The Smartest Way To Get Certified™

MCSD
考前冲刺

The perfect supplement
to all certification
study guides

Over
1 Million
Certification
Insider Press
Guides in
Print!
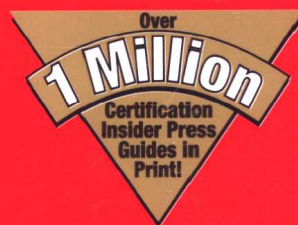
Visual Basic 6
Distributed

Exam 70-175

Microsoft
Certified
Solution
Developer

Includes
Adaptive Testing
Information

Michael Lane Thomas
and Dan Fox

MCSD
考前冲刺
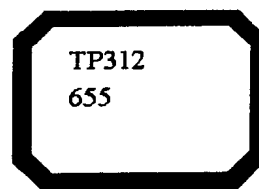
Microsoft
Visual Basic 6
Distributed

Microsoft
Certified
Solution
Developer

Michael Lane Thomas
and Dan Fox

Our readers tell us why
Certification Insider Press *is* #1

"The *MCSE Windows 95 Exam Cram* completely blew away a study guide bought from one of your competitors. This book is well-written and well-structured and has guaranteed my future commitment to this series of books for my future MCP and MCSE study materials."

> —Jose Cardoso

"The *Exam Crams* are by far the best studying companions! Since discovering your study guides, I have been able to cut my study time in half. They are indispensable for those in self-study programs. Thank you!"

> —Michael Dominguez, *MCSE*

"I scored 909 on my NT Server 4 exam and then 882 on my NT Workstation exam. Because of your books, I will achieve my MCSE in two months and my MCSE+I in three months. Thank you, Coriolis!"

> —Harry Yeh

"I just passed the Windows NT Workstation exam with a score of 901/1000, thanks to the *Exam Cram*. Since this was my first exam, I was a bit nervous about the structure of the exam. Thanks and keep up the great work!"

> —Tim O'Rourke, *Data General Corporation*

"I have four of the six courses completed for MCSE. I proudly say that your *Exam Cram* books aided me through each one of them. I highly recommend them for anyone working toward MCSE. The high quality upheld by Coriolis is remarkable."

> —Casey Napier, *MCP*

See for yourself why we're #1 and drop us a line at **craminfo@coriolis.com**— *we look forward to hearing about your testing successes!*

JS816 / 12

MCSD Visual Basic 6 Distributed

The Cram Sheet

This Cram Sheet contains the distilled, key facts about the MCSD Visual Basic 6 Distributed exam. Review this information last thing before you enter the test room, paying special attention to those areas where you feel you need the most review. You can transfer any of these facts onto a blank sheet of paper before beginning the exam.

CONFIGURING THE DEVELOPMENT ENVIRONMENT

1. Conditional compilation directives begin with the symbol #, and include **#If, #ElseIf, #Else,** and **#End If**.

2. **Win32** and **Win16** are known constants used with conditional compilation to determine the system on which the compiling is occurring.

3. Compile code before checking in VSS to ensure maintenance of a functional project.

CREATING WEB-BASED CLIENTS

4. IIS can be used for creating thin-client applications requiring little support.

5. Use the **URLFor** method to determine runtime URLs for Webitems.

CREATING AN ACTIVEX DOCUMENT

6. Navigate to the location of the VBD file to load an ActiveX document.

7. Use the **NavigateTo** method to move from document to document within code.

CREATING BUSINESS SERVICES

8. Unattended Execution does not allow user interface elements; therefore, they are not allowed.

9. **LogPath** and **LogMode** properties are used with COM servers set to Unattended Execution to determine the file location for error logging.

10. Unique properties of objects, such as ID numbers, are best modeled as properties.

11. Setting **MTSTransactionMode** to the following values has the listed effect:
    - **NotAnMTSObject** Ignores MTS; does not participate in any way.
    - **NoTransactions** Does not participate in transactions.
    - **UsesTransactions** Will use client transaction, but will not generate its own.
    - **Requires Transactions** Will use client transaction or create its own.

