

第 3 章

数据库的创建与操作



本章内容提要

数据库是指长期存储在计算机内，有组织的、有结构的、可共享的数据集合，形象地讲，数据库是存储数据的仓库。对于数据的操作，也只有创建了数据库之后才能进行。本章介绍数据库的创建与操作，主要内容包括创建数据库、修改数据库、删除数据库、查看数据库信息等。



微视频

3.1 创建数据库

在 SQL Server 2017 中，除了系统数据库之外，如果要使用其他数据库，第 1 步就是要创建数据库。在创建数据库前，首先需要确定的是数据库存放哪些数据，这样就可以根据数据库的用途来给数据库命名了。

3.1.1 了解数据库的结构

在创建数据库之前，还需要了解一下数据库的结构。在 SQL Server 数据库中，一个数据库通常由数据文件和事务日志组成，一个数据库可以由一个或多个数据文件和事务日志组成。

数据文件是存储数据的地方，事务日志是用来记录存储数据的时间和操作的，可以根据事务日志来恢复数据库中的数据。因此，不能随便将事务日志文件删除。数据文件的扩展名为.mdf，而事务日志文件的扩展名是.ldf。这样我们在看到扩展名后，就可以知道是数据库中哪种类型的文件了。

虽然数据库可以包含多个数据文件与日志文件，但是数据文件是有主次之分的，主要分为主要数据文件与次要数据文件，因此，一个数据库可以使用三类文件来存储信息。

- 主要数据文件：主要数据文件主要存储数据库的启动信息、用户数据和对象，如果有次要数据文件，其引用信息也包含在内。一个数据库只能有一个主要数据文件，默认文件扩展名为.mdf。
- 次要数据文件：如果主要数据文件超过了单个 Windows 文件的最大限制，可以使用次要数据文件存储用户数据。次要数据文件可以将数据分散到不同磁盘中，默认文件扩展名为.ndf。
- 事务日志文件：事务日志文件主要用来恢复数据库日志信息，每个数据库至少应该包括一个事务日志文件，默认文件扩展名为.ldf。

3.1.2 创建数据库的语法

在 SQL Server 2017 中，使用 CREATE 语句可以创建数据库，该语句的语法格式如下：

```
CREATE DATABASE database_name
[ ON
    [ PRIMARY ] [<filespec> [ ,...n ]]
]
[ LOG ON
    [<filespec> [ ,...n ]]
];
```

主要参数介绍如下。

- **database_name**: 数据库名称, 不能与 SQL Server 中现有的数据库实例名称相冲突, 最多可以包含 128 个字符。数据库名称不能以数据开头, 一般是以英文单词或缩写或汉语拼音来命名。
- **ON**: 显示定义, 用来存储数据库数据部分的数据文件。如果省略了 ON 语句, 系统也会默认创建一个数据库。数据库的数据文件和日志文件都与数据库的名称一样, 只是扩展名不同而已。这样文件会存储到数据库安装的默认路径中。
- **PRIMARY**: 主数据文件。所谓主数据文件就是在创建数据库时指定的第一个数据文件。一个数据库中只能有一个主数据文件, 其他的数据文件被称为次要数据文件。如果在创建数据库时没有指定 PRIMARY, 那么 CREATE DATABASE 语句中列出的第一个文件将成为主数据文件。
- **LOG ON**: 指定用来存储数据库日志的日志文件。如果没有指定 LOG ON, 系统也会为其自动创建一个日志文件。

3.1.3 一行语句创建数据库

一行语句其实就是创建数据库的语法只保留必要的关键字, 其余的参数由系统自动创建, 这一行语句就是 CREATE DATABASE database_name。

实例 1: 创建名称为 MR_db 数据库。

在“查询编辑器”窗口中输入以下语句。

```
CREATE DATABASE MR_db;
```

单击“执行”按钮, 即可创建 MR_db 数据库, 这时, 在“对象资源管理器”窗口中可以看到新创建的数据库, 如图 3-1 所示。

提示: 如果刷新 SQL Server 2017 中的数据库节点后, 仍然看不到新建的数据库, 可以重新连接对象资源管理器, 即可看到新建的数据库。

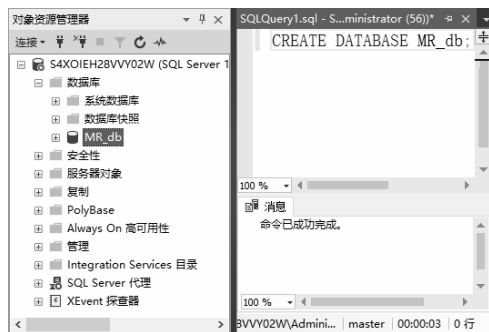


图 3-1 创建 MR_db 数据库

3.1.4 创建带有自定义参数的数据库

有时, 为方便数据库管理员的管理, 也便于查找数据库的位置, 通常会在创建数据库时为数据库指定一个保存位置, 并设置其他的相关参数。为数据库指定位置不仅可以指定数据文件的位置也可以指定日志文件的位置, 一般情况下, 会将数据文件和日志文件指定保存在同一个文件夹下。

实例 2: 创建带有自定义参数的 MR_db 数据库。

创建 MR_db 数据库, 该数据库的主数据文件逻辑名称为 sample-db, 物理文件名称为 sample.mdf, 初始文件大小为 10MB, 最大尺寸为 30MB, 增长速度为 5%; 数据库日志文件的逻辑名称为 sample_log, 保存日志的物理文件名称为 sample.ldf, 初始文件大小为 5MB, 最大尺寸为 15MB, 增长速度为 10%。

在“查询编辑器”窗口中输入以下语句。

```
CREATE DATABASE [MR_db] ON PRIMARY
(
    NAME = 'sample_db',                --数据文件的逻辑名称
    FILENAME = 'D:\database\sample.mdf', --数据文件的存放位置
    SIZE=10MB,                        --数据文件的大小
    MAXSIZE =30MB,                    --数据文件的最大值
    FILEGROWTH=5%                      --数据文件的生长量
)
LOG ON
(
    NAME = 'sample_log',                --日志文件的逻辑名称
    FILENAME = 'D:\database\sample_log.ldf', --日志文件的存放位置
    SIZE=5MB,                          --日志文件的大小
    MAXSIZE=15MB,                      --日志文件的最大值
)
```

```
FILEGROWTH =10% --日志文件的增长量
)
GO
```

