

ضمیمه ۱

نصب نرم‌افزار Visual Basic 6.0

برای نصب زبان برنامه‌نویسی ویژوال بیسیک ابتدا باید دیسک‌های CD نرم‌افزار Microsoft Visual Studio 6 Enterprise Edition نگارش نهایی (Enterprise Edition) را تهیه کنید. در این جا چگونگی نصب این نرم‌افزار از روی دیسک CD، توضیح داده می‌شود. مدل‌های دیگر نیز تقریباً دارای مراحل نصب مشابهی هستند.


CD نصب نرم‌افزار را داخل درایو قرار دهید. اگر CD شما از نوع - Aut Run باشد بعد از قرار گرفتن CD در درایو مربوطه، برنامه نصب ویندوز به طور خودکار اجرا می‌شود و اولین کادر محاوره مربوط به نصب نرم‌افزار ظاهر می‌شود، اما در صورتی که برنامه نصب به طور خودکار اجرا نشد برنامه My Computer یا Window Explorer را اجرا کرده و در ریشه درایو CD خود، فایل Setup.exe را پیدا کنید و سپس روی آن دابل کلیک کنید. برنامه نصب اجرا شده و اولین کادر محاوره نصب نمایش داده می‌شود (شکل ۱).



شکل ۱

اکنون به ترتیب این مراحل را دنبال کنید:

- ۱ - ابتدا روی دکمه Next در اولین کادر محاوره کلیک کنید.
- ۲ - پس از مشاهده شکل ۲ روی دکمه انتخاب I accept the agreement و سپس روی دکمه Next کلیک نمایید.

در صورت تمایل می‌توانید در هر کادرمحاوره با کلیک روی دکمه Back به کادر
نکته 
محواره قبل بازگردید.



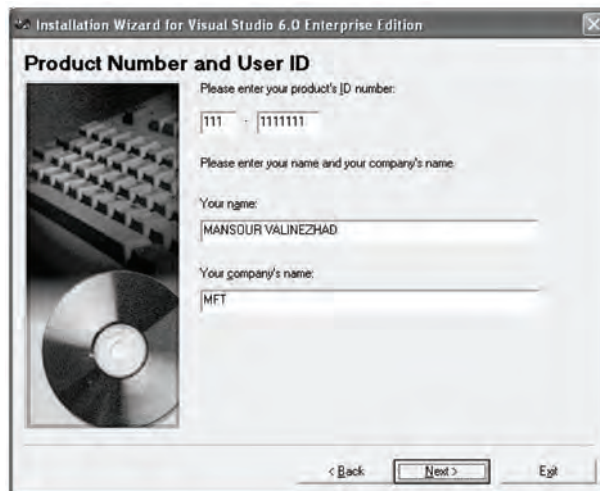
شکل ۲

۳ - در شکل ۳ در دو کادر متنی که در بخش: Please enter your product's ID number:

کادر محاوره قرار گرفته‌اند، شماره (ID) نرم‌افزار را تایپ کنید. سپس در کادر متن‌های

Your name: و : Your company's name اسمی دلخواه خود را بنویسید و در پایان مجدداً

روی دکمه Next کلیک کنید.



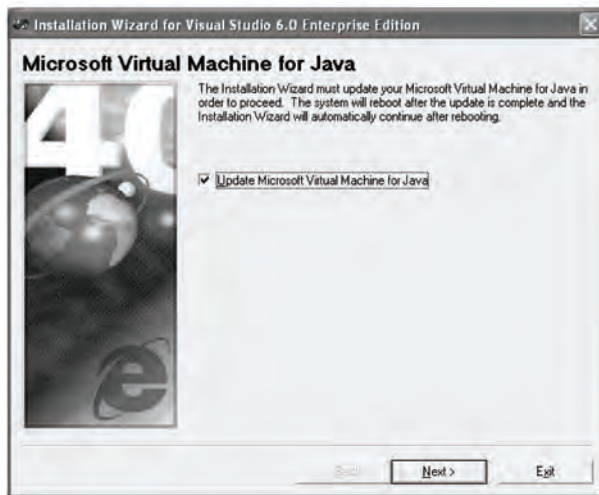
شکل ۳



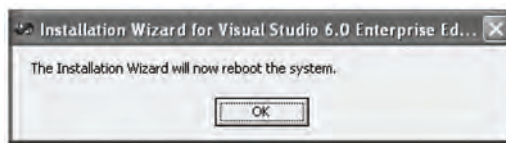
نکته در صورت نادرست بودن شماره ID امکان نصب برنامه وجود ندارد. اگر شماره ID ارایه شده در این جا قابل استفاده نباشد، می‌توانید از فایل‌های موجود در CD، شماره ID را پیدا کنید یا آن را از فروشنده درخواست کنید.

۴ - در کادرمحاوره شکل ۴ روی دکمه Next کلیک کنید.

۵ - در کادرمحاوره شکل ۵ روی دکمه OK کلیک کنید تا کامپیوتر مجدداً راه‌اندازی شود



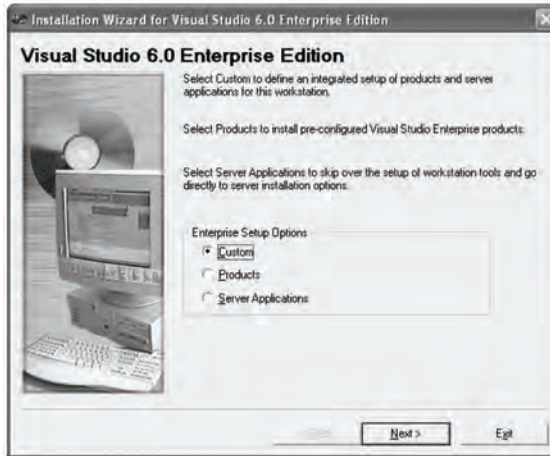
شکل ۴



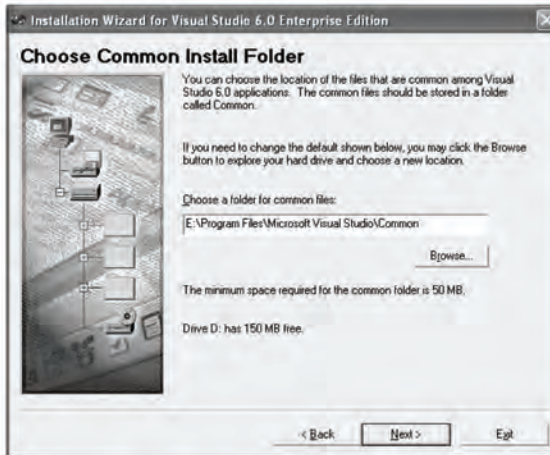
شکل ۵

۶ - پس از راه‌اندازی مجدد کامپیوتر کمی صبر کنید، کادرمحاوره‌ای که ظاهر می‌شود (شکل ۶) که روش‌های مختلفی را برای نصب نرم‌افزار پیشنهاد می‌کند، روی دکمه انتخاب Custom و سپس روی دکمه Next کلیک کنید.

