ένα τυπικό αρχείο HTML αρχίζει με <HTML> και τερματίζει με </HTML>, ενώ έχει δύο μέρη: ένα HEAD και ένα BODY που εμπεριέχονται στα αντίστοιχα ζεύγη (<HEAD>.... </HEAD>) και (<BODY>.... </BODY>), αντίστοιχα.
- Ένα στοιχείο του HEAD είναι το TITLE, το οποίο εμφανίζεται πάνω στη μπάρα τίτλου του φυλλομετρητή (browser)
- Το BODY του αρχείου εμφανίζεται στο κύριο παράθυρο του φυλλομετρητή.
- Τα σχόλια, τα οποία αρχίζουν με <!-- και τελειώνουν με -->, αγνοούνται από τον φυλλομετρητή.

### **Δημιουργία λιστών στην HTML:**

Μπορούμε να δημιουργήσουμε λίστες με bullets με το tag <LI></LI>

Με τον παρακάτω κώδικα δημιουργούμε μια λίστα με τρία στοιχεία:

```
<LI>List 1</LI>
<LI>List 2</LI>
<LI>List 3</LI>
```

### **Προσθήκη εικόνων:**

Χρησιμοποιούμε το tag <IMG> και το επιπλέον χαρακτηριστικό SRC για καθορισμό της τοποθεσίας της εικόνας. Με την πιο κάτω εντολή προσθέτουμε μια εικόνα πάνω στην ιστοσελίδα μας.

```
<IMG SRC="image.gif" height="100" width="100">
```

### **Προσθήκη πίνακα:**

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

Tag	Description
<table>	Define a table
<tr>	Define a row within a table
<th>	Define table header cells
<td>	Define table data cells
<caption>	Define a table caption
	A simple table

Παρακάτω δίνουμε τον κώδικα για δημιουργία του πιο πάνω πίνακα:

```
<table border="2"align="center">
<tr>
<th>Tag</th><th>Description</th>
</tr>
<tr>
<td>&lt;table&gt;</td><td>Define a table</td>
</tr>
<tr>
<td>&lt;tr&gt;</td><td>Define a row within a table</td>
</tr>
<tr>
<td>&lt;th&gt;</td><td>Define table header cells</td>
</tr>
<tr>
<td>&lt;td&gt;</td><td>Define a table data cells</td>
</tr>
<tr>
<td>&lt;caption&gt;</td><td>Define a table caption</td>
</tr>
<caption align="bottom"><B>A simple table</B></caption>
</table>
```

#### **Αλλαγή χρωμάτων:**

Πιο κάτω δίνουμε ένα πίνακα με τα διάφορα στοιχεία που μπορούμε να αλλάξουμε και την αντίστοιχη εντολή.

Element	Associated Tag and Attribute
Document background color	<body bgcolor=" color">
Document background image	<body background=" image">
All document text	<body text=" color">
Active hyperlinks	<body alink="color">

Visited hyperlinks	<body vlink="color">
Regular Hyperlinks	<body link="color">
Small Portion of Text	<font color="color">
Table cells	<table bgcolor="color">

Μερικές τιμές που μπορούμε να δώσουμε στο colour είναι:

Color Name	Color Name
Aqua	Olive
Blue	White
Grey	Black
Lime	Red
Navy	Fuchsia
Purple	Green
Silver	Teal
Yellow	Maroon

Επομένως προσθέτουμε τον πιο κάτω κώδικα για αλλαγή χρωμάτων στο αρχείο μας.

```
<BODY bgcolor="aqua" text="purple" alink="blue" vlink="red" >
<table border="2"align="center" bgcolor="white">
```

Ένα παράδειγμα αλλαγής του μεγέθους και χρώματος των γραμμάτων. Ο κώδικας είναι:

```
<font color="red" size="7"> This</FONT>
<font color="blue" size="+2"> is</FONT>
<font color="purple" size="+1"> a</FONT>
<font color="teal"> small</FONT>
<font color="yellow" size="-1">example</FONT>
```

Το τελικό αρχείο όταν γράψουμε τα πιο πάνω κομμάτια κώδικα είναι:

```
<HTML>
  <!--HEAD-->
<HEAD>
<!--This is a comment. The browser ignores this text-->
<TITLE>Simple Web Page</TITLE>
</HEAD>

<!--BODY -->
<BODY bgcolor="aqua" text="purple" alink="blue" vlink="red" >Blank Line
  <!--Headings -->
<P><H1> Welcome to my Web Page!-Big Heading (H1) </H1>
<P>Blank Line!
<P><H3><center><I><U>Medium Heading (H3) </U></I> </center></H3>
<P><H6> Small Heading (H6) </H6>

  <!--Create a line -->
<HR>Create a line<HR size="10">
```

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής



```
<!--Create URL-->
<A HREF="http://www.ucy.ac.cy/">click here for www.ucy.ac.cy</A>
```

```
<!--Create a List-->
<P><LI>List 1</LI>
<LI>List 2</LI>
<LI>List 3</LI>
```

```
<!--Create Image-->
<!--<IMG SRC="image.gif" height="100" width="100">-->
```

```
<!--CREATE A TABLE -->
<table border="2"align="center" bgcolor="white>

<tr>
  <th>Tag</th><th>Description</th>
</tr>
<tr>
  <td>&lt;table&gt;</td><td>Define a table</td>
</tr>
<tr>
  <td>&lt;tr&gt;</td><td>Define a row within a table</td>
</tr>
<tr>
  <td>&lt;th&gt;</td><td>Define table header cells</td>
</tr>
<tr>
  <td>&lt;td&gt;</td><td>Define a table data cells</td>
</tr>
<tr>
  <td>&lt;caption&gt;</td><td>Define a table caption</td>
</tr>
<caption align="bottom"><B>A simple table</B></caption>
</table>

<!--Change font color and size-->
<font color="red" size="7"> This</FONT>
<font color="blue" size="+2"> is</FONT>
<font color="purple" size="+1"> a</FONT>
<font color="teal"> small</FONT>
<font color="yellow" size="-1">example</FONT>
</BODY>
</HTML>
```

Στην εφαρμογή που δημιουργήσαμε χρησιμοποιήσαμε το πρόγραμμα visual studio .Παρακάτω αναφέρουμε λίγα λογία για το visual studio αλλά και για σύστημα διαχείρισης βάσης δεδομένων access της Microsoft . . [Πληροφορίες απο link\(1,10,20,21,22\)](#)

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

## Visual studio

Στην εφαρμογή που δημιουργήσαμε για την εκπλήρωση της πτυχιακής μας εργασίας χρησιμοποιήσαμε τις γλώσσες c sharp και asp με την χρήση του προγράμματος visual studio .

Το Microsoft Visual Studio είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης (IDE) από τη Microsoft. Χρησιμοποιείται για την ανάπτυξη προγραμμάτων ηλεκτρονικών υπολογιστών για τα Microsoft Windows, καθώς και web sites, εφαρμογών και υπηρεσιών web. Visual Studio χρησιμοποιεί πλατφόρμες ανάπτυξης λογισμικού της Microsoft, όπως τα Windows API, Windows Forms, Windows Presentation Foundation, το Windows Store και το Microsoft Silverlight. Μπορεί να παράγει τόσο εγγενή κώδικα και διαχειριζόμενο κώδικα.

Άλλα ενσωματωμένα εργαλεία που περιλαμβάνουν έναν σχεδιαστή έντυπα για τη δημιουργία εφαρμογών GUI, web designer, τάξη σχεδιαστής, σχεδιαστής και σχήματος βάσης δεδομένων. Δέχεται plug-ins που ενισχύουν τη λειτουργικότητα σχεδόν σε κάθε επίπεδο, συμπεριλαμβανομένων προσθέτοντας υποστήριξη για συστήματα πηγή ελέγχου (όπως η ανατροπή) και την προσθήκη νέων toolsets όπως εκδότες και visual designers για συγκεκριμένους τομείς γλώσσες ή toolsets για άλλες πτυχές του κύκλου ανάπτυξης λογισμικού (όπως τον πελάτη Team Foundation Server: Team Explorer).

