INDICE

INTRODUZIONE A VISUAL BASIC 6.0 6
REQUISITI MINIMI PER INSTALLARE VISUAL BASIC 6.0 7
INSTALLAZIONE DI VISUAL BASIC 6.0 7
AVVIARE VISUAL BASIC 6.0 8
VIDEATA DI VISUAL BASIC 8
BARRA DEL TITOLO: 10
BARRA DEI MENÙ: 10
BARRA STRUMENTI 10
CASELLA DEGLI STRUMENTI 11
FINESTRA PROGETTO 12
FINESTRA DI PROGETTAZIONE FORM 12
FINESTRA IMMEDIATA 12
FINESTRA DISPOSIZIONE FORM 13
FINESTRA PROPRIETÀ 14
FINESTRA GESTIONE PROGETTO 16
IDE 18
HELP 18
GUIDA IN LINEA 20
RICERCA 20
CODICE DI ESEMPIO DA COPIARE 20
GUIDA SENSIBILE AL CONTESTO 21
CONNESSIONE TRAMITE WEB 21
NOZIONI FONDAMENTALI 22
FINESTRA 22
CONTROLLO: 22
COMPONENTE 22
OGGETTO: 22
PROPRIETÀ 22
METODO 23
EVENTO 23
MESSAGGIO 23
MODELLO BASATO SU EVENTI 23
PROGETTO 23
USCIRE DA VISUAL BASIC 24
CREARE UN NUOVO PROGETTO 24
APRIRE UN PROGETTO ESISTENTE 25
SALVARE UN PROGETTO 26
TIPI DI OGGETTI DI UN PROGETTO 27
SALVARE UN OGGETTO DI UN PROGETTO 27
INSERIRE NUOVI FILE NEL PROGETTO 27

Thecla Informatica - Vietata la copia

RIMUOVERE UN FILE DA UN PROGETTO 28
ESEGUIRE UNA APPLICAZIONE 29
PER INTERROMPERE L'ESECUZIONE DI UN APPLICAZIONE 29
CREAZIONE DI UN FILE ESEGUIBILE 29
LE FASI DELLA PROGRAMMAZIONE 32
I CONTROLLI 32
AGGIUNGERE UN COMPONENTE ACTIVEX ED OGGETTI IN UN PROGETTO:33
RIMUOVERE I COMPONENTI DA UN PROGETTO 34
INTERFACCIA UTENTE 35
SELEZIONARE UN CONTROLLO 35
TOGLIERE LA SELEZIONE 35
CANCELLARE UN CONTROLLO DAL FORM 35
RIDIMENSIONARE UN CONTROLLO 36
SPOSTARE UN CONTROLLO 36
ALLINEARE I CONTROLLI SUL FORM 36
BLOCCARE UN CONTROLLO 36
SBLOCCARE UN CONTROLLO 37
CONTROLLO DEGLI OGGETTI TRAMITE PROPRIETÀ 37
COMPONENTE PICTUREBOX 39
COMPONENTE LABEL 40
COMPONENTE TEXTBOX 40
COMPONENTE FRAME 41
CONTROLLO COMMANDBUTTON 42
CONTROLLO CHECKBOX 44
CONTROLLO OPTIONBOX 45
CONTROLLO COMBOBOX 46
CONTROLLO LISTBOX 48
CONTROLLI HSCROLLBAR - VSCROLLBAR 50
CONTROLLO TIMER 51
CONTROLLO DRIVELISTBOX 52
CONTROLLO DIRLISTBOX 53
CONTROLLO FILELISTBOX 54
CONTROLLO SHAPE 56
CONTROLLO LINE 56
CONTROLLO IMAGE 57
CONTROLLO DATA 58
CONTROLLO OLE 60
EDITOR DEL MENU 61
BARRA DEGLI STRUMENTI 63
BARRA DEGLI STRUMENTI MANUALE 64
CONTROLLO BARRA DEGLI STRUMENTI 64
CONTROLLO BARRA DI STATO 66
CONTROLLO BARRA DI STATO 66

