Chapter 9

Using Management Studio Express

In response to many user requests (and complaints), Microsoft reengineered the way you manage and administer tasks in SQL Server Express. A single, consolidated tool—Microsoft SQL Server Management Studio Express (Management Studio)—allows you to access database objects and tools. It really has no predecessor, because Microsoft Database Engine (MSDE) used Microsoft Access's user interface to interact with its SQL Server tables and objects. If you're a database administrator, Management Studio is where you'll spend nearly all your time. This is an extremely rich and powerful application, and you should become familiar with it.

In this chapter, you'll get an overview of how to use Management Studio. We'll start by discussing the general framework that holds Management Studio. Then, we'll look at the objects and tasks within Management Studio. If you're familiar with SQL Server 2005 and SQL Server Management Studio, you'll find the two management tools similar (but not equal).

TIP To download and install Management Studio, visit the following page: http://www.microsoft.com/downloads/details.aspx?FamilyID=82afbd59-57a4-455e-a2d6-1d4c98d40f6e&DisplayLang=en. Fortunately, the process is straightforward.

Introduction to Microsoft SQL Server Management Studio Express

To launch Management Studio, click Start ➢ All Programs, ➢ Microsoft SQL Server 2005 ➢ SQL Server Management Studio Express. Click Connect Using Windows Authentication to open Management Studio's default view, which consists of two windows (see Figure 9.1):

- Object Explorer
- Summary

If you're familiar with Visual Studio (VS), the layout and feel of Management Studio is similar. The VS interface is slowly becoming the environment of choice in Microsoft development applications. This is primarily the case because the design maximizes visibility. You'll find that the tool windows are highly customizable and can be configured to maximize development and management workspace.

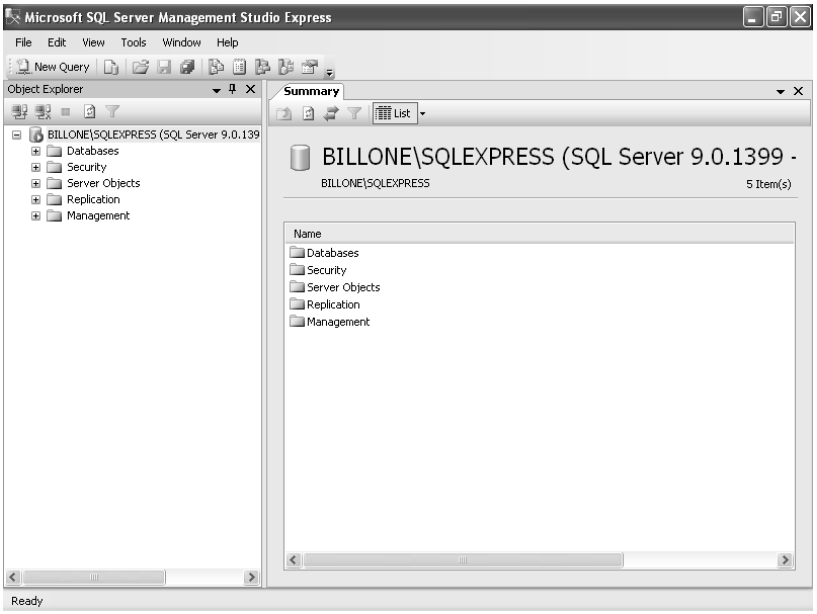


FIGURE 9.1
Default view of SQL Server Management Studio

Through some simple customization, you can readily access the tools and windows you use frequently as well as control how much space you want to allocate to different information and tasks. There are a variety of ways to increase your editing space:

- All the windows can be moved to different locations.
- Most windows can be undocked and dragged out of the Management Studio frame—a valuable attribute when you're using more than one monitor.
- All the windows have an Auto Hide feature that lets you reduce the window to a tab within a bar on the border of the window. When you place the cursor over one of these tabs, the underlying window reveals itself. (You can toggle Auto Hide for a window by clicking the Auto Hide button, represented by a pushpin, in the upper-right corner of the window. You can also choose Window ➢ Auto Hide All.)
- You can configure some components in either tabbed mode (components appear as tabs in the same docking location) or multiple document interface (MDI) mode (each document has its own window). To configure this feature, choose Tools ➢ Options ➢ Environment ➢ General. Then, specify Tabbed Documents (the default) or MDI Documents.

One other feature worth mentioning is that the tools in Management Studio were designed to work together. For instance, you can register a server with Object Explorer or open a SQL Editor window while connected to a specific database. This intertwining of access and functions makes Management Studio powerful and easy to use. With all that said, let's walk through the key windows of Management Studio.

The Main Windows

The windows you'll use most often in Management Studio are Object Explorer and Summary (shown earlier in Figure 9.1). To the left of the screen, Object Explorer provides a treeview of your server and database objects. The Summary window is kind of a placeholder that serves a number of purposes, such as displaying messages and serving as a text editor for code. You can also interact and modify the database objects via the Summary window.

The Object Explorer Window

Object Explorer connects to SQL Server instances. The capabilities of Object Explorer vary depending on the type of server, but they include development features for databases and management features for all server types. In this chapter, we'll focus on the features associated with SQL Server Express instances.

Connecting to a Server Using Object Explorer

Five icons appear on the Object Explorer toolbar (the small toolbar inside the Object Explorer window). Click Connect Object Explorer to display the Connect To Server dialog box, shown in Figure 9.2. To connect, you must provide at least the name of the server and the correct authentication information. You can also configure additional options by clicking the Options button in the Connect To Server dialog box. (The Connect To Server dialog box retains the last-used settings. Any new connections will also use these settings.) For the most part, this step is unnecessary unless you're trying to connect to a second instance of SQL Server Express.