单击“执行”命令  执行(X)，命令执行成功之后，在“消息”窗格中显示命令已成功完成的信息提示。刷新 SQL Server 2017 中的数据库节点，可以在子节点中看到新创建的名称为 MR_db 的数据库，如图 3-2 所示。



图 3-2 创建带有参数的 MR_db 数据库

知识扩展：选择新建的 MR_db 数据库，右击，在弹出的快捷菜单中选择“属性”命令，打开“数据库属性”窗口，选择“文件”选项，即可查看数据库的相关信息。可以看到，这里各个参数值与 SQL 代码中指定的值完全相同，如图 3-3 所示。

注意：在创建数据库之前，要先确保 D 盘下的 database 文件夹已经存在，否则在执行代码时，会在“消息”提示框中显示如图 3-4 所示的错误提示。

3.1.5 创建包含多个文件的数据库

一个数据库只能包含一个数据文件与一个日志文件吗？当然不是了。在创建数据库时，我们可以根据需要定义多个数据文件或日志文件。那么，一个数据库中会包含无数个数据文件吗？当然也不是了，它最大包含 32 767 个文件。

实例 3：创建包含有多个文件的数据库。

创建数据库 FU_db，该数据库包含两个数据文件和两个日志文件，在“查询编辑器”窗口中输入以下语句。

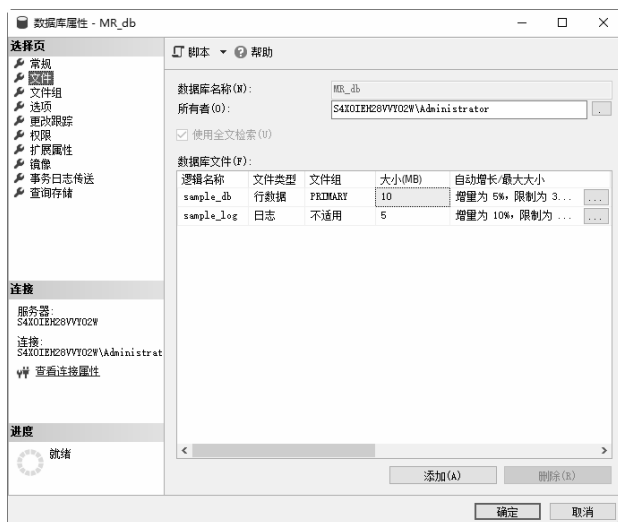


图 3-3 “数据库属性”窗口

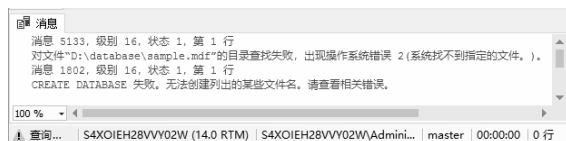


图 3-4 错误提示信息

```

CREATE DATABASE [FU_db] ON PRIMARY
(
    NAME = 'FU_db_1',                                --数据文件的逻辑名称
    FILENAME = 'D:\database\FU_db_1.mdf',             --数据文件的存放位置
    SIZE=10MB,                                         --数据文件的大小
    MAXSIZE =30MB,                                    --数据文件的最大值
    FILEGROWTH = 5%                                   --数据文件的增量
),
(
    NAME = 'FU_db_2',                                --数据文件的逻辑名称
    FILENAME = 'D:\database\FU_db_2.ndf',             --数据文件的存放位置
    SIZE=20MB,                                         --数据文件的大小
    MAXSIZE =50MB,                                    --数据文件的最大值
    FILEGROWTH =10%                                   --数据文件的增量
)
LOG ON
(
    NAME = 'FU_db_log_1',                            --日志文件的逻辑名称
    FILENAME = 'D:\database\FU_db_log_1.ldf',         --日志文件的存放位置
    SIZE=5MB,                                          --日志文件的大小
    MAXSIZE=15MB,                                     --日志文件的最大值
    FILEGROWTH =10%                                   --日志文件的增量
),
(
    NAME = 'FU_db_log_2',                            --日志文件的逻辑名称
    FILENAME = 'D:\database\FU_db_log_2.ldf',         --日志文件的存放位置
    SIZE=10MB,                                        --日志文件的大小
    MAXSIZE=30MB,                                     --日志文件的最大值
    FILEGROWTH=5%                                    --日志文件的增量
)
GO

```

单击“执行”按钮，命令执行成功之后，在“消息”窗格中显示命令已成功完成的信息提示。刷新 SQL Server 2017 中的数据库节点，可以在子节点中看到新创建的名称为 FU_db 的数据库，如图 3-5 所示。

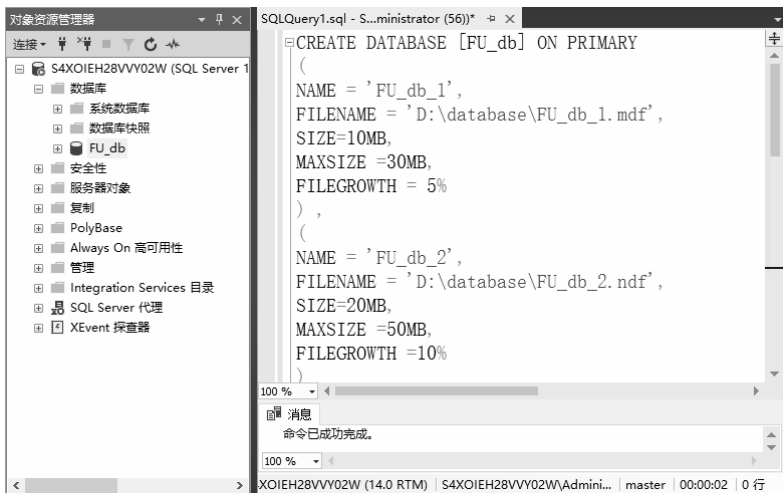


图 3-5 创建 FU_db 数据库

选择 FU_db 数据库，右击，在弹出的快捷菜单中选择“属性”命令，打开“数据库属性”窗口，选择“文件”选项，可以在“数据库文件”窗口中查看该数据库包含的文件信息，包括逻辑名称、文件类型、