Το Visual Studio υποστηρίζει διαφορετικές γλώσσες προγραμματισμού και επιτρέπει το πρόγραμμα επεξεργασίας κώδικα και εντοπισμού σφαλμάτων για την υποστήριξη (σε διάφορους βαθμούς) σχεδόν οποιαδήποτε γλώσσα προγραμματισμού, εφόσον υπάρχει μια γλώσσα-ειδική υπηρεσία. Built-in γλώσσες περιλαμβάνουν C, C ++ και C ++ / CLI (μέσω Visual C ++), VB.NET (μέσω της Visual Basic .NET), C # (μέσω της Visual C #), και F # (όπως του Visual Studio 2010 ). Υποστήριξη για άλλες γλώσσες, όπως η M, Python, Ruby και μεταξύ άλλων είναι διαθέσιμο μέσω των υπηρεσιών του γλώσσα που έχει εγκατασταθεί ξεχωριστά. Υποστηρίζει, επίσης, XML / XSLT, HTML / XHTML, JavaScript και CSS. Java (και J #) στηρίχθηκαν στο παρελθόν.

Η εταιρεία Microsoft παρέχει "Κοινότητα" εκδόσεις του Visual Studio του χωρίς κόστος. Εμπορικές εκδόσεις του Visual Studio μαζί με επιλεγμένα προηγούμενες εκδόσεις διατίθενται δωρεάν στους φοιτητές μέσω του προγράμματος DreamSpark της Microsoft.

Η Microsoft παρέχει μια προεπισκόπηση του Visual Studio Κώδικα για δωρεάν με ένα ιδιόκτητο άδεια. Πρόκειται για ένα πηγαίο κώδικα και επεξεργαστή κειμένου, μαζί με άλλα χαρακτηριστικά, για Linux, OS. . [Πληροφορίες απο link\(17,18,19\)](#)

## Microsoft access

Η Microsoft Access, επίσης γνωστό ως Microsoft Office Access, είναι ένα σύστημα διαχείρισης βάσεων δεδομένων της Microsoft που συνδυάζει την μηχανή σχεσιακής βάσης δεδομένων Microsoft Jet με μια γραφική διεπαφή χρήστη εργαλεία και ανάπτυξης λογισμικού. Είναι μέλος του Microsoft Office σουίτα εφαρμογών, που περιλαμβάνονται στην επαγγελματική και τριτοβάθμια εκδόσεις ή πωλούνται χωριστά. Microsoft Access αποθηκεύει δεδομένα στη δική της μορφή βασίζεται στη μηχανή βάσης δεδομένων της Access Jet. Μπορεί επίσης να εισαγάγετε ή να συνδέσετε απευθείας σε δεδομένα που είναι αποθηκευμένα σε άλλες εφαρμογές και βάσεις δεδομένων.

Οι προγραμματιστές λογισμικού και αρχιτέκτονες δεδομένα μπορούν να χρησιμοποιήσουν τη Microsoft Access για την ανάπτυξη λογισμικού εφαρμογών, και "οι χρήστες δύναμη" μπορούν να το

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

χρησιμοποιήσουν για να δημιουργήσουν εφαρμογές λογισμικού. Όπως και άλλες εφαρμογές του Office, πρόσβαση υποστηρίζεται από τη Visual Basic for Applications (VBA), μια αντικειμενοστρεφής γλώσσα προγραμματισμού που μπορεί να αναφέρονται σε μια ποικιλία αντικειμένων, συμπεριλαμβανομένων DAO (Data Access Objects), το ActiveX Data Objects, και πολλά άλλα στοιχεία ActiveX. Οπτικά αντικείμενα που χρησιμοποιούνται σε φόρμες και εκθέσεις εκθέτουν τις μεθόδους και τις περιουσίες τους στο περιβάλλον προγραμματισμού VBA και ενόητες κώδικα VBA μπορεί να δηλώσει και να καλέσετε λειτουργίες των Windows λειτουργικό σύστημα.

Πίνακες της Access υποστηρίζει μια ποικιλία τυποποιημένων τύπων πεδίου, δείκτες, και την ακεραιότητα αναφορών, συμπεριλαμβανομένων διαδοχικές ενημερώσεις και διαγραφές. Πρόσβαση περιλαμβάνει επίσης μια διεπαφή ερωτημάτων, φόρμες για να εμφανίσετε και να εισαγάγετε δεδομένα, και τις εκθέσεις για την εκτύπωση. Η υποκείμενη βάση δεδομένων Jet, το οποίο περιέχει τα αντικείμενα αυτά, είναι multi-user και χειρίζεται ρεκόρ κλειδώματος.

Επαναλαμβανόμενες εργασίες μπορούν να αυτοματοποιηθούν μέσω μακροεντολές με τις επιλογές point-and-click. Είναι επίσης εύκολο να τοποθετήσετε μια βάση δεδομένων σε ένα δίκτυο και να έχουν πολλαπλά δεδομένα στους χρήστες να μοιράζονται και να ενημερώνει χωρίς να αντικαταστήσετε το έργο του άλλου. Τα δεδομένα είναι κλειδωμένο σε επίπεδο ρεκόρ το οποίο είναι σημαντικά διαφορετική από το Excel το οποίο κλειδώνει το σύνολο του υπολογιστικού φύλλου.

Υπάρχουν πρότυπο βάσεων δεδομένων στο πλαίσιο του προγράμματος και για τη λήψη από την ιστοσελίδα τους. Αυτές οι επιλογές είναι διαθέσιμες κατά την έναρξη πρόσβαση και επιτρέπουν στους χρήστες να ενισχύσει μια βάση δεδομένων με προκαθορισμένους πίνακες, ερωτήματα, φόρμες, εκθέσεις, και μακροεντολές. Πρότυπα υποστήριξη βάσεων δεδομένων κώδικα VBA, αλλά τα πρότυπα της Microsoft δεν περιλαμβάνει κώδικα VBA.

Οι προγραμματιστές μπορούν να δημιουργήσουν λύσεις που χρησιμοποιούν VBA, το οποίο είναι παρόμοιο με το Visual Basic 6.0 (VB6) και χρησιμοποιείται σε όλα τα προγράμματα του Microsoft Office, όπως το Excel, το Word, το Outlook και το PowerPoint. Οι περισσότεροι κώδικα VB6, συμπεριλαμβανομένης της χρήσης των κλήσεων API των Windows, μπορεί να χρησιμοποιηθεί σε VBA. Οι δυναμικοί χρήστες και προγραμματιστές μπορούν να επεκτείνουν τις βασικές λύσεις του τελικού χρήστη σε μια επαγγελματική λύση με την προηγμένη αυτοματοποίηση, την επικύρωση των δεδομένων, παγίδευση σφαλμάτων, και την υποστήριξη πολλών χρηστών.

Η Microsoft Access έχει σχεδιαστεί για να κλιμακώνεται για να υποστηρίξει περισσότερα δεδομένα και τους χρήστες με σύνδεση σε πολλαπλές βάσεις δεδομένων της Access ή χρησιμοποιώντας ένα back-end βάση δεδομένων, όπως ο Microsoft SQL Server. Με τον τελευταίο σχεδιασμό, το ποσό των δεδομένων και οι χρήστες μπορούν να αναβαθμίσουν τις λύσεις σε επίπεδο επιχείρησης.

Το ρόλο της Microsoft Access στην ανάπτυξη ιστοσελίδων, πριν από την έκδοση του 2010 είναι περιορισμένη. Χαρακτηριστικά διεπαφής χρήστη της Access, όπως φόρμες και εκθέσεις, λειτουργούν μόνο στα Windows. Στις εκδόσεις του 2000 μέχρι και το 2003 ένα είδος αντικειμένου της Access που ονομάζεται σελίδες πρόσβασης δεδομένων που δημιουργήθηκε δημοσιεύσιμη ιστοσελίδων. Σελίδων πρόσβασης δεδομένων δεν υποστηρίζονται πλέον. Το Microsoft Jet μηχανισμός διαχείρισης βάσεων δεδομένων, βασικές για την Access, μπορείτε να έχετε πρόσβαση μέσω τεχνολογιών όπως ODBC ή OLE DB. Τα δεδομένα (π.χ., πίνακες και ερωτήματα) μπορεί να προσεγγιστεί από το web-based εφαρμογές που έχουν αναπτυχθεί σε ASP.NET, PHP, ή Java. Με τη χρήση των υπηρεσιών Terminal Services της Microsoft και Remote Desktop εφαρμογή στον Windows Server 2008 R2, οι οργανισμοί μπορούν να φιλοξενήσουν Πρόσβαση σε εφαρμογές, έτσι ώστε να μπορεί να λειτουργήσει πέρα από τον Ιστό. Αυτή η τεχνική δεν κλιμακώνεται τον τρόπο μια διαδικτυακή εφαρμογή θα, αλλά είναι κατάλληλη για ένα περιορισμένο αριθμό χρηστών ανάλογα με τη διαμόρφωση του ξενιστή. . [Πληροφορίες απο link\(14,15,16\)](#).