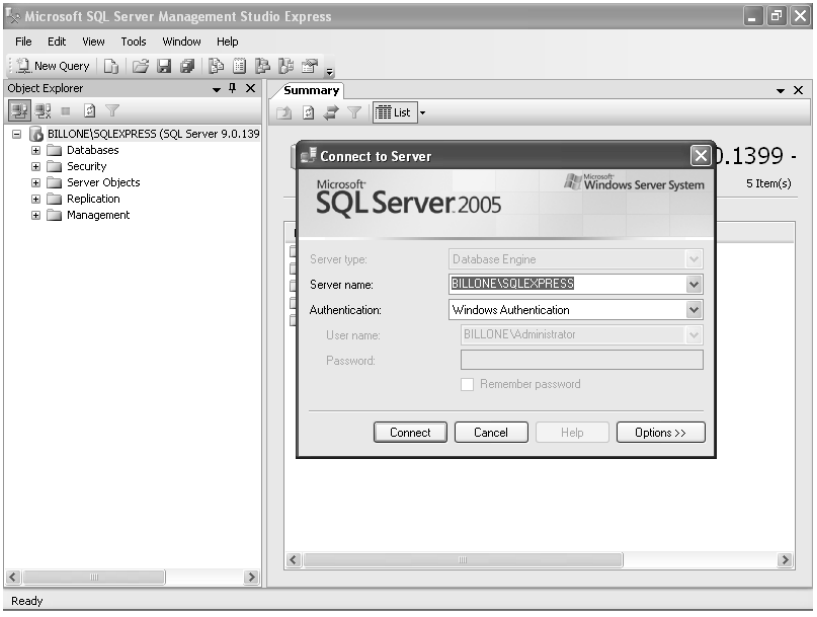


Figure 9.2
Connect To Server dialog box

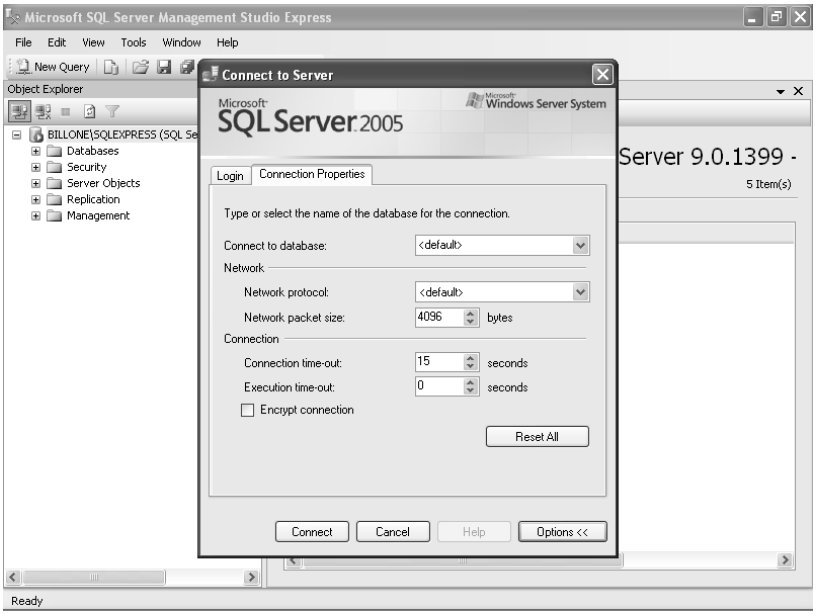


FIGURE 9.3
Clicking the Options button opens a two-tabbed dialog box.

The Login tab let's you do the following:

- Displays Database Engine as the Server type. You won't change this option.
- Select the server name.
- Choose either Windows or SQL Server authentication.
- Specify the login and password for SQL Server authentication.

Under the Connection Properties tab (Figure 9.4), you can do the following:

- Use Connect To Database to choose from the available databases on the server that you have permission to view.
- Select the network protocol.
- Select the network packet size, and configure the size if you want to change the default value of 4096 bytes.
- Set the connection time-out.
- Set the execution time-out in seconds (the default setting [0] means that execution will never time out).

Object Explorer contains a small menu that lets you choose from among the options listed in Table 9.1.

FIGURE 9.4
Connections
Properties tab

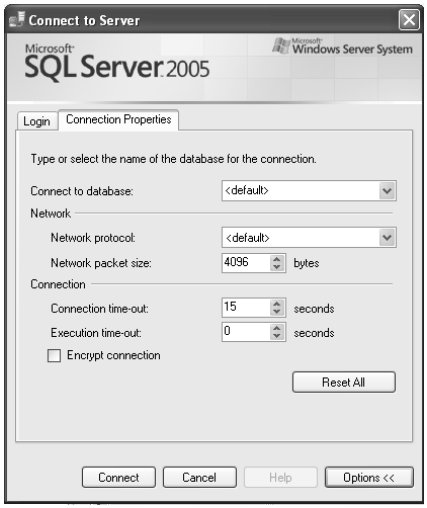


TABLE 9.1: Management Studio Object Explorer Icons

OBJECT	DESCRIPTION
Connect Object Explorer	Connects to server type
Disconnect	Disconnects from the active server
Stop	Stops the current process
Refresh	Refreshes the list of objects in a folder
Filter	Filters the list of objects

FOLDER STRUCTURE AND USAGE

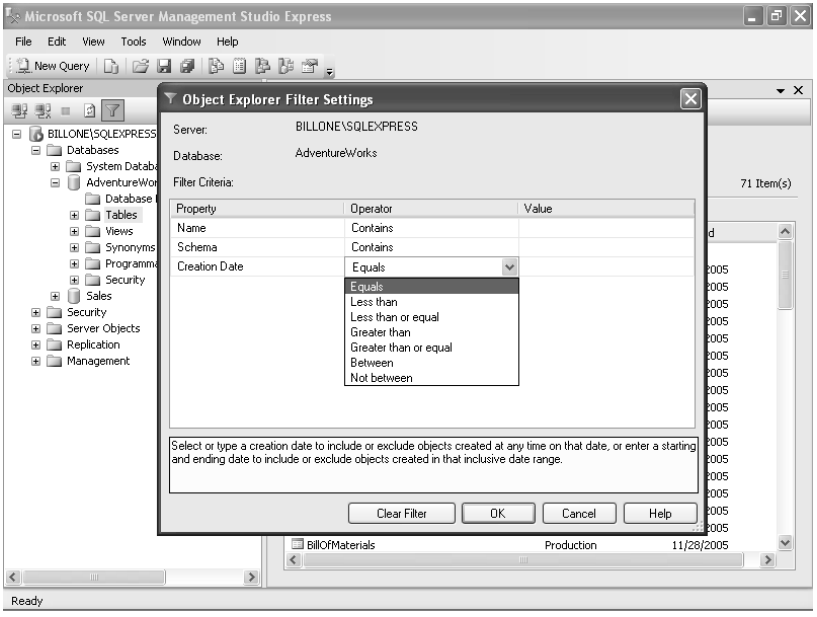
Object Explorer uses a tree structure to group information into folders for each server. In this section, we'll look at what's in that treeview and the contents of those folders for each server.

But first, some general housekeeping and navigation tips. To expand or collapse a folder, click the plus sign or double-click the folder. Usually you'll expand folders for more details. Management Studio is designed so that you can right-click folders or objects to perform common tasks.

The first time you expand a folder, Object Explorer queries the server for information to populate the tree. While Object Explorer is populating the tree, you can click Stop to halt the process. Any actions you take, such as filtering, apply only to the portion of the folder that was populated, unless you refresh the folder to start populating the tree again.

In order to save system resources, the folders in the Object Explorer tree don't automatically refresh their list of contents. To refresh the list of objects within a folder, right-click the folder, and then click Refresh; or click the Refresh button on the Object Explorer menu.

Object Explorer lets you filter the object list to reduce the size of the displayed components for ease of use or to reduce unwanted clutter. For example, you may want to find a specific database user or view only the most recently created tables in lists that contain hundreds of objects. To use the filter feature, select the Table folder (or node) in any database, and then click the Filter button to open the Filter Settings dialog box. Alternatively, you can right-click and select Filter. You can filter the list by name, creation date, and (sometimes) schema, and provide additional filtering operators like Starts With, Contains, and Between.



Folders in Object Explorer

Each server in Object Explorer contains the following folders:

- Databases
- Security
- Server Objects
- Replication
- Management

The Databases Folder

This folder contains the System Databases folder and user databases. The user databases have folders for each type of object they contain, including the following (see Figure 9.5):

- Database Diagrams
- Tables
- Views
- Synonyms
- Programmability
  - Stored Procedures
  - Functions
  - Database Triggers
  - Assemblies
  - Types
  - Rules
  - Defaults
- Security
  - Users
  - Roles

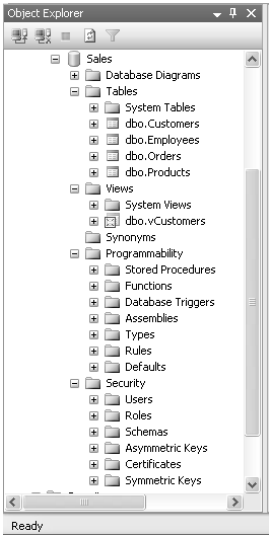


Figure 9.5
Typical Databases folder within Management Studio

- Schemas
- Asymmetric Keys
- Certificates
- Symmetric Keys

Databases

When you select a database in Object Explorer, the Summary pane displays context-specific information. For example, if you select a user database—in this case, AdventureWorks—you're presented with a basic list of database objects, as shown in Figure 9.6.