文件组、大小等，如图 3-6 所示。

另外，打开 D 盘目录下的 database 文件夹，可以看到里面有 4 个与 FU_db 数据库相关的数据文件，如图 3-7 所示。

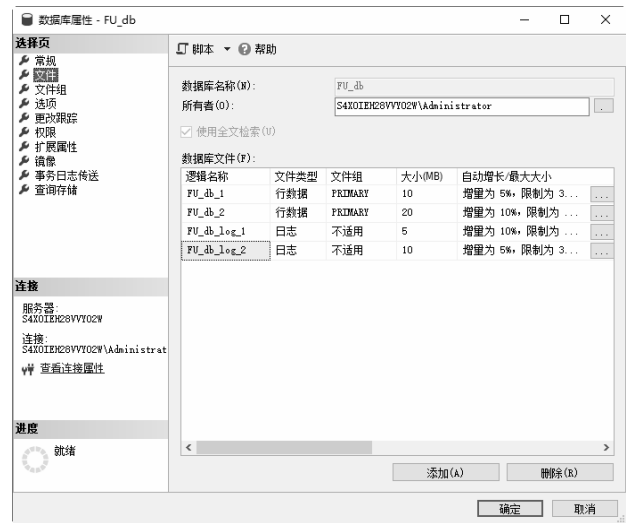


图 3-6 “数据库属性”窗口

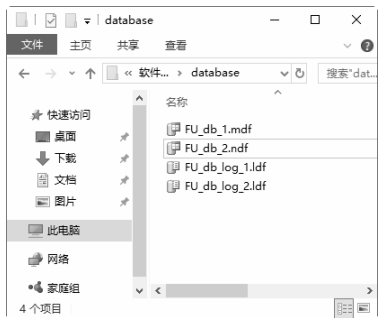


图 3-7 database 文件夹

注意：在数据库中，数据文件与日志文件是不能重名的。实际上，一个数据库只有一个主数据文件，扩展名为.mdf，其他数据文件为次要数据文件，其扩展名为.ndf。

3.1.6 创建包含文件组的数据库

创建包含文件组数据库的语法格式与创建包含多个文件数据库的语法类似。数据库中的文件组从字面上理解，就是在文件组中存放多个文件。在每一个数据库中都可以存在多个文件组，而且一定会有一个主文件组，其他的文件组为用户自定义文件组。

实例 4：创建包含文件组的 EM_db 数据库。

创建数据库 EM_db，该数据库包含一个自定义文件组，在“查询编辑器”窗口中输入以下语句。

```
CREATE DATABASE [EM_db] ON PRIMARY
(
    NAME = 'EM_db_1',
    FILENAME = 'D:\database\EM_db_1.mdf',
    SIZE=10MB,
    MAXSIZE =30MB,
    FILEGROWTH =10%
),
FILEGROUP EM_db_group
(
    NAME = 'EM_db_2',
    FILENAME = 'D:\database\EM_db_2.ndf',
    SIZE=20MB,
    MAXSIZE =50MB,
    FILEGROWTH =10%
)
LOG ON
(
    NAME = 'EM_db_log_1',
    FILENAME = 'D:\database\EM_db_log_1.ldf',
    SIZE=5MB,
```

--数据文件的逻辑名称
--数据文件的存放位置
--数据文件的大小
--数据文件的最大值
--数据文件的增长量

--文件组的名称

--数据文件的逻辑名称
--数据文件的存放位置
--数据文件的大小
--数据文件的最大值
--数据文件的增长量

--日志文件的逻辑名称
--日志文件的存放位置
--日志文件的大小

```

MAXSIZE=15MB,          -- 日志文件的最大值
FILEGROWTH=5%           -- 日志文件的增长量
)

```

单击“执行”按钮，命令执行成功之后，在“消息”窗格中显示命令已成功完成的信息提示。刷新 SQL Server 2017 中的数据库节点，可以在子节点中看到新创建的名称为 EM_db 的数据库，如图 3-8 所示。

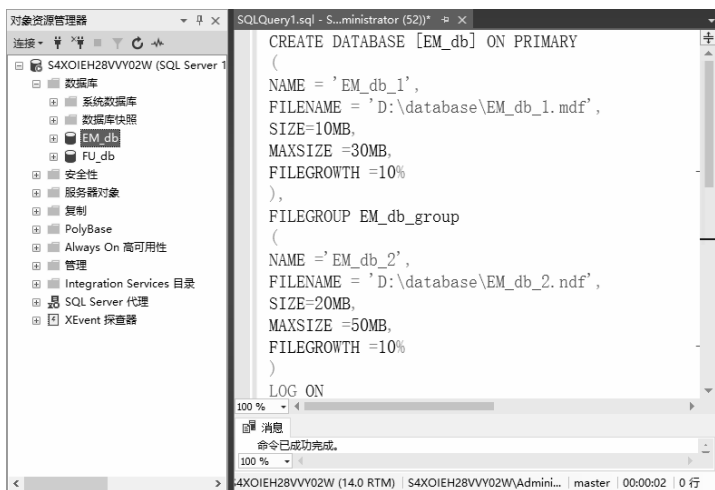


图 3-8 创建 EM_db 数据库

选择 EM_db 数据库，右击，在弹出的快捷菜单中选择“属性”命令，打开“数据库属性”窗口，选择“文件组”选项，可以在右侧的窗口中查看该数据库包含的文件组信息，包括文件组的名称、包含的文件数量等，如图 3-9 所示。

知识扩展：在创建数据库时，数据文件或日志文件的大小一定要大于 512KB，否则就无法成功创建数据库。另外，文件的单位不仅可以为 KB 或 MB，还可以是 GB、TB 等单位。在文件增长量部分也可以使用百分比的形式，可使用 KB 或 MB 作为其单位。如果不能预测数据库的最大容量，可以将 maxsize 的值设置成 Unlimited（无限制）。

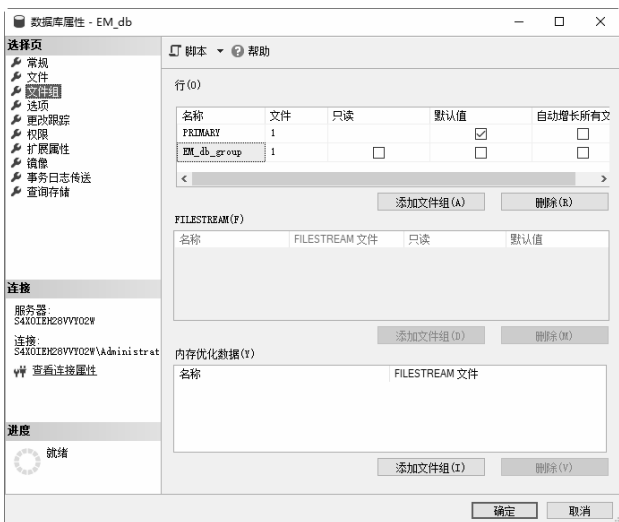


图 3-9 选择“文件组”选项

3.1.7 以图形向导方式创建数据库

在 SQL Server Management Studio 中，我们可以以图形向导方式创建数据库，还可以在创建数据库时更改数据文件的大小、文件的存放位置以及添加文件、使用文件组等操作。