Thecla Informatica - Vietata la copia

CONVENZIONI RELATIVE ALLE DENOMINAZIONE DEGLI OGGETTI: 68
SCRITTURA DEL CODICE 69
PER VISUALIZZARE LA FINESTRA DEL CODICE: 69
PER VISUALIZZARE UN FORM (DALLA FINESTRA DEL CODICE) 71
ROUTINE AD EVENTI 72
NOZIONI FONDAMENTALI SULLA PROGRAMMAZIONE 72
MODULI DI CODICE 72
HELP DURANTE LA SCRITTURA DEL CODICE 73
REGOLE PRINCIPALI PER LA SCRITTURA DEL CODICE 75
VARIABILI 77
DICHIARAZIONE DI VARIABILE 77
TIPI DI DATI: 78
DICHIARAZIONI IMPLICITE 82
DICHIARAZIONI ESPLICITE 84
AREA DI VALIDITÀ DELLE VARIABILI 86
COSTANTI 88
AREA DI VALIDITÀ DELLE COSTANTI 90
VETTORI E MATRICI 91
ISTRUZIONE ISEMPTY 92
ISTRUZIONE ISNULL 92
ISTRUZIONE ISDATE 93
INSERIRE UN MODULO NEL PROGETTO 93
DEFINIZIONE DI ROUTINE 94
OPERATORI 97
ARTIMETICI 97
CONCATENAMENTO 101
DI CONFRONTO 102
LOGICI 104
FUNZIONE INPUTBOX 109
FUNZIONE MSGBOX 110
FUNZIONI DI CONVERSIONE DEL TIPO 112
FUNZIONI DI MANIPOLAZIONE DI STRINGHE 113
STRUTTURE DI DECISIONE 118
ISTRUZIONE IF THEN ELSE 118
ISTRUZIONE SELECT CASE END SELECT 120
CICLO FOR NEXT 121
CICLO DO WHILE LOOP 122
ORDINE DI TABULAZIONE ALL'INTERNO DI UN FORM 124
GESTIONE FORM 125
INSERIRE UN NUOVO FORM 125
IMPOSTARE IL FORM DI AVVIO DEL PROGETTO 128
POSIZIONARE IL FORM 129
ISTRUZIONE CALL 130

Thecla Informatica - Vietata la copia

METODO SETFOCUS 131
FUNZIONE DATE 132
FUNZIONE TIME 132
ISTRUZIONE NOW 132
ISTRUZIONE WEEKDAY 133
METODO ADDITEM 134
METODO REMOVEITEM 134
METODO CREATELINK 135
FUNZIONE LOADPICTURE 136
RECORSET 137
OGGETTO RECORSET DI TIPO TABELLA 137
OGGETTO RECORSET DI TIPO DYNASET 138
OGGETTO RECORSET SNAPSHOT 138
CREARE UNA VARIABILE DI TIPO RECORSET 138
CREARE UN RECORDSET DA UNA QUERY ESISTENTE 140
CREARE UN RECORDSET DA CODICE SQL 140
METODO MOVEFIRST 141
METODO MOVELAST 141
METODO MOVEPREVIOUS 141
METODO MOVENEXT 141
ESTRAZIONE DATI DAI RECORD 142
PROPRIETÀ BOF E PROPRIETÀ EOF 142
METODO ADDNEW 144
METODO DELETE 144
METODO EDIT 145
METODO SEEK 147
METODO CLOSE 148
ACCENNI A SQL 148

Thecla Informatica - Vietata la copia

Introduzione a Visual Basic 6.0

Visual Basic è un linguaggio di programmazione semplice e veloce ma al tempo stesso potente. Con Visual Basic 6.0 è infatti possibile costruire applicativi per sistemi operativi Microsoft Windows in breve tempo.
Le sue principali caratteristiche le si ritrova nel nome stesso del linguaggio:
Visual che significa Visuale. Infatti è possibile costruire interfaccia utente[1] tramite oggetti e non tramite righe e righe di codice.
Basic che è l'acronimo di Beginners All – Purpose Symbolic Instruction Code è il linguaggio più utilizzato e diffuso. Visual Basic è "nato " dal linguaggio Basic.
Visual Basic permette di realizzare progetti 32 bit.
Visual Basic è un linguaggio di programmazione che può essere acquistato singolarmente oppure è possibile acquistare Visual Studio 6.0 è che il tools di sviluppo fornito da Microsoft. Questo tools contiene i seguenti linguaggi di sviluppo:
Visual Basic
Visual C++
Visual Java
Visual Interdev
Visual FoxPro

Ci sono tre differenti versioni di Visual Basic:

Learning: permette di costruire applicazioni per Windows. Contiene i controlli intrinseci ed altri controlli. La documentazione è il testo Manuale di Riferimento del programmatore

Professional: Contiene la versione Learning ed altri controlli quali quelli per Internet ed il generatore di report di Crystal report). La documentazione è il Manuale di Riferimento del programmatore ed la Guida ai componenti ed il manuale di Crystal Report.

