CERTIFICATION STUDY GUIDES

MCSE STUDY GUIDE

Installing, Configuring, and Administering Microsoft SQL Server 2000 Enterprise Edition Exam 070-228

Edition 3

Congratulations!!

You have purchased a ***Testkiller Ltd.*** Study Guide.

This study guide is a selection of questions and answers similar to the ones you will find on the official Installing, Configuring, and Administering Microsoft SQL Server 2000 MCSE exam. Study and memorize the following concepts and keypoints for approximately 15 to 20 hours and you will be prepared to take the exams. We guarantee it!

Remember, average study time is 15 to 20 hours and then you are ready!!!

GOOD LUCK!

Guarantee

If you use this study guide correctly and still fail the exam, send your official score notice and mailing address to:

Testkiller Ltd.
8200 Pat Booker Rd. #368
San Antonio, TX 78233

We will gladly refund the cost of this study guide. However, you will not need this guarantee if you follow the above instructions.

http://www.testkiller.com

Cover & Logo Designed by: The Website Guru Company www.thewebsiteguru.com

Table of Contents

It is important that you read and study the "Keypoint Concepts" portion of this guide. We have identified important "KEYPOINTS" in this section that you will have to know to successfully pass the exam. Please ensure that you absolutely know and understand these. You will find them in double lined boxes throughout the text.

Installing, Configuring, and Administering Microsoft SQL Server 2000 Concepts

Installation and Upgrading

System Requirements

Determining the system requirements for SQL Server 2000 system requires the analysis of what you intend the function of the system to be. One computer may be adequate in small volume database environments. Alternatively, you might find that you need several computers if the system will be operating in an environment containing several thousand users.

Hardware Requirements

SQL Server is designed to run on any Windows NT or Windows 2000 system. Hardware requirements are as follows:

- Pentium 166 MHz or higher processor
- 64 MB RAM minimum, 128MB recommended
- 95 to 270 MB hard drive space for server database components only. Data files will require more space.
- VGA or higher resolution monitor

Software Requirements

SQL Server 2000 Enterprise Edition is designed to run on a system using Windows NT 4 Server (with Service Pack 5) or Windows 2000 Server. SQL Server Enterprise Edition cannot be installed on a system running Windows 98, Windows NT 4.0 Workstation, or Windows 2000 Professional.

To install SQL Server you must have administrator privileges on the target computer. After installation SQL Server Service and SQL Server Agent will, by default, run in the context of a domain administrator account. SQL Server will require a security context that will allow it to perform the tasks that you plan for it. The security context of a domain administrator account allows the programs to perform tasks that require an external security context more easily. If it is not possible for SQL Server to run under a domain administrator account, SQL Server should be configured to run under the context of a local administrator account.

Initial Configuration and Setup

Network Protocols (Network Libraries)

When SQL Server is first installed, it automatically installs the shared memory, TCP/IP, and Named Pipes Net-Libraries. NWLink, AppleTalk ADSP, Multiprotocol, and Banyan VINES can be enabled during and after installation. To enable another network protocol after installation, start the Server Network Utility. Under the General tab you can enable or disable any supported protocol. To configure an enabled protocol, highlight the protocol and click properties.

Collation

When the server is installed, you must choose the default collation for your server. The collation you choose will determine the way that SQL Server will organize and compare data. There are several considerations when deciding on the collation method.

A character set is a standard way to identify characters as integers. Character sets are defined lists of characters that are assigned values that are recognized by hardware and software. The default character set installed in

SQL Server is the Western Hemisphere and Western Europe character set which, like most non-Unicode character sets represents a maximum of 256 unique characters and stores them in eight bits or one byte. Since there are more than 256 characters in all the languages of the world, Unicode is an attempt to define all these characters as integers in the same character set. Unicode stores characters in two bytes or 16 bits and data in that format takes up twice as much space than most non-Unicode character sets. Unicode is not compatible with other 256 character sets.

Sort order determines the way that data is compared and assigned. Sort order determines if the operations that the server performs are case sensitive or accent sensitive. There are different sort orders for Unicode characters and non-Unicode characters.

When you are installing SQL Server, you must choose the default collation for all data. This collation will encompass sort orders and character sets on the data that you use unless you specify a different collation later.

Instances

You can run multiple copies of SQL Server 2000 on the same computer. Each instance is a separate entity with its own settings, valid users, and databases. There is only one default instance, and instances other than the default instance are called named instances. Multiple instances must have different names and are accessed through the name they are assigned. There is a recommended maximum of 16 instances per machine. While AppleTalk, Multiprotocol, and Banyan VINES are supported in the default instance, these protocols are not supported in named instances.

Upgrading from SQL Server 6.5

When installing SQL Server 2000, if the installer detects an installed SQL Server 6.5, you will be given the choice of making the new installation a named instance of SQL Server or upgrading to SQL Server 2000. If you plan to use the databases that you used with SQL Server 6.5, the data must be totally rebuilt to be used with the new installation of SQL Server 2000. If you choose not to create a new named instance, the new installation of SQL Server 2000 will set itself as the default instance. SQL Server 6.5 can only run as the default instance; therefore you can only run either the SQL Server 6.5 or Server 2000 at any given time.

Upgrading from SQL Server 7

If you are installing SQL Server 2000 on a computer that already has an installed instance of SQL Server 7, you will be given the choice of making the new installation a named instance of SQL Server or upgrading the current installation. If you plan to upgrade the installation of SQL Server 7, the installer automatically rebuilds all of the system stored procedures and performs a slight modification of the database files. The installer performs the rebuild to ensure that the most current versions are available. If you plan to use a named instance of SQL Server 2000 with a SQL Server 7 database, you can load database backups from SQL Server 7 and use the *sp_attach_db* stored procedure to connect the database to your installed instance of SQL Server 2000.

Linked Servers

Using linked servers allows the SQL Server to execute commands OLE DB data sources on different sources. Typically linked servers are used to enable SQL Server 2000 to query several different databases from multiple database providers. There are two ways to create a linked server. To create a linked server through stored procedures, execute the sp_addlinkedserver stored procedure. To create a linked server using the Enterprise Manager, use the SQL Enterprise Manager Console Tree and the linked servers node under the Security tab.

SQL Mail and SQLAgentMail

SQL Mail allows a SQL Server to send e-mail. SQL Server 2000 uses two services to send mail, MSSQLServer and SQLServerAgent. MSSQLServer deals with mail for stored procedures. SQLServerAgent uses its own mail resources, SQLAgentMail, to send mail and is configured independently from SQL mail. Mail can be sent by a trigger or a stored procedure.

SQL Mail requires a domain user account, a mail profile, a post office connection, and a mailbox. A mail profile created under the same domain account used to start SQL Server 2000 is required for SQL Mail to run. SQLAgentMail runs under a domain account that is different from the SQL Mail domain account.

Installation and Upgrading Keypoints

1. The SQL Server 2000 computer you administer is named InfoServ and is a member of a Microsoft Windows NT domain. InfoServ contains logins for several domain user groups and is currently configured for Windows Authentication. The server is configured to use the Named Pipes Net-Library. Your company purchases five UNIX client computers and you want to allow users to be able to access InfoServ from these computers. You should **configure InfoServ for Mixed Mode authentication** and **add a login for each of the five UNIX users**.

2. On one of the SQL Server 2000 computers that you administer, you want to retrieve information from an archived inventory database that you have a full tape backup of. The backup's header information shows that the backup uses the SQL_Latin1_General_CR437_BIN collation. The existing SQL 2000 Server computers in your office are configured to use the SQL_Latin1_General_CP1_CI_AS collation. You do not want to join tables in the inventory database with tables in other databases. To restore the inventory database to a SQL Server 2000 computer by using the least amount of administrative effort you should **restore the inventory database to an existing SQL server computer** and **accept the SQL_Latin1_General_CR437_BIN collation for that database**.

3. You are installing SQL Server 2000 on a Windows 2000 Server computer that will run two client/server database applications that will use separate databases. During the installation, you specify SQL_Latin1_General_CP1_CI_AI as the default collation for the SQL server computer. After completing the installation, you discover that one of the new applications, named Financials, is sorting information incorrectly. You contact the application vendor and discovered that the Financials application requires the SQL_Latin1_CP437_BIN collation. To configure the correct collation for the applications without diminishing the performance of the SQL Server computer you should **use the ALTER DATABASE statement to specify the SQL_Latin1_General_CP437_BIN collation for the Financials database**.

4. You create several jobs on the SQL Server 2000 computer you administer and schedule them to run during the evening. You create an operator account on the server and configure the jobs to notify the operator via e-mail if a job fails for any reason. When you test the job, you notice that e-mail messages are not being sent to the operator when a job fails. You use SQL Server Enterprise Manager to confirm that SQL Mail is started. To ensure that e-mail messages are sent to the operator whenever a job fails you should **configure SQLAgentMail to use a valid MAPI profile**.

5. You configure the SQL Server 2000 computer that you administer to send e-mail messages to SQL Server operators whenever a SQL Server Agent job fails or encounters an error. Developers in your company create a set of stored procedures that send query results in e-mail messages but the stored procedures have not been successful. Using a MAPI profile named MS Exchange Settings, you verify that SQLAgentMail is correctly configured and working properly. You should take the following two actions to configure the servers so that the stored procedures function correctly:
   - **Configure the MSSQLServer service to use the same service account as the SQLServerAgent service.**
   - **Configure SQL Mail to use the Default Settings MAPI profile.**

6. The SQLServerAgent service is configured operate under a system account, and you want to configure two SQLServerAgent jobs to perform maintenance tasks. The jobs will copy files from the server to a file share on a corporate server. When testing the jobs, they both fail. To ensure that both jobs to execute successfully you should **configure the SQLServerAgent service to use a domain account**.

7. The SQL server computer you administer is running SQL Server 6.5 and SQL Server 7.0. You install a named instance of SQL Server 2000, and then run the SQL server upgrade wizard. During the upgrade, on the database selection screen, some of the SQL Server 6.5 databases are missing. You cancel the SQL server upgrade wizard. To ensure that the SQL server 6.5 databases are listed in the wizard you should **uninstall SQL Server 2000, and then reinstall SQL Server 2000 as the default instance**.

8. The SQL Server 2000 computer that you administer is named JSASQL1. You want to perform ad hoc distributed queries against a database that is stored on a SQL Server 2000 computer named JSASQL2. JSASQL2 contains several databases, each of which uses a different collation. To make sure that comparisons in distributed queries are evaluated correctly and administrative overhead is minimized you should **configure JSASQL1 to add JSASQL2 as a linked server** and **select the "Use remote collation" check box, and do not specify a collation name**.

9. Your company uses the SQL Server 2000 computer that you administer to store service contract information for its customers. You also administer an Oracle Relational Database Management System (ROEMS) server. The Oracle server is used to store financial information that is updated frequently throughout the day. You need to create a series of reports that will be updated several times a day that combine the service contract information and the financial information. To create the reports while using the minimum amount of disk space you should **set up the Oracle server as a linked server** and **create a view that joins the service contract information and the financial information**.

10. The Windows 2000 Server computer you administer is named TTSRV1 and is a member server in your company's Windows NT 4.0 domain named TTDOMAIN. All of your company's user accounts are contained in a separate Windows NT domain named TTUSERS. A trust relationship exists between the TTDOMAIN and the TTUSERS domains. You install SQL Server 2000 on TTSRV1 and specify a service account for each SQL Server service as shown in the following table:

Service	Service account	Account type
MSSQLServer	TTUSERS\sqlsvr	Domain administrator
SQLServerAgent	TTUSERS\sqlagent	Domain administrator

After the installation is complete, the MSSQLServer service and the SQLServerAgent service do not start. To configure the services to start you should **add TTUSERS\sqlsvr and USERS\sqlagent to the local administrators group on TTSRV1.** This will accomplish your goals without changing any security settings in either domain.

11. You administer two SQL Server computers; one named SQL7, and other is named SQL2000. SQL7 is runs SQL Server 7.0. It utilizes Named Pipes, TCP/IP, NWlink, IPX/SPX, and Mulitprotocol Net-Libraries. SQL2000 is running SQL Server 2000. The actions you should take to ensure that unauthorized users cannot access confidential company information that SQL2000 and SQL7 exchange are:
    - **Select the "Force protocol encryption" check box on SQL2000.**
    - **Enable multipoint encryption on SQL2000 and SQL7.**

12. You are configuring a database on two Microsoft Windows 2000 Advanced Server computers that will store accounting information for your company. The data must be available at all times and interruptions in data connectivity should not last longer than ten minutes. Changes you make to the database must not require you to reconfigure the client computers. To configure the database you should **configure the database on the two servers as a SQL Server 2000 cluster**.

13. On the SQL Server 2000 computer you administer you create a job that performs several database maintenance tasks. You want to set the job to run whenever the server's processor utilization falls below five percent. You create a new schedule for the job and specify the "Start whenever the CPU(s) become idle" option. When you check the log files several days later you notice that the job has never executed. During this time the server's processor utilization has fallen below five percent several times. To enable this job to run you should **modify SQL Server Agent properties and specify a shorter idle time.**

14. Your company has 150 UNIX client computers. Several client computers require access to a database named *Squadinfo* on one of the SQL Server 2000 computers. The server is configured with the TCP/IP and Named Pipes net-libraries. You need to configure the server to allow the client computers to use the *Squadinfo* database while also minimizing the amount of time it takes to configure the server and allow for the tracking of individual user actions. The actions you should take are:
    - **Configure Mixed Mode Authentication.**
    - **Create a SQL server login for each user.**

15. The SQL Server 2000 computer that you administer is named JSASQL. You have configured several SQL Server Agent jobs to perform automated routine maintenance tasks. The jobs execute properly for several weeks, and then stop executing. When you research the situation, you find that the SQLServerAgent service has stopped. You attempt to start the SQLServerAgent service, but are unable to get it to start. In the Windows application event log you discover the following error message:

    ```
    "SQLServerAgent could not be started (reason: Unable to connect to
    server 'JSASQL', SQLServerAgent cannot start)".
    ```

    You verify that the MSSQLServer service is running. To start the SQLServerAgent service and execute the maintenance jobs you should **configure the SQLServerAgent service to start by using the same user account as the MSSQLServer service**.

16. You are preparing to install SQL Server 2000 on the Windows 2000 computer you administer. Your company will use a variety of client computers to connect to the SQL Server 2000 computer. The client computers will use specific net-libraries depending on the type of client. The determining list is as follows:

Client Computer	Net-Library
Microsoft Windows 98	Named pipes
Novell Netware	IPX/SPX
Apple Macintosh	TCP/IP

    To allow client computers to connect to the SQL server computer with the minimum number of configuration changes required on the client computers you should:
    - **Install SQL Server 2000 as the default instance.**
    - **Configure the new instance for Mixed Mode Authentication.**
    - **Configure the server to use the named pipes, IPX/SPX, and TCP/IP Net-Libraries.**

17. There are 12 SQL Server 2000 computers that you administer located in company offices throughout the world. You configure a multiserver job on the server in your company's Houston office. The Houston server is configured as a master server, with the remaining 11 servers designated as target servers. After the job completes successfully, you decide to set up a new SQL Server 2000 computer as the master server. All 12 of the company's current servers will be target servers and the new server will be named JobMaster. To make the Houston server a target server instead of a master server you should **defect all 11 target servers on the Houston server and enlist the 12 servers as target servers on the JobMaster server**.

Creating SQL Server 2000 Databases

Attaching and Detaching Databases

Detaching/attaching is used in moving a database between computers or physical disks. When a database is detached, the links from the server to the data files and transaction logs are removed. When you attach a database, you re-establish those links. When you detach and re-attach, you eliminate the need to restore the database backup manually. When a database is reattached, all files that have changed location must be specified.

Filegroups

Filegroups are database files that have been grouped together for administrative purposes only. There are times when filegroup organization can increase performance, but that is not the primary goal of the filegroup. Filegroups are useful because they increase the flexibility of the database as a whole. Filegroups allow an administrator to partially back up a database. Filegroups also allow greater fault tolerance by granting the ability to spread the database over several physical disks.

The filegroup that contains the primary data file is called the primary filegroup. While other, user defined filegroups can be created, the primary filegroup is the default unless explicitly changed. Also, unless explicitly specified, all files will be placed in the default filegroup. Unless explicitly specified when a database is created, the database will grow as large as necessary until disk space is physically exhausted. Filegroups can be assigned a maximum size and a growth rate that determines how much they will grow when their current space is exhausted. These attributes can be established during filegroup creation and can be changed after the filegroup is created.

Expanding and Shrinking a Database

When a database is created, it is assigned a maximum size and an automatic growth rate to use when the initial file size is exceeded. A database can be manually expanded or shrunk once it has been created.

The CREATE DATABASE Command

The CREATE DATABASE statement allows you to establish many of the options on a database during its creation. The syntax for the CREATE DATABASE statement is as follows:

```
CREATE DATABASE <database_name>
[ON <filegroup_name>
  NAME = <filename_1>,
  FILENAME = 'drive:\directory\filename.mdf',
  SIZE = <initial size in MB>,
  MAXSIZE = <maximum size in MB>,
  FILEGROWTH = <size in MB to add >
]
```

The ALTER DATABASE Command

The ALTER DATABASE command is used to change a data file after it has been created. The syntax for the ALTER DATABASE statement is as follows:

```
ALTER DATABASE <database_name>
{    ADD FILE [TO FILEGROUP filegroup] [FOR RESTORE]
     (FILENAME = 'drive:\directory\filename.mdf'
```

```
        SIZE = <Size in MB>
        MAXSIZE = <size in MB>
        FILEGROWTH = <size in MB to add>
    TO FILEGROUP <filegroup_name>
    }
```

The ADD FILE can also be replaced with the following statements to achieve the following results:

Statement	Purpose
ADD LOG FILE	Add a log file to the database
REMOVE FILE	Remove a data, transaction, or log file from a database
ADD FILEGROUP	Add a filegroup to a database
REMOVE FILEGROUP	Remove a filegroup from a database
MODIFY FILE	Change the attributes (SIZE, MAXSIZE, or FILEGROWTH) of a file in the database
MODIFY FILEGROUP	Change the changes the properties on a specified filegroup (READONLY, READWRITE, DEFAULT)

Transaction Logs

A transaction log records all changes made to a database and allows changes to be rolled back or forward in the event of a system failure or in the case of an erroneous entry. The transaction log is associated with the data files when the database is created. The transaction log actually consists of several files that exactly describe the changes made to the database. These files are treated by the database as one file for the purposes of space and placement.