创建 my_db 数据库的过程可以分为如下几步。

(1) 启动 SQL Server Management Studio 并登录到 SQL Server 2017 数据库，在“对象资源管理器”中打开“数据库”节点，选择“数据库”节点，右击，在弹出的快捷菜单中选择“新建数据库”命令，如

图 3-10 所示。

(2) 打开“新建数据库”窗口，默认选择“常规”选项，在“常规”选项卡中设置创建数据库的参数，这里输入数据库的名称，并设置初始大小等参数，如图 3-11 所示。

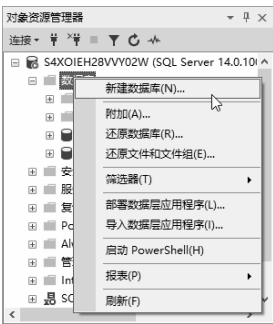


图 3-10 选择“新建数据库”命令



图 3-11 “新建数据库”窗口

注意：数据库名称中不能包含以下 Windows 不允许使用的非法字符：“|” “!” “*” “/” “?” “.” “\” “<” “>” “_”。

(3) 在“选择页”列表中选择“选项”选项，在打开的界面中可以设置有关选项的相关参数，如图 3-12 所示。

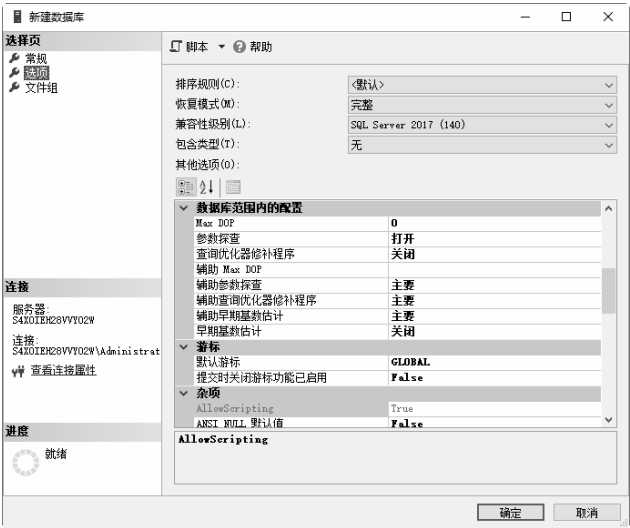


图 3-12 “选项”选项卡

(4) 在“文件组”选项卡中，可以设置或添加数据库文件和文件组的属性，例如是否为只读，是否有默认值，如图 3-13 所示。

(5) 单击“确定”按钮，即可创建 my_db 数据库，在“对象资源管理器”中可以查看新创建的名称为 my_db 的数据库，如图 3-14 所示。

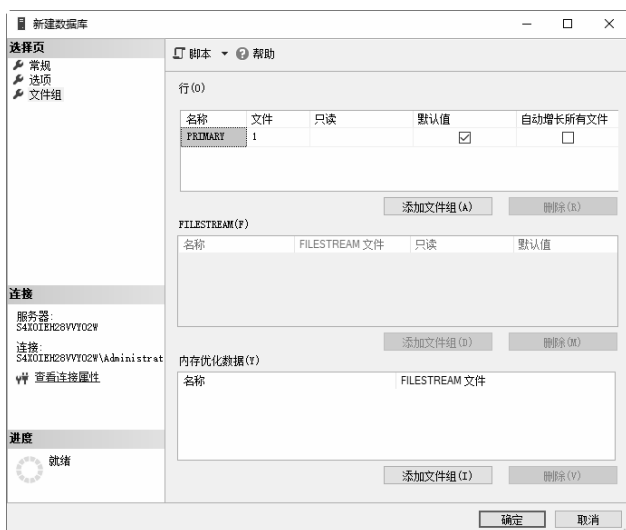


图 3-13 “文件组”选项卡

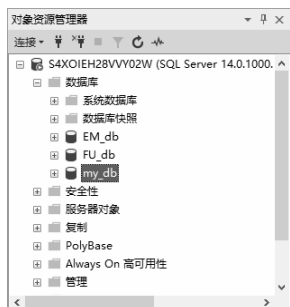


图 3-14 创建的数据库

注意：SQL Server 2017 在创建数据库的过程中，将对数据库进行检验，如果存在一个相同名称的数据库，则创建操作失败，并提示错误信息，如图 3-15 所示。

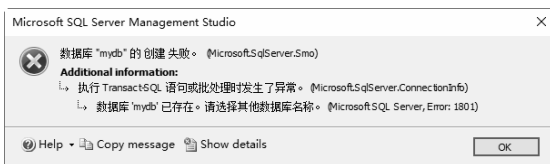


图 3-15 错误信息提示

3.2 修改数据库

数据库创建完毕后，它不是一成不变的，我们可以根据自己的需要对数据库进行修改操作，比如修改数据库的名称、修改数据库的初始大小与最大容量值、添加数据文件或文件组、清理无用的数据文件等。

3.2.1 修改数据库的名称

数据库创建完毕后，如果发现数据库的名称不符合需要，可以通过下面两个方法修改。

1. 使用 ALTER DATABASE 语句修改

具体的语法格式如下。

```
ALTER DATABASE old_database_name
MODIFY NAME=new_database_name
```

主要参数介绍如下。

- old_database_name: 原来数据库的名称。
- new_database_name: 指定新的数据库名称。

实例 5: 将数据库 my_db 的名称修改为 my_dbase，在“查询编辑器”窗口中输入以下语句。

```
ALTER DATABASE my_db          --指定要修改数据库的名称
MODIFY NAME = my_dbase;       --指定新数据库的名称
```

单击“执行”按钮，即可更改数据库的名称，刷新数据库节点，可以看到修改后的新的数据库名称，如图 3-16 所示。



微视频

2. 使用存储过程 sp_renamedb 更改

使用存储过程 sp_renamedb 可以更改数据库的名称，最终结果与 ALTER DATABASE 语句的效果是一样的，不过使用 sp_renamedb 会更简单一些，具体的语法格式如下。

```
sp_renamedb old_database_name,new_database_name
```

