

A

VISUAL BASIC

2 Visual Basic

- Το όνομα **Basic** είναι τα αρχικά γράμματα της φράσης **Beginner's All-purpose Symbolic Instruction Code**.

Η Microsoft βάσισε την Visual Basic σε μια γλώσσα προγραμματισμού γραμμένη ειδικά για του αρχάριους – BASIC. Η BASIC κυκλοφορεί εδώ και 35 χρόνια, με τη μία ή την άλλη μορφή. Όταν η Microsoft αποφάσισε να χρησιμοποιήσει την BASIC σαν την κύρια γλώσσα προγραμματισμού που θα παρέχονταν με το λειτουργικό σύστημα MS-DOS, πρόσθεσε επιπλέον λειτουργικότητα στη γλώσσα δημιουργώντας διαφορετικές παραλλαγές της BASIC με ονόματα όπως τα MBASIC, GWBASIC, BASICA, QuikBASIC, Qbasic.

- Η Microsoft διατήρησε όλες τις εκδόσεις της γλώσσας BASIC σε επίπεδο **interpreter** (διερμηνευτή) και όχι σε επίπεδο **compiler** (μεταγλωττιστή).
- Μια **διερμηνευόμενη (interpreted)** γλώσσα, όπως η BASIC, σας επιτρέπει να τρέχετε τα προγράμματά σας καθώς τα γράφετε. Οι μεταγλωττιζόμενες γλώσσες, αν και τρέχουν γρηγορότερα και είναι πιο κατάλληλες για την ανάπτυξη εμπορικών εφαρμογών, απαιτούν πολύ περισσότερη προσπάθεια. Η Qbasic και όλες οι άλλες παραλλαγές της BASIC βασιζόνταν αποκλειστικά στο κείμενο, η Visual Basic βασίζετε στα γραφικά. Η Visual Basic έχει γίνει μία από τις δημοφιλέστερες γλώσσες προγραμματισμού επειδή λειτουργεί τόσο σαν διερμηνευόμενη, όσο και σαν μεταγλωττιζόμενη γλώσσα.
- Μια **γλώσσα προγραμματισμού** είναι ένα σύνολο εντολών και επιλογών (ορίσματα), τις οποίες χρησιμοποιείτε για να δώσετε οδηγίες στον υπολογιστή. Τα προγράμματα αποτελούνται συχνά από περισσότερα του ενός αρχεία τα οποία επικοινωνούν μεταξύ τους και απευθύνονται στον υπολογιστή, οπότε θα βλέπετε συχνά τον όρο **application (εφαρμογή)** να χρησιμοποιείτε σαν συνώνυμο για όλα τα αρχεία ενός προγράμματος.
- Ένα πρόγραμμα της Visual Basic αποτελείται από κώδικα (**kod**) – σύνολο εντολών και οπτικά στοιχεία τα οποία καθορίζουν την οθόνη και τους μηχανισμούς που χρησιμοποιεί ο χρήστης για να επικοινωνήσει με το πρόγραμμα. Διπλοπατώντας σε οποιοδήποτε σημείο μέσα στο παράθυρο Φόρμας εμφανίζεται ένα παράθυρο Κώδικα το οποίο παρουσιάζει τις ακόλουθες γραμμές κώδικα:

```
Private Sub Form_Load()  
End Sub
```

Αν διπλοπατήσετε σε οποιοδήποτε μηχανισμό (ετικέτα, κουμπί εντολής...) μέσα στη Φορμα θα εμφανιστεί ο αντίστοιχος κώδικας και αντί για Form_Load() θα γράφει το αντίστοιχο όνομα.

Private Sub, End Sub δηλώνουν αρχή και τέλος κάθε διαδικασίας. Όλος ο κώδικας περιέχεται σε **διαδικασίες (procedures)** και όλες οι διαδικασίες χρειάζονται τις γραμμές κώδικα έναρξης και τερματισμού.

- Μία **διαδικασία (procedure)** είναι ένα τμήμα κώδικα της Visual Basic, το οποίο περιέχει εντολές για την εκτέλεση μιας συγκεκριμένης εργασίας.

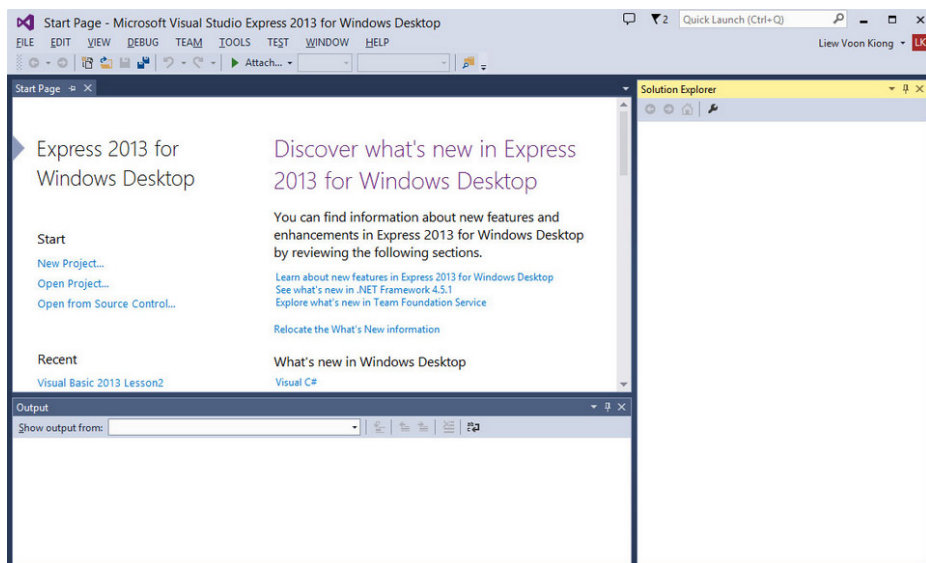
- Καθώς γράφετε ένα πρόγραμμα μπορείτε να το τρέξετε δοκιμαστικά (σε επίπεδο interpreter), για να διορθώσετε (αποσφαλματώσετε – debug) τα σφάλματα (bug). Αφού εξαλείψετε τα σφάλματα και ελέγξετε το πρόγραμμα σας, μπορείτε κατόπιν να το μεταγλωττίσετε σε μια γρηγορότερη και ασφαλέστερη μορφή εκτέλεσιμου προγράμματος.
- Ένα **έργο (project)** είναι μια συλλογή αρχείων τα οποία απαρτίζουν την εφαρμογή σας. Μια εφαρμογή μπορεί να αποτελείται από πολλά αρχεία, και το έργο είναι η συλλογή αυτών των αρχείων.
- Ένα **συμβάν (event)** είναι μια δραστηριότητα η οποία λαμβάνει χώρα κατά τη διάρκεια της εκτέλεσης ενός προγράμματος, όπως π.χ. το κλικ με το ποντίκι, ή το πάτημα ενός πλήκτρου του πληκτρολογίου.

