

**ΣΗΜΕΙΩΣΕΙΣ ΓΙΑ ΤΗ VISUAL BASIC  
Γ' ΓΥΜΝΑΣΙΟΥ**

**Συγγραφική Ομάδα**

**Ευστάθιος Ευσταθίου  
Θεόδουλος Κωνσταντίνου  
Ξένιος Ξενοφώντος  
Χρίστος Μινίκκης**

**Εποπτεία: Ιάκωβος Παπαντωνίου**

# Εισαγωγή στον προγραμματισμό υπολογιστών με την χρήση της γλώσσας MICROSOFT VISUAL BASIC

---

Η **Visual Basic** χρησιμοποιείται για την δημιουργία εφαρμογών (προγραμμάτων) για Microsoft Windows. Περιλαμβάνει εργαλεία (tools) που βοηθούν τον προγραμματιστή να δημιουργήσει προγράμματα που να έχουν χαρακτηριστικά όμοια με αυτά των Windows, χωρίς να πρέπει να γράφει πολλές γραμμές κωδικοποίησης (codes). Δημιουργήθηκε από την εταιρεία Microsoft το 1992.

## Η Visual Basic είναι γλώσσα οπτικού προγραμματισμού

Για να καταλάβετε καλύτερα τι είναι γλώσσα οπτικού προγραμματισμού, θα παρομοιάσω το περιβάλλον της Visual Basic με ένα θέατρο, όπου εσείς, ο προγραμματιστής δηλαδή, είστε ο σκηνοθέτης που θα ανεβάσει ένα έργο.

Θα πρέπει στην πρώτη **σκηνή** (στην VB λέγεται **φόρμα**) να τοποθετήσετε τους ηθοποιούς που θέλετε να παίξουν (δηλαδή να εκτελέσουν κάποιες εντολές που θα τους πείτε εσείς ως σκηνοθέτης).

Αυτοί οι ηθοποιοί, που λέγονται **αντικείμενα** στην Visual Basic, μπορεί να είναι ένα κουμπί (button), μια εικόνα, ένα χρονόμετρο κλπ. Κάθε ηθοποιός (αντικείμενο) ξέρει τι πρέπει να κάνει γιατί του έχει δώσει εντολές ο σκηνοθέτης. Το ίδιο και στην Visual Basic, σε κάθε αντικείμενο ο προγραμματιστής, εσείς δηλαδή, θα δώσετε εντολές να εκτελεί όποτε το επιλέξετε.

Για παράδειγμα έχετε ένα κουμπί πάνω στην φόρμα σας. Μπορείτε να του δώσετε εντολές μόλις το επιλέγετε με το ποντίκι να κάνει την φωτογραφία «μπάλα» να κυλά.

Κάθε αντικείμενο για να εκτελέσει τις εντολές που θα του δώσετε θα πρέπει να προηγηθεί μια ενέργεια. Για παράδειγμα το αντικείμενο κουμπί για να κάνει την μπάλα να κυλήσει πρέπει να πατήσουμε πάνω του με το ποντίκι.

Σε μια γλώσσα οπτικού προγραμματισμού ο προγραμματιστής ξεκινά την ανάπτυξη της εφαρμογής του δημιουργώντας πρώτα απ' όλα τις φόρμες της. Αυτό σημαίνει ότι τοποθετεί πάνω σε μια κενή φόρμα, με τη βοήθεια του πληκτρολογίου και του ποντικιού ή άλλης παρόμοιας συσκευής, όλα τα αντικείμενα που χρειάζεται. Η φόρμα και ότι βρίσκεται πάνω σ' αυτήν έχουν συγκεκριμένες ιδιότητες με προκαθορισμένες τιμές. Ο προγραμματιστής καλείται να αλλάξει όσες από τις τιμές των ιδιοτήτων κρίνει σκόπιμο.

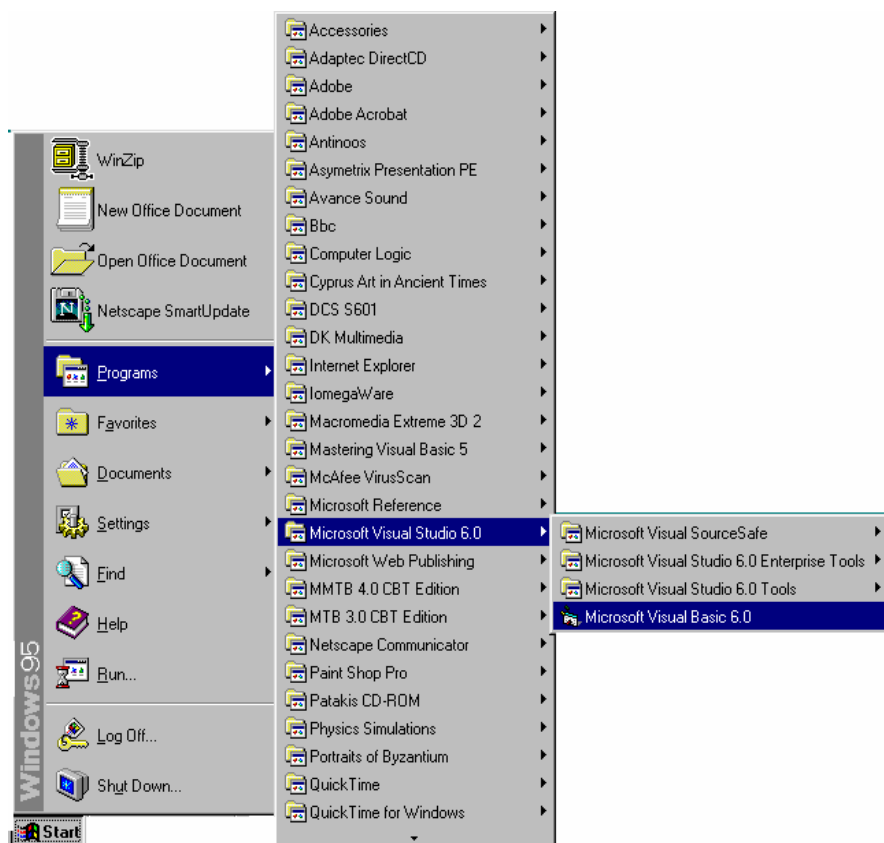
## Κεφάλαιο 1

### Στόχοι

- Ενεργοποίηση της Visual Basic
- Γνωριμία με το περιβάλλον
- Έξοδος από το πρόγραμμα

### Ενεργοποίηση του προγράμματος

Για την ενεργοποίηση του προγράμματος ακολουθήστε τα πιο κάτω βήματα:



---

ΕΙΚΟΝΑ 1: Ενεργοποίηση της Visual Basic

---

Όταν εκκινείτε την Visual Basic για πρώτη φορά, ανοίγει ο οδηγός Project Wizard και θα δείτε το παράθυρο διαλόγου νέου έργου New Project (Βλ. Εικόνα. 2).

Το παράθυρο αυτό θα εμφανίζεται κάθε φορά που θα εκκινείτε την Visual Basic.

Εάν θέλετε πατήστε στην επιλογή **Don't show this dialog in the future**, στο κάτω μέρος του παραθύρου για να μην εμφανιστεί ξανά το παράθυρο διαλόγου. Στη συνέχεια επιλέξετε **Open**.