## Κεφάλαιο 3<sup>ο</sup>

### ASP

Η ASP που σημαίνει Active Server Pages είναι μια γλώσσα προγραμματισμού που έχει σαν χαρακτηριστικό ότι μεταφράζει τον κώδικα στον Server και μετά τον φορτώνει στην HTML έτσι ο χρήστης δεν μπορεί να δει τον αρχικό κώδικα. Αργότερα έγινε γνωστή με το όνομα ASP classic για δυναμική παραγωγή ιστοσελίδων. Η ASP έχει μια ιδιαίτερη τεχνολογία με την οποία μπορούμε να κατασκευάσουμε δυναμικές ιστοσελίδες.

Ένας βασικός κανόνας της ASP είναι ότι το αρχείο έχει τα Tags της HTML όπως επίσης έχει και Server Scripts ανάμεσα στα delimiters (<%.....%>). Εκτελούνται πάντα στον Server και πρέπει να έχουν κάποια έκφραση, διαδικασία, εντολή ή κάποιους έγκυρους τελεστές της γλώσσας που χρησιμοποιούμε. Με το αντικείμενο Response και τη μέθοδο Write της ASP μπορούμε να στείλουμε το περιεχόμενο στον στον φυλλομετρητή.

Τα αρχεία ASP:

- είναι ίδια με τα αρχεία HTML.
- μπορεί να περιέχει tags και scripts όπως στην HTML.
- Εκτελούνται στον Server τα scripts.
- Το αρχείο έχει κατάληξη σε .asp.
- Με την ίδια τεχνολογία δουλεύει και η **PHP** αλλά και η **CFML**.

Οι δυνατότητες της ASP είναι οι παρακάτω:

- Μπορεί να αλλάξει την ιστοσελίδα έτσι ώστε να είναι πιο εύχρηστη από τον χρήστη.
- Είναι πολύ απλή και γρήγορη σε σχέση με την CGI και την Perl.
- Είναι πιο ασφαλής γιατί ο κώδικας δεν μπορεί να διαβαστεί από τον φυλλομετρητή.
- Αλλάζει δυναμικά το περιεχόμενο της σελίδας.
- Σε περίπτωση που χρησιμοποιούμε και την HTML είναι δυνατόν να εμφανιστούν τα αρχεία από το φυλλομετρητή.
- Υπάρχει η δυνατότητα να τροποποιηθεί το περιεχόμενο της σελίδας δυναμικά.
- Υπάρχει η δυνατότητα απάντησης σε ερωτήσεις χρηστών μέσω της HTML.

**Τρόπος λειτουργίας της ASP:**

- Ο Server επιστρέφει τα αρχεία όταν ο φυλλομετρητής κάνει request σε αρχείο HTML.
- Server καλεί την ASP όταν ο φυλλομετρητής κάνει request για ένα αρχείο ASP.
- Η ASP κάνει ανάγνωση του αρχείου ASP και στη συνέχεια εκτελεί τα Scripts.
- Στο τέλος το αρχείο ASP επιστρέφει σαν αρχείο HTML στον φυλλομετρητή.
- Με το αντικείμενο Response και τη μέθοδο Write της ASP μπορούμε να στείλουμε το περιεχόμενο στον φυλλομετρητή. Ένα παράδειγμα θα παραθέσουμε στη συνέχεια:

```
<html>
```

```
<body>
```

---

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

```
<%  
Response.write"Καλημέρα!"  
%>  
</body>  
</html>
```

### Μεταβλητές

Για τις μεταβλητές που δημιουργούνται απο την αρχή της διαδικασίας υπάρχει η δυνατότητα να αλλαχθούν απο τα Script του ASP σε περίπτωση όμως που τις δημιουργήσουμε και τη δηλώσουμε μέσα στην διαδικασία θα χάνεται κάθε φορά που εκτελούμε αυτή τη διαδικασία. Πως δημιουργούμε μεταβλητές θα δούμε στη συνέχεια:

```
<html> . . Πληροφορίες απο link\(4,12\)
```

```
<body>  
<%  
Dim name  
name="Κτωρίδης Αντρέας"  
Response.write("Με λένε:" & name)%>  
</body>  
</html>
```

### Πίνακας:

Για να δημιουργήσουμε ένα πίνακα στην ASP που περιέχει ονόματα κάνουμε το εξής:

```
<html> <body>  
<% Dim famname(4)  
famname(0)="Νικόδημου"  
famname(1)="Φιλίππου"  
famname(2) ="Νεοφύτου"  
famname(3)="Νικολάου"
```

```
famname(4)="Ανδρέου"

for i=0 to 4

response. Write(famname(i) & "<br>")

next %>

</body> <html>
```

### Βρόχος (LOOP):

Υπάρχουν έξι ήδη επικεφαλίδας στα αρχεία της HTML. Στο επόμενο παράδειγμα μπορούμε να δούμε πως γίνεται ένα loop.

```
<Html> <body>

<% Dim i

For i=1 to 6

Response. Write ("<h" & i &"> Header "& i &"</h" & i &">")

Next %>

</body> </html>
```

## Γλώσσα προγραμματισμού C sharp ή C#

Το όνομα της είναι «εμπνευσμένο» απο μια μουσική σημειωγραφία. Έχει καταλήξει το σύμβολο sharp # το οποίο έχει χρησιμοποιηθεί και σε άλλες γλώσσες όπως η A# F# και Eiffel#. Έχει δημιουργηθεί απο την εταιρία Microsoft πάνω στην πλατφόρμα Net και ακολούθως είχε αναγνώριση απο την Emca και την Iso είναι μια απλή αντικειμενοστραφής γλώσσα για γενική χρήση. Δημιουργήθηκε για την Common Language Infrastructure και στα μέσα Αυγούστου του 2012 κυκλοφόρησε η έκδοση S.O. η οποία είναι και η πιο πρόσφατη.

Δημιουργός της είναι ο Anders Hejlsberg ο οποίος μαζί με την ομάδα του το 1999 είχαν δημιουργήσει μια νέα γλώσσα με όνομα Cool C (-like- object oriented Language). Ακολούθος τον Ιούλιο του 2000 δημιουργήθηκε μια νέα πλατφόρμα με το όνομα NET και η γλώσσα είχε μετονομαστεί σε C# πάνω στην οποία εισήχθησαν βιβλιοθήκες της ASP.NET. Το 1994 ο σχεδιαστής της Java James Gosling μαζί με τον Bill Joy συνιδρυτής της Sun Microsystems ανέφεραν ότι η C# είναι κλώνος της Java. Όμως τον Ιούνιο του 2000 ο A.Helisberg υποστήριξε ότι είναι πιο κοντά στην C++ και οτι δεν έχει σχέση με την Java.Απο το 2005 που κυκλοφόρησε η 2<sup>η</sup> έκδοση της C# άρχισε να μοιάζει ολο και λιγότερο της Java και μια απο τις πρώτες διαφορές τους ήταν η δημιουργία των Generic Object.Η C# περιέχει πρώτης κλάσης Generic Objects τα οποία μπορούν να χρησιμοποιηθούν σε οποιοδήποτε άλλη κλάση με τον κώδικα να εκτελείτε ταυτόχρονα. .