Enterprise::Contiene la versione Professional e strumenti per la gestione del database, sistema di controllo per le versioni ecc. La documentazione include quella Professional ed la Creazione di applicazioni cliente/server.

In questo corso si spiegheranno e si tratterà la versione di Visual Basic 6.0 professional.

Requisiti minimi per installare Visual Basic 6.0

Microsoft Windows 95 o NT 3.51 o superiori
Personal Computer 486 o superiori
Tastiera e mouse
Monitor VGA 600*800

[1] Definita anche GUI (Grafic User Interface)

Thecla Informatica - Vietata la copia

Lettore CD-ROM
Almeno 32 Mb di RAM
Almeno 100 Mb di disco fisso libero

Installazione di Visual Basic 6.0

- Inserire il CD-ROM nel lettore
- Attendere il caricamento automatico del programma di installazione
- Se non fosse attivo la proprietà AutoPlay
- Attivare il pulsante

- Selezionare il comando ESEGUI
- A video apparirà la seguente maschera

- Tramite il pulsante SFOGLIA selezionare il programma di installazione che si trova nella root del CD-ROM.
- Seguire le istruzioni.

Thecla Informatica: Vietata la copia

Avviare Visual Basic 6.0

Per avviare Visual Basic eseguire la seguente procedura:

- Selezionare il pulsante

- Attivare il comando PROGRAMMI
- Selezionare

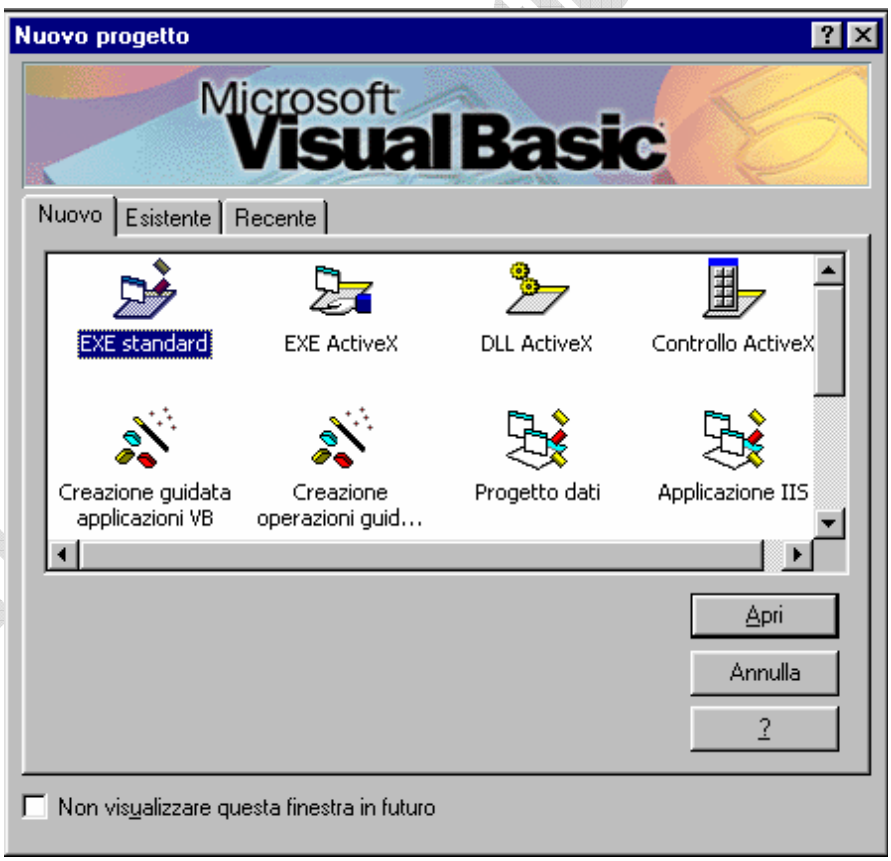
- Successivamente
- oppure

- Selezionare

Nel primo caso Visual Basic è stato installato da una versione di Visual Studio 6.0.