主要参数介绍如下。

- old_database_name: 原来数据库的名称。
- new_database_name: 指定新的数据库名称。

实例 6: 将数据库 my_dbase 的名称修改为 newmy_dbase，在“查询编辑器”窗口中输入以下语句。

```
sp_renamedb my_dbase,newmy_dbase; --使用存储过程修改数据库的名称
```

单击“执行”按钮，即可更改数据库的名称，刷新数据库节点，可以看到修改后的新的数据库名称，如图 3-17 所示。

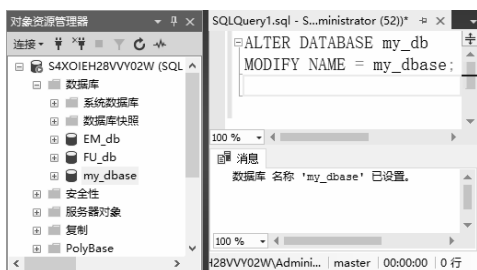


图 3-16 使用 ALTER 语句更改数据库的名称



图 3-17 使用 SP_RENAMEDB 语句更改数据库的名称

3.2.2 修改数据库的初始大小

数据库的初始大小实际上就是数据库中数据文件的初始大小。使用 ALTER DATABASE 语句可以修改数据库的初始大小，具体的语法格式如下。

```
ALTER DATABASE database_name
MODIFY FILE
{
    NAME =datafile_name,
    NEWNAME =new_datafile_name,
    FILENAME ='file_path',
    SIZE=new_size,
    MAXSIZE =new_maxsize,
    FILEGROWTH=new_filegrowth
}
```

主要参数介绍如下。

- database_name: 数据库的名称。
- NAME: 数据文件名，也就是要修改的数据文件的名称。
- NEWNAME: 更改后的数据文件名称，如果不需要修改数据文件的名称，该语句可以省略。
- FILENAME: 设置数据文件保存的位置，如果不需要修改数据文件的保存位置，该语句可以省略。
- SIZE: 数据文件的初始大小，如果不需要修改数据文件的初始大小，该语句可以省略。
- MAXSIZE: 数据文件的最大值，如果不需要修改数据文件的最大值，该语句可以省略。
- FILEGROWTH: 文件自动增长值，如果不需要修改数据文件的自动增长值，该语句可以省略。

实例 7: 将 my_dbase 数据库中的主数据文件 my_db 的初始大小修改为 20MB，在“查询编辑器”窗口中输入以下语句。

```
ALTER DATABASE my_dbase --指定要修改数据库的名称
MODIFY FILE
(
    NAME=my_db, --指定要修改数据文件的名称
```

```
SIZE=20MB --指定要修改数据文件的初始大小
);
```

单击“执行”按钮，my_db 的初始大小将被修改为 20MB，如图 3-18 所示。打开 my_db 数据库的“数据库属性”窗口，在“文件”选项卡中可以看到 my_db 数据文件的初始大小被修改为 20MB，如图 3-19 所示。



图 3-18 输入并执行语句

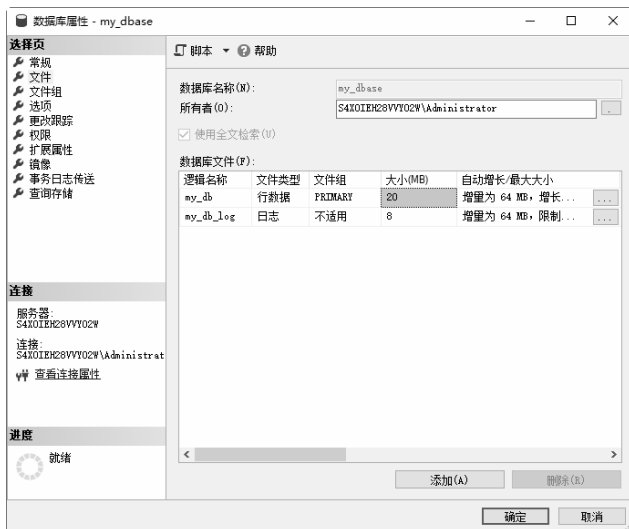


图 3-19 修改数据库的初始大小

注意：修改数据文件的初始大小时，指定的 SIZE 的大小必须大于或等于当前大小，如果小于，代码将不能被执行，并给出如图 3-20 所示的错误提示。

知识扩展：数据库的初始大小修改完毕后，除了可以在“数据库属性”窗口中查询修改后效果，还可以使用存储过程 sp_helpdb 来查看修改后的效果。在“查询编辑器”窗口中输入以下语句。

```
sp_helpdb my_dbase; --查询指定数据库的文件信息
```

单击“执行”按钮，即可在“结果”窗格中显示查询结果，这时可以看到 my_dbase 数据库中的数据文件的初始大小已经更改为 20480KB 了，如图 3-21 所示。在这里，读者可能会有一个疑惑，我们设置的明明是 20MB，为什么显示的是 20480KB？实际上这个结果是正确的，这是因为 1MB=1024KB，那么 20MB 不就是 20480KB 吗！



图 3-20 错误信息提示

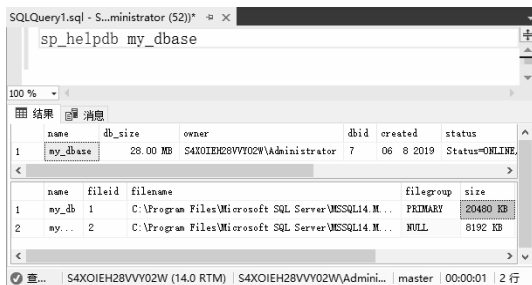


图 3-21 查询结果

3.2.3 修改数据库的最大容量

使用 ALTER DATABASE 语句，不仅可以修改数据库的初始大小，还可以修改数据库的最大容量。

实例 8：将 my_dbase 数据库中的主数据文件 my_db 的最大容量修改为 200MB，在“查询编辑器”窗口中输入以下语句。

```

ALTER DATABASE my_dbase      --指定要修改数据库的名称
MODIFY FILE
(
    NAME=my_db,              --指定要修改数据文件的名称
    MAXSIZE=200MB            --指定要修改数据文件的最大容量
);