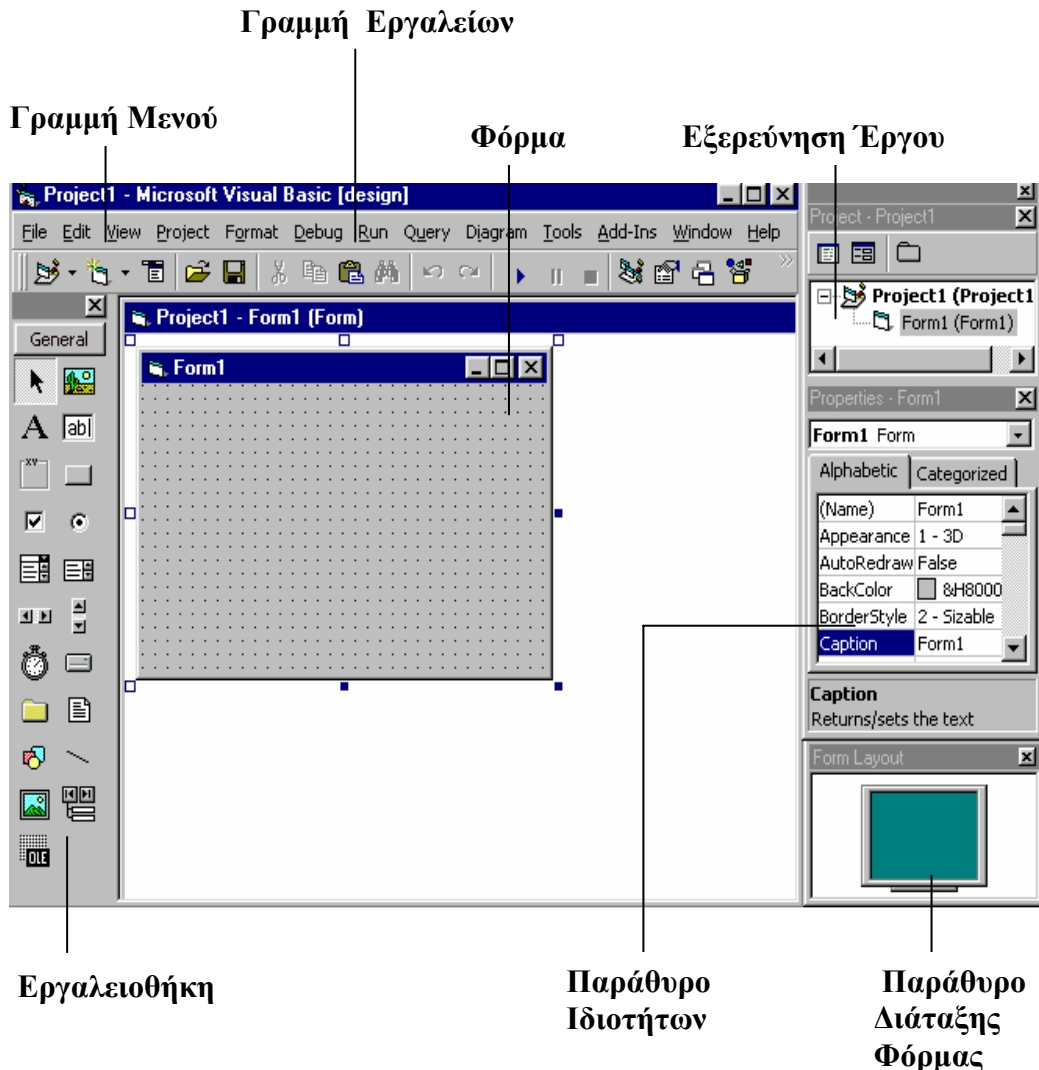
---

ΕΙΚΟΝΑ 2: Το παράθυρο διαλόγου New Project του Project Wizard

---

Τώρα ας δούμε πιο αναλυτικά το **Ολοκληρωμένο Περιβάλλον Ανάπτυξης (IDE)** της Visual Basic.

Το Ολοκληρωμένο Περιβάλλον Ανάπτυξης (IDE) (Βλ. Εικόνα. 3). είναι ένα σημαντικό μέρος της Visual Basic. Είναι το μέρος όπου συνθέτετε τις εφαρμογές σας και όπου θα περάσετε τον περισσότερο χρόνο σας όταν δημιουργείτε εφαρμογές (προγράμματα).



ΕΙΚΟΝΑ 3: Το Ολοκληρωμένο Περιβάλλον Ανάπτυξης (IDE) της Visual Basic

Το περιβάλλον της Visual Basic αποτελείται από διάφορα βοηθητικά εικονίδια και εργαλεία:

- **Γραμμή μενού** (Menu bar) περιλαμβάνει όλα τα μενού που περιέχουν εντολές.
- **Γραμμή εργαλείων** (Toolbar) περιλαμβάνει συντομεύσεις για τις συχνά επαναλαμβανόμενες ενέργειες του χρήστη.
- **Εξερεύνηση έργου** (Project explorer window) περιέχει την λίστα με όλα τα αρχεία του συγκεκριμένου έργου (project.)
- **Παράθυρο ιδιοτήτων** (Properties Window) περιέχει τις ιδιότητες του κάθε αντικειμένου.
- **Παράθυρο διάταξης φόρμας** (Form layout window) δείχνει πώς θα φαίνεται και πώς θα είναι τοποθετημένη η φόρμα κατά τον χρόνο της εκτέλεσης του προγράμματος.
- **Εργαλειοθήκη** (Toolbox) περιέχει εργαλεία που χρησιμοποιούνται για την δημιουργία αντικειμένων.

- **Φόρμα (Form)** είναι ο χώρος στον οποίο τοποθετούνται όλα τα αντικείμενα. Η φόρμα από μόνη της είναι και αυτή ένα αντικείμενο με ιδιότητες. Κάθε έργο (project) έχει τουλάχιστο μία φόρμα.

## Η Γραμμή Μενού

Η γραμμή μενού είναι η γραμμή κειμένου που βρίσκεται στη κορυφή του παραθύρου της Visual Basic. Είναι παρόμοια με τα μενού που έχετε δει σε άλλες εφαρμογές των Windows. Το μενού σας παρέχει πρόσβαση σε πολλά χαρακτηριστικά μέσα στο περιβάλλον ανάπτυξης.

Στα αριστερά βρίσκεται το μενού αρχείου (File). Από το μενού File μπορείτε να δημιουργήσετε, να ανοίξετε, να εκτυπώσετε και να αποθηκεύσετε έργα.

Δίπλα από το μενού File βρίσκεται το μενού επεξεργασίας (Edit). Από εδώ μπορείτε να κάνετε τις συνηθισμένες εργασίες του Clipboard, όπως αποκοπή, αντιγραφή, επικόλληση.

Από το μενού προβολής (View) μπορείτε να προβάλετε διάφορα συστατικά και εργαλεία, μια φόρμα, όπως και άλλα βοηθήματα που σας βοηθούν να αναπτύξετε τον χρόνο ανάπτυξης παραγωγικότερα.

Το μενού έργου (Project) είναι η καρδιά του έργου σας. Από εδώ μπορείτε να προσθέσετε και να αφαιρέσετε φόρμες, μηχανισμούς χρήστη, σελίδες ιδιοτήτων. Πολλές από αυτές τις εφαρμογές μπορούν να προσπελαστούν με δεξιό κλικ στην εργαλειοθήκη.

Οι επιλογές του μενού μορφής (Format) αναφέρονται ειδικά στο μέγεθος και στη τοποθέτηση μηχανισμού και φορμών.

Όταν αποσφαλματώνετε τις εφαρμογές σας θα εξοικειωθείτε ιδιαίτερα με το μενού αποσφαλμάτωσης (Debug). Από εδώ μπορείτε να εκκινείτε και να τερματίζετε τις εφαρμογές σας, όπως και να διακόπτετε στο μέσο της εκτέλεσης ενός προγράμματος, και μετά να συνεχίζετε.

Από το μενού εκτέλεσης (Run) μπορείτε να εκκινείτε και να τερματίζετε τις εφαρμογές σας, όπως και να διακόπτετε στο μέσο της εκτέλεσης ενός προγράμματος και μετά να συνεχίζετε.

Από το μενού εργαλείων (Tools) μπορείτε να προσθέσετε διαδικασίες και να θέσετε ιδιότητες διαδικασίας. Μπορείτε να επιλέξετε Options και να θέσετε προτιμήσεις για το περιβάλλον εργασίας σας.

Το μενού παραθύρων (Window) σας παρέχει επιλογές για τοποθέτηση παραθύρων σε παράθεση ή σε επικάλυψη μέσα στο περιβάλλον εργασίας. Μπορείτε επίσης να διατάξετε τα εικονίδια για ελαχιστοποιημένες φόρμες.

Το μενού βοήθειας (Help), όταν έχετε πρόβλημα, θα είναι το πρώτο, που θα συμβουλευτείτε.

## Η Γραμμή Εργαλείων

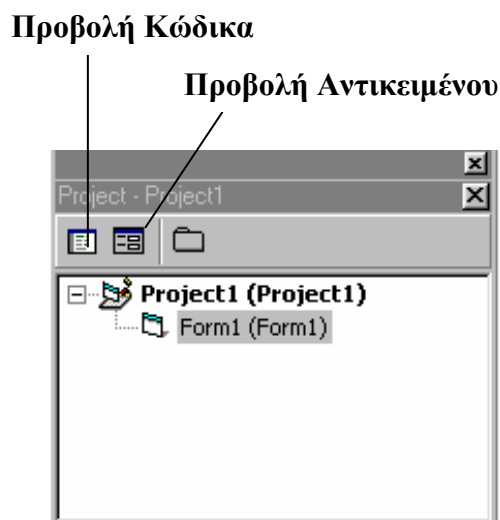
Ακριβώς κάτω από την γραμμή μενού βρίσκεται η γραμμή εργαλείων της Visual Basic. (Βλ. Εικόνα. 4). Αν δεν μπορείτε να δείτε την γραμμή εργαλείων, κάντε κλικ στο View / Toolbars / Standard. Μπορείτε να ελέγξετε όλο το περιβάλλον της Visual Basic από την γραμμή μενού, αλλά η γραμμή εργαλείων σας παρέχει εύκολη πρόσβαση στις εντολές της γραμμής μενού, που θα χρησιμοποιείτε συχνότερα. Θα παρατηρήσετε ότι, αν κρατήσετε τον δείκτη του ποντικιού επάνω σε ένα κουμπί για λίγο, θα δείτε την επεξήγηση εργαλείου γι' αυτό το κουμπί.



EIKONA 4: Η γραμμή εργαλείων της Visual Basic

## Εξερεύνηση Έργου (Project Explorer)

Στη δεξιά πλευρά της οθόνης σας κάτω από την γραμμή εργαλείων βρίσκεται το παράθυρο εξερεύνησης έργου (Project Explorer) (Βλ. Εικόνα 5)

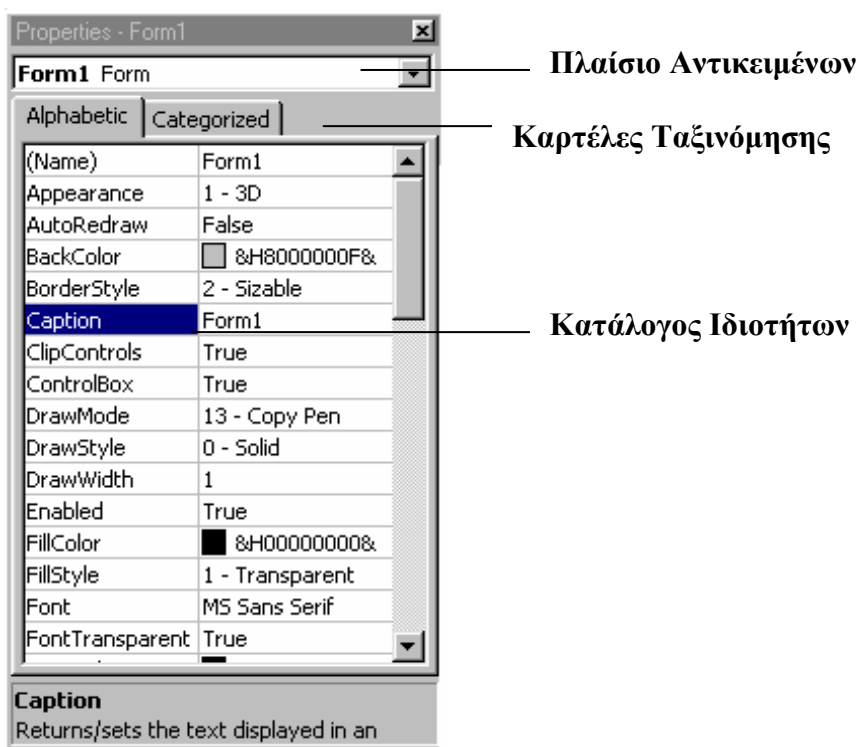


ΕΙΚΟΝΑ 5: Το παράθυρο Project Explorer

Ο Project Explorer είναι η γρήγορη αναφορά στα διάφορα στοιχεία, φόρμες και λειτουργικές μονάδες του έργου σας.