Videata di Visual Basic

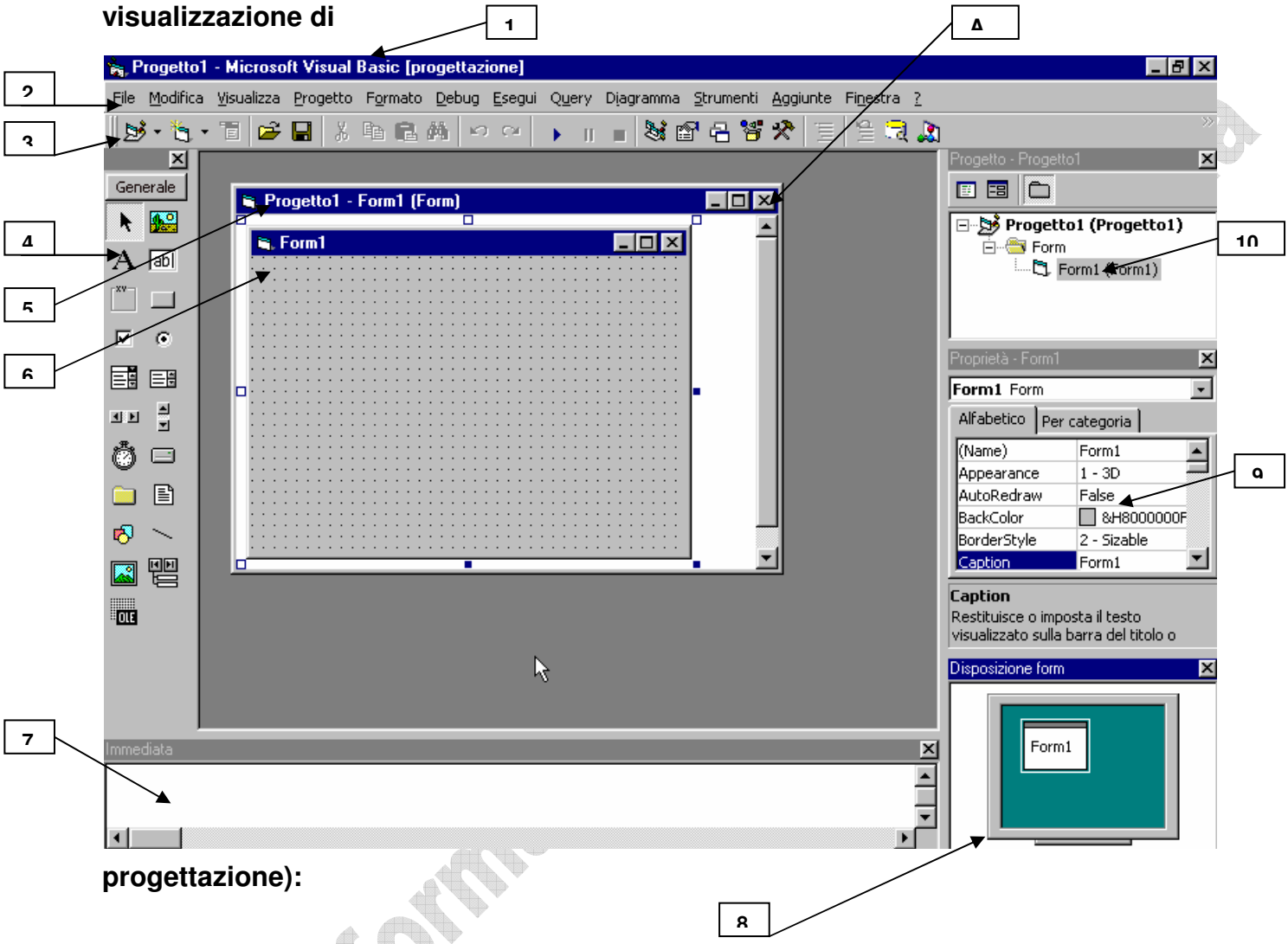
Quando si carica Visual Basic, la procedura innanzitutto chiede il tipo di progetto che si dovrà utilizzare (se nuovo o già esistente).