```

单击“执行”按钮，即可将 my_dbase 数据库的最大容量修改为 200MB，如图 3-22 所示。打开“数据库属性”窗口，在“文件”选项卡中可以看到 my_db 数据文件的自动增长/最大大小被修改为 200MB，如图 3-23 所示。

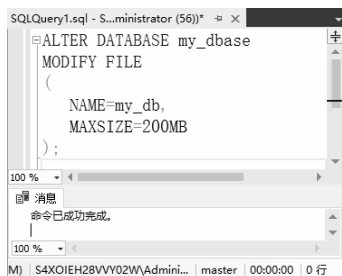


图 3-22 输入并执行语句

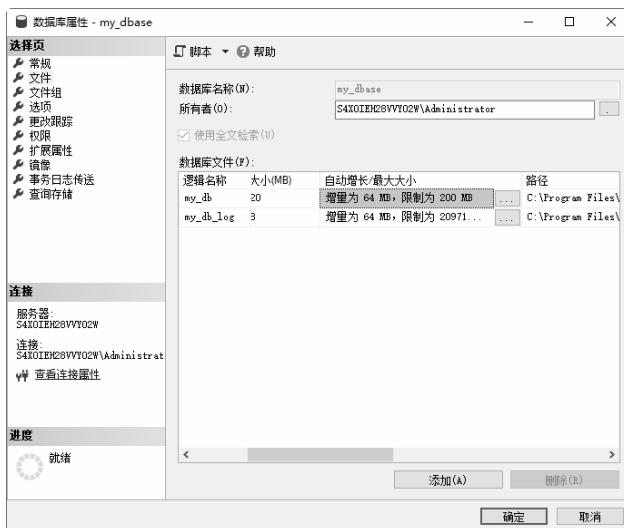


图 3-23 修改数据库的最大容量

提示：缩减数据库的最大容量与增加数据库容量的方法相同，只是在具体设置数值时，一定要大于数据文件的初始大小，例如 my_dbase 数据库的数据文件初始大小为 20MB，如果修改 MAXSIZE 的值为 19MB，则会出现如图 3-24 所示的错误提示。

3.2.4 给数据库添加数据文件

除更改数据库的名称、初始大小等属性外，还可以在数据库中添加数据文件、日志文件或文件组。使用 ALTER DATABASE 语句可以给数据库添加数据文件，具体的语法格式如下。

```

ALTER DATABASE database_name
[ADD FILE|LOG FILE]
(
    NAME =logic_file_name,
    FILENAME ='file_path',
    SIZE=new_size,
    MAXSIZE =new_maxsize,
    FILEGROWTH=new_filegrowth
)
ADD FILEGROUP filegroup_name
[TO FILEGROUP filegroup_name]

```

主要参数介绍如下。

- database_name: 数据库的名称。
- ADD FILE: 添加数据文件，添加数据文件和创建日志文件的文件结构是一样的。

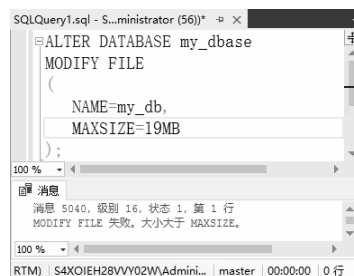


图 3-24 错误信息提示

- ADD LOG FILE: 添加日志文件。
- ADD FILEGROUP: 添加文件组。
- TO FILEGROUP: 为数据文件指定文件组, 如果没有指定文件组, 默认情况下, 数据文件会添加到 PRIMARY 文件组中。

实例 9: 给数据库添加数据文件。

在 my_dbase 数据库中添加一个数据文件 sample_db, 该数据文件的初始大小为 10MB, 最大的文件大小为 100MB, 增长速度为 2MB, 数据库的存放地址为 D 盘下的 database 文件夹。

注意: 在向数据库中添加文件前, 首先需要通过 sp_helpdb 来查看一下现有的文件信息, 以免在向数据库添加新文件时出现重名现象。打开“查询编辑器”窗口, 在其中输入以下语句。

```
sp_helpdb my_dbase; --查询数据库的文件信息
```

单击“执行”按钮, 即可在打开的“结果”窗格中查看当前数据库的数据文件信息, 该数据库中保存了一个数据文件 my_db 和一个日志文件 my_db_log, 如图 3-25 所示。

下面开始给 my_dbase 数据库添加一个数据文件 sample_db, 打开“查询编辑器”窗口, 在其中输入以下语句。

```
ALTER DATABASE my_dbase --指定要修改数据库的名称
ADD FILE
(
    NAME =sample_db, --指定要添加数据文件的名称
    FILENAME= 'D:\database\sample_db.mdf', --指定添加数据文件的存储位置
    SIZE=10MB, --指定添加数据文件的初始大小
    MAXSIZE =100MB, --指定添加数据文件的最大容量
    FILEGROWTH =2MB --指定添加数据文件的自动增长率
);
```

单击“执行”按钮, 即可完成数据文件的添加操作, 如图 3-26 所示。在“对象资源管理器”窗口中选择添加数据文件后的数据库, 右击, 在弹出的快捷菜单中选择“属性”命令, 打开“数据库属性”窗口, 选择“文件”选项, 即可在“数据库文件”列表框中查看添加的数据文件 sample_db, 如图 3-27 所示。另外, 还可以使用 sp_helpdb 来查看一下现有的文件信息, 数据库当前的数据文件组成如图 3-28 所示。

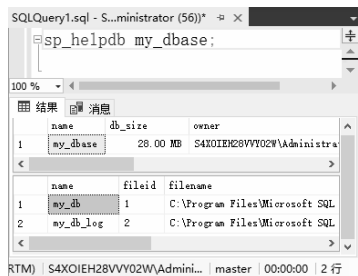


图 3-25 查看数据文件信息

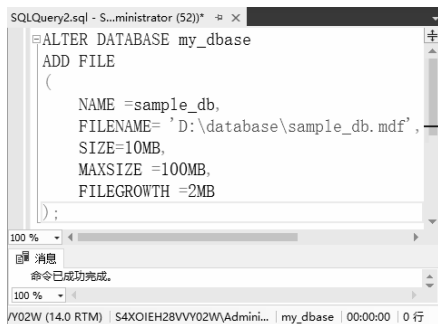


图 3-26 添加数据文件 sample_db

实例 10: 给数据库添加文件组。

在数据库中, 除了可以添加数据文件外, 还可以添加文件组。在 my_dbase 数据库中添加一个文件组, 名称为 my_dbase_group, 然后再为文件组添加一个名称为 sample_db_addnew 的数据文件。