If you right-click the database name, you're presented with a menu that includes these options:

- New Database
- New Query
- Script Database As
  - CREATE To
  - ALTER To
  - DROP To
  - SELECT To

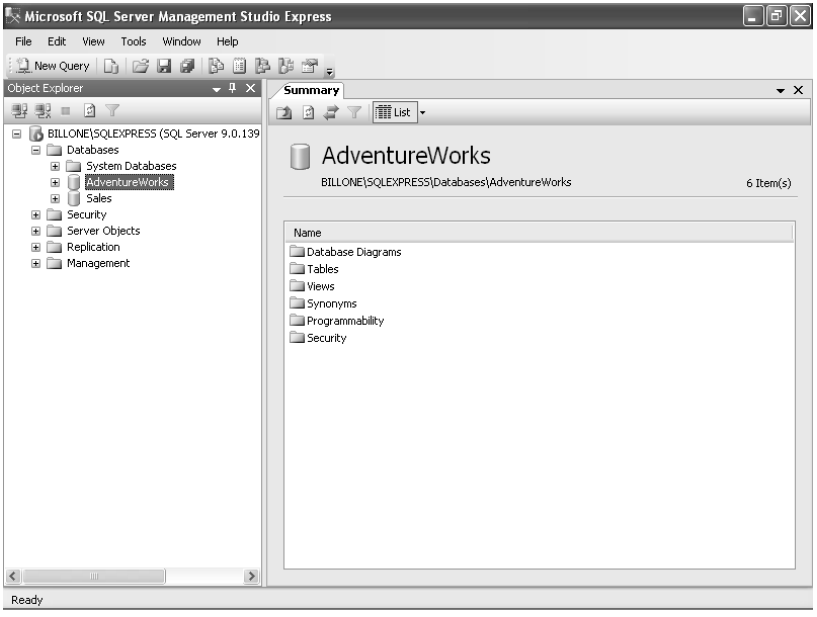


FIGURE 9.6
Summary view of the AdventureWorks database

- INSERT To
- UPDATE To
- DELETE To
- EXECUTE To

- Tasks
  - Detach
  - Shrink
  - Back Up
  - Restore
  - Generate Scripts
- Rename
- Delete
- Refresh
- Properties

The Properties page (Figure 9.7) consists of seven subpages: General, Files, Filegroups, Options, Permissions, and Extended Properties.

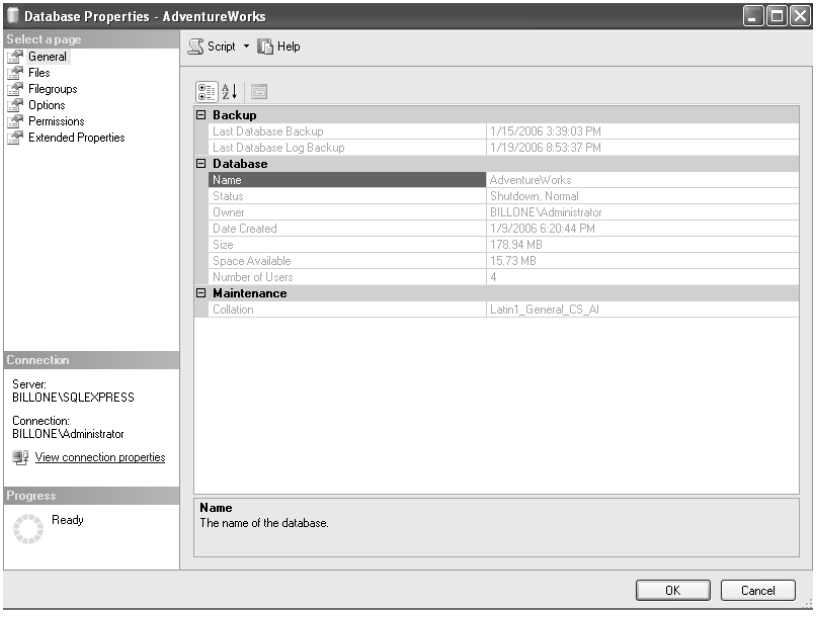


FIGURE 9.7
Database Properties page

The General properties page shows basic information, including the following:

- Dates of last database and database log backups
- Database name
- Status
- Owner
- Creation date
- Size
- Available space
- Number of users
- Collation status

From the Files Properties page, you can view or modify the following:

- Database name (you can't modify this setting)
- Owner
- Whether to enable or disable full-text indexing
- Database files for the associated database
  - Name
  - Type
  - Filegroup
  - Initial size
  - Autogrowth options
  - Path

You can also add a file to or remove a file from the database.

The Filegroups property page lets you add a new filegroup to the selected database.

The Options Properties page contains a number of modifiable settings related to the database. These include:

- Automatic settings for closing the database creating statistics, shrinking the size of the database, and updating statistics
- Cursor behavior
- ANSI settings
- Page verification methods during recovery
- Database state
- Other miscellaneous settings

Clicking Permissions opens a page where you can set user and group permissions for the table.

On the Extended Properties page, you can view, change, or delete the associated extended properties for the object (consisting of a name/value pair of metadata associated with the object). Page contents include the following:

- Database name
- Collation used for the selected database
- Properties window where you can view, create, or change the extended properties
- Delete button

Tables

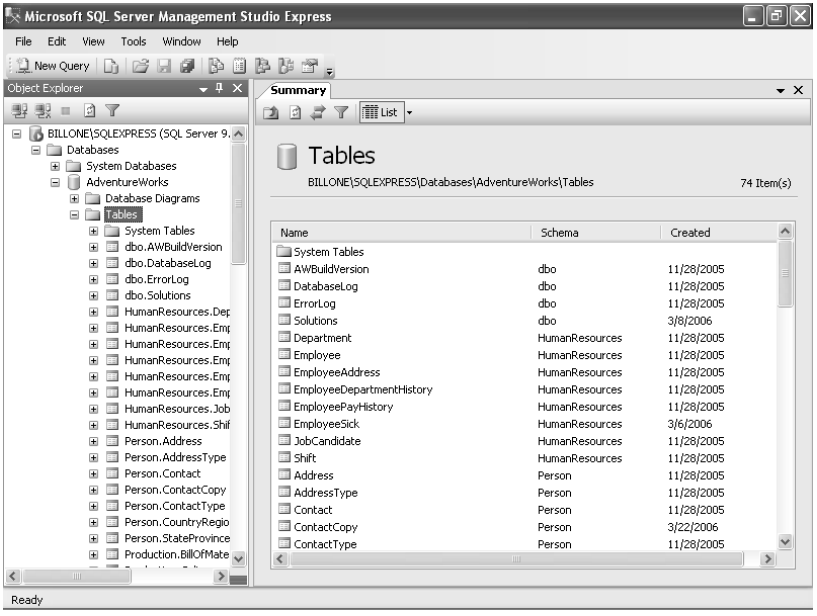
Double-clicking a Tables folder under a database file presents a different summary sheet listing all the tables in the database, as shown in Figure 9.8. Each table is listed alphabetically by name along with its date of creation. By selecting the Tables folder instead of expanding it, you can view the tables in the Summary window without cluttering Object Explorer.

If you right-click the main Tables folder, you can choose to create a new table. You can also apply or remove filtering.