[Πληροφορίες απο link\(23,24\)](#)

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

---

## Αποσπάσματα από κώδικα σου συστήματος που έχουμε δημιουργήσει:

Εδώ δηλώνουμε και καθορίζουμε πιο θα είναι το connection string ( ποια θα είναι βάση δεδομένων μας όταν θα εκτελούμε τις διάφορες εντολές που αφορούν αλλαγές στη βάση μας ) το όνομα του Connection μας είναι **con**.


```
public partial class Grades : System.Web.UI.Page
{
    System.Data.OleDb.OleDbConnection con;
    string path =AppDomain.CurrentDomain.GetData("DataDirectory").ToString();
    protected void Page_Load(object sender, EventArgs e)
    {
        con = new System.Data.OleDb.OleDbConnection();

        con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0; Data
        Source="+path+"\\students.accdb";
    }
}
```

Στο απόσπασμα που ακολουθεί ενδεικτικά έχουμε επιλέξει το σημείο όπου γίνεται ενημέρωση των προσωπικών στοιχείων του μαθητή κάνοντας χρήση της εντολής **cmd.ExecuteNonQuery()** αφού προηγουμένως **con.Open()** έχουμε ανοίξει το Connection. Για την εντολή για την αντιμετώπιση πιθανόν λαθών χρησιμοποιήσαμε την εντολή **try Catch**.

```
protected void GridView1_RowUpdating(object sender, GridViewUpdateEventArgs e)
{
    try
    {
        GridViewRow row = (GridViewRow)GridView1.Rows[e.RowIndex];
        Label lbl = (Label)row.FindControl("lblnum");
        TextBox txtname = (TextBox)row.FindControl("txtname");
        TextBox txtaddress = (TextBox)row.FindControl("txtaddress");
        TextBox txtsurname = (TextBox)row.FindControl("txtsurname");
    }
}
```

Εδώ ορίζονται τις παραμέτρους που θα χρησιμοποιηθούν στην εντολή UPDATE



Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

```

        TextBox txttelephone = (TextBox)row.FindControl("txttelephone");
        TextBox textdad = (TextBox)row.FindControl("txtdad");
        TextBox textNamMun = (TextBox)row.FindControl("txtNamMun");
        TextBox textSurMun = (TextBox)row.FindControl("txtSurMun");
        TextBox textTelDad = (TextBox)row.FindControl("txtTelDad");
        TextBox textclass = (TextBox)row.FindControl("txtclass");

string SqlString = "update student set ΟΝΟΜΑ=?, ΕΠΙΘΕΤΟ=?, ΔΙΕΘΥΝΣΗ=?, ΤΗΛΕΦΩΝΟ=?, [ΟΝΟΜΑ
ΠΑΤΕΡΑ]=?, [ΟΝΟΜΑ ΜΗΤΕΡΑΣ]=? ,[ΕΠΙΘΕΤΟ ΜΗΤΕΡΑΣ]=?, [ΤΗΛΕΦΩΝΟ ΠΑΤΕΡΑ]=?, ΤΜΗΜΑ=? where
ΑΜ=?";

using (OleDbCommand cmd = new OleDbCommand(SqlString, con))
{
    cmd.CommandType = CommandType.Text;
    cmd.Parameters.AddWithValue("ΟΝΟΜΑ", textName.Text);

    cmd.Parameters.AddWithValue("ΕΠΙΘΕΤΟ", textsurname.Text);
    cmd.Parameters.AddWithValue("ΔΙΕΥΘΥΝΣΗ", textaddress.Text);
    cmd.Parameters.AddWithValue("ΤΗΛΕΦΩΝΟ", txttelephone.Text);
    cmd.Parameters.AddWithValue("[ΟΝΟΜΑ ΠΑΤΕΡΑ]", textdad.Text);
    cmd.Parameters.AddWithValue("[ΟΝΟΜΑ ΜΗΤΕΡΑΣ]",
textNamMun.Text);
    cmd.Parameters.AddWithValue("[ΕΠΙΘΕΤΟ ΜΗΤΕΡΑΣ]",
textSurMun.Text);
    cmd.Parameters.AddWithValue("[ΤΗΛΕΦΩΝΟ ΠΑΤΕΡΑ]",
textTelDad.Text);
    cmd.Parameters.AddWithValue("ΤΜΗΜΑ",
textclass.Text.ToString());
    cmd.Parameters.AddWithValue("ΑΜ", lbl.Text);

    con.Open();
    cmd.ExecuteNonQuery();
    con.Close();

    GridView1.DataBind();
}
}

catch (Exception ex)
{
    lblmessage.Text = "Κάποιο πρόβλημα προέκυψε στην ενημέρωση " + ex;
}
}

```

Η εντολή `cmd.ExecuteNonQuery()` δέχεται παραμέτρους τις τιμές της τρέχουσας γραμμής της επιλογής μας αυτές η παραμέτρους ορίζονται μέσα από το κώδικά **Asp**

ΕΠΙΘΕΤΟ	ΔΙΕΘΥΝΣΗ	ΤΗΛΕΦΩΝΟ	ΟΝ_ΠΑΤ	ΕΠΙΘ_ΜΗΤ	ΟΝ_ΜΗΤ	ΤΜΗΜΑ	ΤΗΛ_ΠΑΤ			
Βασιλείου	Κέδρων 27	99888699	Χριστός	Γεωργίου	Μαρίνα	A1	99888846	Ενημέρωση	Ακύρωση	Επιλογή



---

```

<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
    CellPadding="4" DataKeyNames="AM" DataSourceID="GViewStudent"
    onrowediting="GridView1_RowEditing" onrowupdating="GridView1_RowUpdating"
    onselectedindexchanged="GridView1_SelectedIndexChanged"
    onrowcommand="GridView1_RowCommand"
>
    <Columns>
        <asp:TemplateField HeaderText="AM" SortExpression="AM">
            <ItemTemplate>
                <asp:Label ID="lblnum" runat="server" Text='<#
                    Eval("AM") %>'>
            </asp:Label>
        </ItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="ONOMA">
        <ItemTemplate>
            <# Eval("ONOMA") %>
        </ItemTemplate>
        <EditItemTemplate>
            <asp:TextBox ID="txtname" runat="server" Text='<#
                Eval("ONOMA") %>' Width="80">
            </asp:TextBox>
        </EditItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="ΕΠΙΘΕΤΟ">
        <ItemTemplate>
            <# Eval("ΕΠΙΘΕΤΟ") %>
        </ItemTemplate>
        <EditItemTemplate>
            <asp:TextBox ID="txtsurname" runat="server" Text='<#
                Eval("ΕΠΙΘΕΤΟ") %>'
                Width="80">
            </asp:TextBox>
        </EditItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="ΔΙΕΘΥΝΣΗ">
        <ItemTemplate>
            <# Eval("ΔΙΕΘΥΝΣΗ") %>
        </ItemTemplate>
        <EditItemTemplate>
            <asp:TextBox ID="txtaddress" runat="server" Text='<#
                Eval("ΔΙΕΘΥΝΣΗ") %>'
                Width="80">
            </asp:TextBox>
        </EditItemTemplate>
    </asp:TemplateField>
    <asp:TemplateField HeaderText="ΤΗΛΕΦΩΝΟ">

```

```

<ItemTemplate>
    <%# Eval("ΤΗΛΕΦΩΝΟ") %>
</ItemTemplate>
<EditItemTemplate>
    <asp:TextBox ID="txttelephone" runat="server"
        Text='<%# Eval("ΤΗΛΕΦΩΝΟ") %>'
        Width="80">
    </asp:TextBox>
</EditItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="ΟΝ_ΠΑΤ">
    <ItemTemplate>
        <%# Eval("ΟΝΟΜΑ_ΠΑΤΕΡΑ") %>
    </ItemTemplate>
    <EditItemTemplate>
        <asp:TextBox ID="txtdad" runat="server" Text='<%#
            Eval("ΟΝΟΜΑ_ΠΑΤΕΡΑ") %>'
            Width="80">
        </asp:TextBox>
    </EditItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="ΕΠΙΘ_ΜΗΤ">
    <ItemTemplate>
        <%# Eval("ΕΠΙΘΕΤΟ_ΜΗΤΕΡΑΣ") %>
    </ItemTemplate>
    <EditItemTemplate>
        <asp:TextBox ID="txtSurMun" runat="server"
            Text='<%# Eval("ΕΠΙΘΕΤΟ_ΜΗΤΕΡΑΣ") %>' Width="80">
        </asp:TextBox>
    </EditItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="ΟΝ_ΜΗΤ">
    <ItemTemplate>
        <%# Eval("ΟΝΟΜΑ_ΜΗΤΕΡΑΣ") %>
    </ItemTemplate>
    <EditItemTemplate>
        <asp:TextBox ID="txtNamMun" runat="server" Text='<%#
            Eval("ΟΝΟΜΑ_ΜΗΤΕΡΑΣ") %>'
            Width="80">
        </asp:TextBox>
    </EditItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="ΤΜΗΜΑ">
    <ItemTemplate>
        <%# Eval("ΤΜΗΜΑ") %>
    </ItemTemplate>
    <EditItemTemplate>
        <asp:TextBox ID="txtclass" ReadOnly="true"
            runat="server" Text='<%# Eval("ΤΜΗΜΑ") %>'
            Width="20">
        </asp:TextBox>
    </EditItemTemplate>
</asp:TemplateField>