3 Το Περιβάλλον της Visual Basic 2013

Για να φορτώσουμε τη Visual Basic εκτελούμε τα παρακάτω βήματα: **Έναρξη, Προγράμματα, Visual Basic.**

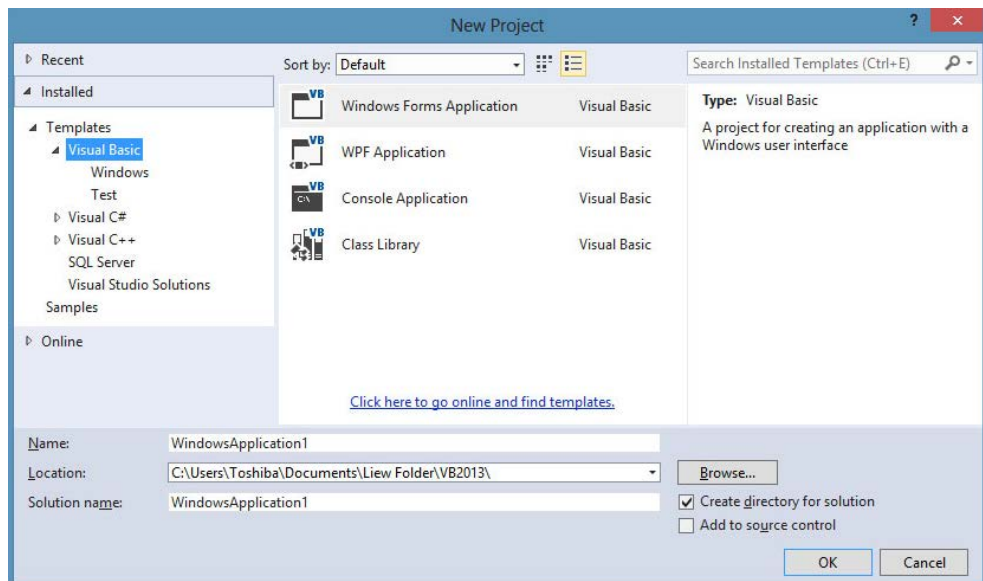
Η Visual Basic Express 2013 είναι η τελευταία έκδοση της Visual Basic που ξεκίνησε από τη Microsoft το 2013. Η Visual Basic Express 2013 είναι σχεδόν παρόμοια με τη Visual Basic Express 2012 αλλά έχει προσθέσει κάποια νέα χαρακτηριστικά. Όπως η Visual Basic Express 2012, είναι τώρα ενσωματωμένο με άλλες γλώσσες προγραμματισμού Microsoft C # και C + + σε ένα πακέτο που ονομάζεται Visual Studio 2013 Visual Studio Express 2013.

Το ολοκληρωμένο περιβάλλον ανάπτυξης, όταν ξεκινήσει η Visual Studio 2013 Express παρουσιάζεται στο παρακάτω διάγραμμα.



Εικόνα 1 Το Περιβάλλον της Visual Basic 2013

Το αρχικό παράθυρο είναι η καρτέλα – Αρχική σελίδα. Για να ξεκινήσετε μια νέα Visual Studio Express 2013 του έργου, κάντε κλικ στο New Project στην ενότητα Έναρξη για να ξεκινήσει η Visual Studio 2013 σελίδα New Project, όπως φαίνεται στο σχήμα παραπάνω. Μπορείτε επίσης να επιλέξετε να ανοίξετε ένα πρόσφατο έργο:



Visual Studio 2013 – Έργο Σελίδα

3.1 Visual Studio 2013 – Έργο Σελίδα

Η Νέα Σελίδα του εγχειρήματος περιλαμβάνει τρία πρότυπα, Visual Basic, Visual C # και Visual C + +. Από τη στιγμή που πρόκειται να μάθετε Visual Basic 2013, θα επιλέξετε Visual Basic. Η Visual Basic 2013 σας προσφέρει τέσσερις τύπους των έργων που μπορείτε να δημιουργήσετε, είναι **Windows Forms Application**, **WPF Application**, **Console Application** και **Class Library**. Από τη στιγμή που πρόκειται να μάθετε να δημιουργήτε εφαρμογές των Windows, θα επιλέξετε τα Windows Έντυπα Αιτήσεων.

Στο κάτω μέρος αυτού του παραθύρου διαλόγου, μπορείτε να αλλάξετε το προεπιλεγμένο όνομα του έργου **Windows Application1** σε κάποιο άλλο όνομα που θέλετε, για παράδειγμα, MyFirstProgram και να ορίσετε τη διαδρομή του φακέλου σας. Αφού έχετε μετονομάσει το έργο, κάντε κλικ στο κουμπί OK για να συνεχίσετε. Το ακόλουθο IDE των Windows θα εμφανιστεί, είναι παρόμοια με τη Visual Basic 2012. Η Εργαλειοθήκη δεν εμφανίζεται μέχρι να κάνετε κλικ στο Εργαλειοθήκη tab. Όταν κάνετε κλικ στην καρτέλα Εργαλειοθήκη ή χρησιμοποιήσετε τα πλήκτρα συντόμευσης Ctrl + Alt + X, η εργαλειοθήκη κοινού ελέγχου θα εμφανιστεί..

7 Λογικοί τελεστές

7.1 Τελεστής And

Χρησιμοποιείται για την εκτέλεση ενός λογικού γινομένου (τομής) σε δύο εκφράσεις.

Σύνταξη

```
αποτέλεσμα = έκφραση1 And έκφραση2
```

Η σύνταξη του τελεστή And έχει τα εξής μέρη:

Μέρος	Περιγραφή
αποτέλεσμα	Απαιτείται. Οποιαδήποτε αριθμητική μεταβλητή.
έκφραση1	Απαιτείται. Οποιαδήποτε έκφραση.
έκφραση2	Απαιτείται. Οποιαδήποτε έκφραση.

Εάν και οι δύο εκφράσεις αποτιμώνται στην τιμή True, το αποτέλεσμα έχει την τιμή True. Εάν οποιαδήποτε από τις δύο εκφράσεις αποτιμάται στην τιμή False, το αποτέλεσμα έχει την τιμή False. Ο παρακάτω πίνακας δείχνει πώς καθορίζεται η τιμή του αποτελέσματος:

Εάν η έκφραση 1 έχει την τιμή:	και η έκφραση 2 έχει την τιμή:	το αποτέλεσμα έχει την τιμή:
True	True	True
True	False	False
False	True	False
False	False	False

Ο τελεστής And εκτελεί επίσης μια bit προς bit σύγκριση μεταξύ bits τα οποία είναι τοποθετημένα με τον ίδιο ακριβώς τρόπο σε δύο αριθμητικές εκφράσεις και καθορίζει την τιμή του αντίστοιχου bit στο αποτέλεσμα σύμφωνα με τον παρακάτω πίνακα:

Εάν το bit στην έκφραση 1 είναι	και το bit στην έκφραση 2 είναι	το αποτέλεσμα είναι
0	0	0
0	1	0
1	0	0
1	1	1

*Dim A, B, C, D, Elegxos1, Elegxos2,
Elegxos3, Elegxos4*

*A = 10: B = 8: C = 6: D = 5
Elegxos1 = A > B And B > C*

*Έκχώρηση αρχικών τιμών σε μεταβλητές.
Έπιστρέφεται η τιμή True.*

$Elegxos2 = B > A \text{ And } B > C$
 $Elegxos3 = A > B \text{ And } B > D$
 $Elegxos4 = A \text{ And } B$

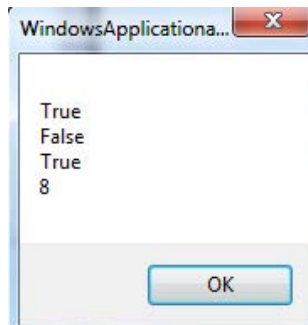
‘Επιστρέφεται η τιμή False.
 ‘Επιστρέφεται η τιμή True.
 ‘Επιστρέφεται η τιμή 8 (σύγκριση bit προς bit).