There comes a point when old log records are no longer needed for recovering or restoring a database and must be deleted to make room for new log records. The process of deleting these log records is called truncating the log.

If possible, transaction logs should be placed on a different physical disk than the data files. This placement will increase performance by removing competition for hard drive access time from the transaction logs and the data files.

Write-Ahead Transaction Log

SQL Server 2000 uses a write-ahead log. A write-ahead log ensures that no data modifications are written to disk before the associated log record. SQL Server 2000 maintains a buffer cache into which it reads data pages. Data modifications are not made directly to disk, but to the copy of the page in the buffer cache. The modification is not written to disk until the lazywriter process schedules a write for the page.

At the time a modification is made to a page in the buffer, a log record is built in the log cache recording the modification. This log record must be written to disk before the associated dirty page is flushed from the buffer cache to disk. If the dirty page were flushed before the log record, it would create a modification on disk that could not be rolled back if the server failed before the log record were written to disk. SQL Server has logic that prevents a dirty page from being flushed before the associated log record. Because log records are always written ahead of the associated data pages, the log is called a write-ahead log.

Managing Database Objects

Tables

Tables are database objects that contain all data in a database. Tables organize data into a row and column format where each row represents a unique record and each column represents a field in that record. Tables are created by the CREATE TABLE command. Changes to a table can be made using the ALTER TABLE command.

Indexes

Indexes are objects that are associated with tables that speed the retrieval of rows in the table containing keys that have been built from one or more columns in a table. Indexes come in two varieties, clustered and nonclustered. A clustered index sorts and stores rows of data in a table based on the key values. Nonclustered indexes do not affect how data rows are sorted in a table. Nonclustered indexes contain values that point to rows in the indexed table.

Constraints

Constraints are conditions placed on data entered into a column that are meant to ensure data integrity. Generally constraints are placed on the column designated as the primary key. Constraints are by no means exclusive to the primary key column but they are a useful way to ensure that all row values in the primary key column are unique.

A foreign key is a column that is used to establish and enforce a link between data in two tables. Foreign key constraints are used to maintain referential integrity between tables by controlling the data that can be entered and the changes made to foreign keys in referring tables.

Stored Procedures

Stored procedures are collections of SQL statements residing on the server that can be executed by users to achieve a desired goal. Stored procedures can have values or variables that can be passed to them by the user or the system that they can act upon. Since they are located on the server, stored procedures can greatly increase performance as there is only one authentication, the authentication for the initial execution of the stored procedure, for all the commands that are executed. You can nest stored procedures and can call other procedures within the body of a stored procedure.

Stored procedures are useful in implementing consistent logic across applications. In a stored procedure SQL statements and logic needed to perform a commonly performed task can be designed, coded, and tested once. Each application needing to perform that task can then simply execute the stored procedure. A structure using a stored procedure offers a single point of control for control of business processes.

Triggers

Triggers are stored procedures that are activated on an event driven basis, instead of being explicitly called. Triggers are typically used to maintain data integrity, cascade referential actions, or to utilize external resources (e.g. send an e-mail, notify a pager, etc.). There are two types of triggers; AFTER triggers and instead-of triggers.

AFTER Triggers

The default type of trigger is the AFTER trigger. On a table there can be multiple triggers defined for an event. You can designate the first and last triggers to fire but you cannot control the order that any other triggers will fire.

INSTEAD OF Triggers

INSTEAD OF triggers are set up to specify the action to take when data is modified. INSTEAD OF triggers must be explicitly declared as instead-of or they will default to AFTER triggers. You can only have one IN STEAD OF trigger for each action.

Views

A view is a filter that dynamically interprets information in one or more tables and creates a composite table that other operations can be performed upon. Rows can be updated, deleted, or inserted from views. A view is not an actual table, but it is presented in table format.

Creating SQL Server 2000 Database Keypoints

1. The SQL Server 2000 computer that you administer contains a 4-GB database named *Marketing*. Your goal is to remove the *Marketing* database from one server and add it to another as quickly as possible. You should **detach the database from the original server by using the sp_detach_db stored procedure**. You should then **copy the database and the transaction log files to the new server, and attach them by using the sp_attach_db stored procedure**.

2. Each night on the SQL Server 2000 computer that you administer, you need to e-mail a copy of the server's MSDB database to the administrator of another SQL server computer. You create a job that contains the following:
   - Detach the msdb database.
   - Send the database to the administrator in an e-mail message.
   - Re-attach the msdb database.

   The job fails on the first step when you attempt to test it. To mail MSDB database to the administrator every night you should **configure the job to backup the MSDB database to a temporary file** and **send the file to the administrator in an e-mail message**.

3. The SQL Server 2000 computer you administer contains a database that stores information about customer complaints. The customer service representatives in your company add and edit the information in the complaints table by using a number of client applications. These applications include a web-based application and a Windows 32-bit application. The customer service manager discovers that some customer complaints are marked as closed before they are resolved to the customer's satisfaction. To notify the customer service manager whenever a compliant is marked as closed without making any changes to client applications you should **create an UPDATE trigger that sends an e-mail message**.

4. You need to create a new database named *Tracking* that employees in your company will use to track inventory data. Users will require immediate responses to queries that help them locate where parts are stored. The database will consume 14 GB of disk space. The courses of action you should take to configure the data files and transaction log to accelerate query response time are:
   - **On drive C, create a transaction log.**
     On drive D, create a data file in the PRIMARY filegroup.
     On drive E, create a data file in the SECONDARY filegroup.
   - **On the PRIMARY filegroup, create all tables.**
     On the SECONDARY filegroup, create all indexes.

5. You want to import a table of geographic information from a Microsoft Access database into a SQL Server 2000 database. The table consists of 12,000 rows and each row averages 5,000 bytes. The table contains lookup data that does not change. You want to minimize the size of the data file and the time required to back up the data. The two actions you should take are:
   - **Create a 95 MB data file named *geography.ndf*.**
   - **Place the table in a new filegroup named LOCATION.**

6. The SQL Server 2000 computer you administer contains a database that consumes about 30 GB of disk space. The data files are configured as shown:

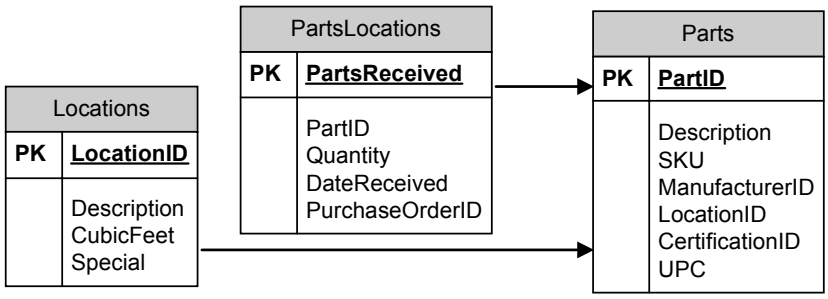
Filename	Locations	Space Allocated (MB)	Filegroup
MedicalRecords_data	E:\Program Files\Micro	10,000	PRIMARY
MedicalRecords_data	F:\data\MedicalRecords	10,000	PRIMARY
MedicalRecords_data	G:\Program Files\Micro	20,000	PRIMARY

You run full database backups each night after business hours. Lately the backup has not been complete by the time the morning shift begins to use the database. You need to minimize the time needed to restore data in the event of a system failure, and you also want to reconfigure the database to allow the backups to complete before the morning shift begins to use the database. The two actions should you take are:

- **Reorganize the data files into three groups.**
  Place the system tables in the PRIMARY filegroup, the indexes in a filegroup, and the tables in the other filegroup
- **Back up the transaction log each night.**
  Run a filegroup backup on a different filegroup each night.

7. The SQL Server 2000 computer you administer contains a database to which you want to import numeric data from other SQL server computers through the use of the data transformation services package. The precision and scale values of this data are not defined consistently on the other servers. To prevent any loss of data during the import operations you should **write Microsoft ActiveX scripts for each DTS transformation**, and then **use the scripts to recast data types to the destinations precision and scale values**.

8. The SQL Server 2000 computer you administer contains a database named *Inventory* that has a Parts table. The Parts table has a field named InStock. When parts have shipped, a table named PartsShipped is updated. When the parts are received, a table named PartsReceived is updated. The relationship of these tables is shown:



To update the InStock field automatically you should **add triggers to the PartsShipped and the PartsReceived tables that update the InStock field in the Parts table**.

9. The SQL Server 2000 computer that you administer contains a database that has data files configured as shown:

File Name	Location	DiskSpace (MB)	Filegroup
Data1	E:\Program files\	8000	Primary

The database has grown to 7 GB. The server has no more available disk space. To make more disk space available in the database without affecting system performance you should **add a new hard disk to the server** and **create a new file on the new hard disk as part of the PRIMARY filegroup**.

10. The SQL Server 2000 computer that you administer contains a database containing 1.5 GB of data. The server has one 9 GB hard disk. You need to import an additional 2 GB of data into the database without adversely affecting database performance. You should **move the transaction log file to drive E, and set the file growth of Acc_Data.mdf by selecting the "Unrestricted file growth" option**.

11. The SQL Server 2000 computer you administer contains a database that your company uses to store the sales department's responses to requests for price quotes. Developers in your company create an application used for saving quotations. A data entry application executes the following transact-SQL statement:

```
UPDATE  QuotationSummary
SETCustomerName = 'Craig Lawrence'
WHERE QuotationID = 13
```

When the application executes the statement, the developers receive the following error:

```
Server: Mrh4402, Level 16, Stats 1, Line 1
View or function 'QuotationSummary' is not updateable because it con-
tains aggregates:
```

To allow developers to use this UPDATE statement without generating errors you should **create an INSTEAD OF trigger on the view to update the composite tables**.

12. The SQL Server 2000 computer you administer contains a database named *Inventory* which has a table named StorageLocations that stores the location of parts contained in your company's warehouses. The StorageLocations table is configured as shown:

Column Name	Data Type	Length	Allow Nulls
StorageLocationID	int	4	
LocationDescription	varchar	50	
PartID	int	4	
UnitsStored	bigint	8	
DateStored	datetime	8	
TagID	int	4	

The LocationDescription field is usually described with a 10 to 25-character name. There are never more than 100,000 units on hand of any given part. You want to modify the table's schema to save space. You cannot lose any existing data. Using the minimum amount of administrative time and server resources, which Transact-SQL statement should you execute?

```
ALTER TABLE dbo.StorageLocations
ALTER FIELD UnitsStored int NOT NULL
```

13. The SQL Server 2000 computer that you administer contains a database named MedicalRecords. The database tables are configured as shown:

Column Name	Data Type	Length	Allow Nulls
PatientID	Int	4	
Title	Char	4	Yes
First Name	Char	25	
Last Name	Char	35	
Address	varchar	50	
Address2	varchar	50	Yes
City	Char	25	
State	Char	2	
ZIP	Char	10	
Phone	Char	15	Yes

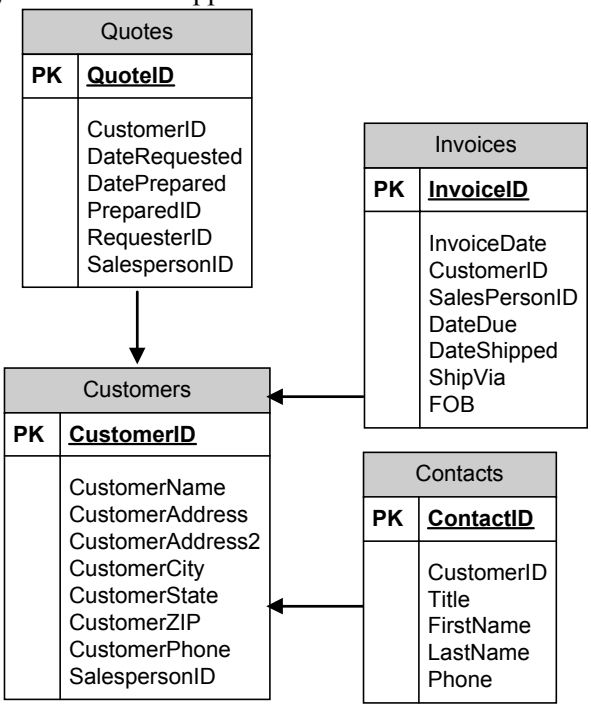
The existing PatientID field is an identity field and according to new government regulations, patient IDs must include a three-character prefix before the numeric portion of the ID defined by the patient's home address. The actions you should take to comply with the new regulations while minimizing changes to the database are:

- **Drop the FOREIGN KEY constraints. In each table, change the data type of the PatientID field to char.**
- **In each table, update the PatientID field.**
  Re-create the FOREIGN KEY constraints.

14. The SQL Server 2000 computer you administer contains a database named *Seaworthy*. Users are reporting that they cannot add new data to the database. You examine the E drive. To allow users to add data while minimizing administrative overhead you should **increase the maximum file size of Sales_Data to 1,500MB**.

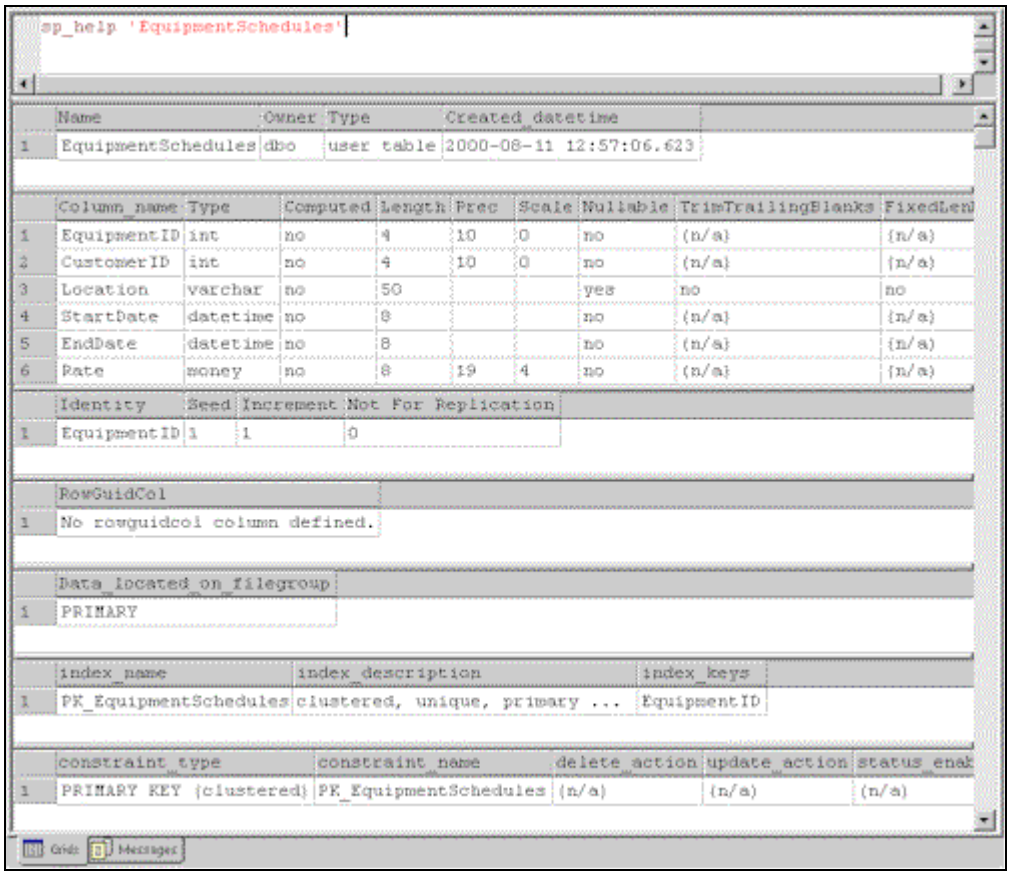
15. The SQL Server 2000 computer that you administer contains confidential information about contracts that your company has bid on. Company policy dictates that information on the bid information be permanently removed from the database one year after the bid closes. To comply with this policy and minimize server overhead you should **create a stored procedure to delete any bids with a closing date older than one year** and **use SQL server agent to schedule the stored procedure to run every night**.

16. The SQL Server 2000 computer that you administer contains a database named *sales*. In the *sales* database you need to change the way customer IDs appear in the customers table. The database schema is as follows:



To automate the process of updating the primary key tools while minimizing record locks and administration within the database during the update process you should **add ON UPDATE CASCADE constraints to the CustomerID field in the Customers table**. You should then **modify the values in the CustomerID field in the Customers table**.

17. The SQL Server 2000 computer you administer contains a database named Inventory that contains a table used to store information about equipment scheduling. The EquipmentSchedules table is configured as shown:



```
sp_help 'EquipmentSchedules'
```

	Name	Owner	Type	Created_datetime
1	EquipmentSchedules	dbo	user table	2000-08-11 12:57:06.623

	Column_name	Type	Computed	Length	Prec	Scale	Nullable	TrimTrailingBlanks	FixedLenN
1	EquipmentID	int	no	4	10	0	no	(n/a)	(n/a)
2	CustomerID	int	no	4	10	0	no	(n/a)	(n/a)
3	Location	varchar	no	50			yes	no	no
4	StartDate	datetime	no	8			no	(n/a)	(n/a)
5	EndDate	datetime	no	8			no	(n/a)	(n/a)
6	Rate	money	no	8	19	4	no	(n/a)	(n/a)

	Identity	Seed	Increment	Not For Replication
1	EquipmentID	1	1	0

	RowGuidCol
1	No rowguidcol column defined.