```

```

</asp:TemplateField>
<asp:TemplateField HeaderText="ΤΗΛ_ΠΑΤ">
  <ItemTemplate>
    <%# Eval("ΤΗΛΕΦΩΝΟ_ΠΑΤΕΡΑ") %>
  </ItemTemplate>
  <EditItemTemplate>
    <asp:TextBox ID="txtTelDad" runat="server"
      Text='<%# Eval("ΤΗΛΕΦΩΝΟ_ΠΑΤΕΡΑ") %>' Width="80">
    </asp:TextBox>
  </EditItemTemplate>
</asp:TemplateField>
<asp:CommandField ButtonType="Button" ShowEditButton="True"
  CancelText="Ακύρωση" EditText="Αλλαγή" SelectText="Αλλαγή"
  UpdateText="Ενημέρωση"/>
<asp:TemplateField>
  <ItemTemplate>
    <asp:Button runat="server" ID="Select"
      Text="Επιλογή"
      CommandName="Select"
      CommandArgument="<%# ((GridViewRow) Container).RowIndex %>" />
  </ItemTemplate>
</asp:TemplateField>

```

---

## ΕΓΧΕΙΡΙΔΙΟ ΧΡΗΣΗΣ

# ΕΡΓΑΛΕΙΑ

Για την δημιουργία του συστήματος διαχείρισης Βαθμών και Απουσιών για Σχολείο Μέσης Εκπαίδευσης χρησιμοποιήσαμε το **Visual Studio 10 με C# και Asp.net**.

Η βάση δεδομένων είναι αποθηκευμένη σε **Access 2010**.

---

# ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

---

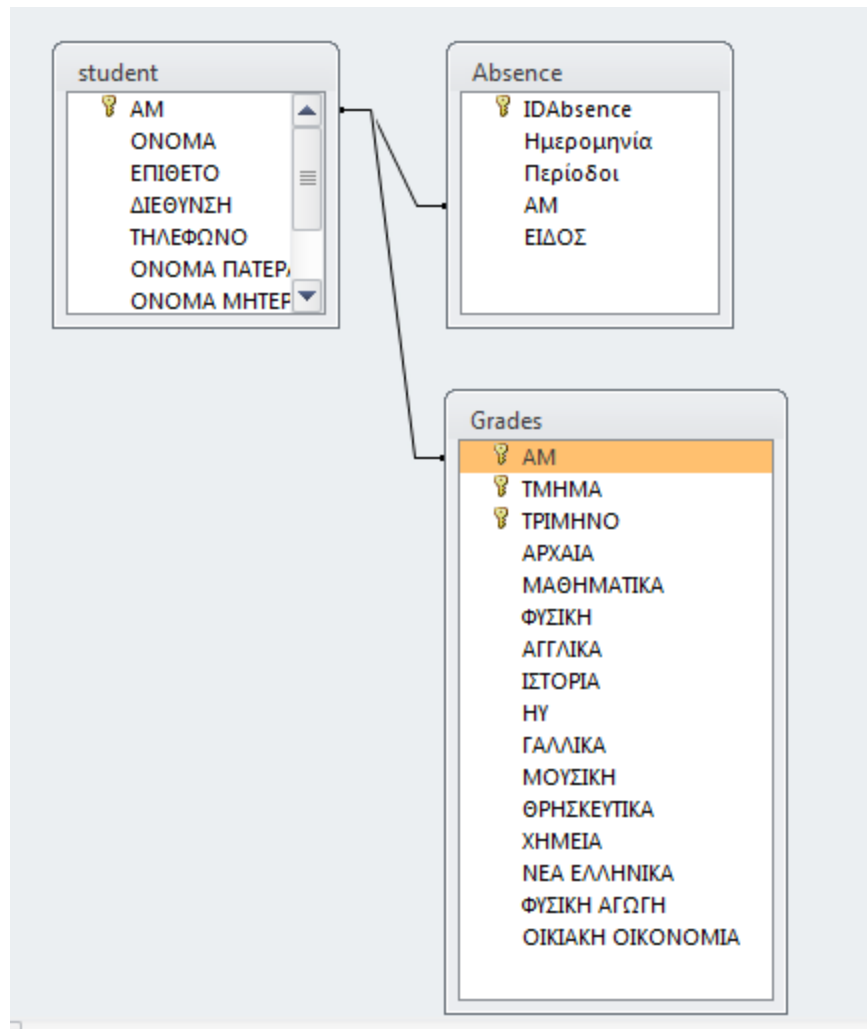
Τρεις είναι οι βασικοί πίνακες της βάσης με όνομα **Student.accdb**.

- ⊙ Student ( Στοιχεία Μαθητών)  
Με πεδία **AM**, ΟΝΟΜΑ, ΕΠΙΘΕΤΟ, ΔΙΕΥΘΥΝΣΗ, ΤΗΛΕΦΩΝΟ, ΟΝΟΜΑ ΠΑΤΕΡΑ, ΟΝΟΜΑ ΜΗΤΕΡΑΣ, ΕΠΙΘΕΤΟ ΜΗΤΕΡΑΣ, ΤΗΛΕΦΩΝΟ ΠΑΤΕΡΑ, ΤΜΗΜΑ
- ⊙ Absence ( Απουσίες )  
Με πεδία **IDAbsence**, Ημερομηνία, Περίοδοι, AM, ΕΙΔΟΣ
- ⊙ Grades ( Βαθμοί )  
Με πεδία **AM**, **ΤΜΗΜΑ**, **ΤΡΙΜΗΝΟ**, ΑΡΧΑΙΑ, ΜΑΘΗΜΑΤΙΚΑ, ΦΥΣΙΚΗ, ΑΓΓΛΙΚΑ, ΙΣΤΟΡΙΑ, Η/Υ, ΓΑΛΛΙΚΑ, ΜΟΥΣΙΚΗ, ΘΡΗΣΚΕΥΤΙΚΑ, ΧΗΜΕΙΑ, ΝΕΑ ΕΛΛΗΝΙΚΑ, ΦΥΣΙΚΗ ΑΓΩΓΗ, ΟΙΚΙΑΚΗ ΟΙΚΟΝΟΜΙΑ

Και ανάμεσα στους τρεις βασικούς πίνακες υπάρχουν σχέσεις.

Student – Absence ( One to Many ) AM-AM

Student – Grades ( One to Many ) AM-AM



Υπάρχουν στη βάση ακόμα τρεις βοηθητικοί πίνακες για τον υπολογισμό **Μέσου Όρου των Βαθμολογιών του κάθε μαθητή ξεχωριστά για κάθε τετράμηνο**. Αυτά τα στοιχεία θα χρησιμοποιηθούν για την παραγωγή των απαραίτητων **Εκθέσεων**.

TempStudentMo

- 🔑 AM
- ΟΝΟΜΑ
- ΕΠΙΘΕΤΟ
- 🔑 ΤΜΗΜΑ
- 🔑 ΤΡΙΜΗΝΟ
- ΜΟ

Settings

Όριο\_Δικαιολογημ

Όριο\_Αδικαιολόγη

lesson

- 🔑 ΟΝΟΜΑ
- ΔΙΔΑΚΤΙΚΕΣ\_ΠΕΡΙΟΔ
- Ποσοστά\_ΜΟ

## ΜΕΝΟΥ ΕΠΙΛΟΓΩΝ

### ΒΑΘΜΟΛΟΓΙΑ

#### 1. Κατά Τμήμα

Τμήμα: A1 ▼

Επιλογή Μαθητή για  
Παρουσίαση της  
Βαθμολογίας του

2424	Ανδρέας	Μιλτιάδους	Καλογερόπουλου 4	25678765	Χρίστος	Πελοπίδα	Πηλενόπη	A1	98767667	Αλλαγή	Επιλογή
9219	Χαράλαμπος	Φράγκου	Μακεδονίας 4	9987644	Μέλιος	Χρυσάνθου	Μαρία	A1	9987665	Αλλαγή	Επιλογή



Επικύρωση καταχώρησης

Αλλαγή Στοιχείων Μαθητή

Στοιχεία Βαθμολογίας του Μαθητή που έχετε επιλέξει. 9219

A' Τετράμηνο 12 8 3 18 9 12 17 12 12 11 11 14 7

B' Τετράμηνο 11 6 9 8 9 16 13 12 11 14 13 15 12 ✔ ✘

Τεχνολογικό Εκπαιδευτικό Ίδρυμα / Τμήμα Μηχανικών Πληροφορικής

**Εισαγωγή** Μαθητή – Με την εισαγωγή νέου Μαθητή Καταχωρείται και αυτόματα μηδενική βαθμολογία για όλα τα μαθήματα για το συγκεκριμένο μαθητή

# 38

Επικύρωση  
βαθμολογίας

Ακύρωση αλλαγών

Η επιλογή του κουμπιού **Αλλαγή** μπορούμε να αλλάξουμε τα προσωπικά στοιχεία του κάθε μαθητή.