Thecla Informatica - Vietata la copia

- La scelta automatica è nuovo Progetto EXE Standard.
- Confermare con il pulsante Apri

A video appare la seguente schermata (che da ora in poi chiameremo visualizzazione di



progettazione):

1. **Barra del titolo**
2. **Menù dei Comandi**
3. **Barra degli Strumenti**
4. **Casella degli Strumenti**
5. **Finestra Progetto**
6. **Finestra di Progettazione Form**
7. **Finestra Immediata**
8. **Finestra Disposizione Form**
9. **Finestra Proprietà**
10. **Finestra Gestione Progetto**

Thecla Informatica

Barra del titolo:

Contiene il nome del linguaggio di programmazione attivo ed il nome del progetto. Nell'esempio il progetto si chiama Progetto1. Come gli applicativi Office Microsoft il nome dell'oggetto attivo è sempre un testo più un numero progressivo.

Barra dei Menù:

Contiene i menù ed i comandi che consentono di operare.
Il menù si aprono o con il clic del mouse oppure con il tasto ALT + lettera sottolineata.

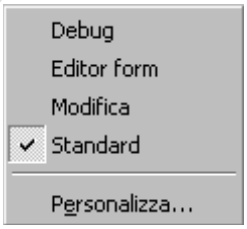
Barra Strumenti

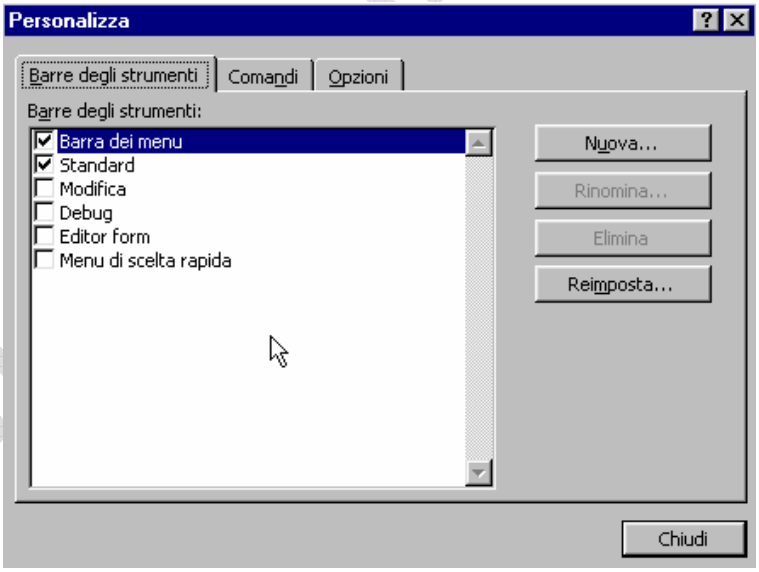
Velocizza le operazioni più frequenti
Come negli applicativi Office Microsoft è possibile personalizzare quest'area.

Aggiungere nuove Barre degli Strumenti:

Procedura:

- Aprire il menù Visualizza
- Selezionare il comando **Barra degli strumenti**
- Nel menù che si attiva:
- Selezionare il comando **PERSONALIZZA**
- A video apparirà la seguente maschera
- Attivare la barra degli strumenti che interessa





- Confermare con il pulsante CHIUDI.

VARIABILI

Quando si realizza un'applicazione, spesso questa deve elaborare dei dati. Ad esempio un'applicazione potrebbe voler calcolare il corrispondente valore di cambio tra Lire ed Euro. Questo programma dovrà ovviamente memorizzare il

Thecla Informatica - Vietata la copia

valore di Lire che l'utente inserirà per poterlo rielaborare. In Visual Basic, come in molti linguaggi di programmazione, si utilizzano le variabili per memorizzare temporaneamente dei valori.
La variabile è un contenitore o segnaposti in memoria che verranno utilizzati per valori sconosciuti.
Le variabili devono avere un nome**. E' necessario cioè assegnare una parola che ne identifica il valore. Inoltre è necessario impostare il** tipo di dati **che verranno inseriti in quello specifico segnaposto. I tipi di casi permettono di controllare la memorizzazione dei dati interna a Visual Basic. Il tipo di dati predefinito, utilizzato da Visual Basic è variant.**
Nota: La matrice è un oggetto che consente di memorizzare insiemi di variabili utilizzabili tramite indici. La matrice può essere utilizzata per memorizzare più informazioni relative ad oggetti legati tra loro. (Es. data di nascita ed età di un gruppo di persone).