	Data_located_on_filegroup
1	PRIMARY

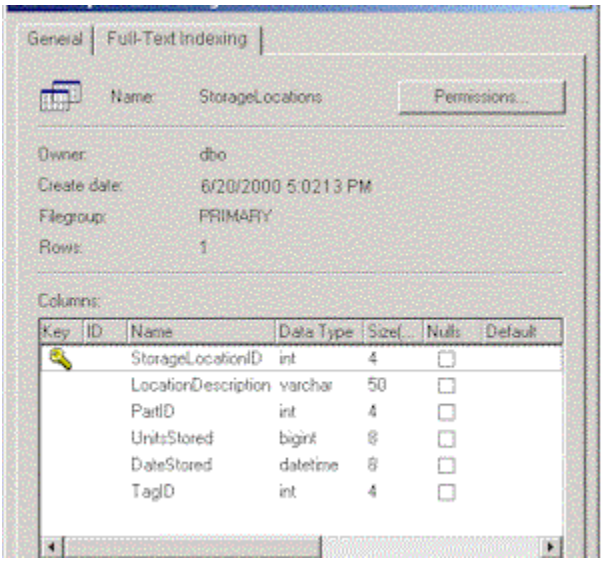
	index_name	index_description	index_keys
1	PK_EquipmentSchedules	clustered, unique, primary ...	EquipmentID

	constraint_type	constraint_name	delete_action	update_action	status_enab
1	PRIMARY KEY (clustered)	PK_EquipmentSchedules	(n/a)	(n/a)	(n/a)

Grids Messages

Users are reporting that some equipment schedules have an end date earlier than the start date. You need to ensure that the start date is always earlier than or equal to the end date while minimizing physical I/O. You cannot allow users to change the Transact-SQL statements they use to modify data within the database. You should **create a constraint that compares the start date to the end date**.

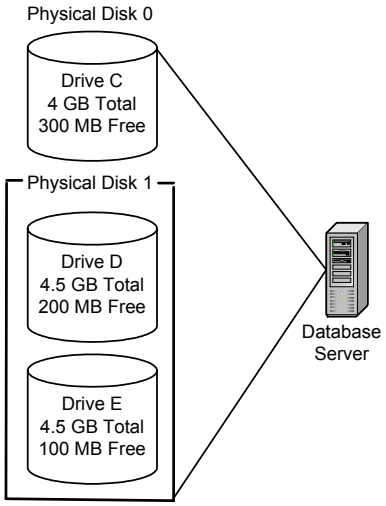
18. The SQL Server 2000 computer that you administer contains a database named Inventory. Users report that several storage locations in the UnitsStored field in the Inventory database contain negative numbers. The table properties are configured as shown:



You correct all the negative numbers in the table, but you must prevent the database from storing negative numbers and minimize use of server resources and physical I/O. The Transact-SQL statement you should execute is:

```
ALTER TABLE dbo.StorageLocations ADD CONSTRAINT
CK_StorageLocations_UnitsStored
CHECK (UnitsStored >= 0)
```

19. The SQL Server 2000 computer that you administer contains a database named *Contracts*. The server is configured as shown:



The database files are configured as shown:

Database Properties-Contracts

File Name	Location	Initial Size(MB)	Filegroup
Contracts_data	E:\data\contracts_data	2000	PRIMARY
Contracts_data2	E:\data\contracts_data2	2000	SECONDARY

As database developers have been creating new tables and indexes, they have not been specifying filegroups for the objects. As a result, the primary filegroup is reaching its maximum capacity. The developers must be able to continue adding new objects, but you do not want then to change the way they create objects. To ensure that you do not run out of disk space and minimize the time it takes to administer the database you should **set the SECONDARY filegroup as the default filegroup**.

Managing, Monitoring, and Troubleshooting SQL Server 2000 Databases

Optimizing Database Performance

Database performance is an important part of administering a SQL Server 2000 environment. There are many ways that performance can be increased including indexing, locking and recompiling.

Indexing

Establishing indexes can dramatically increase query performance, but can also adversely affect data modification performance. Any changes to an indexed table have to be reflected in the indexes to that table, and all changes are logged. These factors combine to produce a slowdown in data modification tasks.

Locking

To ensure that data changed by one user does not unexpectedly affect another user's operations, SQL Server 2000 uses locks. Locking holds data that is in use by another process. The extent of the hold depends on the options that have been configured on the lock. Locks can decrease system performance if over-implemented. There are four transaction isolation levels that are enforced through locking:

Isolation Level	Definition
Read Uncommitted	Transactions are isolated enough to prevent reading physically corrupt data
Read Committed	SQL Server default level
Repeatable Read	Transactions acquire read and write locks on the rows that they utilize
Serializable	Transactions are completely isolated from each other

Recompiling

When a SQL statement is compiled, it is also optimized for the database object that it is working on. Optimization of a SQL statement is based on the information available at the time. Since data or structures may change over time, it may be necessary to recompile SQL statements to maintain performance.

Statistics

Statistics contain information on how key values are distributed in a table. Statistics are useful in increasing query response time as they give the query engine a better idea on where to begin a search for a particular value in a table. As data changes in a table, statistics should be updated to reflect the changes. Tables can be configured to automatically create and automatically update statistics.

Optimizing Data Storage

Optimizing Filegroups

There are situations when certain systems can improve performance by controlling the placement of data and indexes onto specific disk drives. can aid this process by specifying the drives on which files will reside. The system administrator can create filegroups for each disk drive and assign specific tables or indexes from a table to specific filegroups.

Managing Database Fragmentation

There are two types of fragmentation in a SQL database, internal and external fragmentation. Internal fragmentation occurs when indexes are inefficiently using space. External fragmentation occurs when the physical order of pages does not match the logical order. The best way to rid your database of fragmentation is to use the

DBCC INDEXDEFRAG command. This command will defragment the index while still allowing access to the table the index refers to.

Disaster Recovery

Backup Operations

There are three different kinds of backups that can be performed on a SQL 2000 Server database:

- Full backup: copies all database files, including transactions logs, data files, and indexes.
- Differential backup: copies only the files that have changed since the last full backup was made.
- Log backup: copies all log records that have been written to the transaction log since the last full or log backup was made.

Recovery Methods

Setting the recovery method determines how quickly you can restore transaction log backups in the case of hardware failure. Recovery method also dictates the size of the resulting transaction log backup and the degree to which the database is secure from losing committed transactions upon restoration.

SIMPLE Model

The simple recovery model requires the lowest amount of system resources. The system often truncates the transaction log which means that only full and differential backups are allowed.

BULK_LOGGED Model

The recovery model allows you to completely restore your database in case of hardware failure. Operations are fully, but only minimally, logged. This leads to a middle ground compromise between disk space utilization, speed, and safety.

FULL Recovery Model

A transaction log that operates using the full recovery model is has the least risk of losing transactions. All events are fully logged. This logging is useful for restoration but it can lead to massive transaction log sizes and disk space complications.

Log Shipping

Log shipping is a feature that backs up transaction logs and transmits those backups to a fallback server that is to be used in case of failure of the primary server. Log shipping automates these backups at a user-specified interval and automatically restores them on the backup server. To implement log shipping the logon IDs for both the original server and the fallback server must be synchronized to enable users to log into the server in case of original server failure. The SIMPLE recovery mode cannot be used in log shipping as it does not allow transaction log backups.

Integrity checks

Database Consistency Checker (DBCC)

The DBCC is a collection of utilities that are used to check the integrity of a database. These utilities can be used to detect and repair problems in a database. Commands that are included in the Database Consistency Checker are preceded with DBCC. Consistency checking commands in the DBCC include the following:

Statement	Purpose
CHECKALLOC	Checks the allocation and use of all pages in the specified database
CHECKCATALOG	Checks consistency between system tables and specified database
CHECKDB	Checks disk space allocation and integrity of all objects in the database
CHECKFILEGROUP	Checks the disk space allocation and integrity of all tables in the specified filegroup
CHECKIDENT	Checks the current identity value and corrects it if needed
CHECKTABLE	Checks the integrity of the data and indexes for a given page
DBREINDEX	Rebuilds more indexes for a table in the specified database
INPUTBUFFER	Shows the last statement sent to the server from the client
SHOWCONTIG	Shows fragmentation information for the data and indexes for the specified table
SHOWSTATISTICS	Shows the statistics for the current table
SHRINKDATABASE	Shrinks the size of the data files in a database by a specified percentage (if possible)
SHRINKFILE	Shrinks the size of a data or log file to a specified size (if possible)
TRACEON	Enables a specified trace flag
TRACEOFF	Disables a trace flag
TRACESTATUS	Displays the status of trace flags
UPDATEUSAGE	Reports and corrects inaccuracies in the sysindexes table
USEROPTIONS	Returns the user options for the current user connection

Database Maintenance Plan Wizard

The Database Maintenance Plan Wizard is a utility that aids in creating a maintenance plan for a SQL Server 2000 database. You can use the Database Maintenance Plan Wizard to create and schedule a backup scheme that will backup on a designated schedule or will backup based on a defined level of database activity. As part of the backup process, the Database Maintenance Plan Wizard can configure log shipping. The Database Maintenance Plan Wizard can also be used to automatically run maintenance scripts on a predefined schedule.

Troubleshooting Transactions

SQL Profiler

The SQL Profiler is a program that captures events from a server. The captured events are saved in a trace file that can later be analyzed or used to replay a specific series of steps when trying to diagnose a problem. The SQL Profiler is used for activities such as:

- Stepping through problem queries to determine the causes of problems
- Finding and diagnosing slow running queries
- Capturing the series of SQL statements that lead to a problem
- Monitoring the performance of SQL Server to tune workloads

SQL Server Enterprise Manager

SQL Server Enterprise Manager is the primary administrative tool for SQL Server and provides an MMC-compliant user interface that allows users to:

- Define groups of servers running SQL
- Register individual servers in a group

- Configure all SQL Server options for each registered user
- Create and administer all SQL server databases, objects, logins, users, and permissions in each registered server
- Define and execute all SQL Server administrative tasks on each registered server
- Design and test SQL statements, batches, and scripts interactively
- Invoke various wizards defined for SQL server

Managing, Monitoring, and Troubleshooting Keypoints

1. The SQL Server 2000 computer you administer contains a database with two database files and one transaction log file. Each data file is in its own filegroup and is located on its own hard disk. You perform full database, differential, and transaction log backups on a regular basis. The weekly backups are striped across three disk backup devices. A portion of the header information for the current week's backups is as follows:

```
Backup Name          Backup Type   Backup Finish Date
-------------------  -----------   ----------------------
sales_db_10000802    1             2001-08-02 0:57:04.000
Sales_t1_10000803_1  2             2001-08-03 0:04:12.000
Sales_t1_10000803_2  2             2001-08-03 4:04:23.000
Sales_df_20000803    5             2001-08-03 0:15:41.000
```

   On August 5, 2001, at 12:47 p.m., the hard disk that contains the PRIMARY filegroup fails. To recover as much data as possible you should
   - **Restore the most recent full database backup.**
   - **Restore the most recent differential backup.**

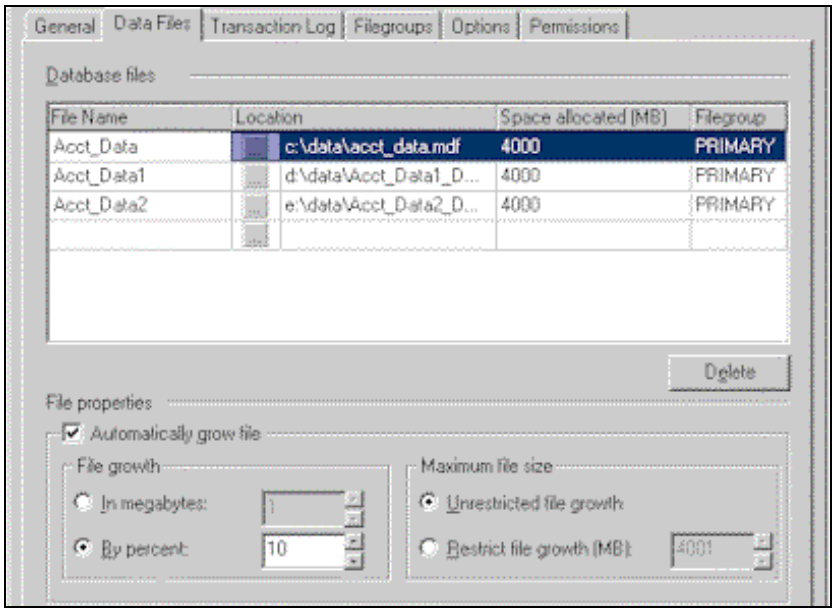
2. On the SQL Server 2000 computer that you administer you import a large amount of data into the *Contracts* database. You perform a full database backup after the import. After the backup, you examine the hard disk and find that the log file has grown from 200MB to 800MB. The Transact-SQL statements that you should execute to recover disk space as quickly as possible without losing backup information are:
   - **BACKUP LOG Contracts TO disk= 'F: \Backups\Contracts_Log.bkp'**
   - **DBCC ShrinkFile (Contracts_Log, 200)**

3. The SQL Server 2000 computer that you administer contains a database named *CustomerManagement* that tracks customer requests for product information. *CustomerManagement* is divided into two filegroups. The PRIMARY filegroup stores current transactional data, and the SECONDARY filegroup stores historical data. Your backup strategy includes full database and transaction log backups. A portion of the header information for the current week's backups is shown:

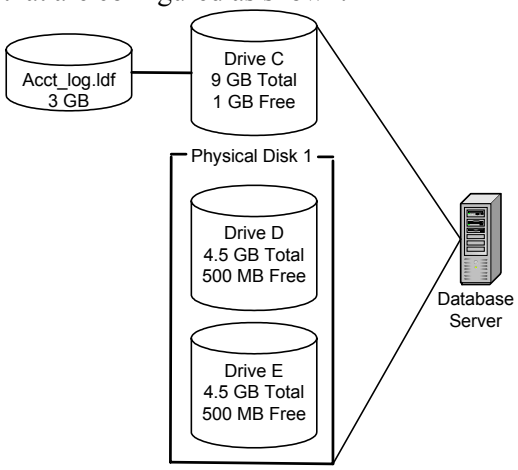
```
BackupName                 BackupType  BackupFinishDate
-------------------------- ----------- -----------------------
custmgmt_full_20010806     1           2001-08-06 21:57:03.000
custmgmt_tlog_20010806_1   2           2001-08-06 10:34:21.000
custmgmt_tlog_20010806_2   2           2001-08-06 14:36:32.000
custmgmt_tlog_20010806_3   2           2001-08-06 18:42:11.000
custmgmt_tlog_20010807_1   2           2001-08-07 10:33:37.000
custmgmt_tlog_20010807_2   2           2001-08-07 14:35:18.000
```

   On August 7, 2001, at 3:57 p.m., an employee accidentally deletes the current customer requests. The employee immediately notifies you of the problem. To recover as much data as you can as quickly as possible you should **back up the transaction log**, **restore the PRIMARY filegroup to a new location, restore subsequent transaction logs in sequence, use the STOPAT option to restore the final transaction log backup,** and **import the deleted data from the new location**. This will also minimize server downtime.

4. The SQL Server 2000 computer you administer contains a database named *Accounting* that you purge of old records and perform a full backup. The database now uses 4 GB of space and the database files are configured as shown:



The server has two hard disks that are configured as shown:



There is a 3 GB transaction log file for the Acct database that is stored on drive C. You have to make room for a new database that contains a 3 GB data file and a 1 GB transaction log file. To optimize database performance on both databases while minimizing administrative overhead you should **shrink the empty data file on drive E** and **backup and shrink the log file on drive C to 2 GB**. You should then **place the new data file on drive E and the new log file on drive C.**

5. The SQL Server 2000 computer that you administer contains a database with torn page detection enabled. Database backups are made daily. Power is lost for one minute. Once power is restored, the server detects torn pages and the database is marked suspect in the SQL Server Enterprise Manager. To correct the problem you should **restore the suspect database from backups**.

6. The SQL Server 2000 computer that you administer contains a database that stores inventory data. Each weeknight the following database maintenance jobs are scheduled to run:
   - A BULK INSERT job imports data at 10:00 p.m.
   - Indexes are rebuilt at 10:15 p.m.
   - Data integrity checks are performed at 10:30 p.m.
   - A differential backup is performed at 10:45 p.m.
   - A DBCC SHRINKDATABASE job runs at 11:00 p.m.

   You notice that the final job often fails and returns the following error message:

   ```
   "Server Msg 3140, Level 16, State 3.  Could not adjust the space al-
   location for file 'inventory_data'."
   ```

   To ensure that the final job runs without errors you should **increase the time between the differential backup and the DBCC SHRINKDATABASE job**.

7. The SQL Server 2000 computer you administer contains your company's Accounts database. There are many users accessing the database each day. Because you have had power failures in the past, you want to ensure the physical integrity of the Accounts database without slowing down server operations. You should therefore **ensure that write caching disk controllers have battery backups**.

8. The SQL Server 2000 computer you administer contains a database that stores product data. You need to execute an existing stored procedure that examines prices for your company's products and will modify them if necessary. You execute the stored procedure after business hours, but it does not complete. You execute the sp_lock stored procedure and receive the following output:

   ```
   spid   dbid   ObjId       IndId   Type Mode
   ------ ------ ----------- ------- ---- --------
   61     7      0           0       DB   S
   64     7      0           0       DB   S
   72     7      0           0       DB   S
   72     7      2145623952  1       PAG  IS
   72     7      2145623952  0       TAB  IS
   72     7      2145623952  1       KEY  S
   78     7      0           0       DB   S
   78     7      2145623952  1       PAG  IX
   78     7      2145623952  0       TAB  IX
   78     7      2145623952  1       KEY  X
   ```

   To allow the stored procedure to complete successfully you should **release the locks held by connections 72 and 78**.