## Το Παράθυρο Ιδιοτήτων

Κάτω ακριβώς από το παράθυρο του Project Explorer βρίσκεται το παράθυρο ιδιοτήτων (properties) (Βλ. Εικόνα 6)



ΕΙΚΟΝΑ 6: Το παράθυρο ιδιοτήτων (properties)

---

Το παράθυρο ιδιοτήτων εμφανίζει τα διάφορα χαρακτηριστικά ή ιδιότητες επιλεγμένων αντικειμένων. Για να διασαφηνίσουμε αυτό το θέμα, σκεφθείτε ότι κάθε φόρμα σε μια εφαρμογή είναι ένα αντικείμενο. Κάθε μηχανισμός (π.χ. ένα κουμπί εντολής), που εμφανίζεται σε μια φόρμα, είναι επίσης ένα αντικείμενο. Τώρα κάθε αντικείμενο στη Visual Basic έχει χαρακτηριστικά όπως χρώμα και μέγεθος. Όλα τα χαρακτηριστικά ενός αντικειμένου καλούνται ιδιότητες. Έτσι μια φόρμα έχει ιδιότητες και κάθε μηχανισμός, που είναι τοποθετημένος σε μια φόρμα, έχει επίσης ιδιότητες. Όταν ένας μηχανισμός, π.χ. ένα κουμπί εντολής τοποθετείται σε μια φόρμα, το παράθυρο ιδιοτήτων δείχνει τις ιδιότητες γι' αυτό τον μηχανισμό, όταν είναι επιλεγμένος.

Στο παράθυρο properties θα δείτε μια λίστα ιδιοτήτων που ανήκουν σε ένα αντικείμενο. Υπάρχουν αρκετές ιδιότητες και θα χρειαστεί να κυλήσετε για να τις δείτε όλες. Επίσης μπορείτε να τις προβάλετε είτε αλφαβητικά είτε βάση της κατηγορίας τους

### Η Εργαλειοθήκη (toolbox)

Η εργαλειοθήκη (Βλ. Εικόνα 7) περιέχει όλα τα εργαλεία που χρειάζεστε, για να δημιουργήσετε την διασύνδεση της φόρμας σας. Όλα τα εργαλεία αντιστοιχούν στα αντικείμενα, που θα θέλατε να τοποθετήσετε σε μια φόρμα. Τα εργαλεία αυτά αναφέρονται σαν μηχανισμοί (controls)



---

ΕΙΚΟΝΑ 7: Η Εργαλειοθήκη (toolbox)

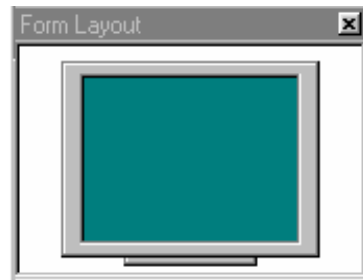
---

### Το παράθυρο διάταξης φόρμας (Form layout)

Το παράθυρο διάταξης φόρμας (Βλ. Εικόνα 8) δείχνει πώς θα φαίνεται και πώς θα είναι τοποθετημένη η φόρμα σας κατά τον χρόνο της εκτέλεσης. Για να χρησιμοποιήσετε το Form layout κάντε τα εξής:



- Κάντε κλικ στο Form layout και μεταφέρετε το στο κέντρο του γραφικού της οθόνης
- Εκτελέστε το πρόγραμμα σας επιλέγοντας Run / Start.




---

ΕΙΚΟΝΑ 8: Το παράθυρο Form layout

---

### Εμφάνιση ή απόκρυψη της γραμμής εργαλείων

Την γραμμή εργαλείων, η οποία βρίσκεται στο πάνω μέρος της οθόνης, κάτω ακριβώς από το μενού, έχετε τη δυνατότητα να την εμφανίζετε ή να την αποκρύψετε. Για να την αποκρύψετε, ακολουθήστε τα πιο κάτω βήματα:

- Από το μενού επιλέξετε View
- Από το View επιλέξετε Toolbars
- Από το Toolbars επιλέξετε Standard

Παρατηρείτε ότι η γραμμή εργαλείων δεν εμφανίζεται. Για να εμφανιστεί, ακολουθήστε τα ίδια βήματα, που κάνατε, για να την αποκρύψετε.

### Εμφάνιση ή απόκρυψη της εργαλειοθήκης

Την εργαλειοθήκη, η οποία βρίσκεται στο αριστερό μέρος της οθόνης, έχετε την δυνατότητα να την εμφανίζετε ή να την αποκρύψετε. Για να την αποκρύψετε επιλέξετε το σύμβολο κλεισίματος **X**, το οποίο βρίσκεται στο πάνω δεξιό άκρο της εργαλειοθήκης.

Για να την εμφανίσετε, ακολουθήστε τα πιο κάτω βήματα:

- Από το μενού επιλέξετε View
- Από το View επιλέξετε Toolbox

### Εμφάνιση ή απόκρυψη του εξερευνητή έργου

Τον εξερευνητή έργου, ο οποίος βρίσκεται στο πάνω δεξιό μέρος της οθόνης, έχετε την δυνατότητα να τον εμφανίζετε ή να τον αποκρύψετε. Για να τον αποκρύψετε, επιλέξετε το σύμβολο κλεισίματος **X**, το οποίο βρίσκεται στο πάνω δεξιό άκρο του εξερευνητή έργου.

Για να τον εμφανίσετε, ακολουθήστε τα πιο κάτω βήματα:

- Από το μενού επιλέξετε View
- Από το View επιλέξετε Project explorer

## Εμφάνιση ή απόκρυψη του παραθύρου ιδιοτήτων

Το παράθυρο ιδιοτήτων το οποίο βρίσκεται κάτω από τον εξερευνητή έργου, έχετε την δυνατότητα να τον εμφανίζετε ή να τον αποκρύψετε. Για να τον αποκρύψετε, επιλέξετε το σύμβολο κλεισίματος **X**, το οποίο βρίσκεται στο πάνω δεξιό άκρο του παραθύρου ιδιοτήτων.

Για να τον εμφανίσετε, ακολουθήστε τα πιο κάτω βήματα:

- Από το μενού επιλέξετε View
- Από το View επιλέξετε Properties Windows.

## Εμφάνιση ή απόκρυψη του παραθύρου διάταξης φόρμας

Το παράθυρο διάταξης φόρμας, το οποίο βρίσκεται κάτω από το παράθυρο ιδιοτήτων, έχετε την δυνατότητα να το εμφανίζετε ή να το αποκρύψετε. Για να το αποκρύψετε, επιλέξετε το σύμβολο κλεισίματος **X**, το οποίο βρίσκεται στο πάνω δεξιό άκρο του παραθύρου διάταξης φόρμας.

Για να το εμφανίσετε, ακολουθήστε τα πιο κάτω βήματα:

- Από το μενού επιλέξετε View
- Από το View επιλέξετε Form Layout Window

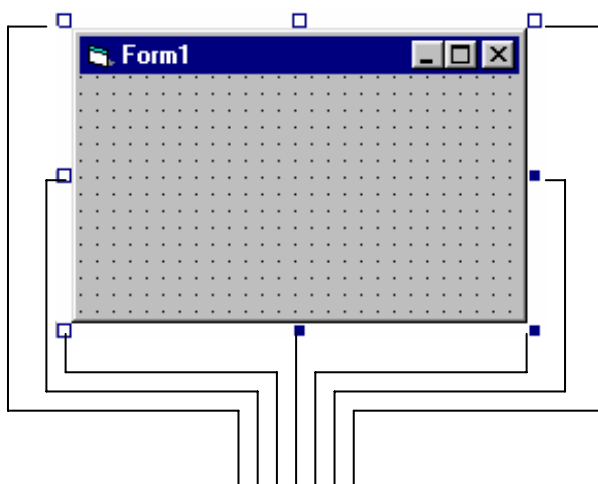
## Αλλαγή διαστάσεων του εξερευνητή έργου, του παραθύρου ιδιοτήτων και του παραθύρου διάταξης φόρμας.

Παρατηρήστε ότι, όταν τοποθετήσετε τον δείκτη του ποντικιού σε μια από τις πλευρές των πιο πάνω παραθύρων, ο δείκτης του ποντικιού αλλάζει. Πατήστε το αριστερό κουμπί του ποντικιού και σύρετε το προς την κατεύθυνση που θέλετε.