Right-clicking a specific table, such as HumanResources.Department, gives you the option to perform basic table operations:

- Create a new table
- Modify or open a table
- Script a table

FIGURE 9.8
Summary sheet view of tables in a database



- View dependencies
- Rename the table
- Delete the table
- Refresh the table contents
- View the table's properties

The Table Properties sheet, shown in Figure 9.9 for the Department table, has three pages: General, Permissions, and Extended Properties. This pane also provides information regarding the connection and progress status.

The General properties page gives you basic information including the following:

- Creation date
- Data disk space
- Index disk space
- Name
- Row count
- Schema
- Whether the table is a system object
- Whether the table is replicated

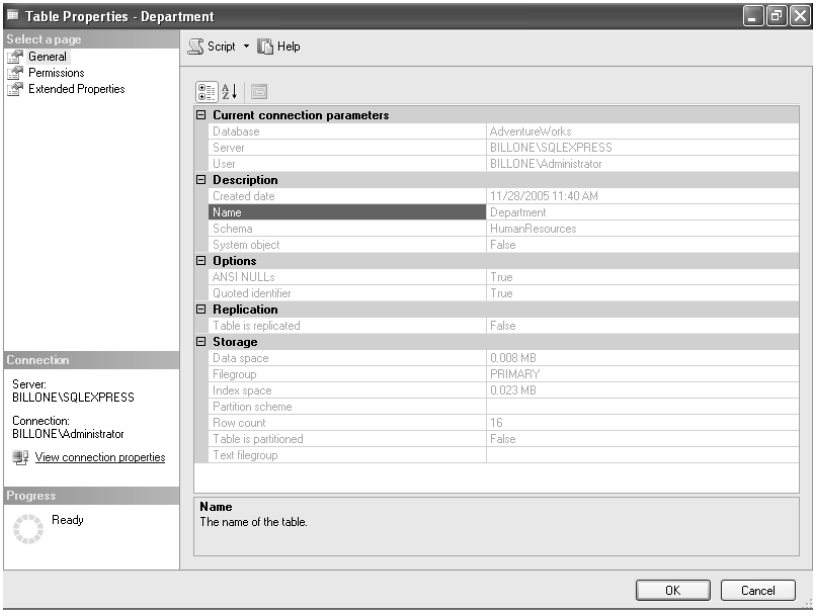


FIGURE 9.9
Table Properties dialog box

- Storage information regarding
  - Filegroup
  - Partition scheme
  - Whether the table is partitioned
  - Text filegroup

Note that you can't change these settings; you can only view them from the Properties sheet.

The Permissions Properties page lets you set user and group permissions for the table. The Extended Properties page is identical to the Database Properties page.

Double-clicking a specific table—HumanResources.Department, in the example shown in Figure 9.10—opens another set of folders:

- Columns
- Keys
- Constraints
- Triggers
- Indexes
- Statistics

Each of these in turn contains information for the specific instance of the object. (You can see this same set of folders by double-clicking a table in Object Explorer.)

You can create new columns, keys, constraints, and triggers by right-clicking the specific folder and selecting New. You can also refresh each of these.

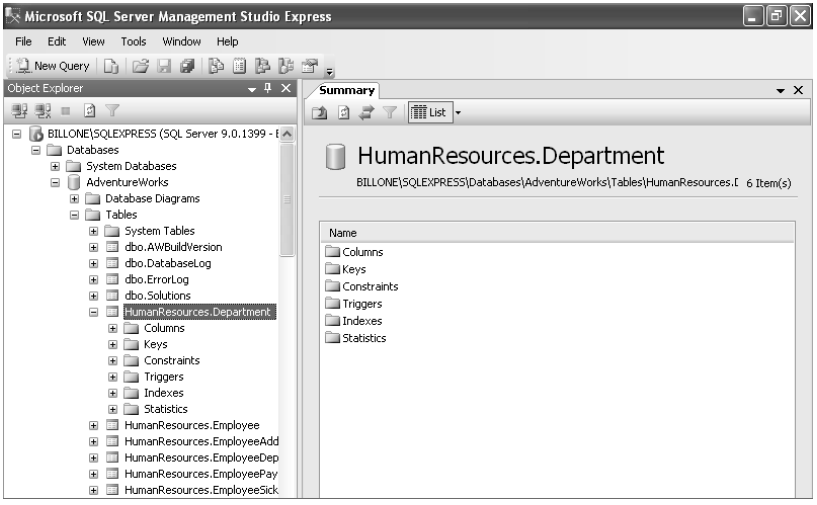


FIGURE 9.10
Double-click a table to learn more about it.

Right-clicking the Indexes folder lets you perform several tasks:

- New Index
- Rebuild All
- Reorganize All
- Disable All
- Filter
- Refresh

NOTE Indexing is covered in greater detail in Chapter 12, "Indexing."

Right-clicking the Statistics folder allows you to create new statistics using the Properties sheet.

Each individual item under a folder can be right-clicked. Doing so opens a menu that gives you the opportunity to select one of the actions listed in Table 9.2.

As you can see, you can rename, delete, or refresh any component by right-clicking the object. You can also create new components (except constraints) and use Script As to launch the Query Editor for all components except columns. Only triggers provide the option to view dependencies by right-clicking an instance.

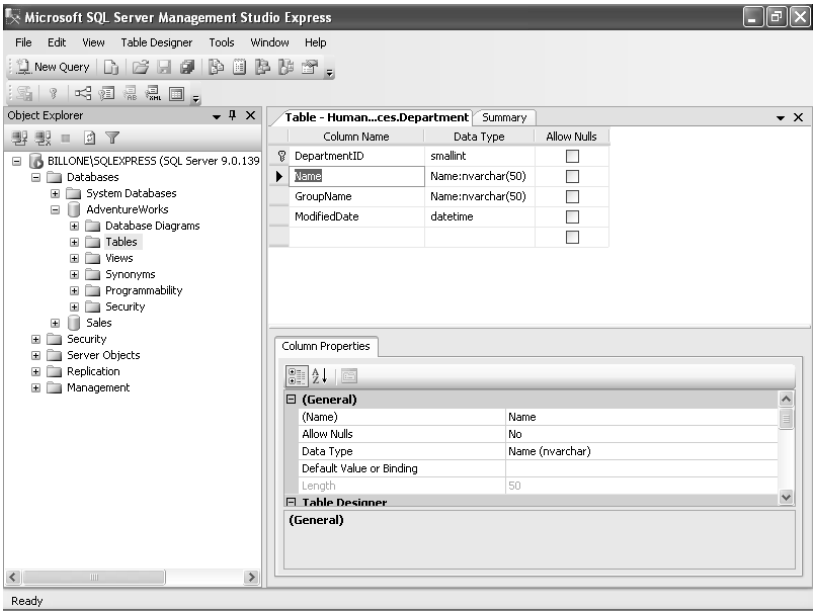
You can modify columns, keys, and triggers by right-clicking. To modify a column, right-click a column and then select Modify. A new window opens instead of the summary sheet (Figure 9.11), where you can change the properties for either the specific column you clicked or any other column in the same table by selecting a different one in the list above the properties list.

You can also view the Properties sheet for columns, indexes, and statistics. For example, right-clicking the Name column under HumanResources.Department and then selecting Properties opens the Column Properties sheet, as shown in Figure 9.12. The General Properties page lets you view a number of the column properties. To change them, however, you need to use the Modify Column option when right-clicking the specific column. The Extended Properties sheet is also available. Similar sheets open when you right-click properties for individual indexes and statistics.