۷ - در کادرمحاوره شکل ۷ می‌توانید مسیر نصب نرم‌افزار را انتخاب کنید. برای این کار مسیر دلخواه خود را در کادر متن تایپ کنید یا با استفاده از دکمه Browse این کار را انجام دهید.

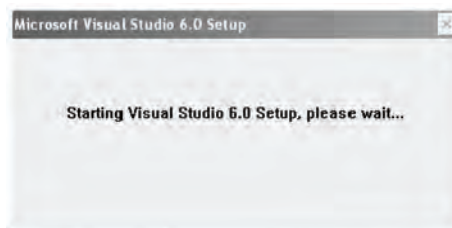


شکل ۶



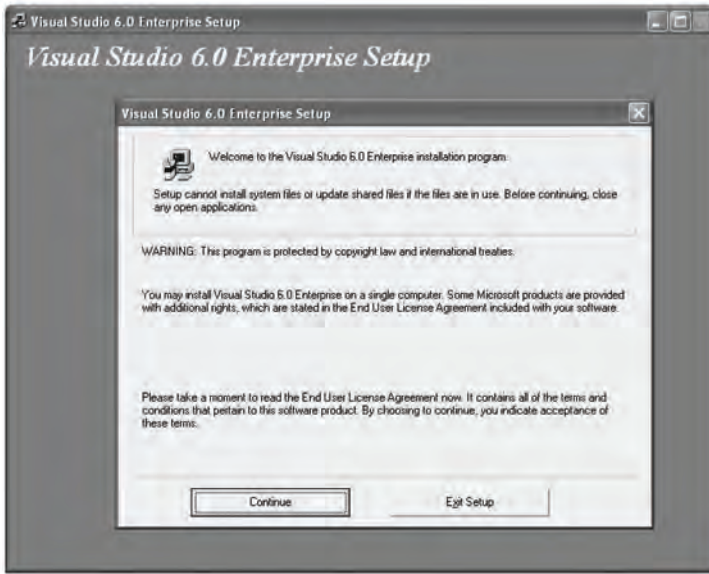
شکل ۷

در این کادر محاوره حداقل فضای مورد نیاز برای نصب نرم افزار و فضای آزاد درایوی که نرم افزار روی آن نصب می شود، قابل مشاهده است. توجه داشته باشید که فضای آزاد روی درایو مورد نظر کمتر از حداقل فضای پیشنهادی نباشد. در پایان این مرحله روی دکمه Next کلیک کنید تا کادر محاوره بعد (شکل ۸) ظاهر شود.



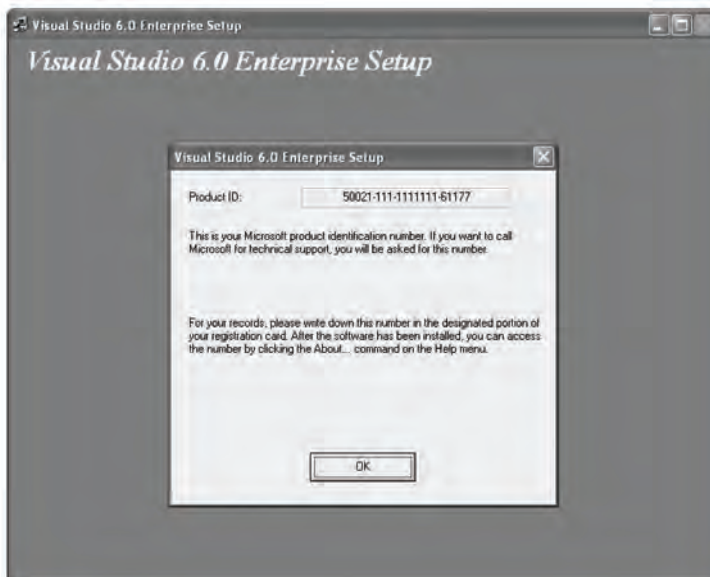
شکل ۸

۸ - در این مرحله، مدتی صبر کنید تا کادر محاوره بعدی نمایش داده شود.



شکل ۹

۹ - در این کادر محاوره (شکل ۹) روی دکمه Continue کلیک کنید تا کادر محاوره مطابق شکل ۱۰ مشاهده شود. در این کادر محاوره روی دکمه OK کلیک کنید تا روند نصب نرم‌افزار ادامه یابد.



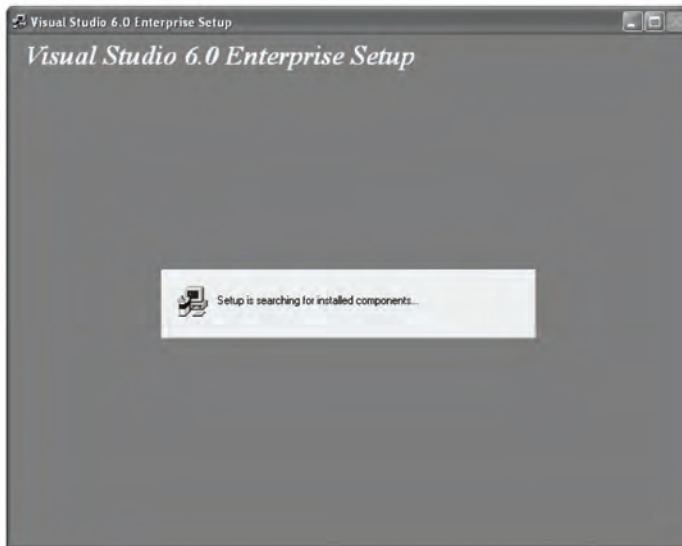
شکل ۱۰

نکته اگر مایل هستید عملیات نصب را خاتمه دهید، می‌توانید در کادر محاوره شکل ۹ روی دکمه Exit Setup کلیک کنید و در کادر محاوره‌ای که ظاهر می‌شود (شکل ۱۱) روی دکمه Exit Setup کلیک نمایید و اگر می‌خواهید مجدداً به کادر محاوره قبل (شکل ۹) بازگردید روی دکمه Resume Setup کلیک کنید.