(Κώδικας)

```

Form1.vb [Design] Form1.vb
General Declarations
Public Class Form1
    Private Sub Form1_Activated(sender As Object,
        Dim A, B, C, D, Elegxos1, Elegxos2,
        Elegxos3, Elegxos4
        A = 10 : B = 8 : C = 6 : D = 5
        Elegxos1 = A > B And B > C
        Elegxos2 = B > A And B > C
        Elegxos3 = A > B And B > D
        Elegxos4 = A And B
        MsgBox(Elegxos1 & vbCrLf & Elegxos2 &
        vbCrLf & Elegxos3 & vbCrLf & Elegxos4)
    End Sub
End Class
  
```

(Αποτέλεσμα)



Εικόνα 13 Τελεστής And – Κώδικας και Αποτέλεσμα

7.2 Τελεστής OrElse

Εκτελεί βραχυκύκλωμα αποκλεισμού λογικής διάζευξης στις δύο εκφράσεις.

Σύνταξη

αποτέλεσμα = έκφραση1 **OrElse** έκφραση2

Η σύνταξη του τελεστή = έχει τα εξής μέρη:

Μέρος	Περιγραφή
αποτέλεσμα	Απαιτείται. Απαιτούμενη. Κάθε Boolean έκφραση.
έκφραση1	Απαιτείται. Απαιτούμενη. Κάθε Boolean έκφραση.
έκφραση2	Απαιτείται. Απαιτούμενη. Κάθε Boolean έκφραση.

Μια λογική λειτουργία λέγεται ότι είναι βραχυκύκλωμα εάν ο κώδικας μπορεί να παρακάμψει την αξιολόγηση μιάς έκφρασης, ανάλογα με το αποτέλεσμα μιας άλλης έκφρασης. Εάν το αποτέλεσμα της πρώτης έκφρασης αξιολογείται καθορίζει το τελικό αποτέλεσμα της λειτουργίας, δεν υπάρχει καμία ανάγκη να αξιολογηθεί η δεύτερη έκφραση, επειδή δεν μπορεί να αλλάξει το τελικό αποτέλεσμα. Το βραχυκύκλωμα μπορεί να βελτιώσει την απόδοση εάν η παράκαμψη της έκφρασης είναι περίπλοκη, ή αν περιλαμβάνει διαδικασία κλήσεις.

Εάν μια ή και οι δύο εκφράσεις είναι True, το αποτέλεσμα είναι True. Ο παρακάτω πίνακας δείχνει πώς προσδιορίζεται αποτέλεσμα.

Εάν η έκφραση 1 έχει τιμή:	και η έκφραση 2 έχει τιμή:	το αποτέλεσμα έχει τιμή:
True	Δεν αξιολογείται	True
False	True	True
False	False	False

Τύποι Δεδομένων

Ο χειριστής OrElse ορίζεται μόνο για την **Boolean Τύπου Δεδομένων (Visual Basic)**. Visual Basic μετατρέπει κάθε τελεστή, όπως είναι απαραίτητο για Boolean και εκτελεί τη λειτουργία εξ ολοκλήρου σε Boolean. Εάν εκχωρήσετε το αποτέλεσμα σε έναν αριθμητικό τύπο, Visual Basic μετατρέπει από Boolean σε αυτό το είδος. Αυτό θα μπορούσε να προκαλέσει απροσδόκητη συμπεριφορά. Για παράδειγμα, 5 OrElse 12 αποτελείσματα σε -1, όταν μετατρέπεται σε Integer.

Το ακόλουθο παράδειγμα χρησιμοποιεί τον τελεστή OrElse να εκτελεί λογική διάζευξη στις δύο εκφράσεις. Το αποτέλεσμα είναι μια Boolean τιμή που αντιπροσωπεύει το αν μία από τις δύο εκφράσεις είναι αλήθεια. Εάν η πρώτη έκφραση είναι αληθής, η δεύτερη δεν αξιολογείται.

Dim a As Integer = 10 'Εκχώρηση αρχικών τιμών σε μεταβλητές.

Dim b As Integer = 8

Dim c As Integer = 6

Dim Elegxos1, Elegxos2, Elegxos3 As Boolean

Elegxos1 = a > b OrElse b > c 'Επιστρέφεται η τιμή True.

Elegxos2 = b > a OrElse b > c 'Επιστρέφεται η τιμή True.

Elegxos3 = b > a OrElse c > b 'Επιστρέφεται η τιμή False

Το προηγούμενο παράδειγμα παράγει αποτελέσματα από True, True και False αντίστοιχα. Κατά τον υπολογισμό των FirstCheck, η δεύτερη έκφραση δεν αξιολογείται επειδή η πρώτη είναι ήδη True. Ωστόσο η δεύτερη έκφραση αξιολογείται κατά τον υπολογισμό του secondCheck.

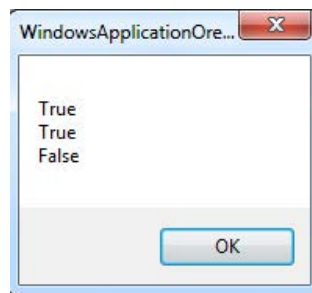
(Κώδικας)

```

Public Class Form1
    Private Sub Form1_Activated(sender As Object,
        e As EventArgs) Handles Me.Activated
        Dim a As Integer = 10
        Dim b As Integer = 8
        Dim c As Integer = 6
        Dim Elegxos1, Elegxos2, Elegxos3 As Boolean
        Elegxos1 = a > b OrElse b > c
        Elegxos2 = b > a OrElse b > c
        Elegxos3 = b > a OrElse c > b
        MsgBox(Elegxos1 & vbCrLf & Elegxos2 & vbCrLf & Elegxos3)
    End Sub
End Class

```

(Αποτέλεσμα)



Εικόνα 14 Τελεστή OrElse – Κώδικας και Αποτέλεσμα

7.3 Τελεστής Andalso

Εκτελεί βραχυκύκλωμα λογικό συνδυασμό στις δύο εκφράσεις.

Σύνταξη

αποτέλεσμα = έκφραση1 Andalso έκφραση2

Η σύνταξη του τελεστή **Andalso** έχει τα εξής μέρη:

Μέρος	Περιγραφή
αποτέλεσμα	Απαιτούμενη. Κάθε Boolean έκφραση. Το αποτέλεσμα είναι η Boolean αποτέλεσμα της σύγκρισης των δύο εκφράσεων.
έκφραση1	Απαιτούμενη. Κάθε Boolean έκφραση.
έκφραση2	Απαιτούμενη. Κάθε Boolean έκφραση.