9. The SQL Server 2000 computer that you administer contains your company's order processing database. Many employees take and enter orders. Your company's disaster recovery policy requires that backups be written to tape and copies of these tapes must be warehoused with a third party. Before orders can be filled they must be entered into the database. You need to create a backup strategy that allows you to recover the order data as quickly as possible in the even of a server failure. The backup strategy must meet company requirements and minimize server workload. The actions should you take are:
   - **Perform a combination of full database, differential, and transaction log backups.**
   - **Back up the data to a network share, and then use enterprise backup software to write the disk backups to tape.**

10. The SQL Server 2000 computer that you administer has two new hard disks on which you will create a database named Inventory. Two actions that you should take to be able to insert, update, and delete data as quickly as possible are:
    - **Configure the hard disks as two independent NTFS volumes**
    - **Place inventory_data.mdf on the first volume and inventory_log.ldf on the second volume**

11. Users are reporting that a database on the SQL Server 2000 computer that you administer is timing out when they attempt to perform queries. Analyzing the server with the System Monitor, you receive the following results:

```
SQLServer:AccessMethods
Full Scans/sec0.000

SQLServer:Latches
Latch Waits/sec14.978

SQLServer:Locks_Total
Lock Requests/sec71371.419

SQLServer:SQL Statistics
Batch Requests/sec60.912
```

    To discover why database performance is slow you should **use the current Activity window to discover who is holding locks in the database**.

12. The SQL Server 2000 computer you administer contains a database named *Inventory*. After developers at your company upgrade an application, users report that when they insert new information in the Locations table the upgraded application returns the following error message:

```
'string or binary data would be truncated.  The statement has been
terminated.'
```

    When you use SQL profiler to trace the activity of the application, you receive the results as shown in the trace exhibit:

Event Class	Test Date	Application Name
SQL: BatchCompleted	Set implicit_transactions of SET T…	MS SQLEM
SQL: BatchCompleted	Use Inventory	MS SQLEM
Audit Login	--Network Protocol : LPC set quoted	Visual Basic
SQL: BatchCompleted	Insert into locations (Descriptions, …	Visual Basic

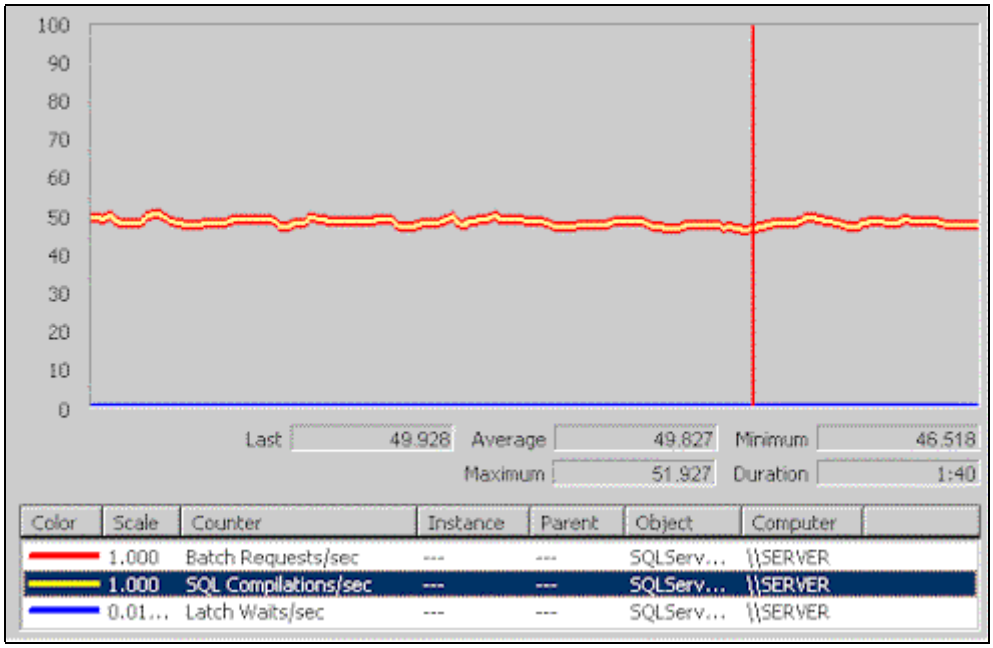
    To configure the database to support both versions of the application without affecting server performance you should **alter the data type of the description field to varchar(50)**.

13. On the multiprocessor SQL Server 2000 computer you administer you are migrating data from other database management systems. You need to load data from two files into a new table on the server. The target table is empty and has no indexes, and you want to complete the data load as quickly as possible. To load the data you should:
    - **Use separate client connections to load the files into the table simultaneously.**
    - **Specify the BULK_LOGGED Recovery model for the database.**

14. The SQL Server 2000 computer that you administer contains a database named MedicalRecords. Users query the database using the following stored procedure:

```
CREATE PROC GetManyVisits
(@PatientID INT,
 @MadeTable TINYINT OUTPUT)
AS
DECLARE @Visits INT
SET @Visits = (SELECT COUNT(*) FROM Cases
                  WHERE PatientID = @PatientID)
IF @Visits > 5
    BEGIN
      SELECT P.PatientName, D.PhysicianName,
       V.DateVisited, V.VisitDuration,
       INTO ##ClientVisits
       FROM Patients P
       JOIN Physician D ON P.PhysicianID = D.PhysicianID
       JOIN Visits V ON P.PatientID = V.PatientID
       WHERE P.PatientID = @PatientID
      SET @MadeTable = 1
    END
ELSE SET @MadeTable = -1
```

Users have been reporting slow query response times. You use System Monitor to measure the system and receive the results shown in the exhibit.



To modify the server to accelerate query response time you should **execute sp_recompile on the stored procedure**.

15. You are the administrator of a SQL Server computer. Users report that the database times out when they attempt to modify data. You use the Current Activity window to examine locks held in the database:
    Spid 41: Locked
    Spid 42: Blocking
    Spid 53: Blocked by 42

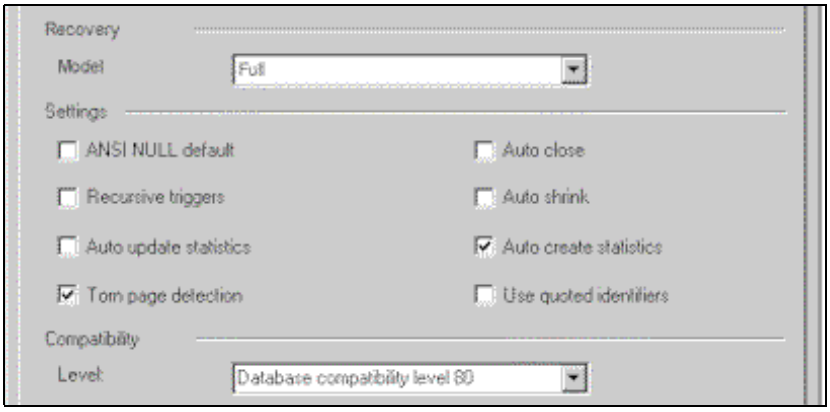
    To discover why users cannot modify data in the database **use the spid 41 icon in the Current Activity window to discover which SQL statement is being executed.** This will also have the effect of not disrupting normal database activities.

16. You are the administrator of a SQL Server 2000 database. Using the default options you configure the database. You drop the indexes to facilitate a large import of data, and rebuild them after the import is complete. Users report that response times from a stored procedure have become slower. In the stored procedure, you examine the following code:

```
CREATE PROCEDURE GetCustomer
( @CustomerID Int )
AS
SELECT FirstName, LastName, Address1, Address2, City,
State, Zip, MAX(PurchaseDate) AS LastPurchase
FROM Customers C
JOIN Addresses A ON A.AddressID = C.AddressID
JOIN Purchases P ON P.CustomerID = C.CustomerID
WHERE C.CustomerID = @CustomerID
GROUP BY FirstName, LastName, Address1, Address2, City, State, Zip
```

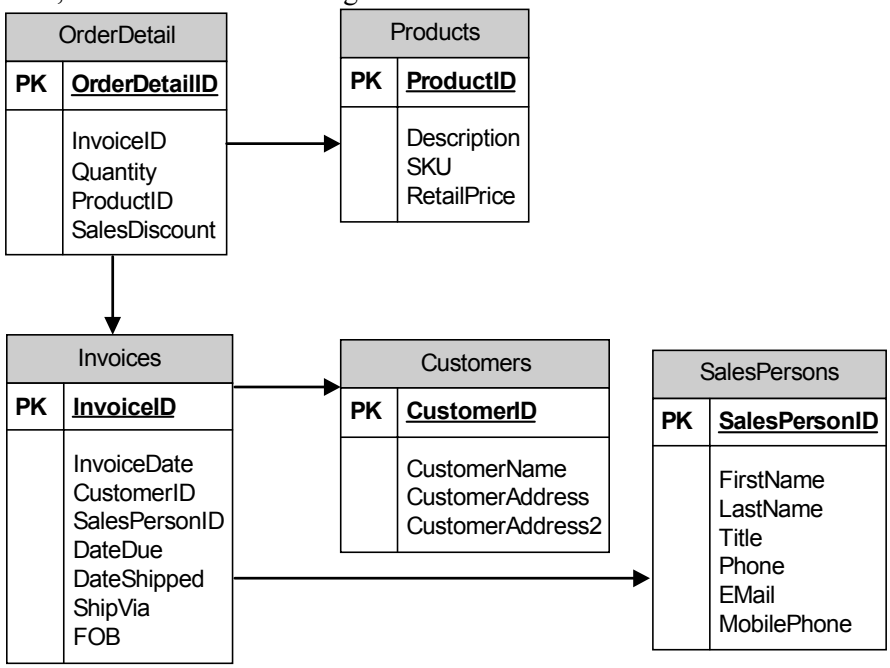
    To improve the performance of the procedure while using the least amount of administrative effort you should **recompile the stored procedure.**

17. The SQL Server 2000 computer that you administer contains a database named *PharmacyRecords*. Users access *PharmacyRecords* by using the CustomerID field, which is the clustered primary key for the Customer table. Users attempting to access *PharmacyRecords* are reporting slow performance. The database options are shown:



    The change that you should make to the database configuration to accelerate query response time while minimizing administrative overhead is to **select the "Auto update statistics" check box**.

18. You are the administrator of a SQL Server 2000 computer. The server contains a database named *Sales*. The database has a group of tables that analysts use to examine sales trends. The database options are set to their default values, and the tables are configured as shown:



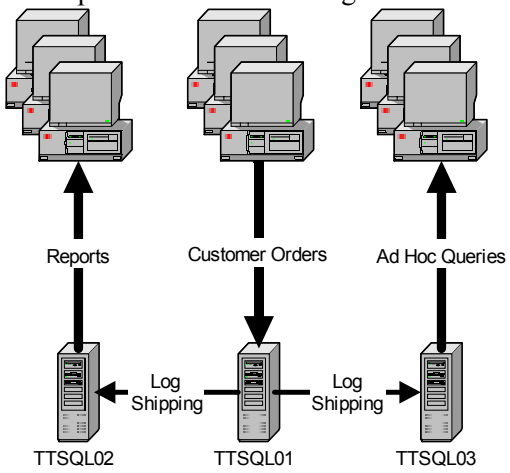
Lately users have been reporting that query performance has become slower. You analyze the clustered primary key on the Invoices table and receive the following results:

```
DBCC SHOWCONTIG scanning 'Invoices' table...
Table: 'Invoices' (21575115); index ID: 1, database ID: 6
TABLE level scan performed.

Pages Scanned.................................: 200
- Extents Scanned.............................: 50
- Extent Switches.............................: 40
- Avg.  Pages per Extent.....................: 4.0
- Scan Density [Best Count:Actual Count].....: 60.00% [3:5]
- Logical Scan Fragmentation..................: 0.00%
- Extent Scan Fragmentation...................: 40.00%
- Avg.  Bytes Free per Page..................: 146.5
- Avg.  Page Density (full)..................: 98.19%

- DBCC execution completed.  If DBCC printed error messages, contact
your system administrator.
```

You want to improve performance of queries that join tables to the Invoices table. Three Transact-SQL statements you can execute to achieve this goal are:

- **CREATE UNIQUE CLUSTERED INDEX PK_Invoices ON Invoices(InvoiceID) WITH DROP_EXISTING**
- **DBCC INDEXDEFRAG ('Sales','Invoices','PK_Invoices')**
- **DBCC DBREINDEX (Sales.dbo.Invoices, 'PK_Invoices')**

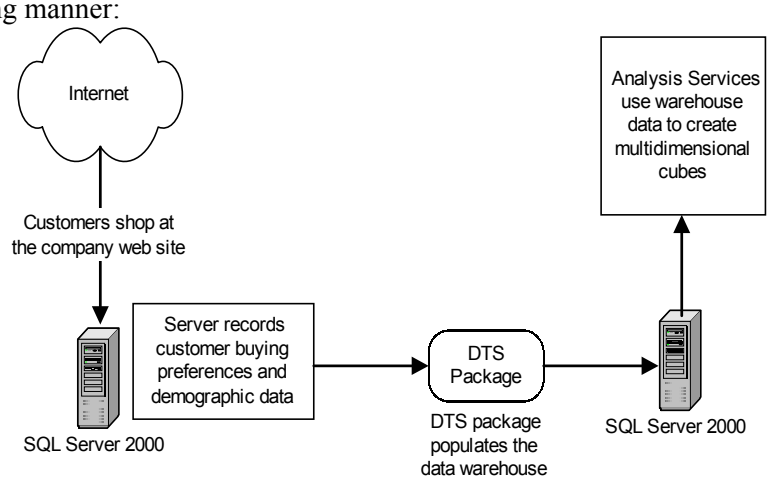
19. You are the administrator of three SQL Server 2000 computers for testkiller.com. One server, TTSQL01, stores other, non-SQL data. You want to be able to use the other two servers, TTSQL02 and TTSQL03, to run reports and answer queries. The planned network configuration is shown:



You want to use the database Maintenance Plan wizard to configure log shipping from TTSQL01 to TTSQL02 and TTSQL03. Users should not be able to add any new data on TTSQL02 and TTSQL03 but should be able to run queries and reports. You should take the following actions:

- **Set the database load state for each destination server to standby mode.**
- **Disable the "Allow database to assume primary role" option for both destination servers.**

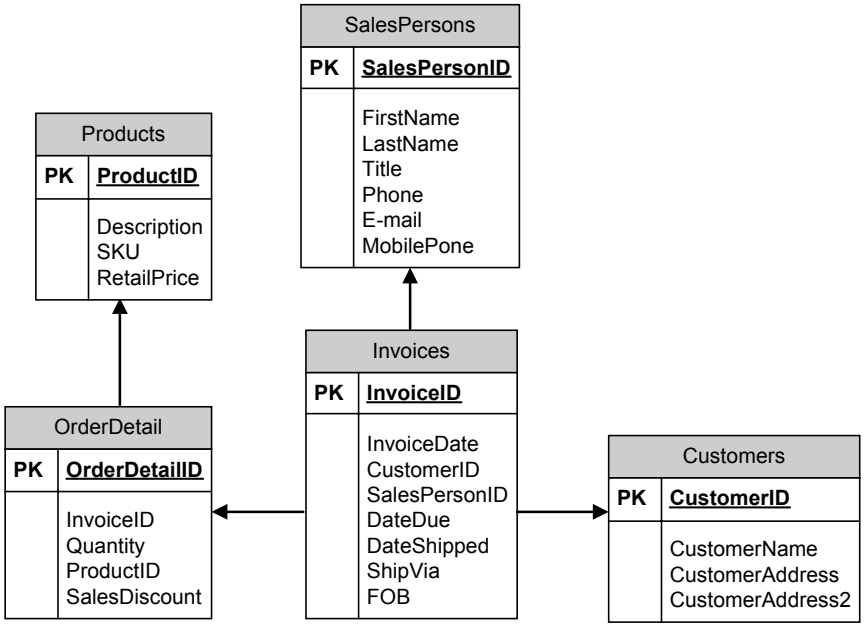
20. You administer two SQL Server 2000 computers for an online retailer. Servers receive and utilize customer data in the following manner:



The Meta Data Services repository on the second SQL Server holds the Data Transformation Services (DTS) package. The actions that you should take to maximize the amount of lineage data that can be recovered if a data file is damaged or lost are:

- **Use the Full Recovery model for the MSDB database.**
- **Back up the transaction log in the MSDB database by using the NO_TRUNCATE option.**

21. You are creating a database that is used as an intermediate data store for a data warehouse. Each night you must import daily data into the database from SQL Server 2000 computers in 120 retail locations. After the data is moved into the data warehouse, the tables are truncated. The database schema is shown:



To configure the data import process to minimize the time needed to import the sales data and administer the database you should **use the Simple Recovery model and the FOR LOAD option to create the database** and **create a Data Transformation Services (DTS) package that uses the BULK INSERT statement to copy the sales data**.

22. The retail company you work for owns 270 stores across the United States. Each store submits approximately 2,500 sales records every month. These records are loaded into a SQL Server 2000 database that you administer at the corporate headquarters. As the records are loaded, a Data Transformation Services (DTS) package writes the transformed sales records to the Sales table, which has a column for integer primary key values. The IDENTITY property automatically assigns a key value to each transformed sales record. While loading incoming sales data for the current month, you discover that a portion of the data contains errors. You cease loading data and identify the problem records. You delete the problem records from the database and you want to reuse the key for the records that you deleted. To assign the deleted key values to the next sales records you load you should **use the DBCC CHECKIDENT statement to reseed the Sales table's IDENTITY property**. This will also have the effect of minimizing disruption to all users' work.

23. The SQL Server 2000 computer that you administer contains a database named Sales. Your company uses an application that uses the Sales database to store sales transactions. Over time, database utilization has steadily increased. You want to automate the collection of data about server resource utilization so that you can plan for future capacity needs. To automate the information gathering process and make sure that it is done as quickly as possible you should **use SQL Profiler to trace server activity and store the results in SQL Server tables**.