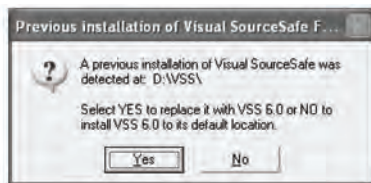
شکل ۱۱

پس از کلیک روی دکمه OK، پنجره نصب نرم‌افزار به صورت شکل ۱۲ در می‌آید. مدتی صبر کنید تا وارد مرحله بعد شوید.



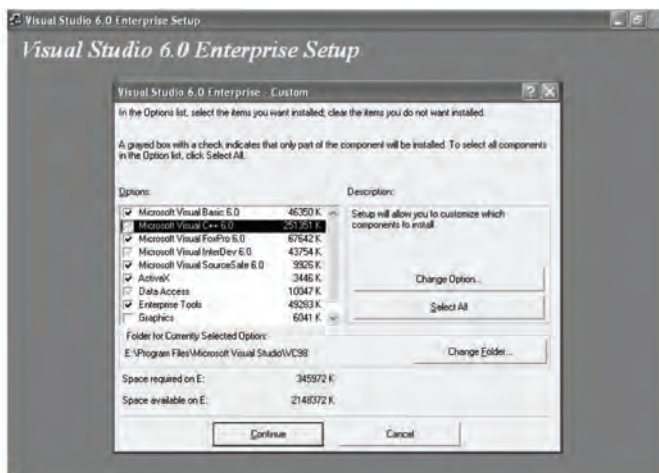
شکل ۱۲

۱۰ - اگر نسخه‌های دیگری از بسته نرم‌افزاری Visual Studio یا همین نگارش را قبلاً نصب کرده باشید، کادر محاوره‌ای مشابه شکل ۱۳ نمایش داده می‌شود. برای اطمینان بیشتر از ایجاد تغییرات در بخش‌های دیگر سیستم خود، در صورت مشاهده این کادر محاوره روی دکمه No کلیک کنید.



شکل ۱۳

۱۱- پس از انجام مرحله ۱۰ پنجره نصب برنامه مجدداً به صورت شکل ۱۲ مشاهده می‌شود، مدتی صبر کنید تا کادر محاوره بعدی مطابق شکل ۱۴ نمایش داده شود.



شکل ۱۴

۱۲- در این مرحله (شکل ۱۴) شما باید زبان‌های برنامه‌نویسی، مانند زبان‌های برنامه‌نویسی VB6، VC++ 6 و VFoxPro6 اجزای گرافیکی مانند انواع فایل‌های گرافیکی که در پروژه‌های برنامه‌نویسی می‌توان از آن‌ها استفاده نمود و سایر ابزارهای مورد نیاز خود را برای نصب انتخاب کنید. چون در این کتاب، زبان برنامه‌نویسی ویژوال بیسیک ۶ مورد نظر است؛ بنابراین با کلیک روی کادر علامت مربوط به موارد زیر آن‌ها را انتخاب کنید یا سایر موارد را در لیست این کادر محاوره از حالت انتخاب خارج کنید. کادرهای علامتی که باید انتخاب شوند عبارتند از:

Microsoft Visual Basic 6.0

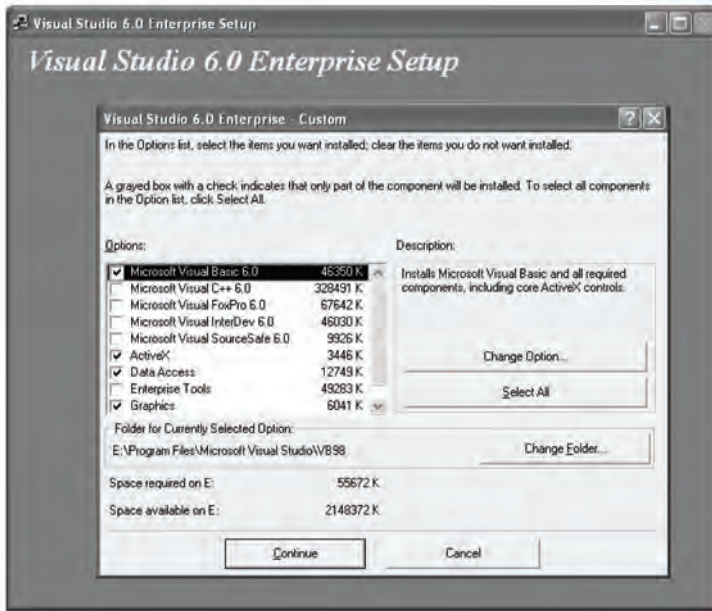
ActiveX

Data Access

Graphics

Tools

پس از انجام تنظیمات گفته شده، کادر محاوره مربوطه به صورت شکل ۱۵ در می آید



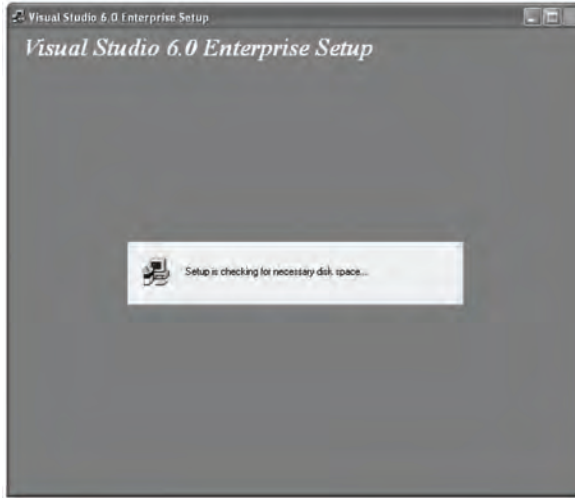
شکل ۱۵

در خاتمه این مرحله روی دکمه Continue کلیک کنید.

نکته • هر یک از گروه‌های موجود در لیست، به چند جزء کوچک‌تر تقسیم می‌شوند که اگر بخواهید در یک گروه فقط بعضی از اجزای آن را نصب کنید، می‌توانید از دکمه Change Option در کادر محاوره شکل ۱۴ استفاده کنید. البته پیشنهاد می‌شود از دستکاری این بخش‌ها بدون آگاهی کافی خودداری نمایید.