## Αλλαγή διαστάσεων της φόρμας

Παρατηρήστε ότι στις γωνίες και στο κέντρο των πλευρών της φόρμας εμφανίζονται μικρά ορθογώνια πλαίσια, που ονομάζονται λαβές αλλαγής μεγέθους. Τοποθετήστε το δείκτη του ποντικιού σε μια λαβή και σύρετε το μέχρι που φόρμα να αποκτήσει το μέγεθος που θέλετε.

Οι γωνιακές λαβές αλλάζουν το μέγεθος της φόρμας οριζόντια και κατακόρυφα, ενώ οι λαβές, που βρίσκονται στο κέντρο των πλευρών, αλλάζουν το μέγεθος της φόρμας προς μια κατεύθυνση.



Λαβές αλλαγής μεγέθους

## Αποθήκευση του προγράμματος

Για να αποθηκεύσουμε ένα έργο (project) στην VB, επιλέγουμε File, Save us . Στην συνέχεια πρέπει να δώσουμε ένα όνομα στην φόρμα και ένα όνομα στο έργο (Project).

## Άσκηση 1

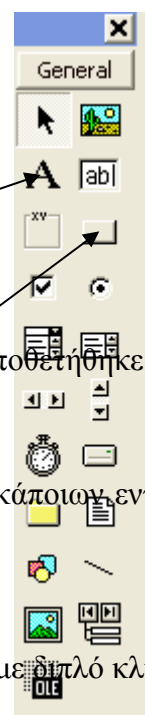
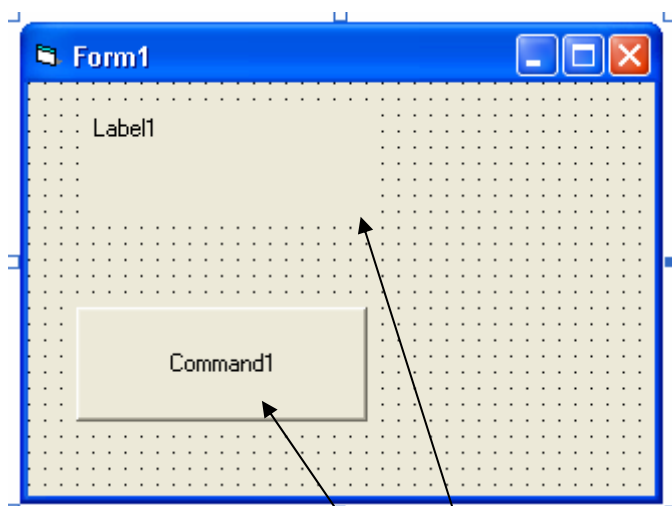
1. Ενεργοποιήστε την Visual Basic.
2. Βρέστε όλα τα μέρη του **Ολοκληρωμένου Περιβάλλοντος Ανάπτυξης (IDE)** όπως έχουμε αναφέρει πιο πάνω, π.χ. form, menu bar, tool box κλπ.
3. Αποθηκεύσετε τη φόρμα με το όνομα **frmfirst** και το έργο σας με το όνομα **first**
4. Κλείστε την εφαρμογή VB.
5. Ενεργοποιήστε την VB πάλι και ανοίξτε το έργο (project) σας.
6. Κλείστε και πάλι την VB.


## Κεφάλαιο 2

Πρόσθεση αντικειμένων (Objects) σε μια φόρμα.


Κάθε έργο περιέχει τουλάχιστο μια φόρμα η οποία περιλαμβάνει τα αντικείμενα του έργου. Π.χ. η πιο κάτω φόρμα form1 περιέχει δύο αντικείμενα

- Την ετικέτα (label)
- Το κουμπί εντολής (command button) Command1



Το αντικείμενο ετικέτα (label) χρησιμοποιείται για να εμφανίζει πληροφορίες. Τοποθετήθηκε στην φόρμα αφού τραβήξαμε το εικονίδιο  από την εργαλειοθήκη.


Το αντικείμενο κουμπί εντολών (command button) χρησιμοποιείται για την εκτέλεση κάποιων εντολών αφού πρώτα κάνουμε πάνω του κλικ με το ποντίκι.

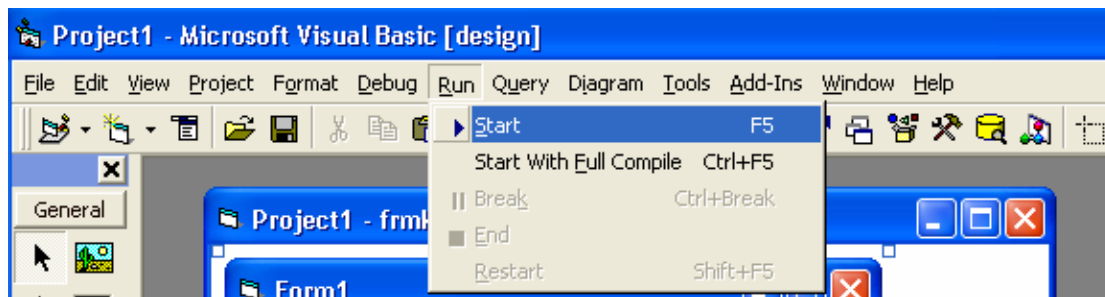
Τοποθετήθηκε στην φόρμα αφού τραβήξαμε το εικονίδιο  από την εργαλειοθήκη.

Ένας δεύτερος τρόπος τοποθέτησης των αντικειμένων πάνω στην φόρμα είναι να κάνουμε διπλό κλικ στο εικονίδιο που θέλουμε από την εργαλειοθήκη.

Η τοποθέτηση των αντικειμένων στην φόρμα γίνεται κατά την διάρκεια του σχεδιασμού της φόρμας (design time).

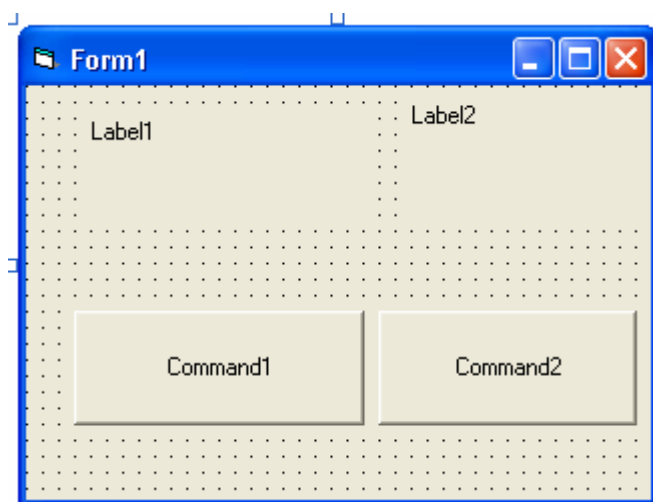
### Εκτέλεση (run) ενός προγράμματος στην VB.

Για την εκτέλεση του προγράμματος μας κάνουμε κλικ πάνω στο μενού Run και επιλέγουμε το Start (F5) ή κάνουμε κλικ στο εικονίδιο .



## Άσκηση 2

1. Ενεργοποιήστε την Visual Basic.
2. Πάνω στην φόρμα του νέου έργου τοποθετήστε τέσσερα αντικείμενα, δύο ετικέτες (label) και δύο κουμπιά εντολών (command buttons) όπως πιο κάτω.

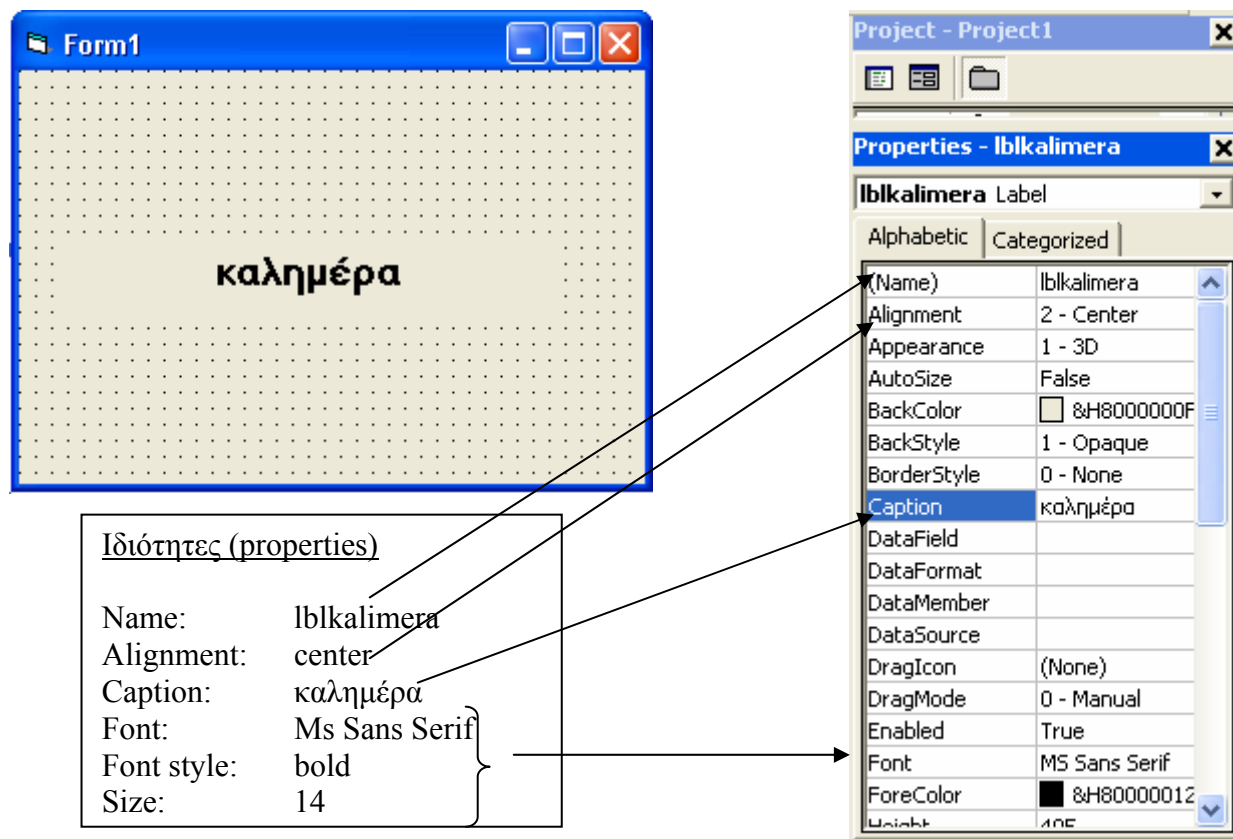


3. Μετακινήστε τα αντικείμενα μέσα στην φόρμα μέχρι να πετύχετε την κατάλληλη θέση που σας αρέσει για το κάθε αντικείμενο.
4. Αλλάξτε το μέγεθος του κάθε αντικειμένου όπως νομίζετε εσείς καλύτερα.
5. Αποθηκεύστε τη φόρμα με το όνομα **frmaskisi2** και το έργο σας με το όνομα **askisi2**
6. Κλείστε την εφαρμογή VB.