12. Out-of-process ActiveX EXEs with the **Instancing** property set to **MultiUse** allow many clients to create many objects from a single component.

13. The following **ObjectContext** object methods perform the listed functions:
    - **SetAbort** Method results unacceptable for completing transaction.
    - **SetComplete** Method results acceptable for completing transaction.
    - **IsCallerInRole** Check participation of component caller in the given or specified role.

- **IsSecurityEnabled** Check if authorization checking is invoked for component.

14. The <out of context> value for Watch variables indicates the value is currently out of the variables' defined scope.

15. The difference between the behavior of Break When Value Is True and Break When Value Changes requires an explicit Boolean statement as the variable definition. If the given expression is like **Name="Lizzy"**, Break When Value Is True should be used to ID a specific application variable value.

16. Watch variable expressions can be simple variable names, Boolean expressions, or any value expression. Simple Boolean statements, such as **Developer="Linda"**, are most commonly used with Break When Value Is True.

17. Specific variable values, object properties, or method results can be printed within the Immediate window.

18. The Local window provides the ability to view properties of the current form and all its constituent controls.

19. Distributed applications can be logically divided into services (User, Business, and Data).

20. The Visual Component Manager is used to publish information about components and other data so that it can be shared by multiple developers.

21. Security can be implemented at the business object or data services layers in Windows NT.

22. The conceptual design identifies the requirements of the application, whereas the physical design determines how the solution will be deployed.

23. To reuse the interface of an existing component, use the **Implements** keyword when creating a class module.

24. Populate elements in different tiers with data by filtering through the layers in order to maintain proper n-tier architecture.

25. The **CausesValidation** property triggers the **Validate** event on the control that is about to receive focus.

26. The **ConnectionString** and **RecordSource** properties of the ADO Data Control are necessary to allow the control to retrieve records.

27. A data environment can reference a stored procedure using a command.

28. The **AddressOf** operator is used to pass the address of a callback procedure to a Win32 API function.

29. Invoking the following error handlers will have the listed effect:
    - **On Error Goto** Followed by a line indicator, such as **ValidateError**, this causes the application to jump to that named line upon encountering an error.
    - **On Error Resume Next** Errors result in immediate execution of the next line of code.
    - **On F or Resume** Errors result in re-execution of the same line. Error is completely ignored.

30. When a procedure does not contain an error handler, errors are handled by either passing the error up to a calling procedure, if it exists, or issuing an application runtime error.

31. DispID and vtable interfaces are both forms of early binding.

32. The **HelpContextID** property of controls and the **HelpFile** property of the **App** object are used to implement user assistance.

33. You can use the **Add** method of the **Controls** collection to create controls dynamically.

34. **Property Get** procedures are used to provide read access to an object property.

35. **Property Let** or **Set** procedures are used to provide write access to an object property. Eliminating these property functions leaves a property read-only.

36. Events in class modules are defined using the **Event** keyword and fired using the **RaiseEvent** statement.

37. Know how to create a Property page for an ActiveX control and use these events:
    - **SelectionChanged** Indicates that a property page element has changed.
    - **ApplyChanges** Saves information changes within the control's property page.
38. Know the following **UserControl** events:
    - **WriteProperties** Used to save persisted component data to the **PropertyBag** object.
    - **ReadProperties** Used to read in persisted component data after having been previously saved in the **WriteProperties** event.
    - **InitProperties** Used when no persisted data exists. Initializes component data prior to any saving to persistent storage (**PropertyBag** object).
39. **Extender** properties are referenced inside an ActiveX control using the **Extender.propertyname** syntax.
40. Testing an ActiveX control is most efficiently done using a project group.
41. Make a control data bound by setting the **DataBindingBehavior** property of the **UserControl** object.
42. Choose Project|Components to add third-party controls to your toolbox