24. The SQL Server 2000 computer that you administer contains a database named *Inventory*. Users report that a query responds slowly. The query is as shown:

```
Select Parts.SKU AS SKU, Parts.Description as PartDescription,
Locations.Description AS Location
FROM PartsLocations INNER JOIN Parts ON PartsLocations.LocationID =
Locations.LocationID AND
Parts.LocationID = Locations.LocationID
```

You examine the indexes in the PartsLocations table. The indexes are configured as shown:

Index Name	Index Description	Index Keys
IX_PartsLocations_LocationID_FK	nonclustered located on PRIMARY	LocationID
IX_PartsLocations_PartID_FK	nonclustered located on PRIMARY	PartID
PK_PartsLocations	clustered, unique, pri…	PartID, LocationID

You also examine the contiguity of the table and receive the following results:

```
DBCC SHOWCONTIG scanning 'PartsLocations' table...
Table: 'PartsLocations' (1); index ID: 1, database ID: 6
TABLE level scan performed.
- Pages Scanned................................: 3000
- Extents Scanned..............................: 750
- Extent Switches..............................: 502
- Avg.  Pages per Extent.......................: 4.0
- Scan Density [Best Count:Actual Count].......: 33.33% [1:3]
- Logical Scan Fragmentation...................: 33.33%
- Extent Scan Fragmentation....................: 33.33%
- Avg.  Bytes Free per Page....................: 1166.0
- Avg.  Page Density (full)....................: 85.77%
```

To improve the performance of the query you should **rebuild the PK_PartsLocations index**.

25. The SQL Server 2000 computer that you administer contains a database named *Infrastructure*. You run regular database consistency checks on the server and perform a full backup on the database every two days. The most recent check of the Sales database returned the following message:

```
CHECKDB found 0 allocation errors and 9 consistency errors in the ta-
ble 'Orders' (object ID 214575782).
```

To correct the data integrity errors while minimizing the amount of data lost you should **disconnect users from the *Sales* database**, **enable the single user database option**, and **execute the DBCC CHECKTABLE statement for the Orders table while specifying the REPAIR_REBUILD option**.

26. The SQL Server 2000 computer you administer is used to store confidential information about company employees. You believe that there may be at least one user accessing the employee information database after business hours without company approval. To discover if users are illegally accessing the database and which users are involved you should **use SQL Profiler to capture security events and audit events**. This will minimize additional overhead on the server during business hours.

27. Your SQL Server 2000 computer contains a database named Orders. The Orders database is used to record customer orders for the products your company sells. Your company processes several hundred thousand orders per day and each order uses approximately 100 KB of space in the database. Users are reporting that the database responds slowly when they enter new orders. You use SQL Profiler to monitor the activity on the database and receive the data shown:

Start Time	Event Class
2001-08-08 01:53:13.033	Data File Auto Grow
2001-08-08 01:53:15.147	Data File Auto Grow
2001-08-08 01:53:17.320	Data File Auto Grow
2001-08-08 01:53:19.124	Data File Auto Grow
2001-08-08 01:53:21.589	Data File Auto Grow
2001-08-08 01:53:23.657	Data File Auto Grow
2001-08-08 01:53:25.157	Data File Auto Grow
2001-08-08 01:53:28.985	Data File Auto Grow
2001-08-08 01:53:31.784	Data File Auto Grow
2001-08-08 01:53:32.057	Data File Auto Grow
2001-08-08 01:53:34.198	Data File Auto Grow

To modify the database to improve performance you should **configure the database to automatically grow by 10 percent**.

28. The SQL Server 2000 computer you administer contains your company's investment-tracking database. There are 100 operators who make huge numbers of changes to customer investments. There are daily and monthly reports created from this data. Your company's developers are going to optimize the database application and need a sample of database query activity to determine if query performance can be increased. The developers also plan to replay the sample data on another SQL Server computer. To capture the sample while minimizing any increase to the workload of the server you should **run SQL Profiler on a client computer** and **configure SQL Profiler to monitor database activity, and log data to a .TRC file**.

29. You create a job to perform several database maintenance tasks on the administrator of a SQL Server 2000 computer you administer. The job should be run whenever the server is restarted. Using SQL Server Enterprise Manager, you create a new schedule entry for the job and specify that the job should start automatically. You restart the server to test the job. After the restart, you check the job's history and discover that the job was not run. To ensure that the job runs automatically when the server restarts you should **configure the SQLServerAgent service to start automatically**.

30. On the SQL Server 2000 computer you administer you create a Data Transformation Services (DTS) package, stored as a local package. The package exports data from an online transaction processing (OLTP) database system to an online analytical processing (OLAP) database system that is located on a second SQL Server 2000 computer. To allow the package to execute automatically each evening at 8:00 p.m. you should **use SQL Server Enterprise Manager to create a new job**, **create a CmdExec job step that runs the DTSRUN utility**, and **schedule the job to run each night at 8:00 p.m**.

31. The SQL Server 2000 computer that you administer contains a heavily indexed database that company users query extensively. Over time, the database has grown and query response time has slowed. The database is stored in a single data file. To accelerate query response time you should **create a new filegroup on a new hard disk** and **drop the existing nonclustered indexes, and re-create them on the new filegroup.**

32. There are eight SQL Server 2000 computers that you administer. You configure alerts on each server to report various problem conditions as they occur. You create 20 operator accounts on one of the servers and configure them to use the e-mail and pager contact information for the employees in your department. You configure the alerts on the server to send e-mail messages and pager messages to the appropriate operators. To configure the same 20 operators on the other seven servers with a minimum amount of administrative time you should **use SQL Server Enterprise Manager to script all of the operators** and **use SQL Query Analyzer to run the script on the other seven servers**.

33. The SQL Server 2000 computer that you administer contains two databases. One client/server application accesses one of the databases and a different client/server application accesses the other. Each application uses several stored procedures to query and update its database. Users are reporting that both applications are responding slowly. You use SQL Profiler to monitor activity on the server and receive results of the trace as shown:

Event Class	Test Date
SQL_BatchCompleted	IF @@TRANCOUNT>0 COMMIT TRAN
SQL_BatchCompleted	Update Customer Set CustomerZip=…
SQL_BatchCompleted	Update Products Set SKU = '2323232323…
SQL_BatchCompleted	IF @@TRANCOUNT>0 COMMIT TRAN
SQL_BatchCompleted	Update Customers Set CustomerZip…

To improve the performance of the applications you should **modify the stored procedures so that they select data into temporary tables**.

34. You administer a Windows 2000 computer that also runs SQL Server 2000. The Server contains a database named *Sales*. *Sales* is accessed by a custom application that has just been upgraded and is used by the sales department of your company. A user named Whitney reports that when she attempts to run the weekly analysis report, she receives the following error message:

```
"Server.Msg 3704, Level 16, State 1, Line 1 User does not have per-
mission to perform this operation on table 'Report'."
```

When you log on to Whitney's computer and run the program, you receive no error message. SQL Profiler provides you with the following trace database activity:

Event Class	Text Data
SQL: Batch Completed	Drop table report
SQL: Batch Completed	Drop table report (Productname C. . .

To allow Whitney to run the report and minimize administrative overhead you should **add Whitney's login to the db_ddladmin role in the Sales database**.

35. The Windows 2000 Server computer that you administer is named HQSQL5 and is a member server in your company's Windows NT 4.0 domain, which is named HQMAIN. You install SQL Server 2000 on HQSQL5 and configure the MSSQLServer service account to use the HQMAIN\sqladmin user account. The HQMAIN\sqladmin account is a member of the local administrators group on HQSQL5. You find that you can use the HQMAIN\sqladmin user account to log on the HQSQL5 but the MSSQLServer service fails to start. To start the service you should **grant "Log on as a service" rights to the HQMAIN\sqladmin user account**.

36. On the SQL Server 2000 computer you administer you configure the SQLServerAgent service as shown:
    Service Startup account
    This Account: CORP\SQLAgent

    Mail Session
    Mail Profile
    Save copies of the sent items in the Sent folder

    Error Log
    File Name: E:\Program files\MicrosoftSQLService

    You create several SQL Server agent jobs and configure them to perform maintenance tasks. These maintenance tasks will delete old database records and copy files from the server to file servers on your network. When you test the maintenance routine you discover that none of the jobs will execute. Through SQL Server Enterprise Manager you check the jobs and discover that the SQLServerAgent service will not start. To allow the SQLServerAgent service to start and execute the jobs you should **grant the SQLServerAgent domain user account the "Log on as a service" rights**.

37. The SQL Server 2000 computer you administer contains a database named *Sales*. Users in your company use a customer relationship management application that has just been upgraded to access the Sales database. There is a table named Customer in the Sales database, which stores information about your company's customers. Users report that they are unable to delete information in the CustomerAddress2 column. The Customer table is configured as shown:
    Table Design

```
Name Customers
OwnerDBO
CreateDate8/25/2000 10:42:06 A.M
FilegroupPRIMARY
Rows1
```

    Columns:

Name	Data type	Size	Allow Null
CustomerID	int	4	No
CustomerName	varchar	50	No
CustomerZip	char	10	No
CustomerAddress2	varchar	50	No
CustomerCity	varchar	50	No
CustomerState	char	2	No

    You use SQL Profiler to capture the activity of the application and receive the result shown in the Trace exhibit:

Event Class	Text Data	Application Name
Audit Login	----Network protocol---	Visual Basic
SQL: Batch Completed	Update Customer se. . .	Visual Basic

    To configure the customer's table to support the upgraded application you should **alter the table to allow NULL values in the CustomerAddress2 column**.

38. Your company has an application modified so it can access data from a SQL server database rather than a Microsoft Access database. You create the new SQL server database. After the database is created and implemented, users report slow query response time. You use System Monitor to analyze database activity. To modify the database to accelerate query response time you should **place primary keys and foreign key indexes on the table**.

39. The SQL Server 2000 computer you administer contains a database named *Infrastructure*. A user named Rex reports that he is unable to modify data in the database. When you use SQL Profiler to capture his activities, you receive a report as shown:

EventClass	TestData
ExistingConnection	-- network protocol: LPC set quoted identifier on set implicit transactions off…
ExistingConnection	-- network protocol: LPC set quoted identifier on set implicit transactions off…
SQL: BatchCompleted	Set transaction level serializable begin transaction select * from…

```
set transaction isolation level serializable
begin transaction
select * from Products
update Products
set UnitPrice = 18.50
where UnitPrice = 18.00
commit transaction
```

Rex must be able to modify data and the Transact-SQL statement must be able to execute without compromising data integrity. Two ways to achieve this goal are:

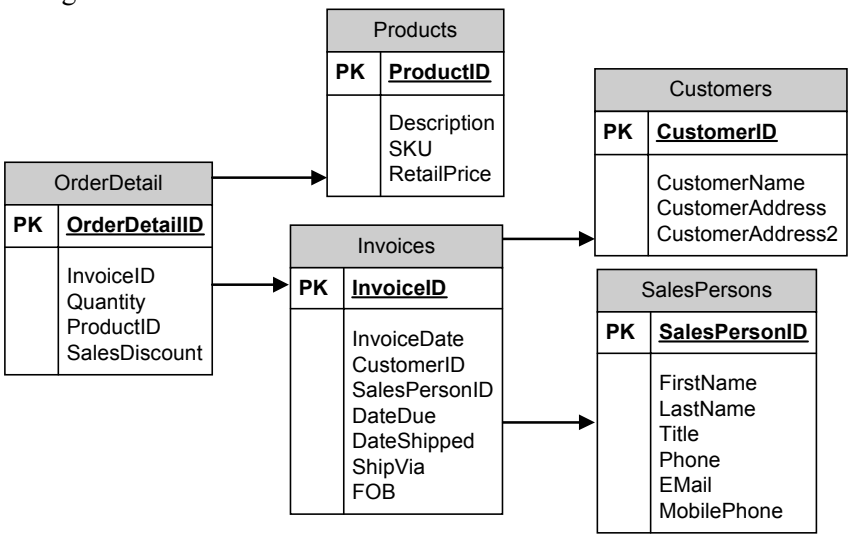
- **Create a stored procedure to execute the T-SQL batch**
- **Add a cascading UPDATE trigger to the products table**

40. The SQL Server 2000 database you administer is named *Articles*. This database is used by a local newspaper as an archive for its articles. Researchers make multiple queries daily to the database. You have created full-text indexes so that users can query the articles quickly. Hardware failures force you to restore the entire database from backup. After the restoration, researchers report that they can no longer run queries without receiving errors. To ensure that researchers are able to query the database you should **rebuild the full-text catalog for the *Articles* database**.

41. The Microsoft Windows NT server 4.0 computer that you administer is the PDC in your domain and also runs SQL Server 2000. The server has four processors and 1GB of RAM. The network consists of 400 client computers that are running Microsoft Windows Millennium Edition. Users report that when they log on in the morning their computers respond slowly and occasionally return error messages. In the System Monitor you check the processor performance and RAM utilization and you discover the %Total Time and %User Time counters average 90% and the %Privileged Time counter averages 20 percent. The available Mbytes counter averages 600MB. To improve server performance for user logons you should **configure SQL server to use only processors 0, 1, and 2**.

42. The Windows NT 4.0 computer that you administer is also running SQL Server 2000 and contains a database named *Sales*. The data file and transaction log for the *Sales* database are configured for automatic file growth and are located on a 9.1 GB hard disk. The data file occupies 7 GB of disk space. To calculate when you will need to add additional disk space you should **configure System Monitor to log disk space utilization**.

43. The SQL Server 2000 computer you administer contains a database named *Infrastructure*. The *Infrastructure* database is configured as shown:



The *Infrastructure* database stores data that is later imported into another database for analysis. Users have been reporting slow response times whenever they enter information. You examine the indexes on the Invoices table that are configured as shown:

Index Name	Index Description	Index Keys
IX_Customers_FK	nonclustered located on PRIMARY	CustomerID
IX_SalesPerson_FK	Nonclustered located on PRIMARY	SalesPersonID
PK_Invoices	clustered, unique, pri…	InvoiceID

To accelerate database response time when users insert new point-of-sale information you should **drop IX_Customers_FK and IX_SalesPerson_FK**.

44. The SQL Server 2000 computer that you administer is a member of a Microsoft Windows NT domain named TKLHQ. Several users who are members of a domain user group named TechWriters need access to this server. The TechWriters group also contains users who should not have access to the server. These users are also members of a domain user group named ContractWriters. You need to allow only the appropriate users to access the server. The Transact-SQL statement should you execute is:

```
EXEC sp_addlogin 'TKLHQ\TechWriters'
EXEC sp_denylogin 'TKLHQ\ContractWriters'
```

45. Users report that queries to a database on a SQL Server 2000 are responding slowly. When you use the System Monitor to examine the subsystems on your server, you find that the %processor time remains at 100 all the time. To modify the server to accelerate query response time you should **add an additional processor**.

46. The SQL Server 2000 computer you administer contains a database named *Exchanges*. The primary table for *Exchanges* is shown:

Column Name	Data Type	Length	Allow Nulls
SalesTransID	int	4	
CustomerName	varchar	50	Yes
CustomerAddress	varchar	50	Yes
CustomerCity	varchar	50	Yes
CustomerState	char	10	Yes
CustomerZip	char	10	Yes
DateSold	datetime	8	Yes
PaymentMethod	char	10	Yes
Item1SKU	char	10	Yes
Item1Qty	int	4	Yes
Item1Price	money	8	Yes
Item2SKU	char	10	Yes
Item2Qty	int	4	Yes
Item2Price	money	8	Yes
Item3SKU	char	10	Yes
Item3Qty	int	4	Yes
Item3Price	money	8	Yes
SalesTax	money	8	Yes

Users have been reporting that when they add new sales, the insert operations often time out. You examine the indexes for the *Exchanges* table and receive the following information:

Index Name	Index Description
IX_CustomerName	nonclustered located on PRIMARY
IX_CustomerState	nonclustered located on PRIMARY
IX_CustomerZip	nonclustered located on PRIMARY
IX_PaymentMethod	nonclustered located on PRIMARY
PK_SalesTransactions_1	nonclustered, unique, primary key located on PRIMARY

You want to decrease the time required to store purchases made at the retail outlets. What are two possible ways to achieve this goal?

- **Normalize the data structure of the table.**
- **Drop the nonclustered indexes of the table.**

47. Users are reporting that a database you administer responds slowly when they insert data. When you check System Monitor you receive the following performance data:

```
Memory
Pages/sec          244.443

PhysicalDisk       _total
%Disk Time        240.198

Processor          _total
% Processor Time   2,000
```

To modify the server to improve its performance you should **increase the amount of RAM**.

48. The SQL Server 2000 computer that you administer contains a database named *Sales*. Users in your company who analyze customer buying patterns frequently use the following query:

```
SELECT Customers.CustomerID,
Invoices.InvoiceID,
Invoices.InvoiceDate,
OrderDetail.Quantity, Products.SKU
FROM Customers
INNER JOIN Invoices ON Customers.CustomerID = Invoices.CustomerID
INNER JOIN OrderDetail ON Invoices.InvoiceID = OrderDetail.InvoiceID
INNER JOIN Products ON OrderDetail.ProductID = Products.ProductID
WHERE CustomerID = ?
```

The indexes in the Invoices table are configured as follows:

Index Name	Index Description	Index Keys
IX_Customers_FK	nonclustered located on PRIMARY	CustomerID
IX_SalesPerson_FK	nonclustered located on PRIMARY	SalesPersonID
PK_Invoices	clustered, unique, primary key located on PRIMARY	InvoiceID

To optimize the database schema to respond quickly to employee queries you should take the following two actions:

- **Recreate the PK_Invoices index as a nonclustered index.**
- **Recreate the IX_Customers_FK index as a clustered index**

49. The SQL Server 2000 computer that you administer contains a database named *Inventory*. Users report that the following is responding slowly:

```
SELECT parts.SKU AS SKU, Parts, Description AS Part Description,
Locations.Description AS Location AND
FROM PartsLocations INNER JOIN Parts ON PartsLocations.LocationID =
Location.LocationID
```

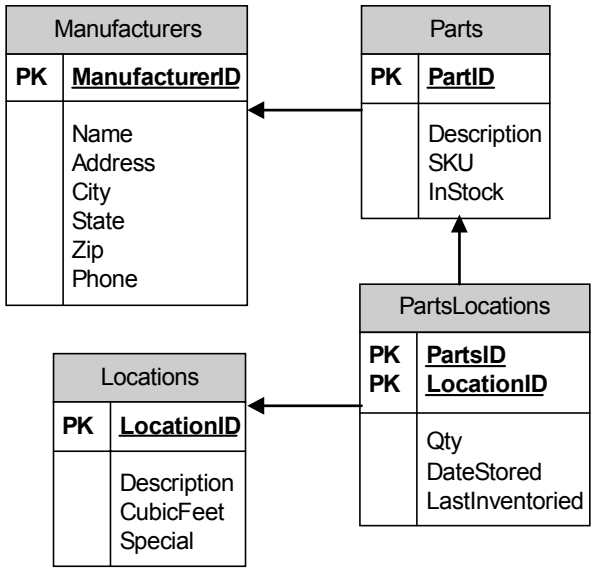
You examine the indexes in the PartsLocations table. You examine the contiguity of the table and receive the following results:

sp_helpindex 'PartsLocations'

Index Name	Index Description	Index Keys
IX_PartsLocations_LocationID_FK	nonclustered located on PRIMARY	LocationID
IX_PartsLocations_PartsID_FK	nonclustered located on PRIMARY	PartsID
PK_PartsLocations	Clustered, unique, primary	PartsID, LocationID

To improve the performance of the query you should **rebuild the PK_PartsLoactions index**.