Per utilizzare le variabili è necessario:
- ✍ Dichiarare la variabile
- ✍ Memorizzare un valore nella variabile
- ✍ Leggere il valore memorizzato nella variabile

Dichiarazione di Variabile

La dichiarazione di una variabile consente di fornire informazioni sulla variabile stessa. Per dichiarare una variabile è necessario sapere il nome con la quale la si vuole utilizzare ed è buona norma sapere il tipo di dati che verranno inseriti in quella specifica variabile.
Per dichiarare una variabile si utilizza la seguente istruzione:

Dim nomevariabile as tipo

Dove
Dim è l'istruzione che consente di dichiarare una variabile

Nomevariabile è il nome con cui si utilizzerà la variabile

As tipo è una proposizione facoltativa che consente di dichiarare il tipo di dati o di oggetto della variabile che si sta dichiarando.

I nomi delle variabili devono:
- **Iniziare con una lettera**
- **Non possono contenere punti (.) o caratteri per la dichiarazione dei tipi**
- **Al massimo possono essere lunghi 255 caratteri**
- **Devono essere univoci nell'area di validità della dichiarazione.**
- **E' buona norma iniziare il nomi della variabile con due o tre caratteri di abbreviazione che corrispondano al tipo di dati memorizzato nella variabile.**

Thecla Informatica - Vietata la copia

Tipi di dati:

Il tipo di dati di una variabile determina la modalità di memorizzazione dei bit che rappresentano i valori nella memoria del computer. Nel caso in cui non si imposti il tipo di dati viene usato il tipo di dati variant. Non è consigliato però impostare tutte le variabili tipo variant poiché non si ottimizza il codice e quindi l'applicazione non risulta efficiente.
I tipi di dati con i quali si può dichiarare una variabile sono i seguenti:
Byte
Integer
Long
Single
Currency
Double
String
Boolean
Date
Object (non gestito in questo manuale)
Variant
Nella tabella sottostante sono indicati le dimensioni di ingombro in byte, il valore di intervallo ed un semplice esempio

Tipo	Dimensioni	Intervallo	Esempio
Byte	1 byte	0-255	Dim bytA as byte bytA = 3
Integer	2 byte	-32.768 fino a 32.767	Dim intVal as integer IntVal = 23
Long	4 byte	-2.147.483.648 fino a 2.147.483.647	Dim lngVal as Long lngval = 268.000
Single	4 byte	-3.402823E38 fino a 3.402823E38	Dim sngVal as Single sngval = 145,555
Currency	8 byte	-922337203685477.5808 fino a 922337203685477.5807	Dim curVal as Currency curval = 4525655,45
Double	8 byte	-1.79769313486232D308 fino a 1.79769313486232D308	Dim dblVal as Double dblval = 3,1415926535
String	1 byte per carattere	0 fino a 65.535 caratteri	Dim strTesto as string strTesto = "Prova"
Boolean	2 byte	True o False	Dim blnVal as boolean bnlVal = False
Date	8 byte	Dal 01/01/100 al 31/12/9999	Dim dtmDataNascita as date dtmDataNascita = #15/05/65#
Object	4 byte	Oggetti dell'applicazione o applicazioni diverse	Dim objDb as Object set objDb = openDatabase(percorso)
Variant	16 byte (per i numeri), 22 byte + 1 byte per carattere (per le stringhe)	Tutti	Dim vntVal as variant vntVal = "Prova"

Thecla Informatica - Vietata la copia

Byte

Le variabili che includono dati binari debbono essere dichiarate come matrici.(Vedere Matrici più avanti). Il tipo Byte è senza segno. Rappresenta valori da 0 a 255 e non può rappresentare valori di segno negativo.

Integer

Il tipo di dati Integer occupa meno spazio in memoria rispetto a variant ed a long. E' quindi consigliato per i valori inclusi tra –32.678 e 32.767. Le operazioni con i valori interi sono eseguite più velocemente. Occupano meno spazio in memoria e sono molto utilizzati ad esempio per i cicli For –Next (vedere comando Ciclo For-Next).