### Κεφάλαιο 3

#### Τα αντικείμενα (objects) και οι ιδιότητες τους (properties)

Κάθε αντικείμενο έχει ιδιότητες οι οποίες καθορίζουν την εμφάνιση, την συμπεριφορά, την θέση κ.λ.π. του αντικειμένου, π.χ **μερικές** ιδιότητες της πιο κάτω ετικέτας (label) που περιέχει την λέξη ‘καλημέρα’ είναι :



Όλες οι ιδιότητες μπορούν να αλλάξουν από το παράθυρο ιδιοτήτων (properties window) αφού πρώτα επιλέξουμε το αντικείμενο και δώσουμε την κατάλληλη τιμή στην ιδιότητα που θα θέλουμε.

Μέχρι τώρα έχουμε παρουσιάσει μερικά σημαντικά αντικείμενα της VB. Ας δούμε μερικές από τις πιο σημαντικές τους ιδιότητες.

#### Αντικείμενο label

Ιδιότητες:

- **Name** καθορίζει το όνομα του label. Καλή τεχνική προγραμματισμού είναι να αρχίζουμε το όνομα κάθε label με lbl. Πχ lblkalimera
- **Caption** καθορίζει το τι θα εμφανίζει το label. Στο προηγούμενο παράδειγμα το Caption είχε τιμή «καλημέρα».
- **Font** καθορίζει την γραμματοσειρά (font), το font style και το μέγεθος των γραμμάτων που εμφανίζει το Caption.
- **Alignment** καθορίζει την θέση του κειμένου μέσα στο label. (left, right, center)

## Αντικείμενο Command Button

Ιδιότητες:

- Name καθορίζει το όνομα του Command Button. Καλή τεχνική προγραμματισμού είναι να αρχίζουμε το όνομα κάθε command button με cmd. Πχ cmdprosthesis
- Caption αλλάζει το κείμενο που βρίσκεται μέσα στο command button.

## Αντικείμενο Form

Ιδιότητες:

- Name καθορίζει το όνομα της φόρμας. Καλή τεχνική προγραμματισμού είναι να αρχίζουμε το όνομα κάθε φόρμας (form) με frm. Πχ frmkalimera
- Caption αλλάζει το κείμενο που βρίσκεται στο **title bar** της φόρμας.

Ένας καλός προγραμματιστής δίνει πάντοτε ονόματα σε όλα τα αντικείμενα του έργου (project) του. Τα ονόματα πρέπει πάντοτε να αρχίζουν με συντομογραφία π.χ. frm, lbl, cmd κλπ για να καθορίζουν το είδος του αντικειμένου και το υπόλοιπο όνομα πρέπει να είναι επεξηγεί τον ρόλο του αντικειμένου.

Π.χ cmdprosthesis Βλέποντας αυτό το όνομα αμέσως καταλαβαίνουμε ότι είναι command button από το cmd και ότι έχει εντολές να κάνει πρόσθεση.

### Άσκηση3

Ανοίξτε το project askisi2. Στην φόρμα frmaskisi2 αλλάξτε τις τιμές των πιο κάτω ιδιοτήτων του κάθε αντικειμένου όπως πιο κάτω:

Δεδομένα για τα αντικείμενα και τις ιδιότητες τους			
A\A	Αντικείμενο	Ιδιότητα	Τιμή
1	Form1	name	frmaskisi2
		caption	Askisi2
2	Label1	name	lblgymnasio
		Caption	Γυμνάσιο
		Font	Ms sans serif, bold, 12
		Alignment	2-center
3	Label 2	Name	lblkitiou
		Caption	Κιτίου
		Font	Ms sans serif, bold, 12
4	Button1	Alignment	0-left justify
		Name	cmdprothesi
		Caption	PROSTHESI
5	Button2	Name	cmdaferesi
		Caption	AFERESI

Εκτελέστε το πρόγραμμα σας και παρατηρήστε πως έχουν αλλάξει οι ιδιότητες της φόρμας σας.



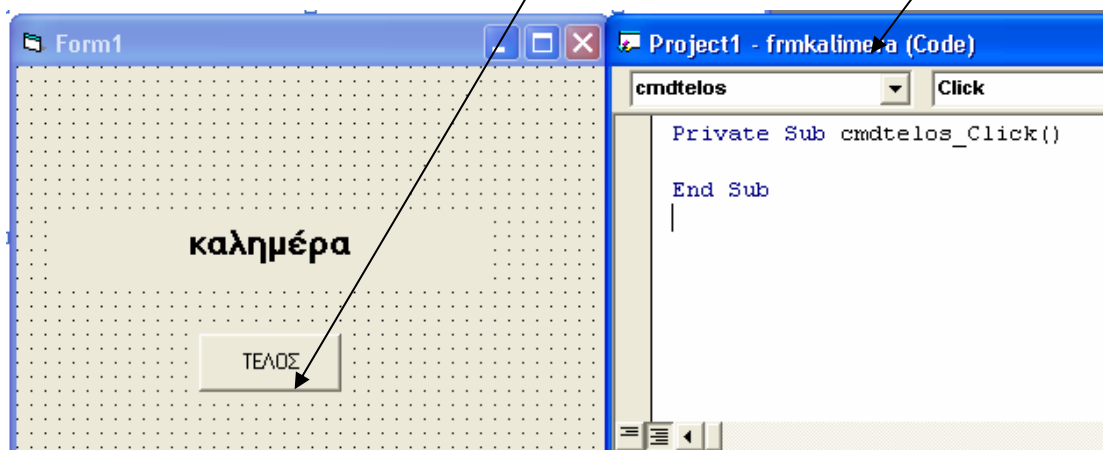
## Κεφάλαιο 4

### Κωδικοποίηση

Ένα πρόγραμμα αποτελείται από μια σειρά εντολών που δίνουμε στον υπολογιστή να μας κάνει κάτι.

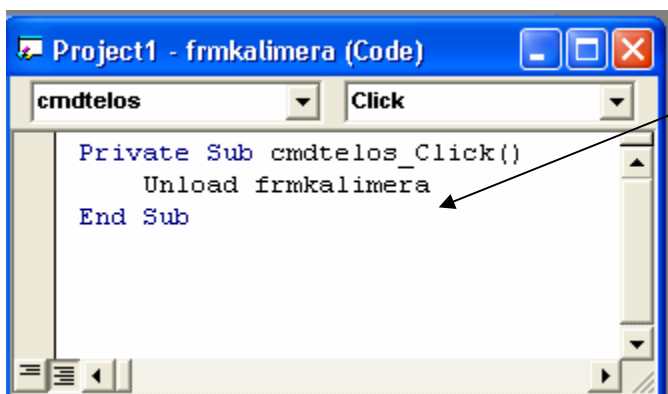
Στην Visual Basic δίνουμε εντολές στα διάφορα αντικείμενα της φόρμας μας κάνοντας διπλό κλικ πάνω τους.

Για να δούμε το παράθυρο κωδικοποίησης του κουμπιού εντολής (command button) ΤΕΛΟΣ κάνουμε διπλό κλικ πάνω στο κουμπί. Τότε εμφανίζεται το παράθυρο κωδικοποίησης.



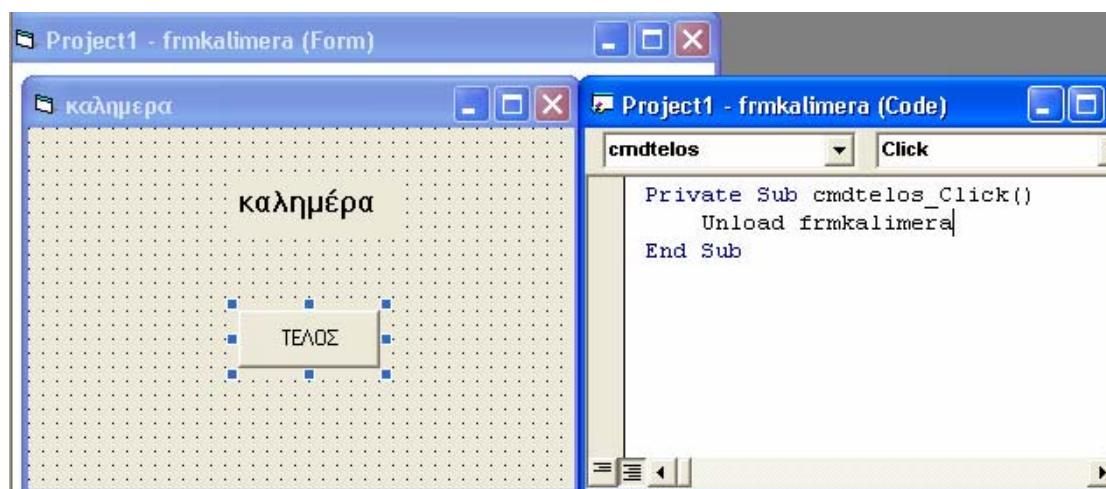
Το **private sub** καθορίζει την αρχή των εντολών που θα γράψουμε και το **end sub** καθορίζει το τέλος των εντολών. Η σειρά εντολών κάθε αντικειμένου λέγεται υπορουτίνα του όλου προγράμματος. Ενδιάμεσα αυτών των δύο γράφουμε τις εντολές που θέλουμε να εκτελούνται όταν κάνουμε κλικ στο κουμπί (button).