43. Review the three methods for creating an ADO recordset (using the **Open** method of the **Recordset** or the **Execute** method of the **Command** or **Connection** objects). In the following, assume **cnSQL** represents an ADO **Connection** object:
    - **Recordset** object **Open** method:

```
rsAuthors.Open _
   "SELECT * FROM authors", cnSQL
```

    - **Connection** object **Execute** method:

```
Set rsAuthors = _
   cnSQL.Execute _
   ("SELECT * FROM authors")
```

    - **Command** object **Execute** method:

```
With cmSQL
    .ActiveConnection = cnSQL
    .CommandText = _
        "SELECT * FROM authors"
    .CommandType = adCmdText
    .CommandTimeout = 10
End With

Set rsAuthors = cmSQL.Execute
```

44. ADO recordsets can be persisted to disk using the **Save** method.
45. Calling a stored procedure using an ADO **Command** object differs from a generic SQL statement by altering the value assigned to the **CommandType** property:

```
cmSQL.CommandType = adCmdStoredProc
```

46. Passing parameters to an ADO **Command** object is performed with the **CreateParameter** method of the **Command** object and the **Append** method of the **Parameters** collection:

```
Set prmSSN = _
   cmSQL.CreateParameter("@au_id", _
   adChar, adParamInput, _
   Len(strSSN), strSSN)
cmSQL.Parameters.Append prmSSN
```

47. Using the **Refresh** method of the **Parameters** collection and directly assigning values to indexed members of the **Parameters** collection can also be used for passing parameters to a stored procedure.
48. Executing a prepared statement using ADO is accomplished by setting the **Prepared** property of the **Command** object to **True** prior to executing. Subsequent uses of the **Command** object will utilize the prepared statement.
49. The **rdExecDirect** RDO option is used to bypass the creation of temporary stored procedures on the SQL database when executing statements.

50. Recordsets can be created using different types of cursors (**adOpenForwardOnly**, **adOpenStatic**, **adOpenDynamic**, and **adOpenKeyset**). Know when to use each type.

51. Benefits of using disconnected recordsets in ADO include:
    - They can be scrolled and modified without maintaining a persistent connection.
    - They free up data service components to be recycled within MTS.

52. Disconnected recordsets require that the **LockType** property be set to **adLockBatchOptimistic** and the **CursorLocation** be set to **adUseClient**.

53. Server-side cursors are best for applications that do not access all the records in a recordset or that need to see the changes made by other users.

54. Joining tables using the **INNER JOIN** syntax of the **SELECT** statement is recommended.

55. Data modifications are typically more efficient when using the **Execute** method of the **Connection** object and passing it the SQL.

56. Multiple errors can be returned in the **Errors** collection of the **Connection** object. The first error is also returned in the VB **Error** object.

57. Know how to create an RDO recordset asynchronously.

58. Use the **BeginTrans**, **CommitTrans**, and **RollbackTrans** methods of the ADO **Connection** object to create logical units of work.

DISTRIBUTING APPLICATIONS

59. The Package and Deployment Wizard can be used to create standard, Internet, and dependency packages.

60. When creating an Internet setup package the **CODEBASE** attribute of the **<OBJECT>** tag contains source and version information.

61. Remember to increment the version number in VB before compiling your application if you want existing clients to upgrade to the newer version.

62. Package and Deployment Wizard Setup packages can be deployed using the network, a floppy disk, or the Internet.

63. The server on which a COM component will be instantiated can be set using the DCOMCNFG utility.

64. The DCOM software is installed with Windows NT Service Pack 3 and must be installed separately on Windows 95 and 98.

65. MTS contains a package export utility that creates an executable for configuring client computers.

66. To configure a distributed application using the Package and Deployment Wizard you must generate and include a VBR file in the setup package.