TABLE 9.2: Actions That Can Be Performed on Individual Table Components

COLUMN	KEY	CONSTRAINTS	TRIGGERS	INDEXES	STATISTICS
New	New	Script As	New	New	New
Modify	Script As	Rename	Modify	Script As	Script As
Rename	Modify	Delete	Script As	Rebuild	Rename
Delete	Rename	Refresh	View Dependencies	Reorganize	Delete
Refresh	Delete		Delete	Disable	Refresh
Properties	Refresh		Refresh	Rename	Properties
				Delete	
				Refresh	
				Properties	

FIGURE 9.11
Expanded Tables folder showing components



Views

If you select the Views folder for a database, the right pane displays a list of all the views in the current database and the date they were created. Figure 9.13 shows this list for the AdventureWorks database. System views can be located in the System Views folder. As you should anticipate by now, right-clicking the Views folder provides you with the option to create a new view, filter the object list, and refresh the object list, as also shown in Figure 9.13.

Each individual user view consists of a set of four subfolders: Columns, Triggers, Indexes, and Statistics. Right-clicking any of these lets you create new instances of that type. Right-clicking a specific view provides you with a menu to perform the following basic operations:

- Create a new view
- Modify an existing view
- Generate SQL scripts (via Script View As)
- View dependencies
- Rename
- Delete
- Refresh
- Open the Properties sheet to view the general properties, set permissions, and view/modify extended properties

NOTE Views will be discussed in greater detail in Chapter 13, "Views."

FIGURE 9.12
Column Properties page

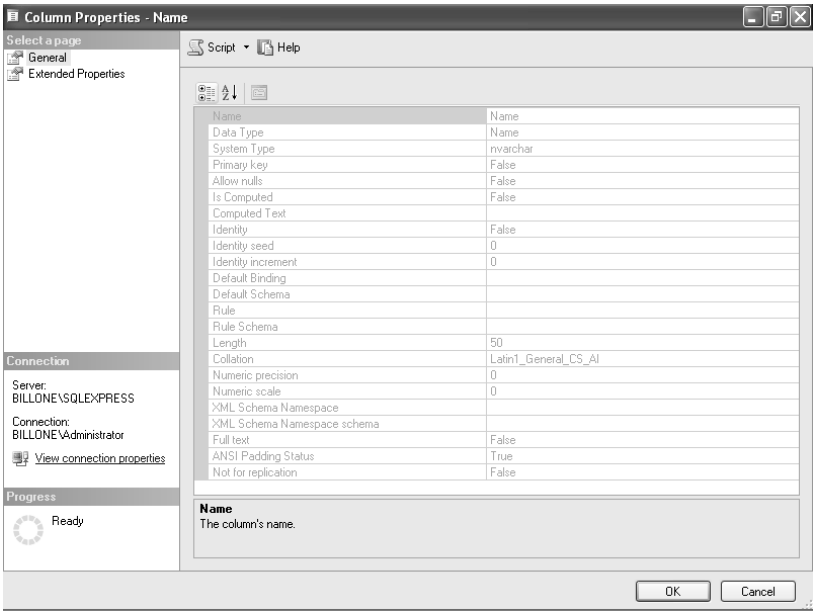
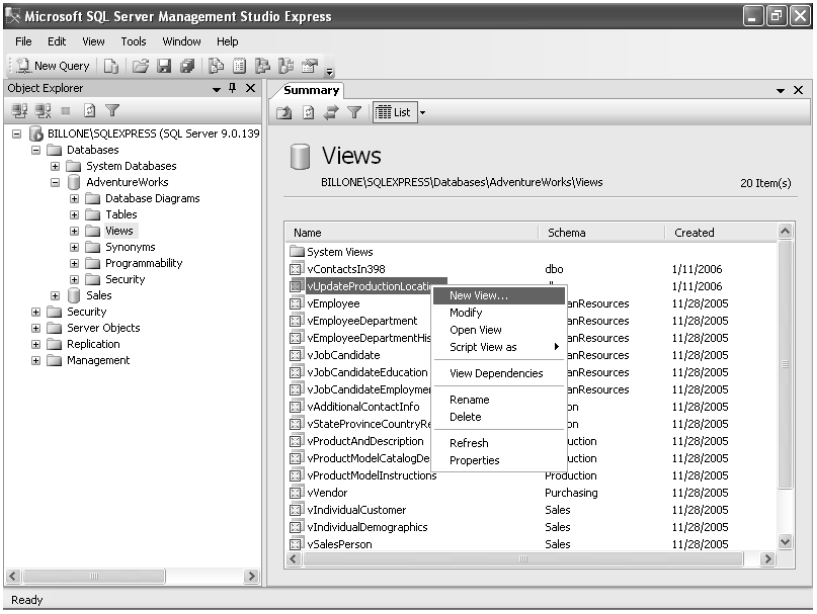


FIGURE 9.13
Views summary in Management Studio



Synonyms

A *synonym* is a database object that does the following:

- Provides an alternate name for another database object, called the *base object*, which can exist on a local or remote server
- Provides an abstraction layer that minimizes the impact of changes made to the name or location of a base object

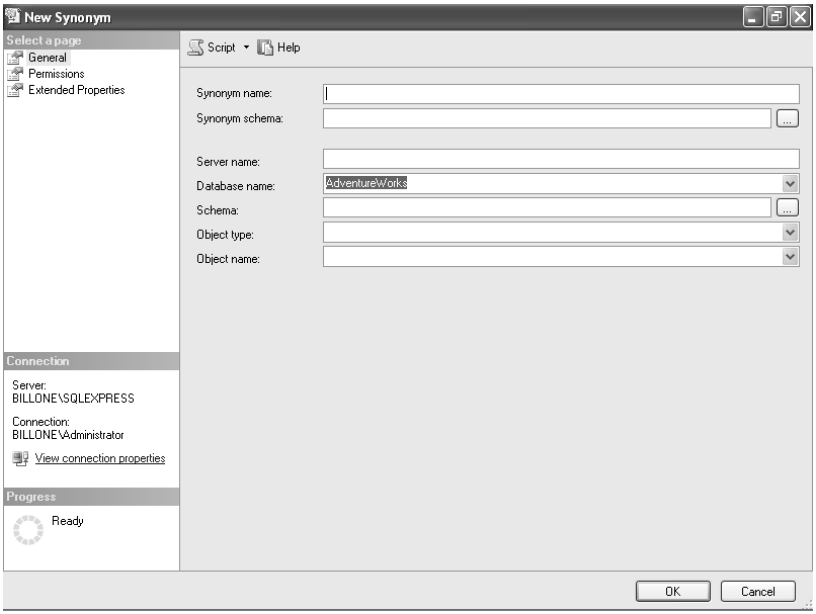
A synonym belongs to a schema; and as is true for other objects in a schema, its name must be unique. Synonyms can be created for the following objects:

- Assembly (CLR) stored procedures
- Assembly (CLR) table-valued functions
- Assembly (CLR) scalar functions
- Aggregate (CLR) functions
- Replication-filter procedures
- Extended stored procedures
- SQL scalar functions
- SQL table-valued functions
- SQL inlined-tabled-valued functions
- SQL stored procedures
- Views
- Tables (user-defined and including local and global temporary tables)

To create a new synonym, right-click the Synonyms folder to open the New Synonym page. Here, as shown in Figure 9.14, you can specify the following:

- Synonym name
- Server name
- Database name
- Schema
- Object type
- Object name

FIGURE 9.14
New Synonym page



Right-clicking a specific synonym (there are none in AdventureWorks) allows you to do the following:

- Create a new synonym
- Generate SQL script
- Delete
- Refresh
- Access the Properties sheet

Programmability

This folder is a placeholder for a series of subfolders related to programming in SQL Server:

Stored Procedures As you should expect by now, when you select the Stored Procedures folder in Management Studio, a list of all the stored procedures in the current database, including their date of creation, appears in the Summary window. Figure 9.15 shows this list for the AdventureWorks database.