Ο παρακάτω πίνακας δείχνει πώς προσδιορίζεται η τιμή του αποτελέσματος:

Εάν η έκφραση 1 έχει την τιμή:	και η έκφραση 2 έχει την τιμή:	το αποτέλεσμα έχει την τιμή:
True	True	True
False	False	False
False	(Δεν αξιολογείται)	False

Τύποι Δεδομένων

Ο χειριστής `andalso` ορίζεται μόνο για την **Boolean Τύπου Δεδομένων (Visual Basic)** Visual Basic μετατρέπει κάθε τελεστή, όπως είναι απαραίτητο για Boolean και εκτελεί τη λειτουργία εξ ολοκλήρου σε Boolean. Εάν εκχωρήσετε το αποτέλεσμα σε έναν αριθμητικό τύπο, Visual Basic μετατρέπει από Boolean σε αυτό το είδος. Αυτό θα μπορούσε να προκαλέσει απροσδόκητη συμπεριφορά. Για παράδειγμα, `5 andalso 12` αποτελέσματα σε `-1`, όταν μετατρέπεται σε Integer.

Το ακόλουθο παράδειγμα χρησιμοποιεί τον τελεστή `andalso` να εκτελέσει ένα λογικό συνδυασμό στις δύο εκφράσεις. Το αποτέλεσμα είναι μια Boolean τιμή που αντιπροσωπεύει εάν ολόκληρο το `conjoined` έκφραση είναι αλήθεια. Εάν η πρώτη έκφραση είναι ψευδής, η δεύτερη δεν αξιολογείται.

Dim a As Integer = 10

Dim b As Integer = 8

Dim c As Integer = 6

Dim Elegxos1, Elegxos2, Elegxos3 As Boolean

Elegxos1 = a > b AndAlso b > c

Elegxos2 = b > a AndAlso b > c

Elegxos3 = a > b AndAlso c > b

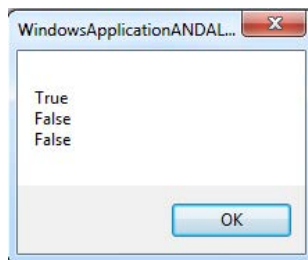
Το προηγούμενο παράδειγμα παράγει αποτελέσματα από **True**, **False** και **False**, αντίστοιχα. Κατά τον υπολογισμό των secondCheck, η δεύτερη έκφραση δεν αξιολογείται επειδή η πρώτη είναι ήδη Λάθος. Ωστόσο, η δεύτερη έκφραση αξιολογείται κατά τον υπολογισμό του thirdCheck.

(Κώδικας)

```

Form1.vb*  x
(Form1 Events)  Activated
Public Class Form1
    Private Sub Form1_Activated(sender As Object,
        e As EventArgs) Handles Me.Activated
        Dim a As Integer = 10
        Dim b As Integer = 8
        Dim c As Integer = 6
        Dim Elegxos1, Elegxos2, Elegxos3 As Boolean
        Elegxos1 = a > b AndAlso b > c
        Elegxos2 = b > a AndAlso b > c
        Elegxos3 = a > b AndAlso c > b
        MsgBox(Elegxos1 & vbCrLf & Elegxos2 & vbCrLf & Elegxos3)
    End Sub
End Class
    
```

(Αποτέλεσμα)



Εικόνα 15 Τελεστής Andalso – Κώδικας και Αποτέλεσμα

7.4 Τελεστής Not

Χρησιμοποιείται για την εκτέλεση μιας λογικής άρνησης σε μια έκφραση.

Σύνταξη

αποτέλεσμα = **Not** έκφραση

Η σύνταξη του τελεστή Not έχει τα εξής μέρη:

Μέρος	Περιγραφή
αποτέλεσμα	Απαιτείται. Οποιαδήποτε αριθμητική μεταβλητή.
έκφραση	Απαιτείται. Οποιαδήποτε έκφραση.

Ο παρακάτω πίνακας δείχνει πώς προσδιορίζεται η τιμή του αποτελέσματος:

Εάν η έκφραση έχει την τιμή:	τότε το αποτέλεσμα έχει την τιμή
True	False
False	True

Επιπλέον, ο τελεστής Not αντιστρέφει τις τιμές των bit οποιασδήποτε μεταβλητής και καθορίζει την τιμή του αντίστοιχου bit στο αποτέλεσμα σύμφωνα με τον επόμενο πίνακα:

Εάν το bit στην έκφραση είναι	τότε το bit στο αποτέλεσμα είναι
0	1
1	0

B

VISUAL BASIC FOR APPLICATIONS

17 Εντολές, Μέθοδοι και Ιδιότητες για το Microsoft Excel

Όλες οι συναρτήσεις, εντολές, μέθοδοι και ιδιότητες της Visual Basic χρησιμοποιούνται στην καταγραφή μακροεντολών στο Excel, με την διαφορά ότι στην καταγραφή μακροεντολών με την επεξεργασία Visual Basic, προσθέτονται μερικές εντολές, μέθοδοι και ιδιότητες που αφορούν τα Worksheet – Φύλλο εργασίας, Workbook – βιβλίο εργασίας, Cells – κελιά, Row – γραμμή, Column –στήλη.

17.1 Ενεργοποίηση του βιβλίου εργασίας.

Η ενεργοποίηση του βιβλίου εργασίας γίνεται χρησιμοποιώντας την μέθοδο **Activating**.

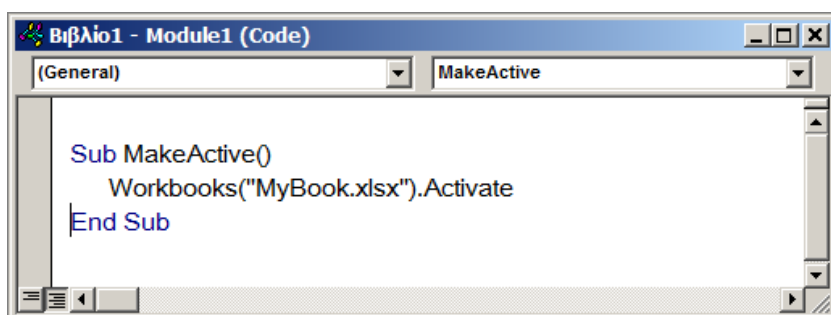
Το παρακάτω παράδειγμα ενεργοποιεί το ανοικτό βιβλίο εργασίας «MyBook.xlsx.»:

```
Sub MakeActive()
```

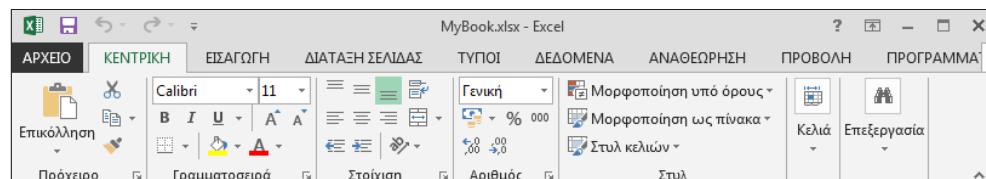
```
    Workbooks("MyBook.xlsx").Activate
```

```
End Sub
```

(Κώδικας)



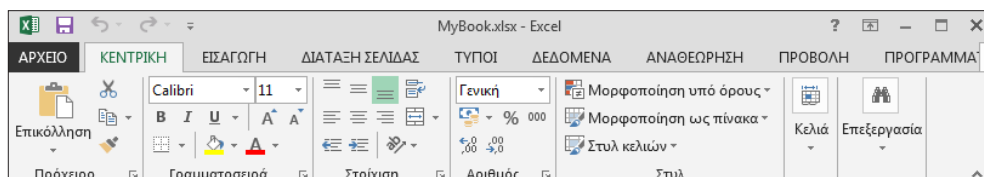
Έχοντας ανοίξει ένα καινούριο βιβλίο στο XL τρέχουμε την μακροεντολή.