• اگر بخواهید تمام گروه‌ها را در لیست ۱۴ انتخاب کنید، روی دکمه Select All کلیک کنید.
• در صورتی که بخواهید مسیر نصب هر یک از گروه‌ها را روی دیسک سخت تغییر دهید می‌توانید از دکمه Change Folder... در کادر محاوره شکل ۱۴ استفاده کنید.

۱۳ - پس از انجام مرحله ۱۲، ابتدا برنامه نصب فضای لازم را برای نصب نرم‌افزار محاسبه می‌کند (شکل ۱۶) و در صورت وجود فضای آزاد به اندازه مناسب، مرحله انجام عملیات کپی فایل‌های نرم‌افزار مطابق شکل ۱۷ آغاز می‌شود.



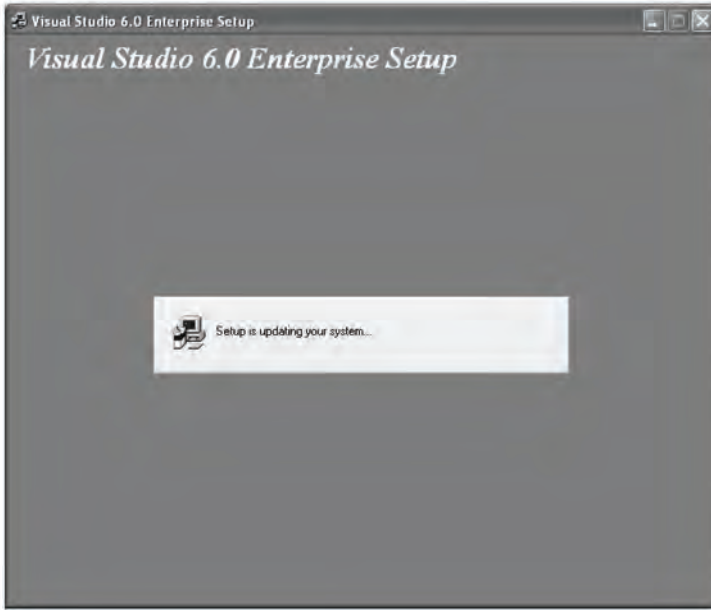
شکل ۱۶

۱۴ - پس از خاتمه مرحله ۱۳ کادر محاوره‌ای مطابق شکل ۱۸ نمایش داده می‌شود که بیانگر به روز کردن اطلاعات مورد نیاز سیستم است.

۱۵ - در مرحله بعد کادر محاوره‌ای مطابق شکل ۱۹ نمایش داده می‌شود، مبنی بر این که عمل نصب با موفقیت انجام شده است. پس از کلیک روی دکمه OK کادر محاوره دیگری مطابق شکل ۲۰ نمایش داده می‌شود، در این کادر محاوره روی دکمه Restart Windows کلیک کنید تا سیستم مجدداً راه‌اندازی شود.



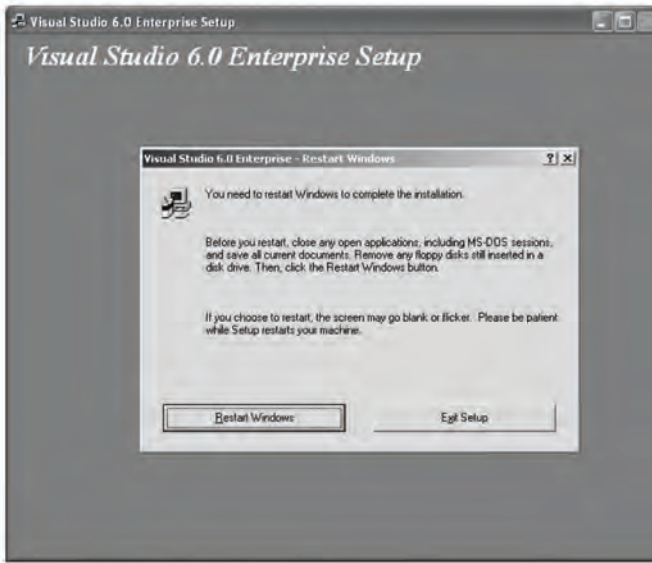
شکل ۱۷



شکل ۱۸

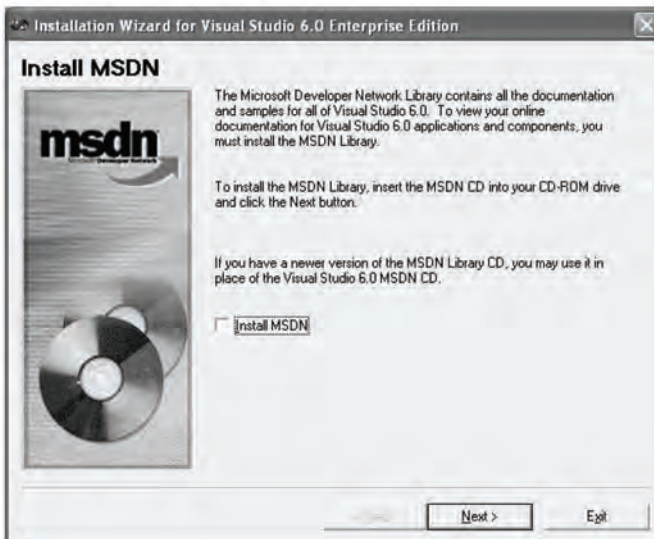


شکل ۱۹



شکل ۲۰

۱۶ - پس از راه‌اندازی مجدد سیستم کادر محاوره دیگری (شکل ۲۱) ظاهر می‌شود. اگر دیسک‌های CD مربوط به راهنمای MSDN مایکروسافت را در اختیار داشته باشید، می‌توانید به وسیله این بخش راهنمای مزبور را نصب کنید تا امکان استفاده از منوی Help ویژوال بیسیک فراهم شود. برای این کار کافی است دیسک‌های CD مربوطه را داخل درایو قرار داده و روی دکمه Next کلیک کنید.