You can right-click the Stored Procedures folder to create a new stored procedure, filter the current object list, or refresh the list.

Right-clicking a specific stored procedure allows you to do the following:

- Create a new stored procedure
- Modify a stored procedure

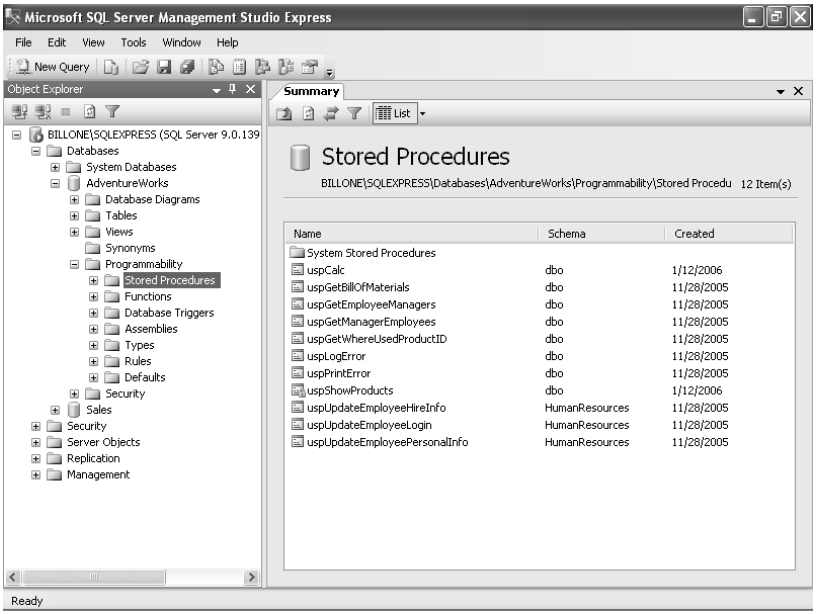


FIGURE 9.15
Stored procedures in Management Studio

- Execute a stored procedure
- Generate SQL scripts
- View dependencies
- Rename a stored procedure
- Delete a stored procedure
- Refresh the stored procedure items

NOTE You'll learn more about stored procedures in Chapter 14," Stored Procedures."

Functions The Functions folder and its subfolders—Table-Valued Functions, Scalar-Valued Functions, Aggregate Functions, and System Functions—contain functions that perform operations on and return values, objects, and settings in SQL Server Express. As you would expect, right-clicking subfolders gives you the opportunity to create new functions or filter or refresh the objects list. Right-clicking individual functions in the Table-Valued or Scalar-Valued Functions folder allows you to do the following:

- Create new functions
- Modify existing functions
- Generate SQL script
- View dependencies

- Rename a function
- Delete a function
- Refresh the functions list

Database Triggers Clicking a Database Triggers folder shows you all the database triggers in the current database. If none are present, then the folder is empty.

Assemblies This folder contains the assemblies in the current database. Managed code must be written and compiled into a .NET assembly in order to create objects. Typically, developers use Visual Studio 2005.NET to formulate a new class library project and compile it into a DLL assembly. It's then loaded using the T-SQL `CREATE ASSEMBLY` command. You can also create an assembly by right-clicking the Assemblies folder in Analysis Server and selecting New Assembly.

Types This folder is composed of four subfolders: System Data Type, User-Defined Data Types, User-Defined Types, and XML Schema Collections.

When you click the User-Defined Data Types folder, Management Studio shows you all the user-defined datatypes in the current database. You can think of user-defined datatypes as aliases for system datatypes.

You can right-click the User-Defined Data Types folder to create a new user-defined datatype. Right-clicking a specific user-defined datatype opens a menu that allows you to do the following:

- Create a new user-defined datatype
- Generate SQL script
- View dependencies
- Rename the datatype
- Delete the datatype
- Refresh the datatype
- Open the datatype Properties sheet

NOTE You'll learn more about user-defined datatypes in Chapter 11, "Tables."

The XML schema collection is a metadata entity that is similar to a table in a database. Schemas are imported into the XML schema collection object when created using the `CREATE XML SCHEMA COLLECTION` statement. You can use the XML schema collection to type XML variables, parameters, and columns. Right-clicking a specific XML schema collection lets you generate SQL script, view dependencies, delete, or refresh.

Rules Clicking a Rules folder shows you all the rules in the current database. *Rules* are conditions expressed in T-SQL syntax (for example, `@salary < 20000`) that can be used to limit the data contained in columns of a table.

NOTE You usually won't find any rules in SQL Server Express databases unless you're working with a database created in an earlier version. Rules are obsolete and have been replaced by constraints.

Defaults When you click a Defaults folder, the Summary window shows you all the defaults in the current database. A *default* is a value that can be attached to one or more table columns for use when a value isn't explicitly supplied for that column in a new row of the table.

TIP Like rules, defaults are largely obsolete. For the most part, you should use default constraints instead of defaults in your database designs. There's further information on defaults in Chapter 4, "Database Design and Normalization."

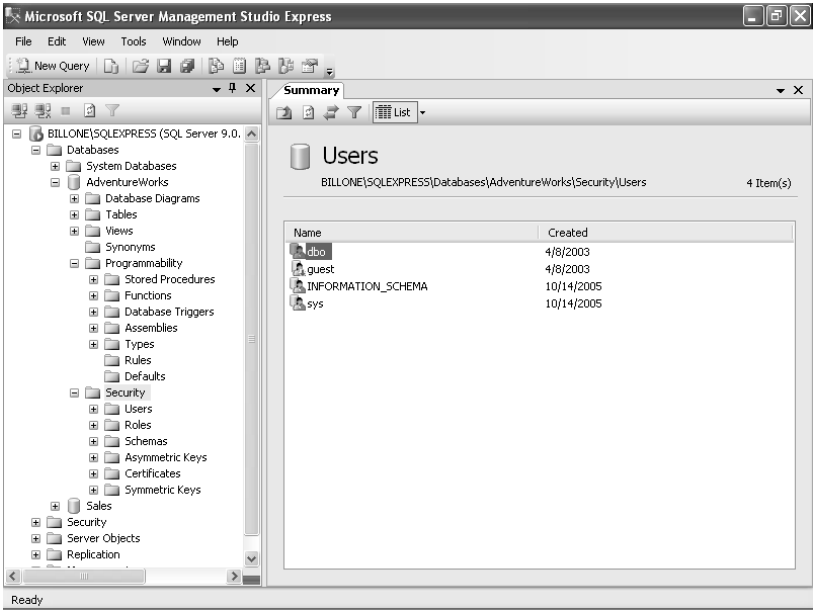
Security

The Security folder consists of six subfolders: Users, Roles, Schemas, Asymmetric Keys, Certificates, and Symmetric keys. The contents of these folders relate to the current database, not the entire server:

Users If you click a Users node, you'll see a list of all the users for the current database, similar to the list shown in Figure 9.16. Users are specific to a database (unlike logins, which apply to entire servers) and are the basis for permissions within that database.

You can create new users by right-clicking the Users folder, as well as filter and refresh the folder objects. Right-clicking an individual user contained within the folder allows you to create a new user, generate SQL script, delete the user, or refresh the user. You can also open the Properties sheet (Figure 9.17) for a user to specify owned schemas and database role membership. In addition, you can specify the login name and default user. You'll find options for setting user permissions on the Secureables page.