67. In order to use DCOM, a COM object must have a reference in the HKEY_CLASSES_ROOT\AppID of the client computer.

MAINTAINING DISTRIBUTED APPLICATIONS

68. MTS does not implement load balancing as of version 2.0.

69. Checking the Project Compatibility option upon compilation does not allow the component to be compatible with existing clients.

70. Binary Compatibility should only be used when method and property signatures remain the same between versions.

71. Your code can create a COM object on a specific server using the **CreateObject** statement.

72. Dynamic load balancing can be implemented using a referral component and algorithms to dynamically select the appropriate server.

73. Static load balancing always instantiates components on specific servers for specific users.

I dedicate this book to my wife, Jennifer, who withstood all the difficulties I imposed on her in enduring the last two to three months. I know these projects have been incredibly taxing on you. Thank you sweetie for hanging in there. Also to my father, Larry, for believing in me and being close enough to let some of his drive rub off on me. And finally, to the Lord Jesus Christ, who seems to replace every closed door with a ballroom of opportunity just down the hall...time and time again.

—*Michael Lane Thomas*

Without the support and encouragement of my wife, Beth, and daughter, Laura, I could not have completed this project. They were both patient and understanding despite the long hours and "mental distance."
I love you both.

—*Dan Fox*

About The Authors

Michael Lane Thomas is a computer industry consultant and technical trainer who also spends his free time writing and speaking. He has spoken publicly on some of the hottest technologies to hit the industry, such as XML, SQL, and Y2K issues, and has been heard at Microsoft-sponsored national technical conferences, special interest groups, and on Kansas City's airwaves on 980KMBZ radio.

As an MCT, he currently teaches Microsoft Official Curriculum (MOC) courses, ranging from all the BackOffice products to the full range of Microsoft development and language technologies. Michael is certified to teach approximately 40 Microsoft courses, with more on the horizon, but he prefers to focus on the most recent development courses because "that's where the fun stuff is!"

When not teaching, Michael spends his time consulting as a SQL DBA, application developer, and general mentor. He prefers the challenge of designing, building, and developing complex intranet, three-tier Web applications, and advanced Web-based solutions. Michael currently holds the titles of MCP, MCP+I, MCSE, MCSE+I, MCT, MCSD, MCP+SB, MSS, and A+.

After graduating from the University of Kansas with a B.A. and B.S. in Mathematics, Michael has continued his traditional academic pursuits with a slow but steady climb towards his M.S. in Engineering Management from the University of Kansas. Michael is a former contributor and technical editor for the *Microsoft Certified Professional Magazine*, and author, contributor, and/or technical editor for six books to date.

Michael lives with his loving wife Jennifer, affectionately known by his nieces and nephew as "Racecar Jenny," for her sporty little blue ZX2. Residing in the western end of southern Johnson County, he occasionally longs for his days traveling around the world, but the smile of his wife Jennifer quickly reminds him why he is so lucky he finally came home.

Dan Fox is a consultant and instructor for Solutech, Inc. in Overland Park, Kansas. In addition to his consulting and teaching work with Visual Basic, Visual InterDev, SQL Server, and Powerbuilder, Dan is also the Database Line of Business Manager for Solutech. In his role at Solutech he also mentors consultants and designs intranet and client/server solutions.

After graduating from Iowa State University with a B.S. in Computer Science, Dan worked for Chevron in Houston, Texas, and the National Association of Insurance Commissioners in Kansas City before joining Solutech in 1995. A Microsoft Certified Solution Developer, Systems Engineer, and Trainer, Dan is a frequent contributor to the *Visual Basic Programmer's Journal* and has spoken at several Developer Days conferences.

Dan lives with his wife Beth and daughter Laura in Overland Park where they root for the Cubs and dream of a trip to the World Series at Wrigley Field.