50. The SQL Server 2000 computer that you administer contains a database named *Inventory*. The database is configured as shown:

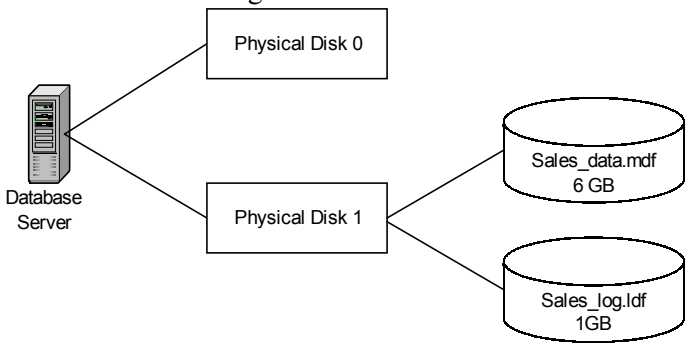


Users frequently need details about parts. Users search for data by using the following query:

```
SELECT Parts.SKU, Locations.Description, Manufacturer.Name, PartsLo-
cations.Qty, PartsLocations.LastInventoried
FROM Parts
INNER JOIN Manufacturer ON Parts.ManufacturerID= Manufac-
turer.ManufacturerID
INNER JOIN Locations ON Parts.LocationID= Loactions.LocationID
WHERE SKU = ?
```

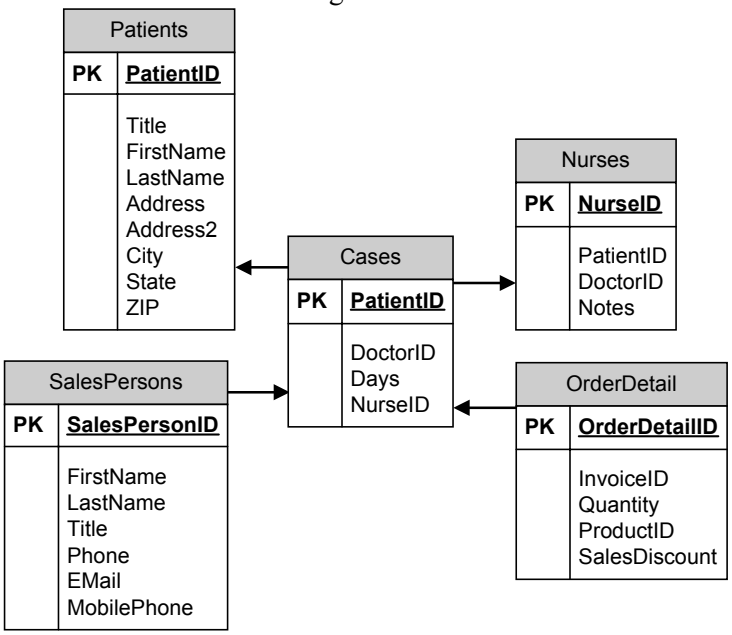
To accelerate query response time you should **create a parameterized stored procedure to retrieve the data**.

51. The Windows 2000 Advanced Server computer that you administer also runs SQL Server 2000 and contains a database named Sales. The server is configured as shown in the exhibit.



The Sales database is configured by using the default options and increases by an average of 50 MB per day. The operator must be notified automatically before the database runs out of disk space. Administrative overhead necessary for this database must be minimized. You should **configure System Monitor to log an alert if the hard disk contains less than 600 MB of free space**. You should then **configure SQL Server Agent to send an e-mail message notifying the operator of the System Monitor alert**.

52. The SQL Server 2000 computer that you administer contains a database named *PharmacyRecords*. You recently added a hard disk to the server and configured the database as shown:



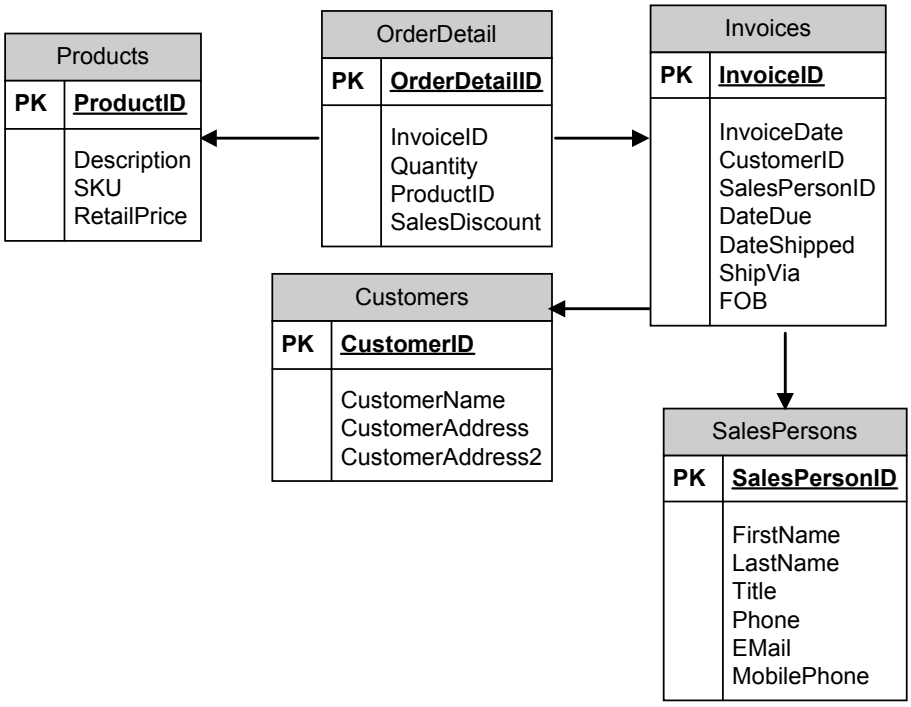
Each primary key is clustered and all foreign keys contain nonclustered indexes. To use the new hard disk to make the database respond more quickly to queries you should **re-create the nonclustered indexes on the SECONDARY filegroup**.

53. The SQL Server 2000 computer you administer contains a database named MedicalRecords. After the data processing department in your company adds new business rules to the database's stored procedures, users report that the medical records application receives time-out errors. You use System Monitor to analyze the server and receive the results shown:

```
SQLServer:AccessMethods
Full Scans/sec              0.000
Index Searches/sec                5782.892

SQLServer:Cache ManagerAdhoc Sql PlansProcedure Plans
Cache Hit Ratio             99.9920.049

SQLServer:Locks_Total
Lock Requests/sec                 9751.032

SQLServer:SQL Statistics
Batch Requests/sec                51.247
SQL Compilations/sec        51.247
SQL Re-Compilations/sec           0.000
```

To tune the stored procedures to improve performance you should **remove the WITH RECOMPILE option from the stored procedure**.

54. The SQL Server 2000 computer you administer contains a database named Sales. Employees use the information stored in Sales to analyze sales data. You configure the database by using the default options and create the database tables as shown:

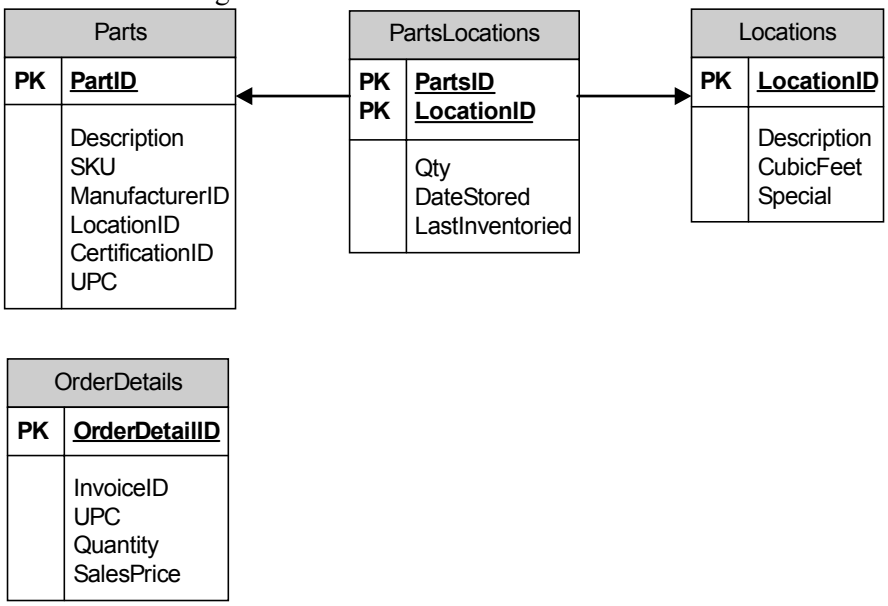


Every morning you import data as a batch process. Users report that the database responds slowly when they run the following query:

```
SELECT Customers.CustomerName, Invoices.InvoiceID, Invoices.InvoiceDate,
OrderDetail.Quantity, Products.Description, Products.SKU, SalesPer-
sons.FirstName, SalesPerson.LastName
FROM Invoices
INNER JOIN Customers ON Invoices.Customer ID = Customer.CustomerID
INNER JOIN OrderDetail ON Invoices.Invoice ID = OrderDetail.InvoiceID
INNER JOIN Products ON OrderDetail.ProductID = Products.ProductID
INNNER JOIN SalesPerson ON Invoices.SalesPersonID = SalesPer-
son.SalesPersonID
```

To improve the performance of this query you should **create indexes on the foreign key fields**.

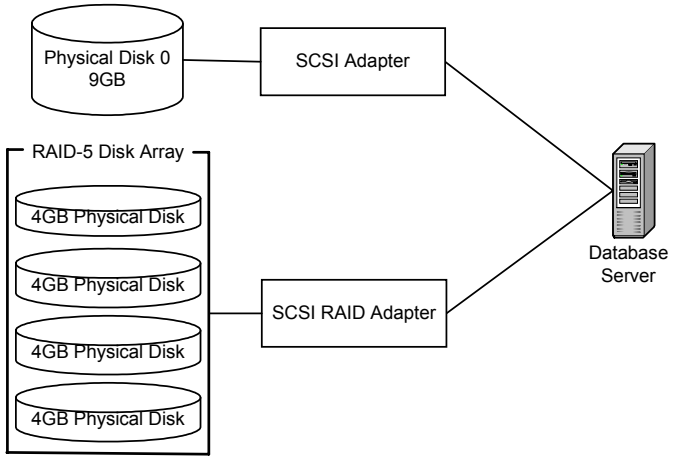
55. The SQL Server 2000 computer that you administer contains a database named *Infrastructure*. In the *Infrastructure* database, the Parts table has a primary key that is used to identify each part stored in the company's warehouse. Each part has a unique UPC code that your company's accounting department uses to identify it. The database is configured as shown:



You want to maintain the referential integrity between the Parts and OrderDetails tables and you want to minimize the amount of physical I/O that is used within the database. The two Transact-SQL statements you should execute are:

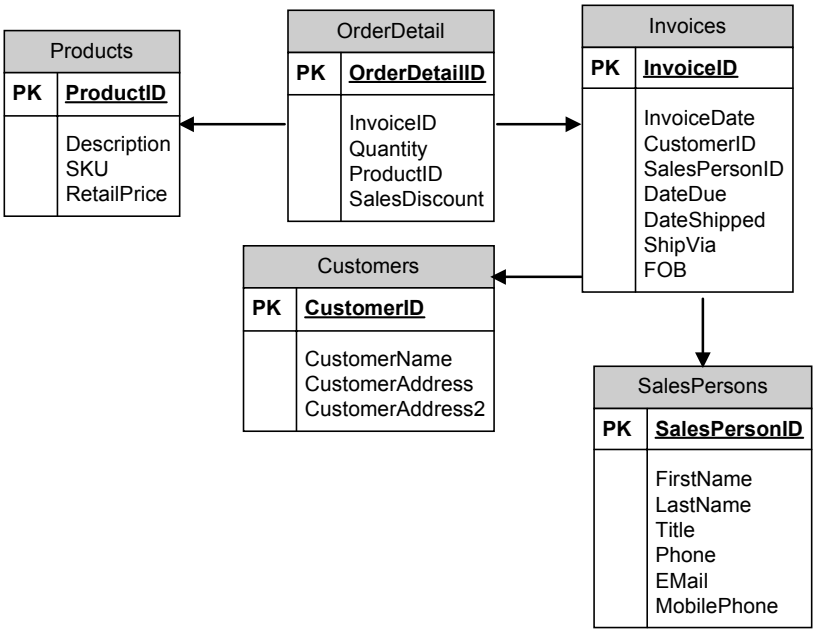
- **`CREATE UNIQUE INDEX IX_UPC On Parts(UPC)`**
- **`ALTER TABLE dbo.OrderDetails ADD CONSTRAINT FK_OrderDetails_Parts FOREIGN KEY(UPC) REFERENCES dbo.Parts(UPC)`**

56. The SQL Server 2000 computer you administer contains a database named Sales that will store sales transactions. Branch offices must be able to add transactions to the database 24 hours a day every day. The server is configured as shown:



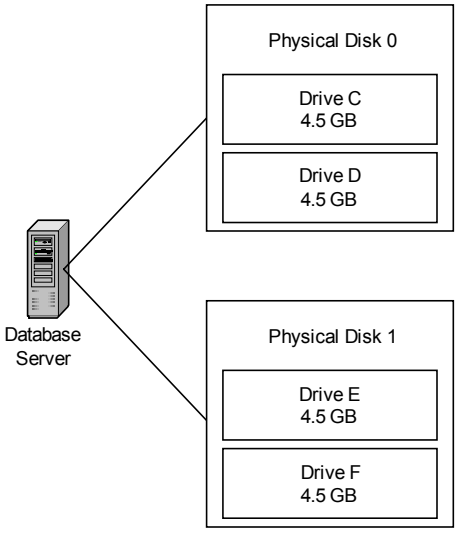
To configure the data files and transaction log for the Sales database you should **place the transaction log and the data file on the RAID-5 disk array.**

57. The SQL Server 2000 computer you administer contains a database named Sales and is configured as shown:



The database has grown by 4 GB in the last six months and now users are reporting that query response time has slowed. The database is set up to have the Full Recovery Model. The "Auto update statistics", "Auto create statistics", and "Torn page detection" check boxes are selected. To accelerate query response time you should **add indexes to the foreign key fields**

58. You are configuring a database for an inventory application. There are two hard disks on your server which are configured as shown:



Operating system files are stored on drive C. Your database requires a maximum of 2 GB of disk space for the transaction log and will store a maximum of 6 GB of data. To optimize database performance you should **add a 2 GB transaction log to drive D, a 3 GB data file to drive E, and a 3 GB data file to drive F**.

Extracting and Transforming Data

Importing and Exporting Data

SQL Server 2000 has several components that support the import and export of data

Data Transformation Services (DTS)

DTS is used to import and export data between like OLE DB and ODBC sources. A DTS package is defined that specifies the source and target data sources which can then be executed on an ad hoc basis or scheduled to run at predefined times. DTS packages can also specify a query as the source of the data which allows packages to transform data by performing an action like running a query that returns aggregate summary values instead of the raw data.

Replication

Replication is used to store synchronized copies of data in separate databases to maintain data recovery integrity. Replication minimizes the overhead required to ensure that all servers have exact copies of current data by removing the need for distributed transactions. In replication there are two parties, the publisher and the subscriber. The publisher sends replication data and the subscriber receives the replication data. There are three types of replication:

- **Snapshot replication:** all data from a publishing site is copied to all subscribing sites and individual changes are not replicated.
- **Transactional replication:** One server is a publisher and other servers subscribe to articles that are published. An article is an individual set of data to be replicated. All changes to the data are made at the publisher's server and are then replicated to the subscribers.
- **Merge replication:** the publisher specifies the articles to be published and all participating parties can make changes. System tables keep track of changes and propagate them to all participating servers.

There are two types of subscriptions, push and pull. Push subscriptions require the publisher to be responsible for the transfer of updates to subscribers. Pull subscriptions allow subscribers to initiate replication.