Για παράδειγμα εάν θέλουμε να σταματάμε την εκτέλεση του προγράμματος της φόρμας frmkalimera με το πάτημα στο κουμπί **τέλος**, θα γράψουμε την εντολή unload frmkalimera



#### Άσκηση 4

Δημιουργήστε ένα νέο έργο όπως φαίνεται πιο κάτω και εκτελέστε τις εντολές του κουμπιού ΤΕΛΟΣ.  
Μην ξεχάσετε να αλλάξετε τις κατάλληλες ιδιότητες για το κάθε αντικείμενο όπως Name, Caption κλπ.



## Κεφάλαιο 5-Είσοδος και έξοδος

Ένας υπολογιστής δεν θα είχε μεγάλη χρήση, εάν δεν υπήρχε τρόπος να του δίνουμε δεδομένα και να παίρνουμε πληροφορίες (αποτελέσματα).

Η διαδικασία της μετακίνησης δεδομένων από και προς έναν υπολογιστή ονομάζεται *είσοδος* και *έξοδος*. Μια οποιαδήποτε γλώσσα προγραμματισμού περιέχει εντολές για είσοδο και έξοδο δεδομένων. Η Visual Basic διαθέτει ένα σύνολο εντολών για να εισάγουμε πληροφορίες στον υπολογιστή και να εξάγουμε αποτελέσματα, επιτρέποντας μας να δημιουργούμε διαλογικά προγράμματα. Τα διαλογικά προγράμματα επιτρέπουν την αμφίδρομη επικοινωνία μεταξύ του χρήστη και του υπολογιστή. Για παράδειγμα, η Visual Basic μπορεί να κάνει μια ερώτηση στο χρήστη εμφανίζοντας την ερώτηση στη οθόνη του υπολογιστή. Ο χρήστης μπορεί να δώσει την απάντηση πληκτρολογώντας.

### Είσοδος δεδομένων

Η Visual Basic μας παρέχει τη δυνατότητα να εισάγουμε δεδομένα στο πρόγραμμα μέσω αντικειμένων. Οι βασικοί τύποι αντικειμένων που μπορούν να λάβουν δεδομένα από τη διασύνδεση χρήστη είναι :

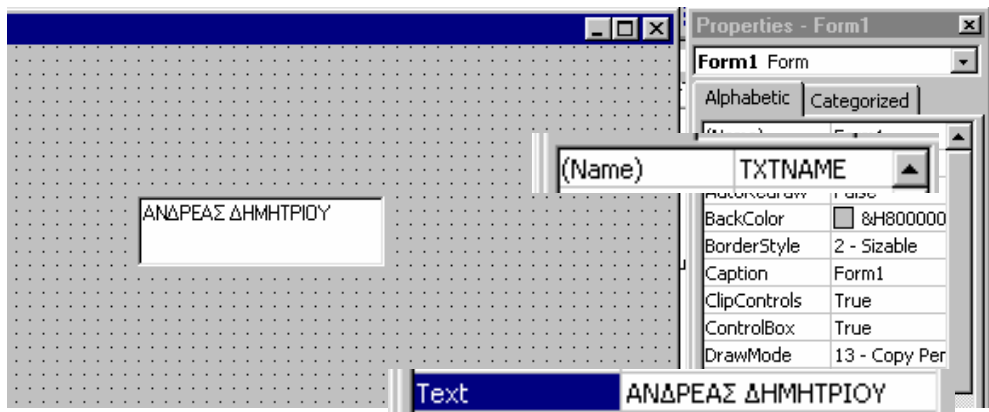
#### Το χειριστήριο TextBox



Ένα αντικείμενο τύπου **TextBox** (πλαίσιο κειμένου) χρησιμοποιείται για να εμφανίζουμε κείμενο πάνω σε μία φόρμα ή για να παίρνουμε δεδομένα από το χρήστη, με τη μορφή κειμένου, καθώς εκτελείται το πρόγραμμα. Η συμπεριφορά του εξαρτάται από τις ιδιότητες που του θέτουμε και από τον τρόπο που το χρησιμοποιούμε στο πρόγραμμα.

#### Χρήσιμες Ιδιότητες

Ιδιότητα	Περιγραφή
(name)	Περιέχει το όνομα του αντικειμένου. Η Visual Basic δίνει αυτόματα ένα όνομα που αποτελείται από τη λέξη Text ακολουθούμενο από έναν αριθμό. Π.χ. Text1. Αν θέλουμε να ονομάσουμε ένα πλαίσιο κειμένου συνήθως χρησιμοποιούμε το πρόθεμα txt. Π.χ. txtName, txtPassword κλπ.
Text	Η ιδιότητα αυτή περιέχει το κείμενο που εμφανίζεται στο πλαίσιο κειμένου.



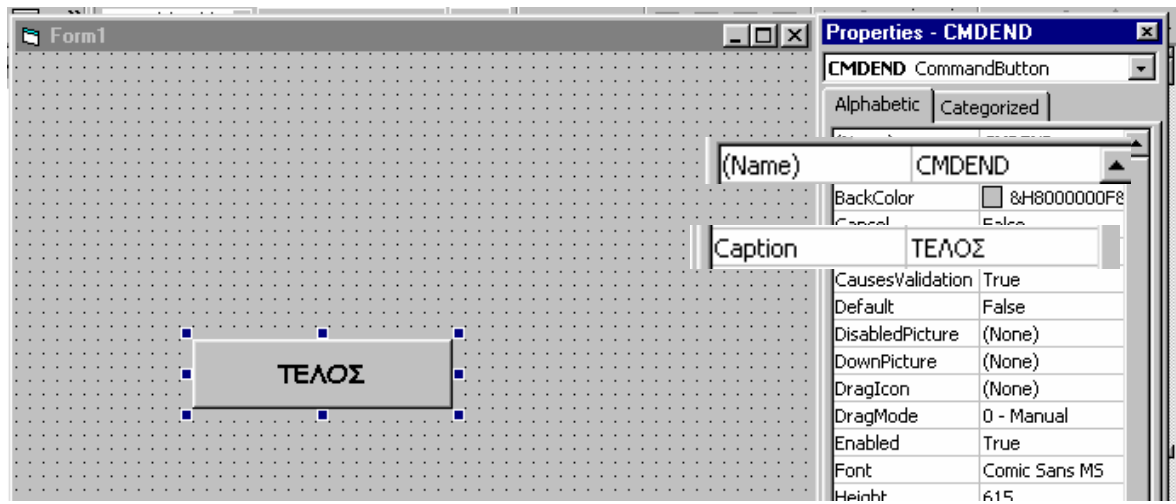
### Το χειριστήριο CommandButton



Ο ευκολότερος τρόπος για να επικοινωνήσουμε με τον χρήστη είναι να χρησιμοποιήσουμε κάποιο κουμπί εντολής. Το κουμπί εντολής χρησιμοποιείται για την εκκίνηση, διακοπή ή λήξη μιας διαδικασίας. Πιο "επίσημα", ο χρήστης χρησιμοποιεί το κουμπί εντολής για να δημιουργήσει ένα γεγονός (event) που πρέπει να χειριστεί η εφαρμογή. Τυπικά παραδείγματα κουμπιών εντολής είναι το κουμπί OK, το κουμπί Cancel και το κουμπί Quit.

### Χρήσιμες Ιδιότητες

Ιδιότητα	Περιγραφή
<b>(name)</b>	Περιέχει το όνομα του κουμπιού, που πρέπει να είναι μοναδικό μέσα στη φόρμα. Η Visual Basic δίνει ένα όνομα που αποτελείται από τη λέξη Command ακολουθούμενο από έναν αριθμό. Π.χ. Command1. Αν θέλουμε να ονομάσουμε ένα command button συνήθως χρησιμοποιούμε το πρόθεμα cmd. Π.χ. cmdOK, cmdCancel κλπ.
<b>Caption</b>	Περιέχει το κείμενο που εμφανίζεται πάνω στο κουμπί. Από το κείμενο αυτό πρέπει ο χρήστης να καταλαβαίνει τι θα γίνει αν κάνει κλικ στο κουμπί αυτό.
<b>Font</b>	Ορίζουμε την γραμματοσειρά (τύπος, μέγεθος κλπ.) του κειμένου που περιέχεται στην ιδιότητα Caption.