Acknowledgments

Without a doubt, I would like to offer the greatest of thanks to Shari Jo and Toni. To Shari Jo, thanks for turning around and walking back up those stairs in San Jose. To Toni, thanks for the patience. I am sure you were convinced I was taking you for granted—nothing could be further from the truth. Thanks also to Ellen Strader, the copy editor, whose job it was to point out the spots in the manuscript where I was more apt to have been sleeping than writing consciously. I would be amiss not to thank the book designers April Nielsen and Jody Winkler for giving a little pizzazz to the project and also big thanks to my Production Coordinator Meg Turecek for providing the all important layout of the book. Thanks to my co-author, Dan Fox, for his help, input, and contribution, without which I would have been a great deal more stressed in getting the project completed.

—Michael Lane Thomas

I would like to say thanks to several of the great people at The Coriolis Group, including Shari Jo Hehr, the acquisitions editor; Toni Zuccarini, the project editor; Ellen Strader, the copy editor; designers April Nielsen and Jody Winkler; and the Production Coordinator Meg Turecek, who was responsible for the layout of the book. Each of them provided invaluable help throughout the entire process. My thanks to Toni especially, as she deserves much of the credit for keeping everything together and making the process go smoothly. I'd also like to thank Michael Lane Thomas for bringing me in on the project and for his contribution to making the book a success.

—Dan Fox

Introduction

Welcome to the *MCSD Visual Basic 6 Distributed Exam Cram*! This book aims to help you get ready to take—and pass—the Microsoft certification test numbered 70-175, "Designing and Implementing Distributed Applications with Microsoft Visual Basic 6.0." This introduction explains Microsoft's certification programs in general and talks about how the *Exam Cram* series can help you prepare for Microsoft's certification exams.

Exam Cram books help you understand and appreciate the subjects and materials you need to pass Microsoft certification exams. *Exam Cram* books are aimed strictly at test preparation and review. They do not teach you everything you need to know about a topic (such as all the trade-offs involved in using Active Server Pages or DHTML or the nitty-gritty technical details of the calling mechanisms used by DCOM). Instead, we (the authors) present and dissect the questions and problems we've found that you're likely to encounter on a test. We've worked from Microsoft's own training materials, preparation guides, and tests, as well as from actual experience implementing distributed applications. Our aim is to bring together as much information as possible about Microsoft certification exams.

Nevertheless, to completely prepare yourself for any Microsoft test, we recommend that you begin your studies with some classroom training, or that you pick up and read one of the many study guides available. We also strongly recommend that you install, configure, and fool around with the software or environment that you'll be tested on, because nothing beats hands-on experience and familiarity when it comes to understanding the questions you're likely to encounter on a certification test. Book learning is essential, but hands-on experience is the best teacher of all.

The Microsoft Certified Professional (MCP) Program

The MCP program currently includes eight separate tracks, each of which boasts its own special acronym (as a would-be certificant, you need to have a high tolerance for alphabet soup of all kinds).

➤ **MCP (Microsoft Certified Professional)** This is the least prestigious of all the certification tracks from Microsoft. Passing any of the major Microsoft exams (except the Networking Essentials exam) qualifies an individual for MCP credentials. Individuals can demonstrate proficiency with additional Microsoft products by passing additional certification exams.

➤ **MCP+I (Microsoft Certified Professional + Internet)** This midlevel certification is attained by completing three core exams: Windows NT Server 4, TCP/IP, and Internet Information Server (3 or 4).

➤ **MCP+SB (Microsoft Certified Professional + Site Building)** This certification program is designed for individuals who are planning, building, managing, and maintaining Web sites. Individuals with the MCP+SB credential will have demonstrated the ability to develop Web sites that include multimedia and searchable content and Web sites that connect to and communicate with a back-end database. It requires passing two of the following three exams: "Designing and Implementing Commerce Solutions with Microsoft Site Server 3.0, Commerce Edition," "Designing and Implementing Web Sites with Microsoft FrontPage 98," and "Designing and Implementing Web Solutions with Microsoft Visual InterDev 6.0."