شکل ۲۱

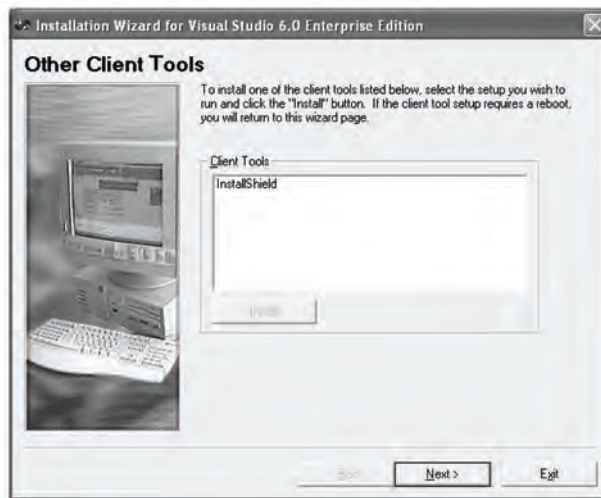
اما اگر دیسک‌های CD مربوطه را در اختیار نداشته باشید باید روی کادر علامت MSDN Install موجود در این کادر محاوره کلیک کنید تا از حالت انتخاب خارج شود؛ سپس روی دکمه Next کلیک کنید.

۱۷ - پس از انجام مرحله ۱۶ (در صورت عدم نصب MSDN) یک کادر پیغام مطابق شکل ۲۲ مبنی بر عدم امکان استفاده از منوی Help در برنامه ویژوال بیسیک نشان داده خواهد شد. روی دکمه Yes کلیک کنید.



شکل ۲۲

۱۸ - در این مرحله کادر محاوره‌ای مطابق شکل ۲۳ برای نصب بعضی از سرویس‌ها نمایش داده می‌شود که به آن‌ها نیازی ندارید. مجدداً روی دکمه Next کلیک کنید.



شکل ۲۳

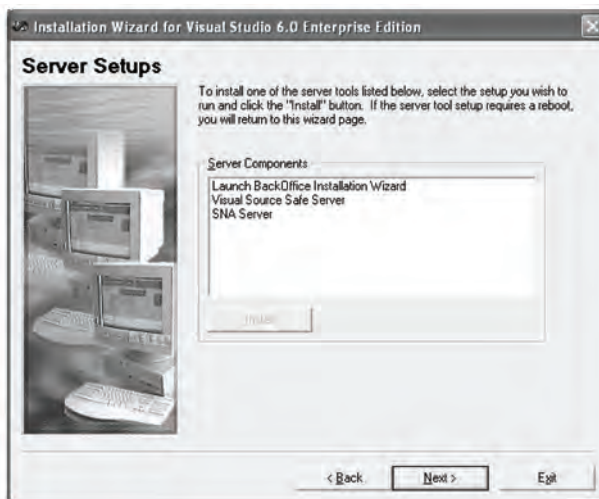
۱۹ - در این مرحله نیز در کادر محاوره شکل ۲۴ روی دکمه Next کلیک کنید، زیرا نیازی به نصب سرویس‌های موجود در لیست این کادر محاوره نخواهید داشت.

۲۰ - اگر مایل هستید در مرحله بعد اطلاعات خود را در سایت اینترنتی مربوطه ثبت کنید (در صورت عدم رعایت حقوق Copy Right امکان استفاده از این بخش میسر نیست).

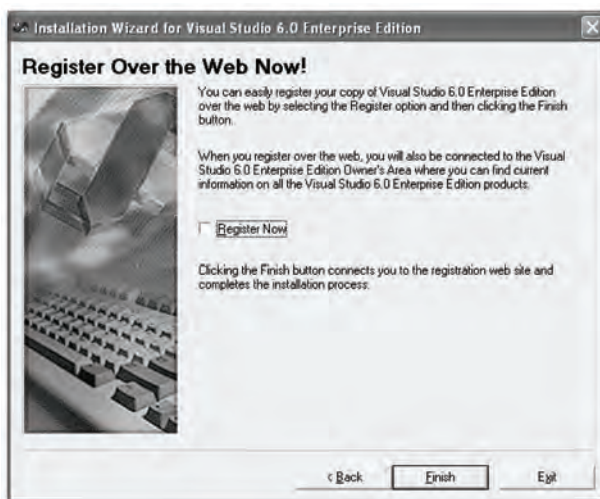
در پایان روی دکمه Finish کلیک کنید تا نصب نرم‌افزار خاتمه یابد سپس سیستم را restart نمایید.

مطابق شکل ۲۵ در کادر محاوره مربوطه، روی کادر علامت Register Now کلیک کنید تا از حالت انتخاب خارج شود، سپس روی دکمه Finish کلیک کنید.

۲۱- در این مرحله عملیات نصب نرم‌افزار خاتمه یافته است و شما می‌توانید زبان برنامه‌نویسی ویژوال بیسیک را از منوی Start ویندوز اجرا کنید.



شکل ۲۴



شکل ۲۵

ضمیمه ۲

کد کلیدها برای رویدادهای صفحه کلید

کدهای مربوط به کلیدهای حرفی در رویدادهای صفحه کلید

ثابت رشته‌ای	کد کلید	توضیح
vbKeyA	65	کلید A
vbKeyB	66	کلید B
vbKeyC	67	کلید C
vbKeyD	68	کلید D
vbKeyE	69	کلید E
vbKeyF	70	کلید F
vbKeyG	71	کلید G
vbKeyH	72	کلید H
vbKeyI	73	کلید I
vbKeyJ	74	کلید J
vbKeyK	75	کلید K
vbKeyL	76	کلید L
vbKeyM	77	کلید M
vbKeyN	78	کلید N
vbKeyO	79	کلید O
vbKeyP	80	کلید P
vbKeyQ	81	کلید Q
vbKeyR	82	کلید R
vbKeyS	83	کلید S
vbKeyT	84	کلید T
vbKeyU	85	کلید U
vbKeyV	86	کلید V
vbKeyW	87	کلید W
vbKeyX	88	کلید X
vbKeyY	89	کلید Y
vbKeyZ	90	کلید Z
vbKeySpace	32	کلید SPACEBAR
vbKeyBack	8	کلید BACKSPACE
vbKeyTab	9	کلید TAB
vbKeyReturn	13	کلید ENTER
vbKeyEscape	27	کلید ESC

کدهای مربوط به کلیدهای رقمی در رویدادهای صفحه کلید

ثابت رشته‌ای	کد کلید	توضیح
vbKey0	48	کلید 0
vbKey1	49	کلید 1
vbKey2	50	کلید 2
vbKey3	51	کلید 3
vbKey4	52	کلید 4
vbKey5	53	کلید 5
vbKey6	54	کلید 6
vbKey7	55	کلید 7
vbKey8	56	کلید 8
vbKey9	57	کلید 9