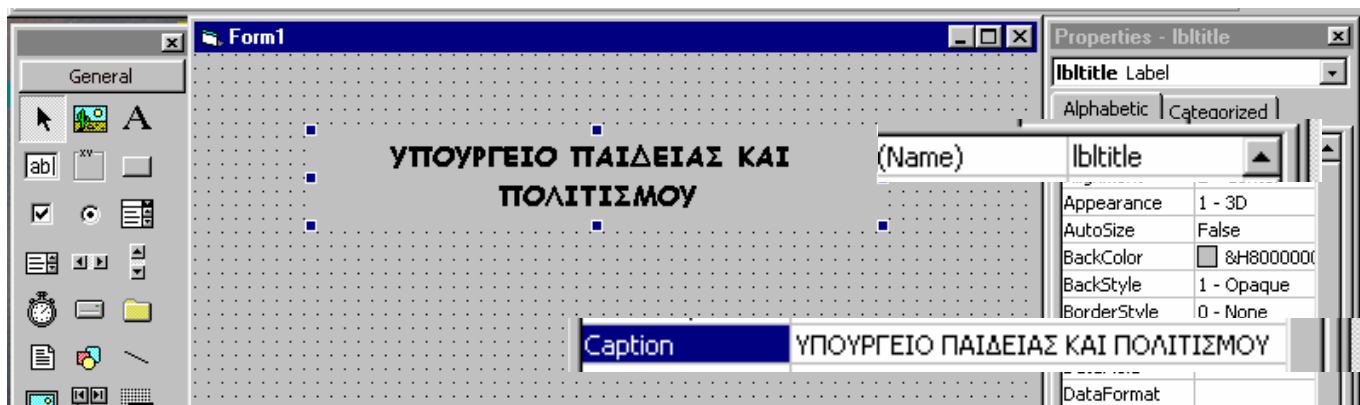
### Το χειριστήριο Label



Με το χειριστήριο αυτό εμφανίζουμε κείμενο που **δεν μπορεί να αλλάξει άμεσα ο χρήστης**. Το χειριστήριο αυτό χρησιμοποιείται συχνά για να βάζουμε ετικέτες σε αντικείμενα που δεν έχουν τη δική τους ιδιότητα Caption ή για να εμφανίσουμε γενικές πληροφορίες για το πρόγραμμα.

### Χρήσιμες Ιδιότητες

Ιδιότητα	Περιγραφή
(name)	Περιέχει το όνομα του αντικειμένου. Η Visual Basic δίνει αυτόματα ένα όνομα που αποτελείται από τη λέξη Label ακολουθούμενο από έναν αριθμό. Π.χ. Label1. Αν θέλουμε να ονομάσουμε ένα text box συνήθως χρησιμοποιούμε το πρόθεμα lbl. Π.χ. lblTitle, lblName κλπ.
Caption	Η ιδιότητα αυτή περιέχει το κείμενο που εμφανίζεται στην ετικέτα.
Font	Ορίζουμε την γραμματοσειρά (τύπος, μέγεθος κλπ.) του κειμένου που περιέχεται στην ιδιότητα Caption.



### Συνάρτηση InputBox

Η συνάρτηση InputBox εμφανίζει ένα διαλογικό παράθυρο που περιέχει ένα πλαίσιο κειμένου (Text Box) και δύο πλήκτρα διαταγής (Command buttons) το OK και το Cancel. Ο χρήστης εισάγει το κείμενο και αν επιλέξει το OK, η συνάρτηση επιστρέφει την τιμή που περιέχει το πλαίσιο κειμένου. Αν η επιλογή είναι Cancel επιστρέφεται μηδενική τιμή.

Η γενική μορφή της εντολής είναι:

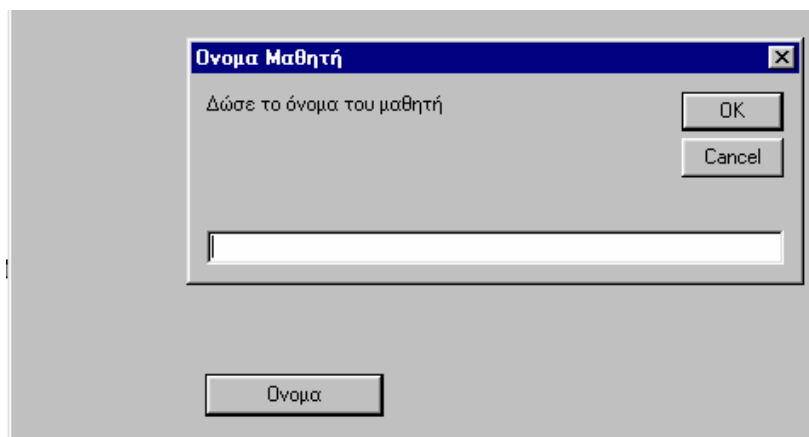
**InputBox (μήνυμα, τίτλος)**

*Μοναδική υποχρεωτική παράμετρος είναι το μήνυμα.*

Παράδειγμα. Αν θέλω με το πάτημα ενός πλήκτρου διαταγής να δημιουργηθεί το πλαίσιο με το οποίο να εισάγει ο χρήστης το όνομα ενός μαθητή και να αποθηκεύεται σε μια μεταβλητή τότε η εντολή είναι:

```
Private Sub cmdGetName_Click()  
Onoma = InputBox("Δώσε το όνομα του μαθητή", "Όνομα Μαθητή")  
End Sub
```

και το αποτέλεσμα είναι το ακόλουθο:



### **Μέθοδος Print**

Επίσης, η εμφάνιση της τιμής μιας μεταβλητής σε μια φόρμα ή σε έναν εκτυπωτή συνδεδεμένο με τον υπολογιστή μπορεί να επιτευχθεί στη Visual Basic με τη μέθοδο Print. Η μέθοδος Print έχει την ακόλουθη σύνταξη:

#### **Print παράσταση(-εις)**

όπου παράσταση είναι μια μεταβλητή, μια ιδιότητα, μια τιμή κειμένου ή/και μια αριθμητική τιμή της διαδικασίας. Για το διαχωρισμό των παραστάσεων χρησιμοποιούνται τα σύμβολα, ερωτηματικό (;) ή κόμμα (.). Το ερωτηματικό (;) τοποθετεί τα στοιχεία το ένα δίπλα στο άλλο ενώ το κόμμα (,) διαχωρίζει τα στοιχεία με ένα χαρακτήρα στηλοθέτη (tab).

## Κεφάλαιο 6-Μεταβλητές και Σταθερές

Με τον όρο μεταβλητή εννοούμε μια πληροφορία (π.χ. αριθμούς, λέξεις, τιμές ιδιοτήτων) η οποία αναφέρεται ονομαστικά και μπορεί να αλλάξει κατά την εκτέλεση του προγράμματος. Για παράδειγμα όταν ο χρήστης εισάγει πληροφορίες σε ένα πλαίσιο κειμένου, οι πληροφορίες αυτές θα πρέπει να αποθηκευτούν μέχρι να χρησιμοποιηθούν. Οι πληροφορίες αυτές αποθηκεύονται σε μεταβλητές. Στη πραγματικότητα η μεταβλητή δεν είναι παρά ένα όνομα που χαρακτηρίζει μια συγκεκριμένη "θέση μνήμης" του υπολογιστή. Στο κώδικα ενός προγράμματος μπορούμε να χρησιμοποιούμε μια ή περισσότερες μεταβλητές στις οποίες αποθηκεύονται λέξεις, αριθμοί, ημερομηνίες ή ιδιότητες. Οι μεταβλητές δημιουργούνται όταν δηλώνονται σε μια διαδικασία ή φόρμα. Στη συνέχεια είναι δυνατόν να τους δοθούν τιμές να ανακτηθούν ή να τροποποιηθούν

Το όνομα μιας μεταβλητής θα πρέπει να ακολουθεί τους παρακάτω κανόνες:

- ☀ Να αποτελείται από λατινικά γράμματα, αριθμούς ή/και την κάτω παύλα (\_). Δεν πρέπει να περιέχει κενά ή άλλα σύμβολα (π.χ. τελεία).
- ☀ Να ξεκινά με γράμμα και να έχει μήκος μέχρι 256 χαρακτήρες.
- ☀ Να μη χρησιμοποιεί δεσμευμένες λέξεις ή ονόματα αντικειμένων ή ιδιοτήτων της Visual Basic (π.χ. End, Sub, Function).
- ☀ Μπορείτε να δίνεται περιγραφικά ονόματα που συνδυάζουν μια ή περισσότερες λέξεις (π.χ. TotalCostOfProduct).
- ☀ Μπορείτε να κάνετε συνδυασμούς πεζών και κεφαλαίων γραμμάτων ή και αριθμών (π.χ. MyBirthDate, Product1).

### Τύποι Δεδομένων

Οι σημαντικότεροι τύποι δεδομένων της Visual Basic δίνονται στον παρακάτω πίνακα όπου η πρώτη στήλη δηλώνει το τύπο δεδομένων, η δεύτερη στήλη το μέγεθος (ή χώρο) αποθήκευσης που απαιτείται στη μνήμη του υπολογιστή και η τρίτη στήλη το εύρος τιμών που καλύπτει αυτός ο τύπος δεδομένων.

Τύπος Δεδομένων	Μέγεθος	Εύρος Τιμών
Boolean	2 bytes	True (αληθής) ή False (ψευδής)
Integer (Ακέραιος)	2 bytes	-32.768 έως 32.767
Currency (Νομισματική τιμή)	8 bytes	-922.337.203.685.477,5808 έως 922.337.203.685.477,5807
String (Αλφαριθμητικό)	1 byte ανά χαρακτήρα	1 έως 65.400 χαρακτήρες αλφαριθμητικών δεδομένων



Date (Ημερομηνία)	8 bytes	1 Ιανουαρίου 100 έως 31 Δεκεμβρίου 9999
-------------------	---------	--

Για την αποθήκευση των τιμών True ή False μπορεί να χρησιμοποιηθεί ο τύπος δεδομένων Boolean. Για να την αποθήκευση ακεραίων αριθμών που βρίσκονται στο διάστημα -32.768 έως +32.767 μπορεί να χρησιμοποιηθούν μεταβλητές με τύπο δεδομένων Integer. Για την αποθήκευση λέξεων και γραμμάτων αριθμών μπορεί να χρησιμοποιηθεί ο τύπος δεδομένων String. Για την αποθήκευση αριθμών που αντιπροσωπεύουν νομισματικές τιμές (π.χ. χρηματικά ποσά) μπορεί να χρησιμοποιηθεί ο τύπος δεδομένων Currency. Για την αποθήκευση ημερομηνιών μπορεί να χρησιμοποιηθεί ο τύπος δεδομένων Date.