➤ **MCSD (Microsoft Certified Solution Developer)** The MCSD credential reflects the skills required to create multitier, distributed, and COM-based solutions, in addition to desktop and Internet applications, using new technologies. To obtain an MCSD, an individual must demonstrate the ability to analyze and interpret user requirements; select and integrate products, platforms, tools, and technologies; design and implement code and customize applications; and perform necessary software tests and quality assurance operations.

To become an MCSD, you must pass a total of four exams: three core exams and one elective exam. The required core exam is "Analyzing Requirements and Defining Solution Architectures." Each candidate must also choose one of these two desktop application exams—"Designing and Implementing Desktop Applications with Microsoft Visual C++ 6.0" or "Designing and Implementing Desktop Applications with Visual Basic 6.0"—plus one of these two distributed application exams— "Designing and Implementing Distributed Applications with Microsoft Visual C++ 6.0" or "Designing and Implementing Distributed Applications with Microsoft Visual Basic 6.0." This book is devoted to the Visual Basic 6 Distributed exam, the last in that list.

Table 1   MCSD Requirements*

Core

Choose 1 from the desktop applications development group	
Exam 70-016	Designing and Implementing Desktop Applications with Microsoft Visual C++ 6.0
Exam 70-176	Designing and Implementing Desktop Applications with Microsoft Visual Basic 6.0
Choose 1 from the distributed applications development group	
Exam 70-015	Designing and Implementing Distributed Applications with Microsoft Visual C++ 6.0
Exam 70-175	Designing and Implementing Distributed Applications with Microsoft Visual Basic 6.0
This solution architecture exam is required	
Exam 70-100	Analyzing Requirements and Defining Solution Architectures

Elective

Choose 1 from this group	
Exam 70-015	Designing and Implementing Distributed Applications with Microsoft Visual C++ 6.0
Exam 70-016	Designing and Implementing Desktop Applications with Microsoft Visual C++ 6.0
Exam 70-029	Designing and Implementing Databases with Microsoft SQL Server 7.0
Exam 70-024	Developing Applications with C++ Using the Microsoft Foundation Class Library
Exam 70-025	Implementing OLE in Microsoft Foundation Class Applications
Exam 70-055	Designing and Implementing Web Sites with Microsoft FrontPage 98
Exam 70-057	Designing and Implementing Commerce Solutions with Microsoft Site Server 3.0, Commerce Edition
Exam 70-165	Developing Applications with Microsoft Visual Basic 5.0
	OR
Exam 70-175	Designing and Implementing Distributed Applications with Microsoft Visual Basic 6.0
	OR
Exam 70-176	Designing and Implementing Desktop Applications with Microsoft Visual Basic 6.0
Exam 70-069	Application Development with Microsoft Access for Windows 95 and the Microsoft Access Developer's Toolkit
Exam 70-091	Designing and Implementing Solutions with Microsoft Office 2000 and Microsoft Visual Basic for Applications
Exam 70-152	Designing and Implementing Web Solutions with Microsoft Visual InterDev 6.0

* This is not a complete listing—you can still be tested on some earlier versions of these products. However, we have tried to include the most recent versions so that you may test on these versions and thus be certified longer. We have not included any tests that are scheduled to be retired.

The MCSD program is being expanded to include FoxPro and Visual J++. However, these tests are not yet available and no test numbers have been assigned.

Core exams that can also be used as elective exams can be counted only once toward certification. The same test cannot be used as both a core and elective exam.

Elective exams cover specific Microsoft applications and languages, including Visual Basic, C++, the Microsoft Foundation Classes, Access, SQL Server, Excel, and more. If you're on your way to becoming an MCSD and have already taken some exams, visit **www.microsoft.com/ train_cert/** for information about how to proceed with your MCSD certification under this new track. Table 1 shows the requirements for the MCSD certification.