Long

Il tipo di dati Long occupa meno spazio in memoria rispetto a variant. E' quindi consigliato per i valori inclusi tra –2.147.483.648 e 2.147.483.647. Le operazioni con i valori long sono eseguite più velocemente rispetto a quelle con valori variant o single o double.

Single

Si sfrutta con i numeri a virgola mobile.
Se si deve utilizzare un tipo di variabili con valori frazionari è necessario dichiararle o Single o Double o Currency. Single ha un intervallo abbastanza ampio (più ampio di Currency) ma più piccolo rispetto a Double.
E' possibile esprimere i single come mmmEeee dove
mmm è la mantissa
eee è l'esponente
Il valore positivo più alto è 3,402823E+38 vale a dire 3,4 per 10 elevato alla 38.

Currency

Il tipo currency supporta fino a quattro cifre a destra del separatore decimale e fino a quindici cifre a sinistra. Questo tipo di dati a virgola fissa adatto a calcoli di valuta.

Double

Si sfrutta con i numeri a virgola mobile.
Double memorizza un range di valori maggiori sia di single sia di Currency.
E' possibile esprimere i double come mmmDeee dove
mmm è la mantissa
eee è l'esponente
Il valore positivo più alto è 1,79769313486232D+308 in pratica 1,8 per 10 elevato alla 308.

Thecla Informatica - Vietata la copia

String

Le variabili che debbono contenere testi o numeri (ma sui quali non si devono eseguire calcoli).
In standard questo tipo di dato è una stringa a lunghezza variabile. Le sue dimensioni aumentano o diminuiscono secondo il valore assegnato.
E' possibile impostare anche variabili di tipo stringa a lunghezza fissa con la seguente sintassi:

dim nomevariabile as string * dimensioni

Dove
Dim è l'istruzione che consente di dichiarare una variabile
Nomevariabile è il nome con cui si utilizzerà la variabile
as string indica che è di tipo stringa
*** dimensioni indica che la variabile è a lunghezza fissa**
dimensioni indica il numero di caratteri.

Es. dim strTesto as string * 6
In questo caso la variabile memorizza sempre una stringa lunga 5.
Se s'impostasse una stringa più corta (es. Doc), automaticamente, a destra della stringa assegnata, si aggiungeranno spazi di riempimento. Gli spazi di riempimento possono essere eliminati con le funzioni TRIM e RTRIM.
Se s'impostasse una stringa più lunga (es. Montanari), automaticamente la stringa sarà troncata.

Boolean

Questo tipo di dati memorizza informazioni di tipo Vero/False, Sì/No e on/ff. Il valore predefinito è FALSE.

Date

Questo tipo di dati memorizza sia le date sia le ore. Nella conversione di tipi di dati numerici nel tipo Date, i valori a sinistra della parte decimale rappresentano la data, mentre i valori a destra rappresentano l'ora. Mezzanotte corrisponde a 0. Mezzogiorno corrisponde a 0,5. I numeri interi negativi rappresentano date precedenti al 30 dicembre 1899.

Object

Le variabili oggetto fanno riferimento agli oggetti dell'applicazione o di un'applicazione diversa. Per questo tipo di variabili l'assegnazione è eseguita tramite l'istruzione SET. Nella dichiarazioni d'oggetto è buona norma utilizzare una classe specifica anziché il tipo generico Object. I riferimenti a proprietà e metodi d'oggetti con tipi specifici possono essere risolti prima dell'esecuzione dell'applicazione; e di conseguenza l'applicazione risulta più veloce.
Le varie classi sono elencate nel visualizzatore oggetti. (non trattato in questo manuale).

Thecla Informatica - Vietata la copia

Variant

E' possibile memorizzare tutti i tipi di dati definiti dal sistema.. Quando si utilizzano le variabili di tipo variant i dati non vanno convertiti, poiché la conversione è automatica.
Nota. Se si utilizzano variabili di tipo variant in funzioni aritmetiche è prima necessario verificare che nella variabile sia memorizzato un valore numerico.
La variabile prima di essere assegnata ha valore Empty.
Per riassegnare ad una variabile Variant il valore Empty si utilizza la parola chiave Empty.