کدهای مربوط به کلیدهای رقمی بخش عددی صفحه کلید در

رویدادهای KeyUp و KeyDown

ثابت رشته‌ای	کد کلید	توضیح
vbKeyNumpad0	96	کلید 0
vbKeyNumpad1	97	کلید 1
vbKeyNumpad2	98	کلید 2
vbKeyNumpad3	99	کلید 3
vbKeyNumpad4	100	کلید 4
vbKeyNumpad5	101	کلید 5
vbKeyNumpad6	102	کلید 6
vbKeyNumpad7	103	کلید 7
vbKeyNumpad8	104	کلید 8
vbKeyNumpad9	105	کلید 9
vbKeyMultiply	106	کلید *
vbKeyAdd	107	کلید +
vbKeySubtract	109	کلید -
vbKeyDecimal	110	کلید 0
vbKeyDivide	111	کلید /

کدهای مربوط به کلیدهای تابعی در رویدادهای KeyDown و KeyUp

ثابت رشته‌ای	کد کلید	توضیح
vbKeyF1	112	کلید F1
vbKeyF2	113	کلید F2
vbKeyF3	114	کلید F3
vbKeyF4	115	کلید F4
vbKeyF5	116	کلید F5
vbKeyF6	117	کلید F6
vbKeyF7	118	کلید F7
vbKeyF8	119	کلید F8
vbKeyF9	120	کلید F9
vbKeyF10	121	کلید F10
vbKeyF11	122	کلید F11
vbKeyF12	123	کلید F12
vbKeyF13	124	کلید F13
vbKeyF14	125	کلید F14
vbKeyF15	126	کلید F15
vbKeyF16	127	کلید F16

کدهای مربوط به سایر کلیدها در رویدادهای KeyUp و KeyDown

ثابت رشته‌ای	کد کلید	توضیح
vbKeyShift	16	SHIFT key
vbKeyControl	17	CTRL key
vbKeyMenu	18	ALT key
vbKeyPause	19	PAUSE key
vbKeyCapital	20	CAPS LOCK key
vbKeyPageUp	33	PAGE UP key
vbKeyPageDown	34	PAGE DOWN key
vbKeyEnd	35	END key
vbKeyHome	36	HOME key
vbKeyLeft	37	LEFT ARROW key
vbKeyUp	38	UP ARROW key
vbKeyRight	39	RIGHT ARROW key
vbKeyDown	40	DOWN ARROW key
vbKeyInsert	45	INS key
vbKeyDelete	46	DEL key
vbKeyNumlock	144	NUM LOCK key

کدهای کاراکتر حرفی کوچک برای رویداد KeyPress

توضیح	کد کلید	ثابت رشته‌ای
کلید a	97	VbKeya
کلید b	98	VbKeyb
کلید c	99	VbKeyc
کلید d	100	vbKeyd
کلید e	101	vbKeye
کلید f	102	vbKeyf
کلید g	103	vbKeyg
کلید h	104	vbKeyh
کلید i	105	vbKeyi
کلید j	106	vbKeyj
کلید k	107	vbKeyk
کلید l	108	vbKeyl
کلید m	109	vbKeym
کلید n	110	vbKeyn
کلید o	111	vbKeyo
کلید p	112	vbKeyp
کلید q	113	vbKeyq
کلید r	114	vbKeyr
کلید s	115	vbKeys
کلید t	116	vbKeyt
کلید u	117	vbKeyu
کلید v	118	vbKeyv
کلید w	119	vbKeyw
کلید x	120	vbKeyx
کلید y	121	vbKeyy
کلید z	122	vbKeyz

کدهای سایر کاراکترها برای رویداد KeyPress

کد کلید	توضیح
60	کلید >
62	کلید <
44	کلید ,
63	کلید ?
47	کلید /
92	کلید \
124	کلید
58	کلید :
59	کلید ;
34	کلید »
39	کلید >
123	کلید }
125	کلید {
91	کلید]
93	کلید [
61	کلید =
95	Underline
40)
41	(
38	&
94	^
37	%
36	\$
35	#
64	@
33	!
126	~
96	`

مقادیر مربوط به آرگومان Shift در رویدادهای KeyUp و KeyDown

ثابت رشته‌ای	ثابت عددی	توضیح
vbShiftMask	1	در صورتی که کلید Shift فشرده شود.
vbCtrlMask	2	در صورتی که کلید Ctrl فشرده شود.
vbAltMask	4	در صورتی که کلید Alt فشرده شود.
vbShiftMask+vbCtrlMask	3	در صورتی که کلیدهای Shift+Ctrl فشرده شوند.
vbShiftMask+vbAltMask	5	در صورتی که کلیدهای Shift+Alt فشرده شوند.
vbCtrlMask+vbAltMask	6	در صورتی که کلیدهای Ctrl+Alt فشرده شوند.
+vbShiftMask+vbCtrlMask vbAltMask	7	در صورتی که کلیدهای Shift+Ctrl+Alt فشرده شوند.

ضمیمه ۳

کد و پیام‌های خطا در ویژوال بیسیک

Code	Message
3	Return without GoSub
5	Invalid procedure call
6	Overflow
7	Out of memory
9	Subscript out of range
10	This array is fixed or temporarily locked
11	Division by zero
13	Type mismatch
14	Out of string space
16	Expression too complex
17	Can't perform requested operation
18	User interrupt occurred
20	Resume without error
28	Out of stack space
35	Sub, Function, or Property not defined
47	Too many DLL application clients
48	Error in loading DLL
49	Bad DLL calling convention
51	Internal error
52	Bad file name or number
53	File not found
54	Bad file mode
55	File already open
57	Device I/O error
58	File already exists
59	Bad record length
61	Disk full
62	Input past end of file
63	Bad record number
67	Too many files

Code	Message
68	Device unavailable
70	Permission denied
71	Disk not ready
74	Can't rename with different drive
75	Path/File access error
76	Path not found
91	Object variable or With block variable not set
92	For loop not initialized
93	Invalid pattern string
94	Invalid use of Null
97	Can't call Friend procedure on an object that is not an instance of the defining class
98	A property or method call cannot include a reference to a private object, either as an argument or as a return value
298	System DLL could not be loaded
320	Can't use character device names in specified file names
321	Invalid file format
322	Can't create necessary temporary file
325	Invalid format in resource file
327	Data value named not found
328	Illegal parameter; can't write arrays
335	Could not access system registry
336	Component not correctly registered
337	Component not found
338	Component did not run correctly
360	Object already loaded
361	Can't load or unload this object
363	Control specified not found
364	Object was unloaded
365	Unable to unload within this context
368	The specified file is out of date. This program requires a later version
371	The specified object can't be used as an owner form for Show
380	Invalid property value
381	Invalid property-array index
382	Property Set can't be executed at run time
383	Property Set can't be used with a read-only property
385	Need property-array index
387	Property Set not permitted
393	Property Get can't be executed at run time
394	Property Get can't be executed on write-only property
400	Form already displayed; can't show modally
402	Code must close topmost modal form first
419	Permission to use object denied
422	Property not found
423	Property or method not found
424	Object required
425	Invalid object use
429	Component can't create object or return reference to this object