打开“查询编辑器”窗口, 在其中输入以下语句。

```
ALTER DATABASE my_dbase
ADD FILEGROUP my_dbase_group --添加文件组

ALTER DATABASE my_dbase --指定要修改数据库的名称
ADD FILE
(
```

```

NAME =sample_db_addnew,
FILENAME= 'D:\database\sample_db_addnew.ndf',
SIZE=20MB,
MAXSIZE =200MB,
FILEGROWTH =5MB
)
to filegroup my_dbase_group

```

--指定要添加数据文件的名称
 --指定添加数据文件的存储位置
 --指定添加数据文件的初始大小
 --指定添加数据文件的最大容量
 --指定添加数据文件的自动增长率
 --将添加的数据文件指定到添加的文件组中

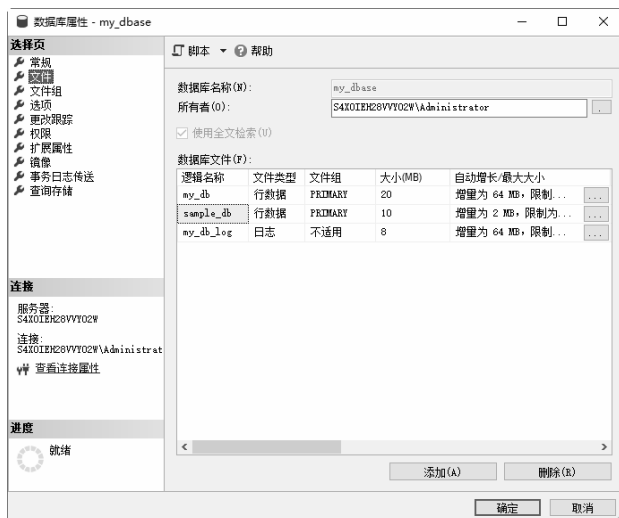


图 3-27 “数据库属性”窗口

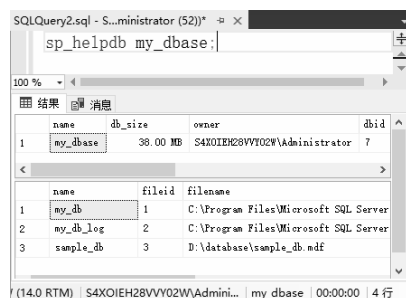


图 3-28 数据文件组成信息

单击“执行”按钮，即可完成文件组的添加操作，如图 3-29 所示。在“对象资源管理器”窗口中选择添加数据文件后的数据库，右击，在弹出的快捷菜单中选择“属性”命令，打开“数据库属性”窗口，选择“文件组”选项，即可在右侧窗格中查看添加的文件组 my_dbase_group，如图 3-30 所示。选择“文件”选项，可以在“数据库文件”类别中查看添加的数据文件信息，如图 3-31 所示。

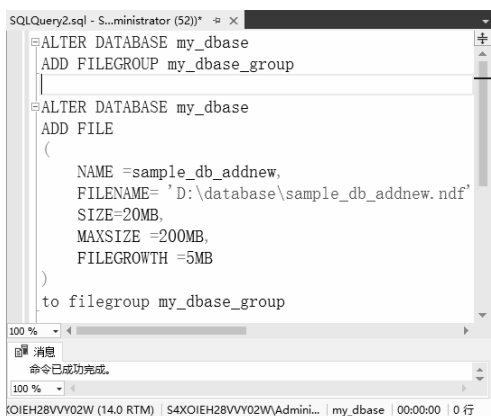


图 3-29 文件组的添加

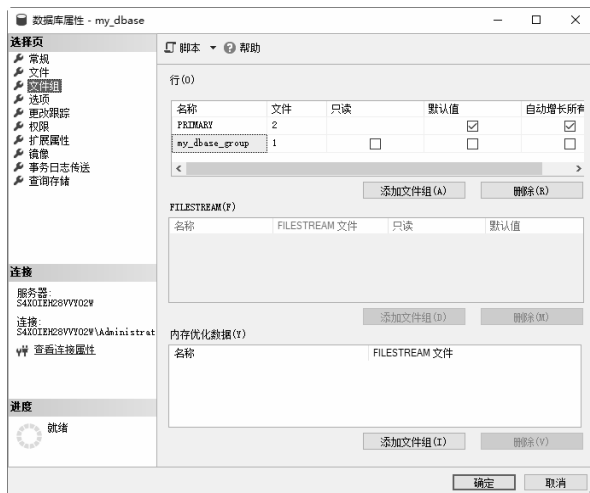


图 3-30 “文件组”选项

另外，还可以使用 sp_helpdb 来查看一下现有的文件信息，数据库当前的数据文件组成如图 3-32 所示。从这里可以看出在 my_dbase 数据库中数据文件列表中多出了一个名称为 sample_db_addnew 的数据文件，并且该数据文件在 my_dbase_group 文件组中。

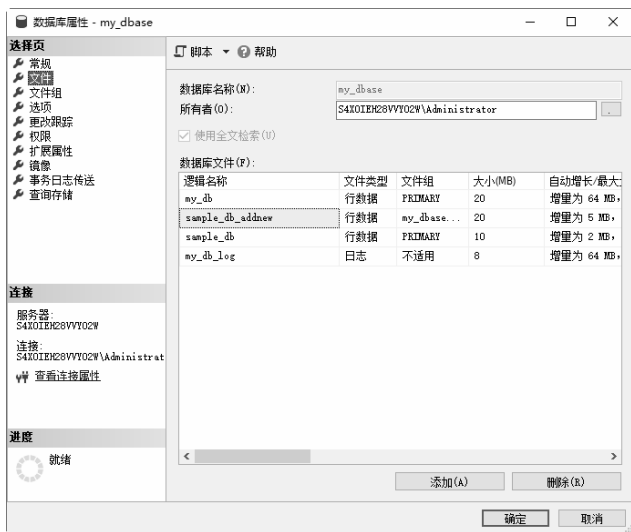


图 3-31 “文件”选项

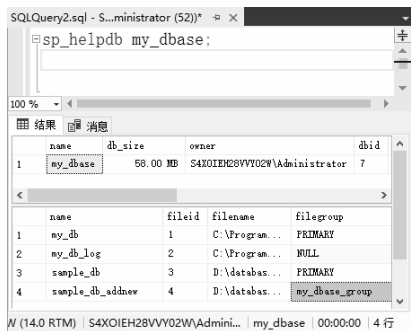


图 3-32 使用 sp_helpdb 查看文件信息

3.2.5 清理数据库中无用的文件

数据库中的文件不能一直添加，对于那些无用的数据文件、日志文件或文件组，我们要及时清理。与如何添加数据文件、日志文件或文件组相比，删除这些文件的操作就比较简单了，具体的语法格式如下。

```
ALTER DATABASE database_name
REMOVE FILE|FILEGROUP file_name|filegroup_name
```