Εικόνα 114 Ενεργοποίηση του βιβλίου εργασίας – Κώδικας

(Αποτέλεσμα)

Παρατηρούμε ότι έχει ενεργοποιηθεί το βιβλίο με όνομα MyBook.xlsx.



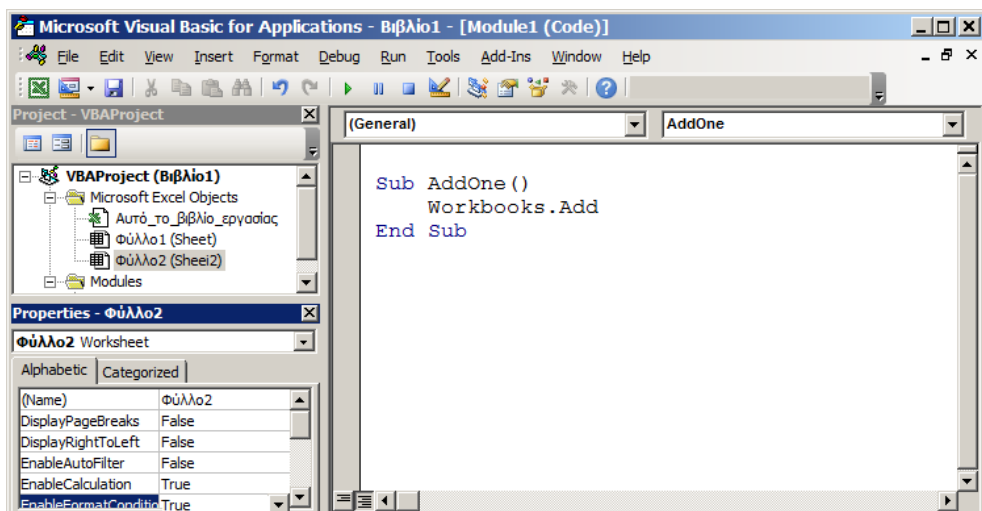
Εικόνα 115 Ενεργοποίηση του βιβλίου εργασίας – Αποτέλεσμα

17.2 Δημιουργία καινούργιου βιβλίου εργασίας

Για να δημιουργήσετε καινούργιο βιβλίο εργασίας θα χρησιμοποιήσετε την μέθοδο Add. Στο παρακάτω παράδειγμα δημιουργείται ένα καινούργιο βιβλίο εργασίας. Το Microsoft Excel αυτομάτως αριθμεί τα βιβλία εργασίας. Αν είχατε μόνο ένα βιβλίο εργασίας, η παρακάτω εντολή θα ανοίξει καινούργιο βιβλίο2:

```
Sub AddOne()  
    Workbooks.Add  
End Sub
```

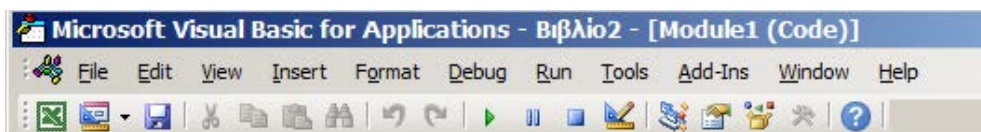
(Κώδικας)



Εικόνα 116 Δημιουργία καινούριου βιβλίου εργασίας – Κώδικας

Παρατηρούμε ότι το όνομα του τρέχοντος βιβλίου είναι Βιβλίο1(Module1-Code). Μόλις πατήσουμε το πλήκτρο **Run** αλλάζει ονομασία το βιβλίο μας και αυτό γίνεται γιατί προστίθεται ένα νέο βιβλίο εργασίας με όνομα Book 2.

(Αποτέλεσμα)

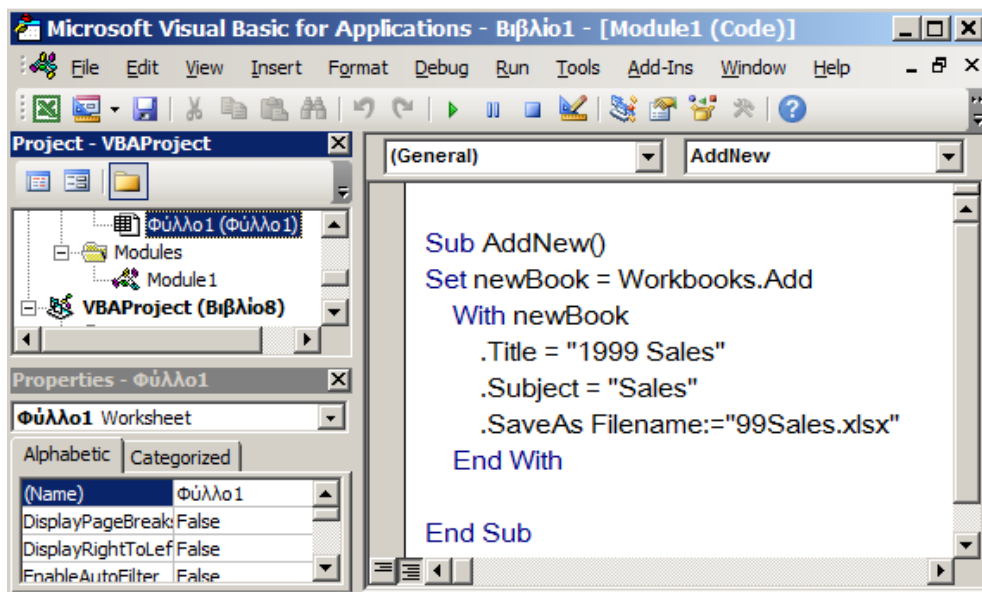


Εικόνα 117 Δημιουργία καινούριου βιβλίου εργασίας – Αποτέλεσμα

Χρησιμοποιώντας μερικές από τις ιδιότητες, μπορείτε να δημιουργήσετε καινούργιο βιβλίο εργασίας δίνοντας, για παράδειγμα, τον τίτλο, το θέμα και με τι όνομα να αποθηκευτεί το βιβλίο εργασίας:

```
Sub AddNew()  
Set newBook = Workbooks.Add  
    With newBook  
        .Title = "1999 Sales"  
        .Subject = "Sales"  
        .SaveAs filename:= "99Sales.xlsx"  
    End With  
End Sub
```

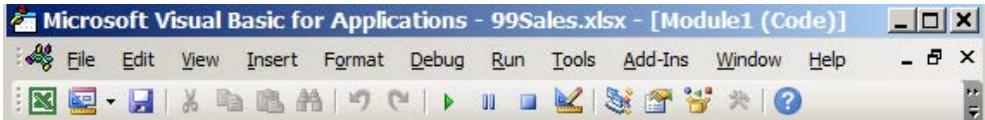
(Κώδικας)



Εικόνα 118 Δημιουργία καινούργιου βιβλίου εργασίας και ορισμός τίτλου θέματος και ονόματος του αρχείου –Κώδικας

Όπως και στο προηγούμενο παράδειγμα έτσι και σ'αυτό προσθέτουμε ένα βιβλίο εργασίας με τη διαφορά ότι στο καινούριο βιβλίο δίνουμε όνομα, θέμα και εντολή για αυτόματη αποθήκευση.