Code	Message
430	Class doesn't support Automation
432	File name or class name not found during Automation operation
438	Object doesn't support this property or method
440	Automation error
442	Connection to type library or object library for remote process has been lost
443	Automation object doesn't have a default value
445	Object doesn't support this action
446	Object doesn't support named arguments
447	Object doesn't support current locale setting
448	Named argument not found
449	Argument not optional or invalid property assignment
450	Wrong number of arguments or invalid property assignment
451	Object not a collection
452	Invalid ordinal
453	Specified not found
454	Code resource not found
455	Code resource lock error
457	This key is already associated with an element of this collection
458	Variable uses a type not supported in Visual Basic
459	This component doesn't support the set of events
460	Invalid Clipboard format
461	Method or data member not found
462	The remote server machine does not exist or is unavailable
463	Class not registered on local machine
480	Can't create AutoRedraw image
481	Invalid picture
482	Printer error
483	Printer driver does not support specified property
484	Problem getting printer information from the system. Make sure the printer is set up correctly
485	Invalid picture type
486	Can't print form image to this type of printer
520	Can't empty Clipboard
521	Can't open Clipboard
735	Can't save file to TEMP directory
744	Search text not found
746	Replacements too long
31001	Out of memory
31004	No object
31018	Class is not set
31027	Unable to activate object
31032	Unable to create embedded object
31036	Error saving to file
31037	Error loading from file

Invalid procedure call or argument (Error 5)

Some part of the call can't be completed. This error has the following causes and solutions:

An argument probably exceeds the range of permitted values. For example, the Sin function can only accept values within a certain range.

Positive arguments less than 2,147,483,648 are accepted, while 2,147,483,648 generates this error.

Check the ranges permitted for arguments.

This error can also occur if an attempt is made to call a procedure that isn't valid on the current platform.

For example, some procedures may only be valid for Microsoft Windows, or for the Macintosh, and so on.

Check platform-specific information about the procedure.

For additional information, select the item in question and press F1.

Overflow (Error6)

An overflow results when you try to make an assignment that exceeds the limitations of the target of the assignment.

This error has the following causes and solutions:

The result of an assignment, calculation, or data type conversion is too large to be represented within the range of values allowed for that type of variable.

Assign the value to a variable of a type that can hold a larger range of values.

An assignment to a property exceeds the maximum value the property can accept.

Make sure your assignment fits the range for the property to which it is made.

You attempt to use a number in a calculation, and that number is coerced into an integer, but the result is larger than an integer. For example:

Dim x As Long

```
x = 2000 * 365 ' Error: Overflow
```

To work around this situation, type the number, like this:

Dim x As Long

```
x = CLng(2000) * 365
```

For additional information, select the item in question and press F1

Out of memory (Error 7)

More memory was required than is available, or a 64K segment boundary was encountered.

This error has the following causes and solutions:

You have too many applications, documents, or source files open.

Close any unnecessary applications, documents, or source files that are open.

You have a module or procedure that's too large.

Break large modules or procedures into smaller ones. This doesn't save memory, but it can prevent hitting 64K segment boundaries.

You are running Microsoft Windows in standard mode.

Restart Microsoft Windows in enhanced mode.

You are running Microsoft Windows in enhanced mode, but have run out of virtual memory.

Increase virtual memory by freeing some disk space, or at least ensure that some space is available.

You have terminate-and-stay-resident programs running.

Eliminate terminate-and-stay-resident programs.

You have many device drivers loaded.

Eliminate unnecessary device drivers.

You have run out of space for Public variables.

Reduce the number of Public variables.

Subscript out of range (Error 9)

Elements of arrays and members of collections can only be accessed within their defined ranges. This error has the following causes and solutions:

You referenced a nonexistent array element.

The subscript may be larger or smaller than the range of possible subscripts, or the array may not have dimensions assigned at this point in the application.

Check the declaration of the array to verify its upper and lower bounds.

Use the UBound and LBound functions to condition array accesses if you're working with arrays that are redimensioned.

If the index is specified as a variable, check the spelling of the variable name.

You declared an array but didn't specify the number of elements. For example, the following code causes this error:

```
Dim MyArray() As Integer
```

```
MyArray(8) = 234 ' Causes Error 9.
```

Visual Basic doesn't implicitly dimension unspecified array ranges as 0 - 10. Instead, you must use Dim or ReDim to specify explicitly the number of elements in an array.

You referenced a nonexistent collection member.

Try using the For Each...Next construct instead of specifying index elements.

You used a shorthand form of subscript that implicitly specified an invalid element.

For example, when you use the ! operator with a collection, the ! implicitly specifies a key.

For example, object!keyname.

value is equivalent to object.item(keyname).value. In this case, an error is generated if keyname represents an invalid key in the collection.

To fix the error, use a valid key name or index for the collection.

Division by zero (Error 11)

Division by zero isn't possible. This error has the following cause and solution:

The value of an expression being used as a divisor is zero.

Check the spelling of variables in the expression. A misspelled variable name can implicitly create a numeric variable that is initialized to zero.

Check previous operations on variables in the expression, especially those passed into the procedure as arguments from other procedures.

Type mismatch (Error 13)

Visual Basic is able to convert and coerce many values to accomplish data type assignments that weren't possible in earlier versions. However, this error can still occur and has the following causes and solutions:

The variable or property isn't of the correct type. For example,

a variable that requires an integer value can't accept a string value unless the whole string can be recognized as an integer.

Try to make assignments only between compatible data types. For example, an Integer can always be assigned to a Long,

a Single can always be assigned to a Double, and any type (except a user-defined type) can be assigned to a Variant.

An object was passed to a procedure that is expecting a single property or value.

Pass the appropriate single property or call a method appropriate to the object.

A module or project name was used where an expression was expected, for example:

```
Debug.Print MyModule
```

Specify an expression that can be displayed.

You attempted to mix traditional Basic error handling with Variant values having the Error

subtype (10, vbError), for example:

```
Error CVErr(n)
```

To regenerate an error, you must map it to an intrinsic Visual Basic or a user-defined error, and then generate that error.