FIGURE 9.16
User list in Management Studio



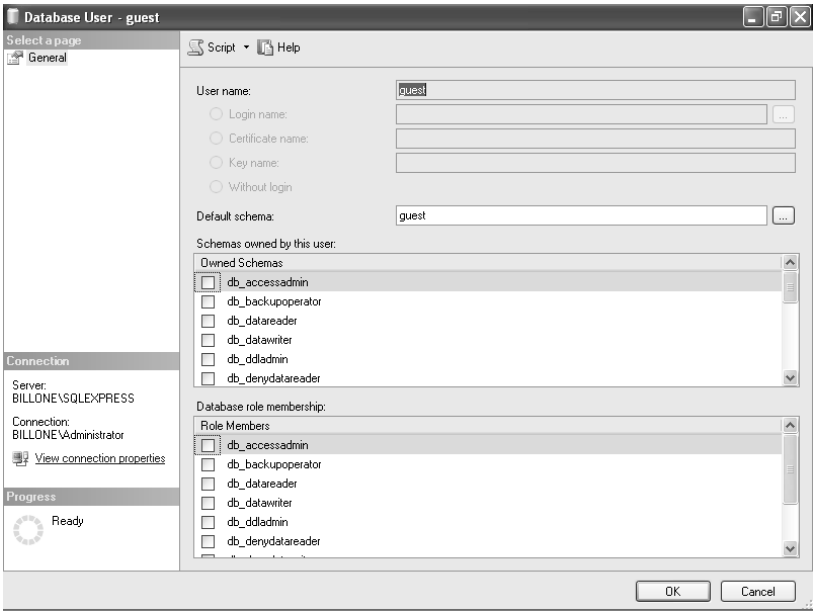


FIGURE 9.17
Database User Properties sheet

Roles Roles allow you to manage permissions for groups of users rather than for individual users. The Roles folder consists of two subfolders, because there are two types of roles: Database Roles contain SQL Server users, and Application Roles are designed for client-side validation of the user's identity. Clicking either subfolder shows you a list of all the roles in the current database, as shown in Figure 9.18 for Database Roles in the AdventureWorks database.

By right-clicking the Roles folder, you can create either a new database or an application role. Right-clicking a specific subfolder enables you to create a new role of that type. Choosing New launches an empty Properties sheet (Figure 9.19). You can use this to create a new role and specify all pertinent settings. For an existing role, you can use this page to view or modify role properties. This page is accessed in two ways: by right-clicking Database Roles (or Application Roles) in Object Explorer and clicking New Database (or Application) Role, or by right-clicking an existing role and clicking Properties. In the latter case, some of the options aren't editable.

NOTE You'll learn more about roles in Chapter 17, "Security and SQL Server Express."

Schemas Schemas contain database objects like tables, views, stored procedures, and so on. This folder contains all the schemas associated with the database. Right-clicking the Schemas folder lets you create a new schema. Right-clicking an existing schema enables you to create a new schema, generate SQL script, or delete or refresh the schema. Selecting Properties opens the same sheet as when you create a new schema, but some options (such as the schema name) aren't editable. You can also set permissions and extended properties.

Asymmetric Keys The Asymmetric Keys folder contains the asymmetric keys associated with the particular database. If the database has no associated asymmetric keys, the folder is empty.

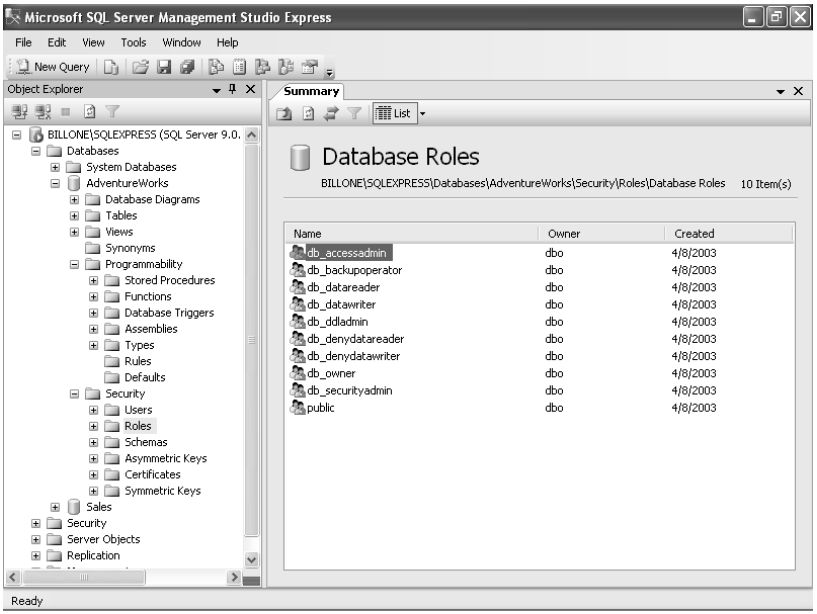


FIGURE 9.18
Database Roles list in Management Studio

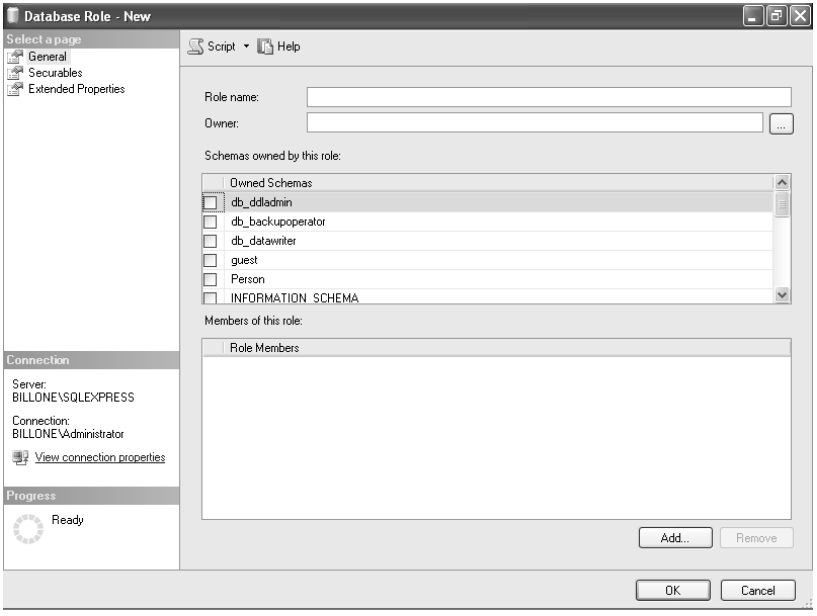


FIGURE 9.19
New Database Role Properties sheet

Certificates The Certificates folder contains the certificates associated with the particular database. If the database has no associated certificates, the folder is empty.

Symmetric Keys The Symmetric Keys folder contains the symmetric keys associated with the particular database. If the database has no associated symmetric keys, the folder is empty.

The Security Folder

The server Security folder (not to be confused with the database Security folder, which we just reviewed) contains settings that effect the entire server, not just the particular database. There are three subfolders: Logins, Server Roles, and Credentials.

Logins

Logins provide the security context for the users on SQL Server. When you click the Logins folder, Management Studio displays a list of all the logins known to the current server. Right-clicking the main Logins folder lets you create a new log-in, or filter, or refresh the objects list. When you create a new login, you're directed to a blank Properties sheet.