DICHIARAZIONI IMPLICITE

La dichiarazione implicita consente di non dichiarare la variabile.
Esempio:

```
Sub Prova
        StrVal = "Prova di scrittura"
        Msgbox (StrVal)
End sub
```

In questo esempio la variabile StrVal non è stata dichiarata.
Questo tipo di dichiarazione può essere più veloce, però può portare ad errori nel codice nel caso in cui il nome della variabile non venga scritto correttamente.

Thecla Informatica - Vietata la copia

Es.

```
Sub Prova
        StrVal = "Prova di scrittura"
        Msgbox (StrVal)
End sub


Sub Prova
        StrVal = "Prova di scrittura"
        Msgbox (StVal)
End sub
```

METODO SETFOCUS

Permette di spostare lo stato attivo su un determinato oggetto oppure un form.

Sintassi:

oggetto.setfocus

Dove
Oggetto: Obbligatorio. E' il componente che dovrà subire lo stato attivo.

Lo stato attivo può essere spostato solo su un form o un controllo visibile
È necessario che l'oggetto sia un oggetto Form**, o un controllo che può ricevere lo stato attivo.**
Lo stato attivo può essere spostato solo su un form o un controllo visibile.
Lo stato attivo non può essere spostato su un oggetto che ha la proprietà ENABLED uguale a False.
Poiché il form e i corrispondenti controlli non sono visibili fino al termine dell'evento Load del form, non è possibile utilizzare il metodo SetFocus per spostare lo stato attivo sul form mentre questo viene caricato, a meno che prima non sia stato utilizzato il metodo Show per visualizzare il form prima del termine della routine dell'evento Form_Load.

Thecla Informatica - Vietata la copia

FUNZIONE DATE

La funzione date restituisce la data del sistema.

Sintassi:

oggetto.proprietà = date()

oppure

nomevariabile = date()

FUNZIONE TIME

La funzione time restituisce l'ora del sistema.

Sintassi:

oggetto.proprietà = time()

oppure

nomevariabile = time()

ISTRUZIONE NOW

Restituisce un valore Variant che specifica la data e l'ora correnti in base all'impostazione dell'ora e della data di sistema.

Sintassi:

oggetto.proprietà = now()

oppure

nomevariabile = now()

ISTRUZIONE WEEKDAY

Restituisce un valore intero che contiene il numero corrispondente al giorno della settimana.

Sintassi:

Thecla Informatica - Vietata la copia

nomevariabile = Weekday(data, firstday)

dove
data: obbligatoria. Qualsiasi combinazione che rappresenti una data.
Firstday: non obbligatoria. Specifica il primo giorno della settimana

Valori restituiti
I valori restituiti dalla funzione Weekday **possono essere i seguenti:**

Costante	Valore	Descrizione
vbSunday	1	Domenica
vbMonday	2	Lunedì
vbTuesday	3	Martedì
vbWednesday	4	Mercoledì
vbThursday	5	Giovedì
vbFriday	6	Venerdì
vbSaturday	7	Sabato

Thecla Informatica - Vietata la copia

METODO ADDITEM

**Il metodo AddItem è utilizzato con i componenti LISTBOX e COMBOBOX.
Permette di caricare una lista di valori nella proprietà LIST da codice.
La sintassi è la seguente:**

oggetto.additem elemento, indice

**dove
oggetto: obbligatorio. Indica l'oggetto listbox o combobox.
Elemento: obbligatoria. E' una stringa che indica il valore che deve essere aggiunto.
Indice: Non obbligatoria. Indica la posizione nella quale inserire l'elemento. Se omessa (e la proprietà Sorted è impostata a True) inserisce l'elemento nella posizione corretta in relazione all'inserimento. Se omessa (e la proprietà Sorted è impostata a False) inserisce l'elemento alla fine dell'elenco.**

METODO REMOVEITEM

**Il metodo RemoveItem è utilizzato con i componenti LISTBOX e COMBOBOX.
Permette di eliminare un valore da una lista di valori nella proprietà LIST da codice.
La sintassi è la seguente:**

oggetto.removeitem indice

**dove
oggetto: obbligatorio. Indica l'oggetto listbox o combobox.
Indice: obbligatoria. Numero intero che rappresenta la posizione nell'oggetto dell'elemento o della riga da rimuovere.**

Thecla Informatica - Vietata la copia