Bulk Copy

The bulk copy feature is a method for the efficient transfer of large amounts of data. Bulk copy operations transfer data into or out of one table at a time. Bulk copying supports the following transfers:

- From one table or view to another table or view
- From a table or view to a data file
- Query results into a table, view, or data file
- From a data file into a table or view

Distributed Queries

Distributed queries allow Transact-SQL statements to reference data from a data source. These data sources can be another server's database, or a heterogeneous data source such as Microsoft Access or Oracle. SELECT INTO and INSERT statements can be used to:

- Export data from a SQL database to an OLE DB data source
- Import data from an OLE DB data source

Extracting and Transforming Data Keypoints

1. A vendor supplies your company with an application that stores its data in a database on the SQL Server 2000 computer that you administer. The tables in this database contain columns that function as primary keys, but Primary Key and Foreign Key constraints are not used. You need to replicate data from this database to another SQL Server computer that will use the data to generate reports. The bulk of the reports will be run each month. The accounting department needs to have the ability to run reports at any time, and reports should be accurate through the last full working day. You need to implement replication but cannot make any changes to the database. The actions you should take are:
   - **Implement snapshot replication.**
   - **Schedule replication to run during off-peak hours.**

2. You administer a SQL Server 2000 computer at a bank's headquarters. Customers are able to access their accounts from headquarters or from branch offices. The server at headquarters maintains a master copy of all account data for all customers. Any deposits or withdrawals of funds at branch offices must be reflected in the customer's balance on the computers at that office and at headquarters. You want to minimize latency between the headquarters and branch offices. To copy account data between the server at headquarters and the servers at the branch offices you should **implement transactional replication**, **allow immediate-updating subscribers**, and **schedule replication to run continuously**.

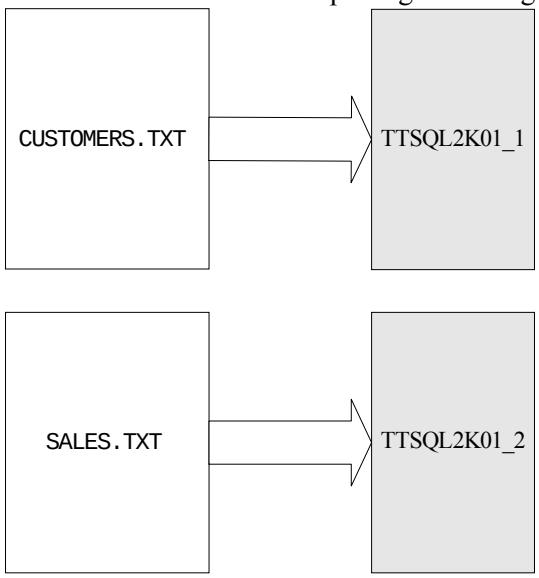
3. You want to set up snapshot replication on the SQL Server 2000 computer you administer. Your server will act as Publisher and Distributor for a minimum of 50 Subscribers. You want to publish 4GB of data, but the data is expected to grow in the near future. Subscribers will receive a new snapshot each month but due to the volume of data to be transferred, you want to minimize the workload on the Publisher/Distributor. The actions you should take to configure snapshot replication are:
   - **Store the snapshot in a shared folder on a file server.**
   - **Create pull subscriptions.**

4. The SQL Server 2000 computer you administer contains a Data Transformation Services (DTS) package that queries multiple databases and writes results to a text file. You run this package by using a Microsoft Windows batch file that uses the DTSRUN utility to execute the DTS package. The actions you should take to ensure that connection properties like login names and passwords cannot be read or modified by users are:
   - **Save the DTS package so that it has an owner password.**
   - **Encrypt the DTS package details in the command line of the DTSRUN utility.**

5. You administer a database in the Chicago branch office of a specialty foods supplier. A mainframe database at the main office contains all company data, and each branch office contains a SQL Server 2000 computer that imports regional data from the mainframe database. The server in Chicago contains a Data Transformation Services (DTS) package that uses an OLE DB to connect to the company's mainframe database. The DTS package extracts and transforms data for that region and then writes the data to the SQL Server database in Chicago. You need to prepare a new SQL Server computer bound for the branch office in Dublin. You want to copy the Chicago package and modify it so that it writes data to the SQL Server database in Dublin. You use the DTS Designer to modify the package to import the regional data from the main office database to the server in Dublin. The modified DTS package runs successfully on the server in Chicago. You transfer this DTS package to the server in Dublin, but the server in Dublin is unable to connect to the main office database. To enable the server in Dublin to connect to the mainframe database you should **install an OLE DB provider for the mainframe database on the server in Dublin**.

6. Each evening after business hours, you perform database maintenance tasks on the SQL Server 2000 computer you administer. You create a Data Transformation Services package to populate the data warehouse. MAPI is not installed on the server. The DTS package should be scheduled to run each night but you want to attend to other duties while the DTS package is executing, and you also want to be notified if the DTS package fails. You should **select the "DTS packages write completion status to event log" check box** and **create a SQL Server event alert to notify you if the package fails**.

7. You administer a SQL Server 2000 computer named TTSQL2K01. You create a data transformation services package that defines two transform data tasks. The transform data tasks get data from two text files named CUSTOMERS.TXT and SALES.TXT. The DTS package is configured as shown:



Unless both transform tasks complete successfully, you do not want the DTS package to add any data to the server. You select the "Join transactions of present" check box of both transform data tasks and the "Use transactions" check box of the DTS package. You then select the "fail package on step failure" check box of both transform data tasks. To ensure that changes are committed if all imported data is successfully transformed you should **select the "commit on successful completion" check box of the DTS package**.

8. You are creating a data transformation services package on the SQL Server 2000 computer you administer. To begin the process you need to load data from text files that contain information on new stores that join your franchise. These text files list the data columns in the following format: StoreID, StoreName, Address, City, State, ZIPCode, ManagerID, StoreTypeID, FacilityID. To load the data into the table as quickly as possible you should **create and edit a format file to select the columns you want to import then use a BULK INSERT task, and specify the format file to import the appropriate data into the destination table**.

9. There are several SQL Server 2000 computers that you administer. A Data Transformation Services (DTS) package uses native OLE DB providers to transfer data between the servers. There are .UDL files that store connection details for the servers and are frequently updated as connection details change. You want to make sure connection details are available and DTS packages are distributed as files to developers in your company. The tasks you should perform are:

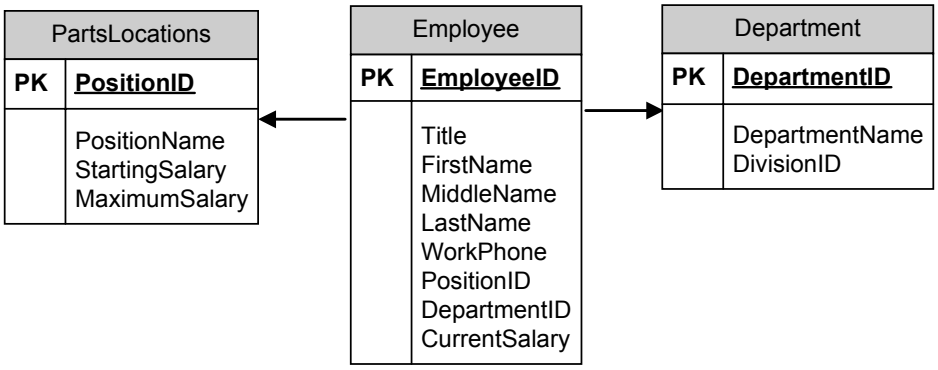
- **Enable the "Always read properties from UDL file" option in the Connection Properties dialog box.**
- **Make the .UDL files available on a network share.**

10. You administer the SQL databases for a financial services company. Employees enter data 24 hours a day into a SQL Server 2000 database. When new account information is gathered from branch offices and added to the database, employees report slow query response time. You currently use the following statements to add account information:

```
BULK INSERT finance.dbo.customers
FROM 'bulk\acct14_10141999.txt'
WITH DATAFILETYPE = 'char',
FIELDTERMINATOR = '\t',
ROWTERMINATOR = '\n,'
TABLOCK
```

To ensure that the response time does not slow when new account information is added to the database you should **remove the TABLOCK option from the BULK INSERT statement**.

11. The SQL Server 2000 computer that you administer stores employee data. A portion of the database schema is shown:



As quickly as possible, you want to create a text file that lists these data columns in the following format: Title, FirstName, LastName, WorkPhone, PositionName, and DepartmentName. You do not expect to recreate this file, and you want to avoid creating new database objects if possible. You should create a **SELECT query that joins the data from the appropriate columns in the three tables** and **add an INTO clause to the query to create a global temporary table**. You should then **use the BCP utility to export the data from the global temporary table to a text file**.

12. The SQL Server 2000 computer you administer contains a database named Customers. Customers is used by a custom client/server application and contains more than 1 million rows. Users report that the database responds slowly when they use the application to search for customers. Users search for customers by scrolling through an alphabetical list of customer names. SQL provides you with the following information on application database use:

Event Class	Text Data
SLQ: Batch completed	Select * from customers

To improve the application's performance you should **rewrite the application so that it uses a query with a WHERE clause to search for customer names**.

13. The SQL Server 2000 computer you administer is located at the warehouse that your company uses to ship all product orders. There are 30 sales offices located throughout the country that offer a range of products specific to their regions. Each sales office contains one SQL Server 2000 computer that is connected to the warehouse through a dial-up connection. The servers dial into the warehouse server as needed, typically once a day. Each sales office needs data pertaining only to its region and inventory data from the server at the warehouse need to be replicated to the sales office servers. The actions you should take to minimize the amount of time needed to replicate the data are:
    - **Create one publication for all subscribers.**
    - **Enable horizontal filtering.**
    - **Use pull subscriptions.**

Managing and Monitoring SQL Server 2000 Security

Authentication

There are two methods of authentication offered SQL Server 2000, Windows authentication and SQL Server authentication. Windows authentication allows the SQL server authentication to be integrated directly with security on a Windows 2000/NT system. Authentication will depend on the security mode that SQL Server 2000 runs under, Windows Authentication mode, or Mixed Mode.

If you are running SQL Server 2000, you will need to configure the SQL server to run in mixed mode to allow the non-Windows 2000/NT clients to connect where it will interact with computers running operating systems other than Windows 2000 or Windows NT. Mixed mode allows clients who are not operating Windows 2000/NT to connect to a SQL 2000 server using SQL Server authentication.

If you are running in a fully Windows 2000/NT environment then you should configure the SQL server to run under the Windows Authentication security mode. In Windows Authentication mode the SQL server relies on Windows 2000/NT for security. While running under the Windows Authentication Mode, a server can utilize security features such as password aging, complexity requirements, and encryption. The SQL server grants access based on the Windows authentication of the user.

Auditing

The basis of auditing in SQL Server 2000 is the server side SQL Trace and the client side viewing utilities. Auditing is configured using the SQL Profiler. With the SQL Profiler you can create a custom auditing scheme that allows you to audit nearly every event that is applied to server data and security.

SQL Server 2000 allows login auditing similar to auditing available on Windows 2000/NT. Administrators can select the level of auditing that they require via the Enterprise Manager. Auditing keeps track of whenever a user is successful or unsuccessful in their attempt to log on.

C2 Auditing

C2 auditing is a special type of auditing required by some government agencies that requires that every event occurring be logged. SQL 2000 Server is capable of C2 auditing, but the overhead for maintaining this level of auditing is very high.

Creating Logins

Logins are used to prevent intentional or unintentional tampering from unauthorized users. Each login has a set of sub-accounts that are linked to it that provide aliases for it. Logins can be created in three ways; in SQL Server 2000, through the Enterprise Manager, or by using T-SQL commands.

Users

A user is defined for a particular database and linked to a login via a sub-account. Users can be created through the Enterprise Manager, the Login Wizard or through the use of T-SQL. Once a user is created, permissions should be assigned to that reflect the job for which it was created. Permissions are assigned to a specific user login. There are two types of permissions that can be assigned to a user. Server permissions are granted to allow certain users to perform database administration tasks. Database permissions are used to control access to database objects.

Permissions and Ownership of Database Objects

Every object in a SQL database has a user who is designated as an owner or dbo. When an object is first created, the only person who has the ability to perform any operations on the object is the dbo. The dbo will grant other users permission to perform certain operations on the object.

For a user to be able to manipulate, perform operations on, or even see a database object the user must have permissions to do so. Permissions on tables and views are:

- INSERT
- UPDATE
- DELETE
- SELECT
- REFERENCES

Alternatively, the owner of a database object may assign ALL permissions to grant a user all of the above. A user must have the appropriate permissions before they can execute the corresponding statement. Owners of a stored procedure can grant EXECUTE permission to a user to enable the user to run the stored procedure.

Database Roles

The task of managing permissions to many database users can be simplified through the use of database roles. Using database roles allows like permissions to be assigned to many users without having them being individually assigned. Roles can be created and managed using the Enterprise Manager or through the use of T-SQL Statements. There are several predefined roles that are established when SQL Server 2000 is installed:

- **sysadmin:** performs any server activity
- **dbcreator:** creates and alter databases
- **diskadmin:** manages disk files
- **setupadmin:** manages linked servers and startup procedures
- **bulkadmin:** performs bulk inserts
- **serveradmin:** sets any server option and shut down the database
- **securityadmin:** manages logins and create database permissions
- **processadmin:** manages SQL processes

Other roles can be defined and altered by the administrator after installation. Fixed roles cannot be modified.

Managing Security through Other Means

Security can be managed through other means through limiting access to objects that could affect sensitive data. Like permissions, access to functions and stored procedures can be explicitly restricted or explicitly granted. In addition there are several system stored procedures that allow an administrator to manage security more easily:

Statement	Purpose
sp_grantlogin	Allows a Windows 2000/NT user to connect with Windows authentication
sp_addapprole	Adds an application security role
sp_changeobjectowner	Changes the owner of an object in a database
sp_changedbowner	Changes the owner of a database
sp_defaultlanguage	Changes the default language of a login
sp_addlogin	Adds a SQL Server login that uses SQL Server Authentication
sp_helplogins	Supplies information about logins and users associated to the database
sp_addrolemember	Adds a security account as a member of an existing database role
sp_grantdbaccess	Adds a security account to a database and allows an administrator to grant permissions

Maintaining security and protecting information from triggers that may negatively alter sensitive information can be achieved by establishing constraints. Properly established constraints can ensure that data integrity and security is maintained across a database.

Managing, Monitoring, and Troubleshooting SQL Server 2000

SQL Server Agent Jobs

SQL Server Agent is a job scheduling agent that aids in the management of a SQL server. SQL Server Agent can be started when the operating system starts and is configured through the SQL Service Manager or Enterprise Manager. The SQL Server Agent is responsible for the execution of jobs. Jobs are made up of multiple steps that have instructions on what to do depending on the success or failure of each step in the job. Jobs are assigned to an operator, a user who is responsible for that job. Clients never directly interact with the SQL Server Agent to create or change scheduled jobs as it is controlled purely from data in SQL server tables. When the SQL Server Agent is started, it connects to its instance of SQL Server 2000 and retrieves scheduled jobs, the steps that encompass those jobs, and the possible alerts that may need to be sent.

Notification with SQL Server Agent

The SQL Server Agent can be configured to send alerts to operators on the success or failure of a job. Alerts can be sent to operators through paging systems, via e-mail, or through the *NET SEND* command to operators. Instead of an alert, failed job results can be written to an application log. To send an e-mail notification, the SQL Server Agent must have a MAPI-1 compliant e-mail client and a valid e-mail profile. An operator for the job must be named and the SQL Server Agent must have access to contact information for that operator. Alerts can be fired for reasons other than success or failure. The SQL Server Agent can be configured to alert an operator when designated events occur. Alerts can be sent when performance conditions, such as running out of disk space.

SQL Profiler

The SQL Profiler is a tool that is useful for analyzing the performance of a SQL Server. The SQL Profiler can capture activity that takes place on the SQL server and play it back later for further analysis. SQL Profiler can also simulate load or stress testing. SQL Profiler can view all statements that are running on a system and display graphical information about them. The SQL profiler can be used to determine which SQL statements are consuming the most CPU time and I/O resources.

System Monitor

The System monitor is the best tool to monitor overall system performance. SQL Server 2000 is able to export performance data to the System Monitor so a picture of the entire system can be seen. This is useful because an administrator can view the SQL server in the context of the entire system instead of by itself. Important counters to monitor are:

- **Disk Transfers/sec:** measures the amount of read/writes that the system does per second
- **Pages/sec:** measures RAM paging on the system
- **% Processor Time:** measures CPU usage. There are two spearate instances of this counter. One measures CPU usage over the entire system and the other measures CPU usage for a particular process.

Managing and Monitoring SQL Server 2000 Keypoints

1. The SQL server you administer is used to store information for your company's sales department. The permissions on the *SalesLeads* table are configured as shown:

Database Role	Select	Insert	Update	Delete
Accountants	Yes	Yes	No	No
Managers	Yes		Yes	Yes
Salespeople	Yes			
Guest				

Dinah is a user in the sales department who needs to review the data in the *SalesLeads* table and must also be able to add new items to the table. To allow Dinah to perform these tasks without giving her additional permissions in the database you should:

- **Add Dinah to the Salespeople database role.**
- **Grant Dinah INSERT permissions on the table**

2. The SQL Server 2000 computer that you administer is used to store information for your company's finance department in the *InterestData* database. The permissions on the *InterestData* database are configured as shown:

User/Role	Create Table	Create View	Create Procedure
Administrators	Yes		
Public			
Managers	Yes	Yes	Yes
Arno		No	
Guest			
Jennifer	Yes		

The actions that you should take to allow Arno to add a new view to the database are:

- **Add Arno to the Managers database role**
- **Revoke CREATE VIEW permissions for Arno**

3. The SQL Server 2000 computer that you administer is used to store sales information for your company. Your company hires a new sales manager named Heinrich. You create a user account named Heinrich in your company's Microsoft Windows NT domain, which is named TESTKILLER. You also create a Windows authenticated login for him on the TESTKILLER domain. The permissions on the sales table are shown:

User/Database Role	Select	Insert	Update	Delete
Accountants	Yes	Yes	Yes	
Administrators	Yes			
Reporters	Yes			
SalesManagers	Yes	Yes	Yes	Yes
Salespeople	Yes	Yes		
Guest				

The Transact-SQL statement that you should execute to give Heinrich permission to view, add, edit, and remove information in the database tables is:

EXEC sp_grantdbaccess 'TESTKILLER\Heinrich', 'Heinrich'
EXEC sp_addrolemember 'SalesManagers', 'Heinrich'

4. The SQL Server 2000 computer that you administer contains a database named *InterestData* that is used by a client/server application. To access the server, the application uses a SQL server login that each employee has. Your company hires a new employee named Alexa. You create a SQL server login named Alexa and also create a database user named Alexa in the database and grant the user full permissions in the database. Alexa receives an "Invalid object name" error message when she runs the client/server application. The Transact-SQL statement you should execute to resolve this problem is:

```
EXEC sp_defaultdb 'Alexa', 'Interestdata'
```

5. The SQL Server 2000 computer you administer is named CorpSQL and is a member of a Windows NT domain named CORPDOMAIN. CorpSQL contains three databases named *Sales*, *Finance*, and *Research*. CorpSQL is configured for Windows authentication. Dan is an administrative assistant in your company who uses several different client/server applications to access the databases on CorpSQL. Dan is then transferred to a different position where he requires access only to the *Research* database. You need to configure the appropriate permissions for Dan. The batch of transact-SQL statements should you execute is:

```
USE Finance
GO
EXEC sp_revokedbaccess 'Dan'
USE Sales
GO
EXEC sp_revokedbaccess 'Dan'
```

6. The SQL Server 2000 computer that you administer contains seven databases that provide data to several client/server applications. A different user maintains each database. To configure server and database permissions so that each user who maintains a database has full permissions on that database you should **create a login for each user's domain user account**, **map each login to a database user in the appropriate databases**, and **add the database users to the db_owner database role in each database**.