Opening the Properties sheet for an existing login (by right-clicking the login and selecting Properties) shows you (and in some cases allows you to modify) the following:

- The login name
- Whether the login is permitted or denied access to the server
- The default database for the login
- The default language for the login

Right-clicking an existing login lets you create a new login, generate SQL script, delete the login, or refresh the login.

Server Roles

Server *roles* are built-in sets of permissions that SQL Server Express supplies. For example, there's a Server Administrator role that allows its members to configure server-wide settings. When you click the Server Roles folder, Management Studio displays all the server roles on that server.

Right-clicking a server role lets you manage membership by adding logins to or removing logins from the server role.

NOTE Unlike most other objects displayed in Management Studio, you can't create or delete server roles.

Credentials

Credentials prove that you are who you say you are, so you can gain access to local and networked resources. Credentials include usernames, passwords, smart cards, and certificates.

The Replication Folder

The Replication folder serves as a central location to organize and administer subscriptions. If you right-click the Replication folder, you can change replication passwords. To add a new subscription, right-click the Local Subscriptions subfolder.

The Management Folder

This folder provides access to traditional database administrator information, including the following:

- SQL Server logs
- Activity Monitor, including
  - Process information
  - Locks by process
  - Locks by object

SQL Server Logs

The SQL Server Express Logs folder contains nodes for the current activity log and for the six most recent activity logs before that. Whenever you start SQL Server, it begins writing events to the Windows application event log (the same one found in Event Viewer). These events are also available in the SQL Server Express log.

NOTE You'll learn more about interpreting SQL Server logs in Chapter 16, "Basic Administrative Tasks."

Activity Monitor

Activity Monitor can be used to analyze server performance, view user connections, and resolve deadlocks. You can also apply filters to restrict the display to items of interest as well as change the refresh rate to watch activity as it occurs. There are three pages in Activity Monitor: Process Info, Locks By Process, and Locks By Object. For each process, the Process Info page shows the following:

- Process ID (this is the unique ID that SQL Server Express assigns to each process when it's started—also known as a *spid*)
- System Process
- User
- Database
- Status
- Open Transactions
- Command
- Application
- Wait Time
- Wait Type
- Resource
- CPU
- Physical IO

- Memory Usage
- Login Time
- Last Batch
- Host
- Net Library
- Net Address
- Blocked By
- Blocking
- Execution Context (a unique ID for each of the subthreads operating on behalf of a single process)

Double-clicking a process lets you see the most recent SQL batch submitted by that process. You can also kill the process from the same windows.

The Locks By Process page contains one entry for each process running on the server. Clicking one of these nodes shows information about all the locks being maintained by the process. The following information is displayed about each lock:

- Object
- Type
- Subtype
- Object ID
- Description
- Request Mode
- Request Type
- Request Status
- Owner Type
- Owner ID
- Owner GUID
- Database
- Process ID
- Context
- Batch ID

NOTE The Locks By Object page lists current locks on objects. You'll learn more about locking in Chapter 21, "Locking."

Template Explorer

Template Explorer is a collection of folders containing templates. *Templates* are standardized files containing SQL scripts that you can use to create objects in the database. SQL Server Express comes with a number of different templates that are installed by default in the `C:\Program Files\Microsoft SQL Server\90\Tools\Binn\VSShell\Common7\IDE\sqlworkbenchnewitems` directory.

To open Template Explorer, choose View ➢ Template Explorer or press Ctrl+Alt+T.

Templates are available for solutions, projects, and various types of code editors. They're also available to create objects such as databases, tables, views, indexes, stored procedures, triggers, and functions. In addition, there are templates that help you to manage your server by creating extended properties, linked servers, logins, roles, and users.

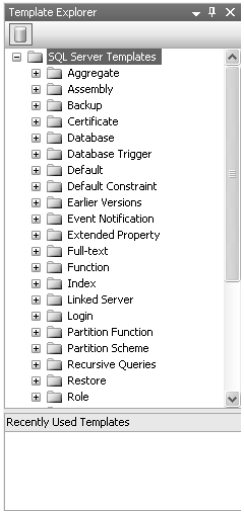
You can open a template from the File menu or from Template Explorer. To open a template from Template Explorer, Choose View ➢ Template Explorer. A new window opens, as shown in Figure 9.20.

Template Explorer contains shortcuts for three server types: SQL Server, Analysis Server, and Mobile Server. Each type of server has different folders associated with it by default.

You can create a new custom template in Template Explorer as follows:

1. Right-click a folder, subfolder, or template; click New; and then select Template.
2. Right-click the new template, and select Edit.
3. When you're prompted to connect to SQL Server Express (this is optional), click Connect.
4. In the subsequent windows, use Query Editor to create the script.
5. Right-click the Template tab, and click Save <*template name*>.

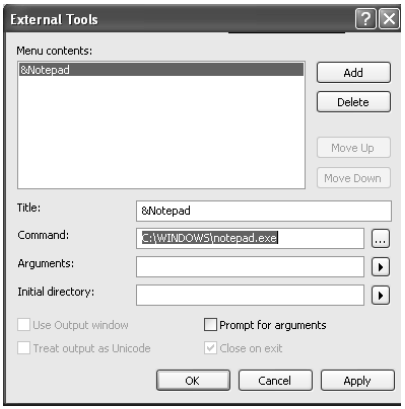
FIGURE 9.20
Template Explorer



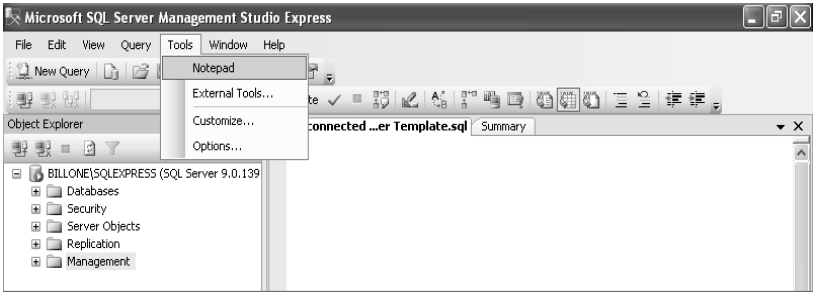
External Tools

You can launch any Microsoft Windows 2000 application from Management Studio by adding external applications (such as Notepad) to the Tools menu. To add an external tool to the Tools menu, complete these steps:

1. Choose Tools ➢ External Tools.
2. In the Title box, type the name as you want it to appear on the menu bar. (Hint: If you put an ampersand [&] before a letter in the name of the tool, you can use that letter as an accelerator key.) For example, typing **&Notepad** displays Notepad on the Tools menu and makes the letter *N* into an accelerator key.
3. In the Command box, enter the path to the executable file.



4. In the Arguments box, specify any value you want passed to the tool, or command-line switches if appropriate.
5. In the Initial Directory box, specify the tool's working directory.
6. Set the following options, when available: Use Output Windows, Prompt For Arguments, and Close On Exit.
7. Click OK. The new program is added to the Tools menu.



Summary

In this chapter, we introduced you to the immensely rich Management Studio. Styled on the look of Visual Studio, this is the central master control system for all SQL Server Express operations. As you've seen, you can perform many common SQL Server Express operations without ever leaving the Management Studio window.

In addition to displaying information about SQL Server Express objects and operations, Management Studio is lavish in the ways it provides for you to perform basic tasks. Nearly everything you need to do can be accomplished through shortcuts menus in Object Explorer. You also learned how to add external tools so that you can easily access other applications such as Notepad.

Now it's time to look at the objects within Management Studio more closely. In the next chapter, we'll start with databases.