➤ **MCSE (Microsoft Certified Systems Engineer)** Anyone who has a current MCSE is warranted to possess a high level of expertise with Windows NT (version 3.51 or 4) and other Microsoft operating systems and products. This credential is designed to prepare individuals to plan, implement, maintain, and support information systems and networks built around Microsoft Windows NT and its BackOffice family of products.

To obtain an MCSE, an individual must pass four core operating system exams, plus two elective exams. The operating system exams require individuals to demonstrate competence with desktop and server operating systems and with networking components.

You must pass at least two Windows NT-related exams to obtain an MCSE: "Implementing and Supporting Microsoft Windows NT Server" (version 3.51 or 4) and "Implementing and Supporting Microsoft Windows NT Server in the Enterprise" (version 3.51 or 4). These tests are intended to indicate an individual's knowledge of Windows NT in smaller, simpler networks and in larger, more complex, and heterogeneous networks, respectively.

You must pass two additional tests as well. These tests relate to networking and desktop operating systems. At present, the networking requirement can be satisfied only by passing the Networking Essentials test. The desktop operating system test can be satisfied by passing a Windows 95, Windows NT Workstation (the version must match whichever core NT curriculum you are pursuing), or Windows 98 test.

The two remaining exams are elective exams. An elective exam may fall in any number of subject or product areas, primarily BackOffice components. These include tests on Internet Explorer 4, SQL Server, IIS, SNA Server, Exchange Server, Systems Management Server, and the like. However, it's also possible to test out on electives by taking advanced networking tests such as "Internetworking with Microsoft TCP/IP on Microsoft Windows NT" (but here again, the version of Windows NT involved must match the version for the core requirements taken).

Whatever mix of tests is completed toward MCSE certification, individuals must pass six tests to meet the MCSE requirements. It's not uncommon for the entire process to take a year or so, and many individuals find that they must take a test more than once to pass. Our primary goal with the *Exam Cram* series is to make it possible, given proper study and preparation, to pass all Microsoft certification tests on the first try.

➤ **MCSE+Internet (Microsoft Certified Systems Engineer + Internet)** This is a newer Microsoft certification and focuses not just on Microsoft operating systems, but also on Microsoft's Internet servers and TCP/IP.

To obtain this certification, an individual must pass seven core exams, plus two elective exams. The core exams include not only the server operating systems (NT Server and Server in the Enterprise) and a desktop operating system (Windows 95, Windows 98, or Windows NT Workstation), but also include Networking Essentials, TCP/IP, Internet Information Server, and the Internet Explorer Administration Kit (IEAK).

The two remaining exams are electives. These elective exams can be in any of four product areas: SQL Server, SNA Server, Exchange Server, and Proxy Server.

➤ **MCDBA (Microsoft Certified Database Administrator)** The MCDBA credential reflects the skills required to implement and administer Microsoft SQL Server databases. To obtain an MCDBA, an individual must demonstrate the ability to derive physical database designs, develop logical data models, create physical databases, create data services by using Transact-SQL, manage and maintain databases, configure and manage security, monitor and optimize databases, and install and configure Microsoft SQL Server.

To become an MCDBA, you must pass a total of five exams: four core exams and one elective exam. The required core exams are "Administering Microsoft SQL Server 7.0," "Designing and Implementing Databases with Microsoft SQL Server 7.0," "Implementing and Supporting Microsoft Windows NT Server 4.0," and "Implementing and Supporting Microsoft Windows NT Server 4.0 in the Enterprise."

The elective exams that you can choose from cover specific uses of SQL Server and include "Designing and Implementing Distributed Applications with Visual Basic 6.0," "Designing and Implementing Distributed Applications with Visual C++ 6.0," "Designing and Implementing Data Warehouses with Microsoft SQL Server 7.0 and Microsoft Decision Support Services 1.0," and two exams that relate to NT "Internetworking