7. The SQL Server 2000 computer that you administer is used to store information for your company's finance department. The permissions on the FinanceData table are configured as shown:

Database Role	Select	Insert	Update	Delete
Accountants	Yes	Yes	Yes	No
Administrators	Yes	Yes	Yes	Yes
Reporters	Yes			Yes
Guest				

Jackson is a member of the Accountants and Administrators group. He reports that he is unable to delete information from the FinanceData table. You need to allow him to delete information from the table. The Transact-SQL statement should you execute is:

```
EXEC sp_droprolemember 'Accountants', 'Jackson'
```

8. Shiera is an employee in your company's marketing department who has a SQL Server login named Shiera, and has access to a database named MarketingData. Shiera has been assigned to a research project and needs to view and edit information stored in a database named ResearchData that is on the same server as the MarketingData database. You need to configure the appropriate permissions for Shiera on the ResearchData database. The Transact-SQL statement should you execute is:

```
EXEC sp_grantdbaccess 'Shiera,' ShieraH'
GO
EXEC sp_addrolemember 'db_datareader,' ShieraH'
EXEC sp_addrolemember 'db_datawriter,' ShieraH'
```

9. The SQL Server 2000 computer you administer contains a database named *EmployeeData* that your company's human resources department uses. *EmployeeData* tables are used as follows:

Table Name	Used For
Employees	Employee names, addresses, departments, and base salaries
Bonuses	Bonuses that have been paid to employees
Awards	Awards presented to employees

The human resources department wants to make employee names, addresses, and awards available to anyone who has permission to access the server, but access to salary and bonus information should be limited to human resources department employees only. To allow company users to access only the appropriate employee information you should **create a view that contains the name, address, and award information** and **grant the Guest database user SELECT permissions on the view**.

10. One of the databases on the SQL Server 2000 computer that you administer contains a table named Performance. Performance stores information about sales orders that is used to create sales reports and to calculate commissions for sales personnel. The permissions on the Performance table are configured as shown:

Database Role	Select	Insert	Update	Delete
Reporters	Yes			
SalesManagers	Yes	Yes	Yes	Yes
Salespeople		Yes		
Guest				

Sales managers monitor the entries made by sales personnel and correct any errors. The sales department will hire several assistants to assist in the monitoring process. Sales managers make any necessary corrections on a hard copy, and then the assistants will enter the changes in the table. The assistants must not be allowed to delete any data from the table. To configure permissions on the table to support these requirements and reduce the amount of time needed to manage these permissions you should **create a database user account for each assistant**, **create a database role named SalesAssistants**, **grant SELECT, UPDATE, and INSERT permissions to the SalesAssistants database role**, and **add each database user account to the SalesAssistants database role**.

11. The SQL Server 2000 computer you administer is configured for Windows Authentication and is a member of a Windows NT domain named CORP. Users in the DataUsers user group can connect to the server by using SQL Query Analyzer. There is a login for CORP\DataUsers on the server. Your company purchases a client/server data access application that implements its own user security and will access the server by using a login named DAapp1. You create a domain user account named DAapp1 and add it to the DataUsers domain user group, but when you try to run the data access application, you receive an error message that states "The required SQL Server database cannot be accessed." To ensure that the application runs properly you should **configure the server for Mixed Mode authentication** and **create a login named DAapp1, and configure it to access the application's database**.

12. The database you administer contains 64 lookup tables that store static data that should not change. Users are reporting that some of this data is being changed. You need to prevent users from modifying the data and you want to minimize changes to your security model and to your database applications. To modify the database you should **create a filegroup named LOOKUP**, **move the lookup tables to this filegroup**, and **select the "Read only" check box for the filegroup**.

13. The SQL Server 2000 computer that you administer is named InfoServ and is a member of a Microsoft Windows NT domain named EXPLORATION. Users who need access to InfoServ all belong to a domain user group named Domain Researchers and a local user group named Local Repair on the InfoServ computer. There are 25 new employees who will need access to InfoServ will be added to the Domain Researchers group by domain administrators and who will also need access to InfoServ. The Transact SQL statement that you should execute is

    ```
    EXEC sp_grantlogin EXPLORATION\Domain Researchers
    ```

    This will allow all of the appropriate users to access InfoServ and minimize the amount of time you have spend maintaining security on InfoServ.

14. One of the two of your Microsoft Windows 2000 computers is running Internet Information Services (IIS), and the other is running SQL Server 2000. Employees in your company need to connect over the Internet and query the SQL Server computer. Some employees have computers that do not use Microsoft operating systems or Web browsers. You need to configure the IIS and SQL Server 2000 computers to allow access to data by means of the IIS virtual directory. IIS uses Basic Authentication to allow access to the virtual directory. Only company employees should have access to the databases through the Internet. Employees should be allowed to make ad hoc queries, and you want to be able to audit successful and failed logins to the SQL Server computer. The actions you should take to allow employees access to the SQL Server computer while keeping your security scheme as simple as possible are:
    - **Configure the IIS virtual directory to allow URL queries.**
    - **Create a Windows Authenticated login for each employee on the SQL Server computer.**
      Configure the IIS virtual directory to use Windows Integrated Authentication when connecting to the SQL Server computer.

15. The SQL Server 2000 computer you administer will be used to contain data from several of your company's departments. Each department is contained in a Microsoft Windows domain users group. The groups are named Accounting, Engineering, and Production. Members of each department need to read and modify their data in their department alone. The actions you should take to configure the database so that it meets your company's security requirements and to minimize administration time and the consumption of server resources are:
    - **Create a database for each department.**
    - **Create a Windows authenticated login for each department.**
    - **Map each domain users group to the db_datareader and db_datawriter database roles.**

16. On the SQL Server 2000 computer you administer, you configure a set of alerts that will notify you whenever certain operations fail or encounter errors. Notifications are sent to a Windows 2000 Professional computer via the *NET SEND* command. The alerts are successfully tested, but after several days you stop receiving notifications. You know that there are operations that are still failing and errors are being encountered. You verify that SQLServerAgent service is started and you need to make sure that you continue to receive alert notifications. You should **clear the application log on the SQL server computer**.

17. The SQL Server 2000 computer you administer is used to store confidential company information. Company policy requires that every action and change of permission on the server be logged, and that the server can run only when logging is enabled. To configure the server to comply with this policy you should **set the c2 audit mode option to 1**, and **restart the MSSQLServer service**.

18. The SQL Server 2000 computer you administer is used to store sales and product information for your company. To access the server, employees use a client/server application by activating an application role named SalesAppRole. Several users report that they are unable to use the client/server application. When you test the application, you discover that it cannot log on to the server. You examine your security audit logs and find the following:

Event Class	Text Data	Login Name	SPID
Audit Object Permission…	SELECT * FROM [customers]	libby	52
Audit Object Permission…	SELECT [productname] = [productname] FR…	libby	52
Audit Object Permission…		heinrich	55
Audit Object Permission…	-- sp Mshasdsaccess select name as '…	heinrich	55
Audit Object Permission…	UPDATE [Customers] SET [CompanyName] =…	libby	52
Audit Object Permission…	UPDATE [Customers] SET [CompanyName] =…	libby	52
Audit Object Permission…	-- 'sp_approlepassword' was found in …	heinrich	55
Audit App Role Change Pa…		heinrich	55
Audit Object Permission…	-- 'sp_approlepassword' was found in …	heinrich	55
Audit Object Permission…	-- 'sp_approlepassword' was found in …	heinrich	55
Audit Object Permission…	-- 'sp_approlepassword' was found in …	heinrich	55
Audit Object Permission…	UPDATE [Customers] SET [CompanyName] =…	libby	52
Audit Object Permission…	UPDATE [Customers] SET [CompanyName] =…	libby	52
Audit Object Permission…	-- 'sp_setapprole' was found in t…	heinrich	55
Audit Object Permission…	SELECT * FROM [products]	heinrich	55

To enable the application to log on to the server you should **reset the password on the SalesAppRole application role**.

19. You configure a SQL Server 2000 computer that you administer to perform C2 auditing. Audit logs are saved in the M:\MSSQL\Audit folder. The server functions normally for several days, but then the SQL Server Service shuts down without warning and you cannot restart the service. To restart the SQL Server you should **delete the audit logs from the M:\MSSQL\Audit folder** and **start the MSSQLServer service**.

20. You administer two SQL Server 2000 computers, each of which contain data for one of your company's two regional divisions. Currently users have a single data entry application on client computers that they use to make changes to the database. You install a new data entry application. After the installation users are reporting slow query responses. To monitor server traffic during business hours to help diagnose this problem you should **execute the sp_monitor stored procedure** and **save the output to a text file**.

21. The SQL Server 2000 computer that you administer is a member of a Windows NT domain and is configured for Windows authentication. Many users need to access the database by means of the Internet by using the guest login. Some users are reporting that they cannot connect to the server by means of the Internet. To allow anyone to be able to access the server by means of the Internet you should **configure the server for mixed mode authentication**.

22. Your company has client computers that run Microsoft Windows 2000 Professional and UNIX. All client computers require access to the SQL Server 2000 computer that you administer. Users with domain accounts will access the server through the company LAN, and external users will access the server by means of the Internet. To configure the server so that all authorized users can connect to it you should:
    - **Configure Mixed Mode Authentication.**
    - **Create a login for the Domain Users group.**
    - **Create a login for each UNIX user, and set a password for each login.**

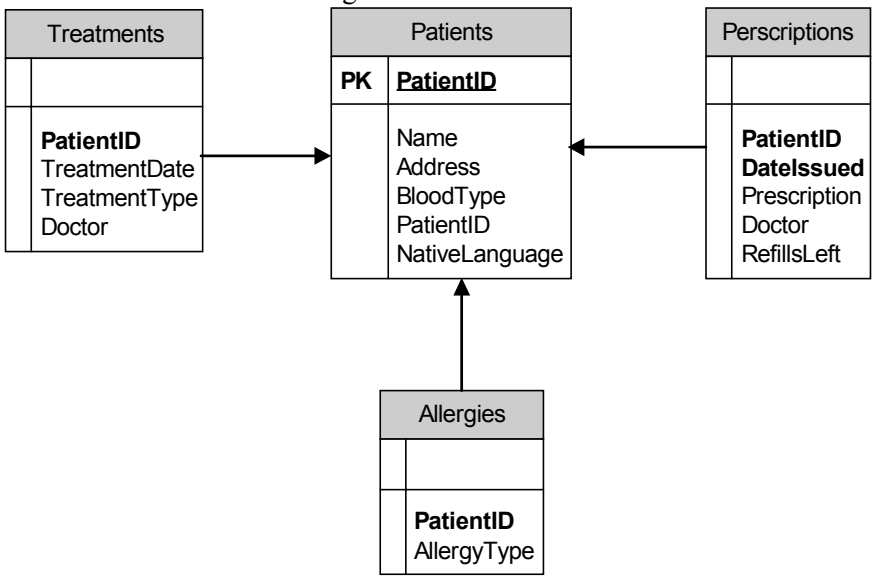
23. The SQL Server 2000 computer you administer is named JSASQL and is a member of a Microsoft Windows NT domain named MUSEUM. Yolanda maintains a local user account on JSASQL and is a member of the local Power Users group. She requests access to the SQL Server databases that are stored on JSASQL and you need to allow her to access JSASQL. The Transact-SQL statement you should execute is:

```
EXEC sp_grantlogin JSASQL\Yolanda
```

24. The client computers operated by your company require access to the SQL Server 2000 computer that you administer. The client computers run Windows 2000 Professional. Company policy dictates that passwords must be at least 10 characters long and change every 30 days. To implement these policies and allow client computers to connect you should:
    - **Configure Windows Authentication.**
    - **Create a login for the Domain Users group.**

25. The SQL Server 2000 computer that you administer is configured for Windows authentication. You implement an integrated logon for the domain users group. There are 20 users in the domain users group who should not be able access to the server. To prevent the unauthorized users from accessing the server while making the least number of changes to the server you should **add the unauthorized users to a domain users group, add a login for the group, and then deny access to the login**.

26. The SQL Server 2000 computer you administer contains a database named *Medical*, which stores patient and treatment information. *Medical* is configured as shown:



The company accountant needs access to the total number of treatments that have been performed in the office, but should not be able to access to any other patient data or be able to change any information in the database. You create a SQL Server Authenticated login and database user account named AccountUser for the accountant. The Transact-SQL statements you should execute to make the appropriate information available to the accountant, while protecting the other information in the database are:

```
CREATE PROCEDURE TotalTreatments AS
SELECT COUNT(*) AS Total FROM dbo.Patients
GO
GRANT ALL ON TotalTreatments TO AccountUser
```

27. The SQL Server 2000 computer you administer is used to store information for your company's financing division. The permissions on the Financing table are configured as shown:

Database Role	Select	Insert	Update	Delete
Accountants	Yes	Yes	No	No
Geoff	No			
Managers	Yes		Yes	Yes
Salespeople	Yes			
Guest				

Geoff is a member of the Managers and Salespeople database roles. While working in the table, he reports that he can edit and remove information. However, he cannot view any of the information in the table. You need to allow Geoff to view the information in the Financing table. The Transact-SQL statement you should execute is:

```
REVOKE SELECT ON Financing FROM Geoff.
```

28. The SQL Server 2000 computer that you administer belongs to a Microsoft Windows NT domain named DOMAIN1 and contains a database named SalesData. Libby is a sales representative who uses the database. In order for her to perform her job, you create a login named DOMAIN1\Libby and manually configure the permissions necessary. Libby is temporarily reassigned to the marketing department, and during her assignment she must not be able to access the server. You must restore her access when she returns to the sales department. You need to configure the appropriate database permissions for Libby. The Transact-SQL statement you should execute is:

```
EXEC sp_denylogin 'Libby'
```

29. The SQL Server 2000 computer that you administer contains multiple databases. One of these databases is named *RadioData*. Your company hires a new employee named Courtney who should be able to create new tables in the *RadioData* database. Courtney should not have access to any other databases on the server. You create a SQL Server login named Courtney. Courtney reports that she can log on to the server but is unable to access the *RadioData* database. To allow Courtney access to the *RadioData* database you should **create a new database user in the *RadioData* database**, **map the database user to Courtney's login**, and **grant CREATE TABLE permissions to the database user**.

30. On the SQL Server 2000 computer you administer you create a new database named EmployeeData. The database will store confidential employee information for your company. Only authenticated users should be able to access the EmployeeData database. The Transact-SQL statement you should execute is:
EXEC sp_revokedbaccess 'guest'

31. The SQL Server 2000 computer you administer has a database that contains a table named *Orders* that is used to store information about your company's customers. Some users should be allowed to modify only the table's Address column, and other users should be allowed to modify only the table's CreditCard column. Neither group of users has permission on the *Orders* table. To make sure that only authorized users make changes to the Address and CreditCard columns you should **create a stored procedure and require users to enter information by means of the stored procedure**.

32. There are 20 SQL Server 2000 computers that you administer located in ten company offices around the world. You work in the Houston office and need to configure several dozen alert responses on each server. You plan on making periodic changes to these alert responses over the course of time. What you should do to reduce the amount of time you spend managing alert responses and be able to manage all responses from the Houston office is to **configure alert forwarding on the remote servers**.

33. The SQL Server 2000 computer that you administer is used to store information for your company's finance department. The permissions on the database are configured as shown:

Database Role	Create Table	Create View	Create SP
Accountants	No		No
Administrators	Yes		
Public			
AppRole1			
Managers	Yes	Yes	Yes
Salespeople			
Heinrich		No	
Guest			
Libby	Yes		

Heinrich is a member of the Accountants database role and needs to add a new view to the database. The actions you should take to allow Heinrich to add a new view you should:

- **Add Heinrich to the Managers database role.**
- **Revoke the CREATE VIEW permissions for Heinrich.**

34. The SQL Server 2000 computer you administer is named JSASql1 and is a member of a Microsoft Windows NT domain named CORP. Your company purchases a client/server application that will use a database named *Marketing* stored on JSASql1. Users named Libby, Heinrich, and Rex belong to a domain user group named Marketers and will use the application to access the *Marketing* database. In the future, several additional users who will also use the new application will be added to the Marketers group. The logins on JSASql1 are configured as shown:

Name	Type	Server Access
BUILTIN\Administrators	Windows	Permit
Sa	Standard	Permit
CORP\Heinrich	Windows	Permit
CORP\Rex	Windows	Permit
CORP\Libby	Windows	Permit
Marketers	Standard	Permit

You need to allow Libby, Rex, and Heinrich access to the *Marketing* database as quickly as possible. The batch of Transact-SQL statements you should execute is:

```
EXEC sp_grantlogin 'CORP\Marketers'
GO
USE Marketing
EXEC sp_grantdbaccess 'CORP\Marketers'
```

Index