Η επιλογή του κουμπιού **Εισαγωγή** μπορούμε να εισάγουμε ένα νέο μαθητή και την συνέχεια επιλέγουμε **Επικύρωση**.

Με την επιλογή του κουμπιού **Επιλογή** θα μπορέσουμε να δούμε τη βαθμολογία του μαθητή και να την επεξεργαστούμε.

## 2. Κατά Μαθητή

### Στοιχεία και Βαθμολογία ΜΑΘΗΤΗ

Τμήμα: A1

Κωδικός	Επίθετο	Όνομα	Τμήμα
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Όνομα Πατέρα	Όνομα Μητέρας	Τηλ. Πατέρα	Διεύθυνση
<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
Επίθ. Μητέρας	Τηλέφωνο		
<input type="text"/>	<input type="text"/>		

	Αρχαία	Μαθηματικά	Φυσική	Αγγλικά	Ιστορία	Η/Υ	Μουσική	Γαλλικά	Θρησκευτικά	Χημεία	Νέα Ελληνικά	Φυσική Αγωγή	Οικιακή Οικονομία
A' Τετράμηνο	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
B' Τετράμηνο	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
<<	<	>	>>	>*	0 / 0	Αριθμός Μαθητών							
Πρώτος	Τελευταίος	Νέος Μαθητής	Επικύρωση - Αλλαγής στοιχείων ή Επικύρωση - Εισαγωνή στοιχείων										

!!! Καμία αλλαγή δεν γίνεται αν οι βαθμολογίες των μαθητών δεν είναι μεταξύ των τιμών 0 έως 20.



## ΑΠΟΥΣΙΕΣ

Για να μπορέσουμε να δούμε τις απουσίες του μαθητή θα πρέπει να επιλέξουμε τον μαθητή. Αν έχει απουσίες θα παρουσιαστεί μια λίστα με τις απουσίες **ταξινομημένη** κατά **ημερομηνία**. Στη συνέχεια αν επιλέξουμε το κουμπί **εισαγωγής νέας** θα μπορούμε να εισάγουμε νέα απουσία επιλέγουμε **Δ** για δικαιολογημένη απουσία ή **A** για αδικαιολόγητη.

Μας δίνεται αυτόματα ο μεγαλύτερος αριθμός (Κωδικός απουσίας ) αυξανόμενος κατά ένα.

*!!! Καμία αλλαγή δεν γίνεται αν ο αριθμός περιόδων που θα δώσουμε δεν είναι νούμερο ή δεν επιλέξουμε ημερομηνία.*

9219	Χαράλαμπος	Μακεδονίας 4	Φράγκου	9987644	Μέλιος	Χρυσάνθου	Μαρία	A1	9987665	Επιλογή
------	------------	--------------	---------	---------	--------	-----------	-------	----	---------	---------

Κωδικός Μαθητή που έχετε επιλέξει: 9219

Αρ.Απουσιών	Ημερομηνία	Περίοδοι	AM	ΕΙΔΟΣ(A ή Δ)		
29	12/05/2015	4	9219	A	Αλλαγή	Διαγραφή
30	20/05/2015	3	9219	Δ	Αλλαγή	Διαγραφή

Καμία εργασία σε εξέλιξη!!!



Επέλεξε Ημερομηνία Απουσίας:

36 9219  Δ  Σύνολο Απουσιών Μαθητή: 7

Μεγαλύτερος Αριθμός - (Κωδικός απουσίας )  
αυξανόμενος κατά **ΈΝΑ**.

Συνολικό Αριθμός Απουσιών  
- Αδικαιολόγητες +  
Δικαιολογημένες.

Το **SqlSource** για τον να υπολογίζεται ποιος θα είναι ο επόμενος κωδικός απουσίας.

```
Select MAX(IDAbsence) as CountA from Absence
```

## ΡΥΘΜΙΣΕΙΣ

### Μαθήματα

Καθορίστε τα Κύρια/Δευτερεύοντα Μαθήματα για τον Υπολογισμό Μέσου Όρου Βαθμολογιών:

ΟΝΟΜΑ	ΔΙΔΑΚΤΙΚΕΣ ΠΕΡΙΟΔΟΙ	Ποσοστά_ΜΟ	
ΧΗΜΕΙΑ	2	20	Αλλαγή
ΑΓΓΛΙΚΑ	1	10	Αλλαγή
ΑΡΧΑΙΑ	1	10	Αλλαγή
ΓΑΛΛΙΚΑ	2	20	Αλλαγή
ΗΥ	2	20	Αλλαγή
ΦΡΗΣΚΕΥΤΙΚΑ	1	10	Αλλαγή
ΙΣΤΟΡΙΑ	4	40	Αλλαγή
ΜΑΘΗΜΑΤΙΚΑ	4	40	Αλλαγή
ΜΟΥΣΙΚΗ	1	10	Αλλαγή
ΝΕΑ ΕΛΛΗΝΙΚΑ	4	40	Αλλαγή
ΟΙΚΙΑΚΗ ΟΚΟΝΟΜΙΑ	1	10	Αλλαγή
ΦΥΣΙΚΗ	4	40	Αλλαγή
ΦΥΣΙΚΗ ΑΓΩΓΗ	2	20	Αλλαγή

Το πεδίο Διδακτικές Περίοδοι χρησιμοποιείται για τον υπολογισμό του μέσου όρου

-

( Σύνολο Διδακτικών Ορών)

$2/29$ \*Βαθμός Χημείας +  
 $1/29$ \*Βαθμός Αγγλικών +  
 $1/29$ \*Βαθμός Αρχαίων +  
 $2/29$ \*Βαθμός Γαλλικών +

## Καταστάσεις(Εκθέσεις)

Έχουν δημιουργηθεί 4 καταστάσεις για την παρουσίαση πληροφοριών που αφορούν την βάση δεδομένων **Student.acddb**.

- ⊙ Λίστα με Απουσίες
- ⊙ Δελτίο Προόδου Μαθητή
- ⊙ Λίστα Καλύτερων ανά Τμήμα
- ⊙ Λίστα Καλύτερων ανά Τάξη (Α,Β,Γ)

### ΗΛΕΚΤΡΟΝΙΚΗ ΚΑΤΑΧΩΡΗΣΗ ΒΑΘΜΩΝ ΚΑΙ ΑΠΟΥΣΙΩΝ

Αρχική

Πληροφορίες

Βαθμολόγια

Απουσίες

Ρυθμίσεις

Καταστάσεις

Τμήμα: A1

Λίστα με Απουσίες

Δελτίο Προόδου Μαθητή

Λίστα Καλύτερων ανά Τμήμα

Λίστα Καλύτερων ανά Τάξη( Α,Β,Γ)

ΑΜ	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΔΙΕΘΥΝΣΗ	ΤΗΛΕΦΩΝΟ	ΟΝ_ΠΑΤ	ΕΠΙΘ_ΜΗΤ	ΟΝ_ΜΗΤ				
2333	Νικόλαος	Βασιλείου	Κέδρων 26	99888699	Χριστός	Γεωργίου	Μαρίνα	A1	99888846	Αλλαγή	Επιλογή
5656	Μάρκος	Νικολάου	Πλατεών 45	97888888	Θεόδουλος	Μάρκου	Νίκη	A1	99999999	Αλλαγή	Επιλογή

## 1. ΛΙΣΤΑ ΜΕ ΑΠΟΥΣΙΕΣ

Επέλεξε το όριο Απουσιών :