(Αποτέλεσμα)



Εικόνα 119 Δημιουργία καινούργιου βιβλίου εργασίας και ορισμός τίτλου, θέματος και ονόματος και αρχείου – Αποτέλεσμα

17.3 Άνοιγμα βιβλίου εργασίας

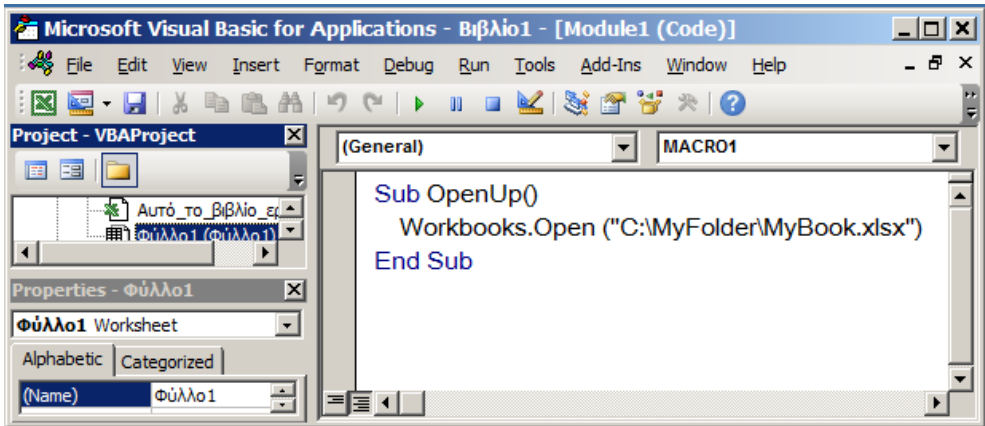
Για να ανοίξετε ένα βιβλίο εργασίας με την μέθοδο `Open`. Το παρακάτω παράδειγμα ανοίγει ένα βιβλίο εργασίας με όνομα `MyBook.xlsx` στον φάκελο «`MyFolder`» του σκληρού δίσκου `C:`

```
Sub OpenUp()
```

```
    Workbooks.Open("C:\MyFolder\MyBook.xlsx")
```

```
End Sub
```

(Κώδικας)



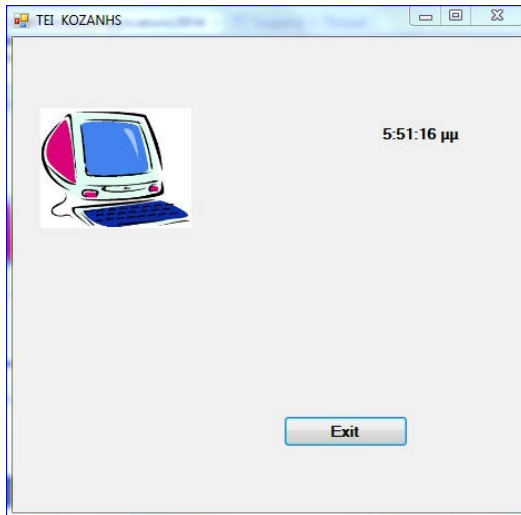
С

ΔΗΜΙΟΥΡΓΙΑ ΕΦΑΡΜΟΓΩΝ

18 Εισαγωγή εικόνας και τρέχουσας ώρας – αρχείο Form1

Η πρώτη εφαρμογή που θα δημιουργήσετε είναι απλή – το πρόγραμμα **Form1** που θα γράψετε, θα εμφανίζει μια φόρμα με τίτλο TEI KOZANHS, μια εικόνα αριστερά και δεξιά την ώρα. Θα τερματίζει το πρόγραμμα πατώντας το πλήκτρο EXIT:

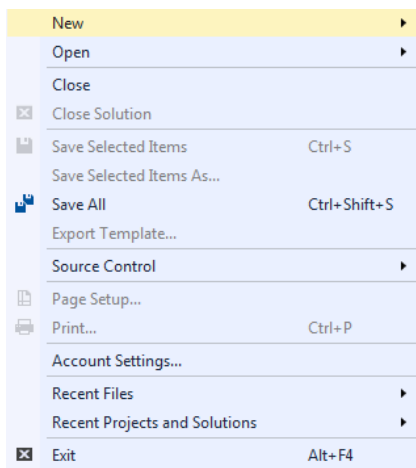
Φορτώστε τη Visual Basic και ακολουθήστε τα παρακάτω βήματα:



Εικόνα 194 Εισαγωγή εικόνας και τρέχουσας ώρας (Form1)

18.1 Δημιουργία νέου έργου

Επιλέξτε την εντολή File, New Project για να εμφανίσετε το πλαίσιο διαλόγου



Εικόνα 195 Δημιουργία νέου έργου

Η Visual Basic θα εμφανίσει μόνο ένα παράθυρο φόρμας με όνομα Form1 (στη γραμμή τίτλου του παραθύρου)