A CVErr value can't be converted to Date. For example:

```
MyVar = CDate(CVErr(9))
```

Use a Select Case statement or some similar construct to map the return of CVErr to such a value.

At run time, this error typically indicates that a Variant used in an expression has an incorrect subtype,

or a Variant containing an array appears in a Print # statement.

To print arrays, create a loop that displays each element individually.

Out of string space (Error 14)

Visual Basic permits you to use very large strings. However, the requirements of other programs and

the way you manipulate your strings may cause this error. This error has the following causes and solutions:

Expressions requiring that temporary strings be created for evaluation may cause this error. For example,

the following code causes an Out of string space error on some operating systems:

```
MyString = "Hello"
```

```
For Count = 1 To 100
```

```
    MyString = MyString & MyString
```

```
Next Count
```

Assign the string to a variable of another name.

Your system may have run out of memory, which prevented a string from being allocated.

Remove any unnecessary applications from memory to create more space.

Sub, Function, or Property not defined (Error 35)

A Sub, Function, or Property procedure must be defined to be called. This error has the following causes and solutions:

You misspelled the name of your procedure.

Check the spelling and correct it.

You tried to call a procedure from another project without explicitly adding a reference to that project in the References dialog box.

To add a reference

Display the References dialog box.

Find the name of the project containing the procedure you want to call. If the project name doesn't appear in the References dialog box, click the Browse button to search for it.

Click the check box to the left of the project name.

Click OK.

The specified procedure isn't visible to the calling procedure.

Procedures declared Private in one module can't be called from procedures outside the module.

If Option Private Module is in effect,

procedures in the module aren't available to other projects. Search to locate the procedure.

You declared a Windows dynamic-link library (DLL) routine, but the routine isn't in the specified library or code resource.

Check the ordinal (if you used one) or the name of the routine. Make sure your version of the

DLL is the correct one.

The routine may only exist in later versions of the DLL.

If the directory containing the wrong version precedes the directory containing the correct one in your path, the wrong DLL is accessed.

You gave the right DLL name, but it isn't the version that contains the specified function.

File not found (Error 53)

The file was not found where specified. This error has the following causes and solutions:

A statement, for example, Kill, Name, or Open, refers to a file that doesn't exist.

Check the spelling of the file name and the path specification.

An attempt has been made to call a procedure in a dynamic-link library (DLL) or Macintosh code resource,

but the library specified in the Lib clause of the Declare statement can't be found.

Check the spelling of the file name and the path specification.

In the development environment, this error occurs if you attempt to open a project or load a text file that doesn't exist.

Check the spelling of the project name or file name and the path specification.

File already exists (Error 58)

This error has the following causes and solutions:

This error occurs at run time when the new file name, for example, one specified in a Name statement, is identical to a file name that already exists.

Specify a new file name in the Name statement or delete the old file before specifying it in a Name statement.

You used the Save As command to save a currently loaded project, but the project name already exists.

Use a different project name if you don't want to replace the other project.

Disk full (Error 61)

This error has the following causes and solutions:

There isn't enough room on the disk for the completion of a Print #, Write #, or Close operation.

Move some files to another disk or delete some files.

There isn't enough room on the disk to create required files.

Move some files to another disk or delete some files.

Disk not ready (Error 71)

:This error has the following causes and solutions

There is no disk in the specified drive.

.Put a disk in the drive and retry the operation

The drive door of the specified drive is open.

Close the drive door and retry the operation.

For loop not initialized (Error 92)

For loop counters must be initialized. This error has the following cause and solution:

You jumped into the middle of a For...Next loop.

Remove the jump into the loop. Placing labels inside a For...Next loop isn't recommended.

Invalid pattern string (Error 93)

The pattern string specified in the Like operation of a search is invalid. This error has the

following cause and solution:

A common example of an invalid character list expression is [a-b , where the right bracket is missing.

Review the valid characters for list expressions.

Invalid use of Null (Error 94)

Null is a Variant subtype used to indicate that a data item contains no valid data. This error has the following cause and solution:

You are trying to obtain the value of a Variant variable or an expression that is Null. For example:

```
MyVar = Null
```

```
For Count = 1 To MyVar
```

```
...
```

```
Next Count
```

Make sure the variable contains a valid value.

The specified object can't be used as an owner form for Show() (Error 371)

You must use an appropriate object with the Show method.

Invalid property value (Error 380)

Most properties only accept values of a certain type, within a certain range. This error has the following cause and solution:

An inappropriate value has been assigned to a property.

See the property's Help topic to determine what types and range of values are appropriate for the property.

Property Set can't be executed at run time (Error 382)

It may not be possible to obtain a reference to a property at run time. This error has the following cause and solution:

You tried to get a reference to a property that's either read-only or write-only at run time. Since you can use a reference for both reading and writing, the property must provide run-time support for both operations for a reference to be obtained at run time.

Property Set can't be used with a read-only property (Error 383)

It may not be possible to obtain a reference to a property at run time. This error has the following cause and solution:

You tried to get a reference to a property that's read-only at run time. Since you can use a reference for both reading and writing, the property must provide run-time support for both operations for a reference to be obtained at run time. You can only use a Property Get with this property.

Property not found (Error 422)

Not all objects support the same set of properties. This error has the following cause and solution:

This object doesn't support the specified property. Check the spelling of the property name. Also, you may be trying to access something like a «text» property when the object actually supports a «caption» or some similarly named property. Check the object's documentation.

Property or method not found (Error 423)

The spelling of an object name must exactly match the definition in its object library. This error

You may have misspelled the name of the object. To see what properties and methods are defined for an object, display the Object Browser. Select the appropriate object library to view a list of available properties and methods.

Named argument not found (Error 448)

A named argument may not be used in a procedure invocation unless it appears in the procedure definition. This error has the following cause and solution:

You specified a named argument, but the procedure was not defined to accept an argument by that name.

Check the spelling of the argument name.

Code resource not found (Error 454)

This error can only occur on the Macintosh. This error has the following cause and solution:

A call was made to a procedure in a code resource, but the code resource could not be found.

Check to be sure the resource is available and properly referenced.

For additional information, select the item in question and press F1.

Out of memory (Error 31001)

Your system could not allocate or access enough memory or disk space for the specified operation.

فهرست منابع

۱- راهنمای MSDN

2- Microsoft Visual Basic 6 Programmer's Guide (Microsoft press)