Για να προετοιμαστεί το Report

AM	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΔΙΕΘΥΝΣΗ	ΠΑΤΕΡΑΣ	ΤΗΛΕΦΩΝΟ	ΔΙΚΑΙΟΛΟΓΗΜΕΝΕΣ	ΑΔΙΚΑΙΟΛΟΓΗΤΕΣ	ΣΥΝΟΛΟ ΑΠΟΥΣΙΩΝ
<b>A1</b>								
1000	Αντρέου	Μάρκος	Πειρεώς	Κωνσταντίνος	9988888	0	8	8
1009	Μυρτώ	Λουκάς	Αθανασίου 55	Νικόλαος	99099887	0	3	3
2333	Νικόλαος	Βασιλείου	Κέδρων 26	Χριστός	99888846	1	25	26
2424	Ανδρέας	Μιλτιάδους	Καλογερόπουλου 4	Χρίστος	98767667	2	0	2
5656	Μάρκος	Νικολάου	Πλατεών 45	Θεόδουλος	99999999	11	7	18
5659	Μαρία	Νικοδήμου	Βαλθαλομαίου 22	Χρίστος	9877777	9	0	9
9219	Χαράλαμπος	Φράγκου	Μακεδονίας 4	Μέλιος	9987665	3	4	7
<b>B1</b>								
1004	Μαρίνα	Νεοφύτου	Μάρκου 20	Χριστόδουλος	99887764	0	2	2
1111	Μαρία	Νικολάου	Μεσολογγίου	Σπύρος	9900000	2	0	2
2222	Νέστωρας	Κυριαζή	Λεμεσού	Μανώλη	24999991	2	0	2
4321	Μέλιος	Θεμιστοκλέους	Καρπάθου 2	Δημήτρης	9999999	2	0	2
5677	Μαρία	Νικολάου	Ριζοκαρπάσου 44	Μάρκος	99975321	2	0	2
8000	Αντρέας	Νεοφύτου	Θεσσαλονίκης 98	Χριστόφορος	98888889	2	0	2

Προσδιορίζουμε τον αριθμό των απουσιών πχ έχουν συμπληρώσει μεγαλύτερο ή ίσο αριθμό απουσιών από 2

Για τη δημιουργία της έκθεσης έχουν χρησιμοποιηθεί οι πίνακες **Students** και **Absence**. Το **DataSet** που έχει σχεδιαστεί από τους δύο πίνακες έχει το εξής **SqlSource**.

```
SELECT student.AM, student.ΟΝΟΜΑ, student.ΕΠΙΘΕΤΟ, Absence.IDAbsence AS
ΑΠΟΥΣΙΑ, Absence.Ημερομηνία AS ΗΜΕΡΟΜΗΝΙΑ, Absence.Περίοδοι AS ΠΕΡΙΟΔΟΙ,
Absence.ΕΙΔΟΣ,
student.ΔΙΕΘΥΝΣΗ, student.ΤΜΗΜΑ, student.[ΟΝΟΜΑ ΠΑΤΕΡΑ] AS ΠΑΤΕΡΑΣ,
student.[ΤΗΛΕΦΩΝΟ ΠΑΤΕΡΑ] AS ΤΗΛΕΦΩΝΟ
FROM (Absence INNER JOIN
student ON Absence.AM = student.AM)
```

Στη στήλη ΔΙΚΑΙΟΛΟΓΗΜΕΝΕΣ απουσίες έχει χρησιμοποιηθεί ο πιο κάτω τύπος.  
=SUM(IIf(Fields!ΕΙΔΟΣ.Value="Δ", CDb1(Fields!ΠΕΡΙΟΔΟΙ.Value), 0.00))

Στη στήλη ΑΔΙΚΑΙΟΛΟΓΗΤΕΣ απουσίες έχει χρησιμοποιηθεί ο πιο κάτω τύπος.  
=SUM(IIf(Fields!ΕΙΔΟΣ.Value="Α", CDb1(Fields!ΠΕΡΙΟΔΟΙ.Value), 0.00))

Στη στήλη ΣΥΝΟΛΟ απουσίες έχει χρησιμοποιηθεί ο πιο κάτω τύπος.  
=Sum(CDb1((Fields!ΠΕΡΙΟΔΟΙ.Value)))

## 2. ΔΕΛΤΙΟ ΠΡΟΟΔΟΥ ΜΑΘΗΤΗ

Για τη δημιουργία της έκθεσης έχουν χρησιμοποιηθεί οι πίνακες **Students** και **Grades**. Το **DataSet** που έχει σχεδιαστεί από τους δύο πίνακες έχει το εξής **SqlSource** (το *Report* είναι παραμετρικό).

```
SELECT student.AM, student.ΟΝΟΜΑ, student.ΕΠΙΘΕΤΟ, student.ΤΜΗΜΑ, Grades.ΤΡΙΜΗΝΟ,
Grades.ΑΡΧΑΙΑ, Grades.ΜΑΘΗΜΑΤΙΚΑ, Grades.ΦΥΣΙΚΗ, Grades.ΑΓΓΛΙΚΑ, Grades.ΙΣΤΟΡΙΑ,
Grades.ΗΥ, Grades.ΓΑΛΛΙΚΑ, Grades.ΜΟΥΣΙΚΗ, Grades.ΘΡΗΣΚΕΥΤΙΚΑ, Grades.ΧΗΜΕΙΑ,
Grades.[ΝΕΑ ΕΛΛΗΝΙΚΑ] AS ΕΛΛΗΝΙΚΑ, Grades.[ΦΥΣΙΚΗ ΑΓΩΓΗ] AS ΓΥΜΝΑΣΤΙΚΗ,
Grades.[ΟΙΚΙΑΚΗ ΟΙΚΟΝΟΜΙΑ] AS ΟΙΚΟΝΟΜΙΑ, Grades.AM AS Expr1 FROM (student INNER
JOIN Grades ON student.AM = Grades.AM)
```

Στη στήλη ΜΟ έχει χρησιμοποιηθεί ο πιο κάτω τύπος:

```
=Round(CDb1(((Fields!ΑΡΧΑΙΑ.Value * Parameters!rptparAnc.Value)/
Parameters!rptparTot.Value + (Fields!ΜΑΘΗΜΑΤΙΚΑ.Value*
Parameters!rptparMat.Value)/ Parameters!rptparTot.Value +
(Fields!ΦΥΣΙΚΗ.Value * Parameters!rptparPhy.Value)/
Parameters!rptparTot.Value + (Fields!ΑΓΓΛΙΚΑ.Value*
Parameters!rptparEng.Value)/ Parameters!rptparTot.Value +
(Fields!ΙΣΤΟΡΙΑ.Value* Parameters!rptparHis.Value)/
Parameters!rptparTot.Value + (Fields!ΗΥ.Value *
Parameters!rptparCom.Value)/ Parameters!rptparTot.Value +
(Fields!ΓΑΛΛΙΚΑ.Value * Parameters!rptparFre.Value)/
Parameters!rptparTot.Value + (Fields!ΜΟΥΣΙΚΗ.Value *
Parameters!rptparMus.Value) / Parameters!rptparTot.Value +
(Fields!ΘΡΗΣΚΕΥΤΙΚΑ.Value * Parameters!rptparRel.Value)/
Parameters!rptparTot.Value + ( Fields!ΧΗΜΕΙΑ.Value *
Parameters!rptparChe.Value)/ Parameters!rptparTot.Value +
(Fields!ΕΛΛΗΝΙΚΑ.Value * Parameters!rptparHel.Value)/
Parameters!rptparTot.Value + (Fields!ΓΥΜΝΑΣΤΙΚΗ.Value *
Parameters!rptparGym.Value)/ Parameters!rptparTot.Value +
(Fields!ΟΙΚΟΝΟΜΙΑ.Value* Parameters!rptparOik.Value)/
Parameters!rptparTot.Value)),2)
```

Επέλεξε για προετοιμασία Ελέγχων Προόδου



Για να προετοιμαστεί το Report

1 of 2 ? Find   Next																
<b>ΤΜΗΜΑ: A1</b>																
AM:	1000	ΤΡΙΜΗΝΟ:	ΑΡΧΑΙΑ	ΜΑΘΗΜΑΤΙΚΑ	ΦΥΣΙΚΗ	ΑΓΓΛΙΚΑ	ΙΣΤΟΡΙΑ	ΗΥ	ΓΑΛΛΙΚΑ	ΜΟΥΣΙΚΗ	ΘΡΗΣ/ΤΙΚΑ	ΧΗΜΕΙΑ	ΕΛΛΗΝΙΚΑ	ΓΥΜΝΑΣΤΙΚΗ	ΟΙΚΟΝΟΜΙΑ	
ΟΝΟΜΑ:	Αντρέου															
ΕΠΙΘΕΤΟ:	Μάρκος															
M.O:	15.725															
	A	18	16	17	17	12	16	19	16	13	13	12	11	20	14.83	
	B	17	19	18	16	18	18	19	18	16	18	13	11	11	16.62	