18.2 Αλλαγή μεγέθους φόρμας

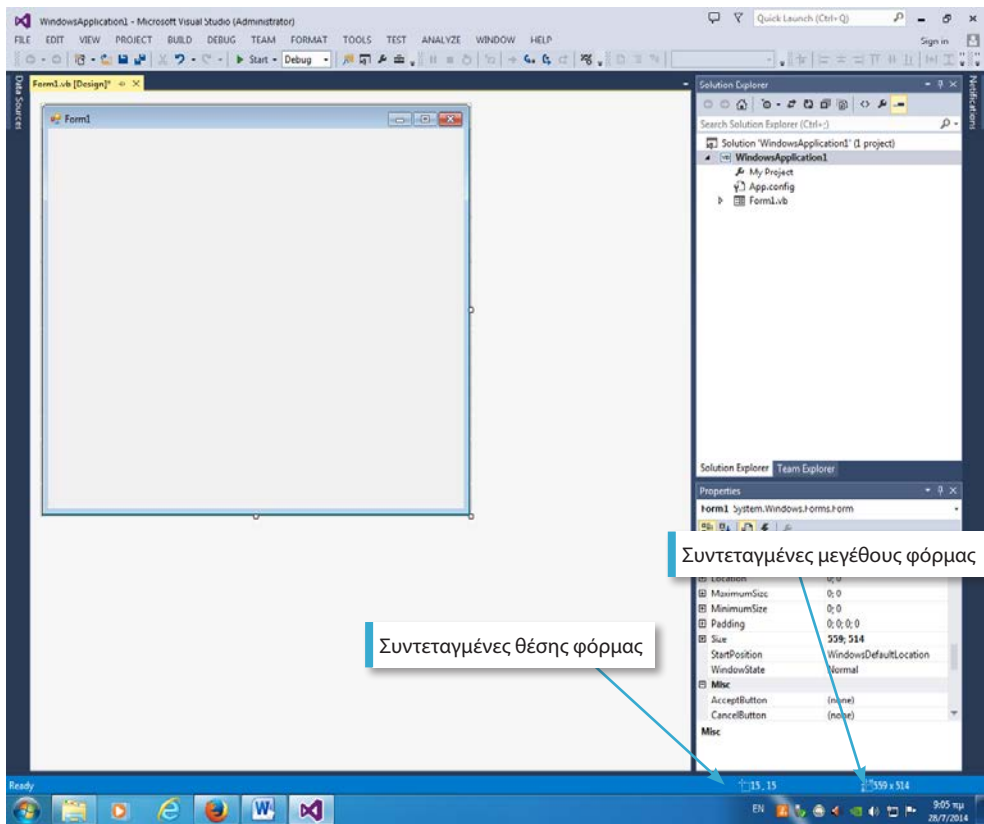
Σύρετε την κάτω δεξιά λαβή του παραθύρου προς τα κάτω και δεξιά. Καθώς σέρνετε, παρατηρήστε ότι οι ενδείξεις για το πλάτος και το ύψος, στο κάτω μέρος του παραθύρου, αλλάζουν. Προσαρμόστε το μέγεθος της φόρμας έτσι ώστε να έχει διαστάσεις περίπου 7125 x 5220 twips. Όταν αλλάζετε το μέγεθος του παραθύρου Φόρμας, ουσιαστικά αλλάζετε το παράθυρο της εφαρμογής σας.

Το twip είναι μια μονάδα μέτρησης για την οθόνη. Μπορείτε να θεωρήσετε ένα twip σαν μια «κουκκίδα» της οθόνης σας, αλλά διαφορετικές συσκευές και κάρτες οθόνης έχουν διαφορετικές αναλύσεις, και κατά συνέπεια διαφορετικό αριθμό κουκκίδων.

Καθώς προσαρμόζετε τη θέση και το μέγεθος των παραθύρων των φορμών σας, καλό θα είναι να προσέχετε τις συντεταγμένες θέσης και μεγέθους της φόρμας που εμφανίζονται στα δεξιά της γραμμής εργαλείων. Οι τιμές αυτές εμφανίζονται πάντα σε ζεύγη.

Η πρώτη τιμή στο ζεύγος συντεταγμένων θέσης αντιπροσωπεύει την απόσταση του αριστερού άκρου της φόρμας από το αριστερό άκρο της οθόνης. Η δεύτερη τιμή αντιπροσωπεύει την απόσταση του πάνω άκρου της φόρμας από το πάνω άκρο της οθόνης.

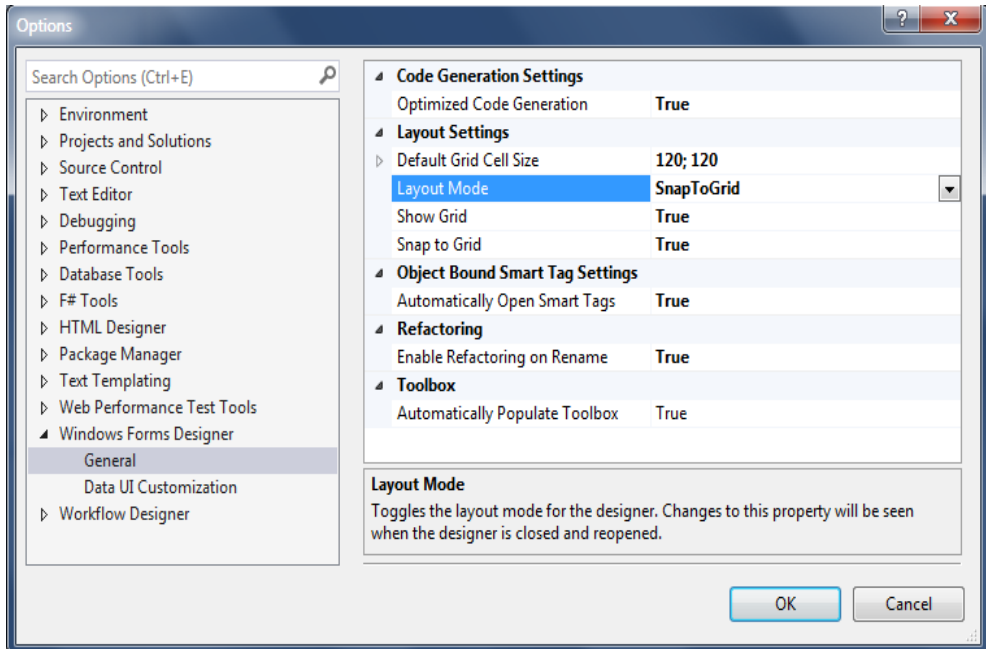
Το δεύτερο ζεύγος συντεταγμένων αντιπροσωπεύει το πλάτος και το ύψος του παραθύρου της φόρμας. Οι ιδιότητες Left και Top αντιπροσωπεύουν τις συντεταγμένες θέσης της φόρμας και εκφράζουν την απόσταση από το πάνω και αριστερό άκρο της οθόνης. Οι ιδιότητες Width Height αντιπροσωπεύουν το πλάτος και το ύψος του παραθύρου της φόρμας. Η Visual Basic ενημερώνει αυτόματα τις τιμές αυτών των ιδιοτήτων στο παράθυρο Ιδιοτήτων όταν μετακινείτε ή αλλάζετε τη θέση της φόρμας στη περιοχή επεξεργασίας.



Εικόνα 196 Συντεταγμένες μεγέθους και θέσης φόρμας

18.3 Τοποθέτηση και ταξινόμηση κατά μέγεθος ελέγχου – Χρήση του Grid

Όταν προσθέτετε στοιχεία ελέγχου σε μια φόρμα είναι δυνατό να ρυθμίσετε τον τρόπο ελέγχου που τοποθετούνται και το μέγεθος με την ενεργοποίηση ενός πλέγματος. Υπάρχουν διάφοροι τρόποι για να ρυθμίσετε τη διακριτότητα και τη συμπεριφορά του δικτύου Visual Studio. Οι ρυθμίσεις αυτές είναι παγκόσμιες, εφαρμόζονται σε όλες τις μορφές και τα σχέδια, όχι μόνο για τη σημερινή του μορφή ή έργου. Οι ρυθμίσεις δικτύου αλλάζουν χρησιμοποιώντας την οθόνη Visual Studio Επιλογές. Για να αποκτήσετε πρόσβαση σε αυτή την οθόνη, επιλέξτε Επιλογές ... από το μενού Εργαλεία. Όταν εμφανιστεί το παράθυρο διαλόγου, κάντε κλικ στο **Windows Forms Designer** στο αριστερό δέντρο χέρι. Μόλις επιλεγεί ο διάλογος θα πρέπει να εμφανίζεται ως εξής:



Εικόνα 197 Εμφάνιση και απόκρυψη ευθυγράμμισης στοιχείων ελέγχου μέσα στη φόρμα

Ένας από τους βασικούς στόχους στο σχεδιασμό αισθητικά ευχάριστο, είναι οι μορφές να πάρουν τους ελέγχους ευθυγράμμισης. Ένας τρόπος για να κάνει το καθήκον της ευθυγράμμισης των ελέγχων ευκολότερη περιλαμβάνει τη χρήση του «Snap Lines» στο Visual Studio. Όταν ενεργοποιηθεί, η λειτουργία Snap Lines προκαλεί μια διαχωριστική γραμμή ανάμεσα σε μια άκρη του ελέγχου που κινείται και το αντίστοιχο άκρο του πλησιέστερου ελέγχου στη φόρμα όταν οι άκρες είναι σε ευθυγράμμιση.

Για ενεργών γραμμών Snap επιλέξτε Επιλογές ... από το μενού Εργαλεία, κάντε κλικ στο **Windows Form Designer** στο δέντρο στα αριστερά του διαλόγου **Options** και να διαμορφώσετε τις παρακάτω ρυθμίσεις:

Layout Mode: SnapLines

SnapToGrid: False

ShowGrid: False

και πατώντας το κουμπί Ο.Κ.

Όταν εφαρμόσετε τις ρυθμίσεις προσθέστε ένα στοιχείο ελέγχου κουμπιού σε μια φόρμα. Στη συνέχεια, κάντε κλικ στο στοιχείο ελέγχου πλαισίου λίστας στην εργαλειοθήκη και σύρετε πάνω στη φόρμα. Όπως μπορείτε να μετακινήσετε το πλαίσιο λίστας κάτω από το κουμπί μια γραμμή θα εμφανιστεί μεταξύ των ελέγχων σε κάθε σημείο στις ακμές ευθυγράμμισης. Για παράδειγμα, τα ακόλουθα στοιχεία δείχνουν μια γραμμή που εμφανίζεται στο σημείο τα αριστερά και δεξιά άκρα – χέρι των ελέγχων ευθυγράμμιση:

D

Καταγραφή μακροεντολών με
την Επεξεργασία Visual Basic

31 Χρήση των εντολών For Each ... Next

Στο παρακάτω παράδειγμα θα δημιουργήσετε μια μακροεντολή η οποία σε ένα φύλλο εργασίας θα ελέγχει κάποια ορθογώνια περιοχή κελιών και αν η τιμή κάποιου κελιού της περιοχής θα είναι μικρότερη από 0.001 θα αντικαθιστάται με μηδέν.

31.1 Εισαγωγή δεδομένων στο φύλλο εργασίας

Ακολουθήστε τα παρακάτω βήματα:

- Φορτώστε το Microsoft Excell με τις εντολές:
Έναρξη,
Προγράμματα,
Microsoft Excel
- Ονομάστε το φύλλο 1 Sheet1: διπλατώνοντας στο όνομα φύλλο1 και γράφοντας το καινούριο όνομα **Sheet1**.
- Γράψτε στις πρώτες 10 γραμμές και 4 στήλες τα δεδομένα που βλέπετε παρακάτω στην εικόνα:

	A	B	C	D	E	F	G	H	I	J	K
1	25	32	0,003	24							
2	34	1124	0,007	-25							
3	0,5	-2	2E+09	-245							
4	0,001	-6	5,678	3,45							
5	0,0003	354		0,078							
6	1	234	34	0,0012							
7	234	678	125	0,0019							
8	0,009	-15	200	0,02							
9	-15	24	0,576	0,354							
10	-24	0,001	13	0,0017							
11											
12											

Εικόνα 273 Εισαγωγή δεδομένων στο φύλλο εργασίας

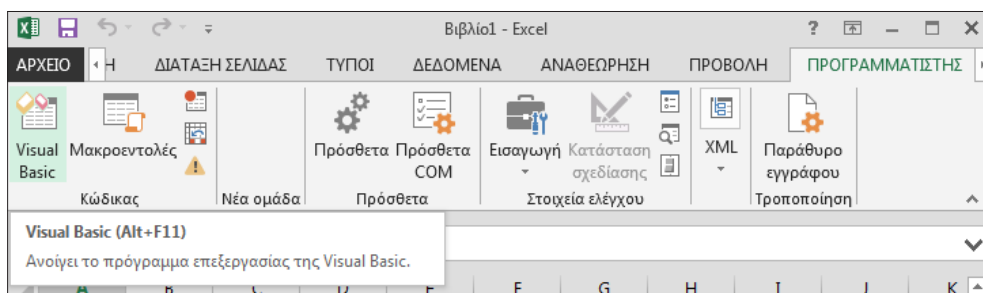
- Αποθηκεύστε το αρχείο με τις εντολές **Αρχείο, Αποθήκευση** ως, στον φάκελο σας με όνομα **βιβλίο1**.

31.2 Παράθυρο επεξεργασίας Visual Basic

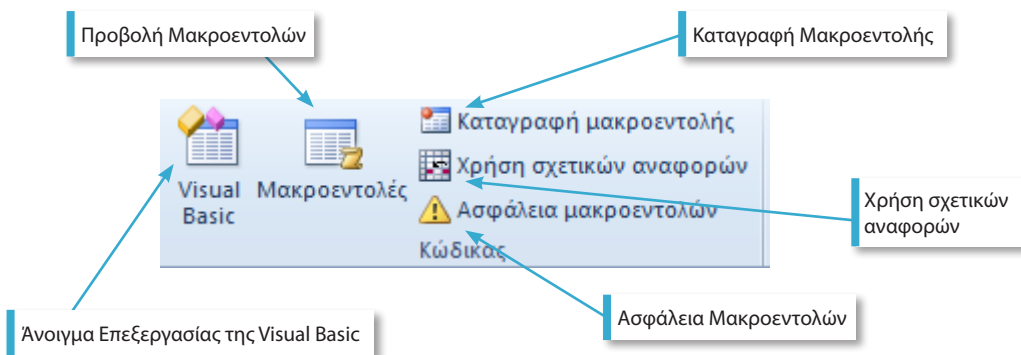
Μπορείτε να γράψετε τον κωδικό μιας μακροεντολής χρησιμοποιώντας την επεξεργασία Visual Basic.

- Από την Γραμμή μενού φύλλου εργασίας του EXCEL διαλέξτε τις εντολές:
**Προγραμματιστής,
Γραμμες Εργαλείων**

Στην οθόνη σας θα εμφανιστούν οι Εντολές Μακροεντολών της Visual Basic



31.3 Καταγραφή μακροεντολής – Macro1



Εικόνα 274 Παράθυρο επεξεργασίας Μακροεντολών

- Πατήστε στην Εντολή **Προβολή Μακροεντολών** και στο Όνομα μακροεντολής δώστε το όνομα **macro1**.