Οι βασικοί κανόνες για τύπους δεδομένων είναι οι ακόλουθοι:

- ☀ Κάθε μεταβλητή έχει μόνο ένα τύπο δεδομένων.
- ☀ Ο τύπος κάθε μεταβλητής δηλώνεται πριν από τη χρήση της μεταβλητής.

Πριν χρησιμοποιήσετε μια μεταβλητή θα πρέπει να την ορίσετε. Ο ορισμός μιας μεταβλητής είναι μια δήλωση που καθορίζει το όνομά της και τον τύπο της..

Οι **τοπικές μεταβλητές** ορίζονται στην αρχή μιας διαδικασίας και μπορούν να διαβαστούν, να τροποποιηθούν και να εμφανιστούν μόνο μέσα σε αυτή τη διαδικασία. Η σύνταξη του ορισμού μια τοπικής μεταβλητής δίνεται από την παρακάτω εντολή:

*Dim ΌνομαΜεταβλητής As Τύπος Δεδομένων*

### **Παραδείγματα:**

*Dim TotalTax As Single*

*Dim MyBirthDate As Date*

**Παράδειγμα:** Εάν θέλετε να δηλώσετε πολλές μεταβλητές σε μια γραμμή, θα πρέπει να δηλώσετε τις μεταβλητές μια προς μια: *Dim SurName As String, FirstName As String*

### **Σταθερές**

Αν μια μεταβλητή περιέχει μια τιμή που δεν αλλάζει ποτέ (π.χ. η μαθηματική σταθερά 3,14159), τότε μπορείτε να αποθηκεύσετε τη τιμή αυτή ως σταθερά (constant) και όχι ως μεταβλητή. Μια σταθερά δηλώνεται με την παρακάτω εντολή:

*Const ΌνομαΣταθεράς = Τιμή*

όπου *ΌνομαΣταθεράς* είναι το όνομα της σταθεράς το οποίο ακολουθεί τους κανόνες των ονομάτων των μεταβλητών. Οι σταθερές χρησιμοποιούνται όταν η ίδια τιμή χρειάζεται σε διάφορα σημεία του προγράμματος. Εάν χρειαστεί να αλλάξετε τη τιμή μια σταθεράς τότε μόνο ο ορισμός Const θα χρειαστεί να αλλάξει. Οι σταθερές σας βοηθούν να διατηρήσετε το πρόγραμμά σας πιο ευανάγνωστο. Επίσης, μειώνουν το κίνδυνο λαθών, μια και η τιμή μιας μεταβλητής **δεν μπορεί να τροποποιηθεί**.

### **Παραδείγματα:**

Const *Pi* = 3,14159

Const *MyAddressNumber* = 132

### **Εκφράσεις**

Στις μεταβλητές αποδίδονται τιμές με τη χρήση του τελεστή απόδοσης τιμής (=) και εκφράσεων. Με τον όρο *έκφραση* εννοούμε είτε μια συγκεκριμένη τιμή είτε συνδυασμό τιμών και μεταβλητών συνδεδεμένων με μαθηματικούς τελεστές. Η σύνταξη της εντολής απόδοσης τιμής σε μια μεταβλητή είναι η εξής:

*ΌνομαΜεταβλητής* = *Έκφραση*

Ο τύπος δεδομένων της έκφρασης πρέπει να συμφωνεί με τον τύπο δεδομένων της μεταβλητής στην οποία αποδίδεται.

### **Παραδείγματα:**

Dim Country As String

Dim Cost As Integer, Tax As Integer, TotalCost As Integer

Country = "Greece"

Cost = 1000

Tax = 100

TotalCost= Cost + Tax + 50

## Κεφάλαιο 7-Δομές Αποφάσεων IF ... THEN και IF ... THEN ... ELSE

Όταν ένα πρόγραμμα εκτελείται (τρέχει) στον υπολογιστή, τότε οι εντολές του κώδικα προγράμματος εκτελούνται η μια μετά την άλλη σύμφωνα με τη σειρά που έχουν διατυπωθεί, μέχρι να ολοκληρωθεί και η τελευταία εντολή.

Πολλές φορές όμως απαιτείται η εκτέλεση όχι της επόμενης εντολής, αλλά κάποιας άλλης, σε διαφορετικό σημείο του προγράμματος υπερπηδώντας ενδεχόμενα τις εντολές που παρεμβάλλονται. Αυτό επιτυγχάνεται με τις δομές αποφάσεων (επιλογών).

Οι δομές αποφάσεων της Visual Basic που θα χρησιμοποιήσετε είναι οι ακόλουθες:

**IF ... THEN**

**IF ... THEN ... ELSE**

### **Η δομή IF ... THEN**

Η δομή IF ... THEN χρησιμοποιείται για την εκτέλεση μιας ομάδας εντολών, αν η συνθήκη ικανοποιείται.

Η σύνταξη της δομής έχει την ακόλουθη μορφή:	Σχηματική παράσταση της δομής
<pre>IF συνθήκη THEN     εντολή(-ές) END IF</pre>	

### **Η λειτουργία της δομής γίνεται ως εξής:**

1. Ελέγχεται η *συνθήκη*.
2. Αν η *συνθήκη* είναι αληθής (True), τότε η Visual Basic εκτελεί όλες τις εντολές που ακολουθούν τη δεσμευμένη λέξη THEN.
3. Αν όμως η *συνθήκη* είναι ψευδής (False), τότε εκτελείται η εντολή που ακολουθεί μετά το END IF.

Η δεσμευμένη λέξη END IF προσδιορίζει το τέλος της δομής απόφασης, είναι μη εκτελέσιμη εντολή και πρέπει απαραίτητα να υπάρχει. Είναι προαιρετική μόνο στην περίπτωση που μετά τη δεσμευμένη λέξη THEN ακολουθεί μια και μόνο εντολή, τότε η σύνταξη της δομής IF...THEN μπορεί να έχει την ακόλουθη μορφή:

**IF συνθήκη THEN εντολή**

Οι δύο παρακάτω δομές είναι ισοδύναμες. Η πρώτη δομή χρησιμοποιεί τη δεσμευμένη λέξη END IF ενώ η δεύτερη δεν την χρησιμοποιεί.

Δομή A:	Δομή B:
<pre>IF MemberNo&lt;=120 THEN     Price=10000 END IF</pre>	<pre>IF MemberNo&lt;=120 THEN Price=10000</pre>

<b>Παράδειγμα 1:</b>	<b>Παράδειγμα 2:</b>
<b>IF</b> Score >= 20 <b>THEN</b> Ward=10000 Label1.Caption = "Κερδίσατε!" <b>END IF</b>	DaysInFebruary=28 <b>IF</b> ((NumMonth=2) <b>AND</b> (LeapYear="Yes")) <b>THEN</b> DaysInFebruary=29 <b>END IF</b>

### Η δομή IF ... THEN ... ELSE

Η δομή IF...THEN...ELSE χρησιμοποιείται για την εκτέλεση μιας ομάδας εντολών, αν η *συνθήκη* ικανοποιείται.

Στην περίπτωση που δεν ικανοποιείται η *συνθήκη*, τότε εκτελείται μια άλλη ομάδα εντολών.

<b>Η σύνταξη της δομής έχει την ακόλουθη μορφή:</b>	<b>Σχηματική παράσταση της δομής</b>
<pre> <b>IF</b> συνθήκη <b>THEN</b>     εντολή(ές) <b>ELSE</b>     εντολή(ές) <b>END IF</b> </pre>	<pre> graph TD     Entry(( )) --&gt; Decision{συνθήκη}     Decision -- Αληθής --&gt; Then[εντολή(ές)]     Decision -- Ψευδής --&gt; Else[εντολή(ές)]     Then --&gt; Exit(( ))     Else --&gt; Exit </pre>

### Η λειτουργία της δομής γίνεται ως εξής:

1. Ελέγχεται η *συνθήκη*.
2. Αν η *συνθήκη* είναι αληθής (True), εκτελούνται οι εντολές που βρίσκονται μεταξύ γραμμής IF ... THEN και γραμμής ELSE. Στο σημείο αυτό τελειώνει και η συγκεκριμένη δομή IF...THEN...ELSE και η Visual Basic συνεχίζει με τις εντολές που ακολουθούν την END IF.
3. Αν όμως η *συνθήκη* είναι ψευδής (False), αγνοείται η πρώτη ομάδα εντολών και τότε εκτελείται η δεύτερη ομάδα εντολών που βρίσκονται μετά το ELSE.

<b>Παράδειγμα 1:</b>	<b>Παράδειγμα 2:</b>
<b>IF</b> LeapYear = "Yes" <b>THEN</b> DaysInFebruary = 29 <b>ELSE</b> DaysInFebruary = 28 <b>END IF</b>	<b>Dim</b> Vathmos <b>As</b> Integer <b>IF</b> Vathmos < 10 <b>THEN</b> Label1.Caption = " Δεν Προάγεστε! " <b>ELSE</b> Label1.Caption = " Προάγεστε!" <b>END IF</b>