### 3. ΛΙΣΤΑ ΚΑΛΥΤΕΡΩΝ ΑΝΑ ΤΜΗΜΑ

Στη βάση δεδομένων έχει δημιουργηθεί ένας προσωρινός πίνακας **TempStudentMo** όπου καταχωρούνται τα στοιχεία του μαθητή και ο **Μ.Ο** του μαθητή σε κάθε τρίμηνο σύμφωνα με το πίνακα **Lesson** και τις διδακτικές περιόδους που αντιστοιχούν σε κάθε μάθημα ο πίνακας θα είναι **ταξινομημένος** ανά **τμήμα** και μετά σε **ΦΘΙΝΟΥΣΑ ΣΕΙΡΑ** ανά **ΜΟ** για να γίνει δυνατή η παρουσίαση μόνο των πρώτων 3 μαθητών ανά τμήμα.

```
"SELECT student.AM, student.ONOMA, student.ΕΠΙΘΕΤΟ, student.ΤΜΗΜΑ, AVG(TempStudentMo.MO) AS MO_a, TempStudentMo.AM" +  
" FROM      (TempStudentMo INNER JOIN " +  
"student ON TempStudentMo.AM = student.AM) " +  
"GROUP BY  TempStudentMo.AM, student.ONOMA, student.ΕΠΙΘΕΤΟ, student.ΤΜΗΜΑ, student.AM " +  
"ORDER BY student.ΤΜΗΜΑ, AVG(TempStudentMo.MO) DESC";
```

## ΗΛΕΚΤΡΟΝΙΚΗ ΚΑΤΑΧΩΡΗΣΗ ΒΑΘΜΩΝ ΚΑΙ ΑΠΟΥΣΙΩΝ

Αρχική

Πληροφορίες

Βαθμολογία

Απουσίες

Ρυθμίσεις

Καταστάσεις

Επέλεξε για προετοιμασία 3 Καλύτερων σε Τάξη



ΤΜΗΜΑ	ΑΜ	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΜΟ
<b>A1</b>				
	5659	Μαρία	Νικοδήμου	19.4 *
	8888	Μιχάλης	Μενελάου	17.2 *
	1000	Αντρέου	Μάρκος	15.7 *
<b>A2</b>				
	5000	Γιώργος	Χριστοφή	20.0 *
	1414	Χριστάκης	Παλαιολόγος	18.7 *
	7777	Ελένη	Φιλίππου	12.2 *
<b>B1</b>				
	8000	Αντρέας	Νεοφύτου	18.0 *
	6565	Μαρίνα	Σοφοκλέους	16.6 *
	5677	Μαρία	Νικολάου	15.5 *
<b>Γ1</b>				

Για να προετοιμαστεί το Report

Group By  
Τμήμα

Τεχνολογικό

Στη στήλη ΜΟ έχει χρησιμοποιηθεί ο πιο κάτω τύπος:

=IF(RowNumber("ΤΜΗΜΑ") = 1 or RowNumber("ΤΜΗΜΑ") = 2 or RowNumber("ΤΜΗΜΑ") = 3, "\*", "-")

#### 4. ΛΙΣΤΑ ΚΑΛΥΤΕΡΩΝ ΑΝΑ ΤΑΞΗ (Α,Β,Γ)

Έχει χρησιμοποιηθεί ο ίδιος πίνακας TempStudentMo αλλά διαφορετική ομαδοποίηση.

ΤΑΞΗ	ΑΜ	ΟΝΟΜΑ	ΕΠΙΘΕΤΟ	ΤΜΗΜΑ	ΜΟ
<b>A</b>					
	5000	Γιώργος	Χριστοφή	A2	20.0
	5659	Μαρία	Νικοδήμου	A1	19.4
	1414	Χριστάκης	Παλαιολόγος	A2	18.7
<b>B</b>					
	8000	Αντρέας	Νεοφύτου	B1	18.0
	6565	Μαρίνα	Σοφοκλέους	B1	16.6
	5677	Μαρία	Νικολάου	B1	15.5

Στο σημείο Visibility καθορίζουμε το πιο κάτω τύπο

= RowNumber(«ΤΜΗΜΑ») = 1 and RowNumber(«ΤΜΗΜΑ») = 2 and RowNumber(«ΤΜΗΜΑ») = 3



## Κεφάλαιο 4<sup>ο</sup>

### Μελλοντικές Προεκτάσεις

Η εργασία μας έχει δημιουργηθεί για την διευκόλυνση των καθηγητών στα τμήματα της δευτεροβάθμιας εκπαίδευσης και των ιδιωτικών εκπαιδευτηρίων. Στο μέλλον θα μπορούσαμε να αναπτύξουμε σε ένα καλύτερο βαθμό την λειτουργία της σελίδας μας και της βάσης δεδομένων για την καταχώριση περισσότερων στοιχείων από τους καθηγητές. Επίσης μπορούμε να προεκτείνουμε τις δυνατότητες της εφαρμογής έτσι ώστε να μπορούν περισσότεροι χρήστες να την χρησιμοποιούν όπως οι γονείς, οι μαθητές και διάφοροι φορείς έτσι ώστε να υπάρχει μια πλήρης ενημέρωση σε όλους για τις επιδόσεις των μαθητών και γενικότερα του συνόλου του σχολείου.

Παρά τις αλλαγές που θα μπορούσαμε να πραγματοποιήσουμε η πρόσβαση για την καταχώριση και αλλαγή των στοιχείων ή βαθμών θα επιτρέπετε μόνο από τους εκπεδευτικούς.

# Βιβλιογραφία

- (1) <https://el.wikipedia.org/wiki/HTML>
- (2) <http://dide.flo.sch.gr/Plinet/Tutorials-PHP-Bible/Tutorials-PHP-Bible-2.html>
- (3) <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-Php-Analytical.html>
- (4) <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-ASP.html>
- (5) [https://en.wikipedia.org/wiki/Active\\_Server\\_Pages](https://en.wikipedia.org/wiki/Active_Server_Pages)
- (6) <https://el.wikipedia.org/wiki/SQL>
- (7) <http://www.w3schools.com/sql/default.asp>
- (8) <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-SQL.html>
- (9) <https://el.wikipedia.org/wiki/MySQL>
- (10) [https://www3.ntu.edu.sg/home/ehchua/programming/sql/MySQL\\_Beginner.html](https://www3.ntu.edu.sg/home/ehchua/programming/sql/MySQL_Beginner.html)
- (11) [http://www.dblab.upatras.gr/download/courses/DATABASES%20LABORATORY/2012\\_13/lect4.pdf](http://www.dblab.upatras.gr/download/courses/DATABASES%20LABORATORY/2012_13/lect4.pdf)
- (12) [http://www.w3schools.com/css/css\\_syntax.asp](http://www.w3schools.com/css/css_syntax.asp)
- (13) <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-CSS.html>
- (14) [https://en.wikipedia.org/wiki/Microsoft\\_Access](https://en.wikipedia.org/wiki/Microsoft_Access)
- (15) <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorial-Office-Access.pdf>
- (16) <http://www.pre.aegean.gr/documents/stufffiles/tsolak/Access01.pdf>
- (17) <http://blogs.msdn.com/b/visualstudio/>
- (18) <https://msdn.microsoft.com/en-us/vstudio/aa718325.aspx>
- (19) [https://en.wikipedia.org/wiki/Microsoft\\_Visual\\_Studio](https://en.wikipedia.org/wiki/Microsoft_Visual_Studio)
- (20) <http://www.eng.ucey.ac.cy/CEE500/webPageInstructions.html>
- (21) <http://www.icbnet.ntua.gr/website/Mathimata/IntroInternetTech/HTML%20Notes.pdf>
- (22) <http://www.wlearn.gr/index.php/html-html-33>

(23)[https://el.wikipedia.org/wiki/C\\_Sharp](https://el.wikipedia.org/wiki/C_Sharp)

(24)<http://www.cprogramming.com/tutorial/csharp.html>