- 主要参数介绍如下。
- database_name: 要修改数据库的名称。
 - REMOVE FILE: 移除文件。移除的文件包括数据文件和日志文件。
 - REMOVE FILEGROUP: 移除数据库中的文件组。注意，要想成功移除文件组，前提是文件组中没有任何文件。

实例 11: 移除数据库 my_dbase 中的 sample_db_addnew，打开“查询编辑器”窗口，在其中输入以下语句。

```
ALTER DATABASE my_dbase --指定要修改数据库的名称
REMOVE FILE sample_db_addnew; --指定要移除的数据文件
```

单击“执行”按钮，即可将数据文件 sample_db_addnew 从数据库中移除，如图 3-33 所示。

实例 12: 移除数据库 my_dbase 中的 my_dbase_group 文件组，打开“查询编辑器”窗口，在其中输入以下语句。

```
ALTER DATABASE my_dbase --指定要修改数据库的名称
REMOVE FILEGROUP my_dbase_group; --指定要移除的文件组
```

单击“执行”按钮，即可将文件组 my_dbase_group 从数据库中移除，如图 3-34 所示。

注意：如果数据文件或文件组下存在有数据信息，那么在执行移除操作时，会出现如图 3-35 所示的错误提示，这就需要事先把数据文件或文件组清空，然后再执行移除操作。

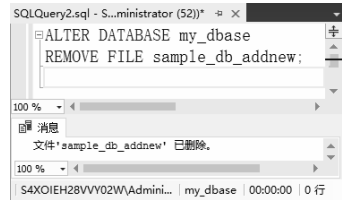


图 3-33 移除数据文件



图 3-34 移除文件组

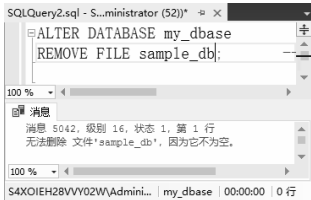


图 3-35 错误信息提示

3.2.6 以图形向导方式修改数据库

在 SQL Server Management Studio 中，我们可以以图形向导方式修改数据库，包括修改数据库的名称、初始大小、最大容量，添加文件或文件组等。

1. 修改数据库的名称

(1) 启动 SQL Server Management Studio 并登录到 SQL Server 2017 数据库，在“对象资源管理器”窗口中打开“数据库”节点，选择需要更改名称的数据库，右击，在弹出的快捷菜单中选择“重命名”命令，如图 3-36 所示。

(2) 在显示的文本框中输入新的数据库名称 newmy_dbase，然后，按 Enter 键确认或在对象资源管理器中的空白处单击，即可完成名称的更改，如图 3-37 所示。



图 3-36 选择“重命名”命令



图 3-37 修改数据库名称

2. 修改数据库的所有者信息

(1) 数据库连接成功之后，在“对象资源管理器”窗口中打开“数据库”节点，选择需要修改的数据库，右击，在弹出的快捷菜单中选择“属性”命令，如图 3-38 所示。

(2) 打开“数据库属性”窗口，在“选择页”列表中选择“文件”选项，进入“文件”设置界面，如图 3-39 所示。



图 3-38 选择“属性”命令

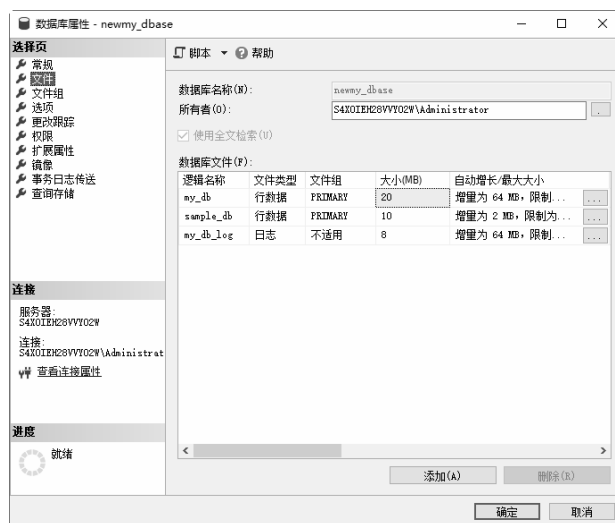


图 3-39 “数据库属性”窗口

(3) 单击“所有者”右侧的“浏览”按钮，打开“选择数据库所有者”对话框，如图 3-40 所示。

(4) 单击“浏览”按钮，打开“查找对象”对话框，在其中选择需要匹配的对象，如图 3-41 所示。

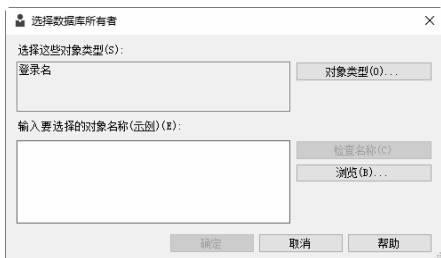


图 3-40 “选择数据库所有者”对话框



图 3-41 “查找对象”对话框

(5) 单击“确定”按钮，返回到“选择数据库所有者”对话框中，在“输入要选择的对象名称”列表框中可以看到添加的所有者信息，如图 3-42 所示。

(6) 单击“确定”按钮，返回到“数据库属性”窗口中，可以看到数据库的所有者发生了改变，如图 3-43 所示。

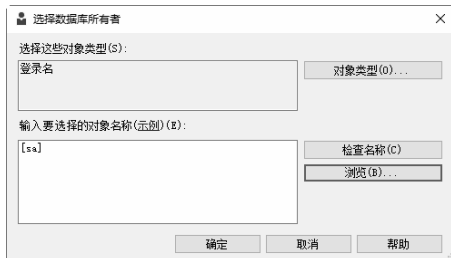


图 3-42 输入要选择的对象名称

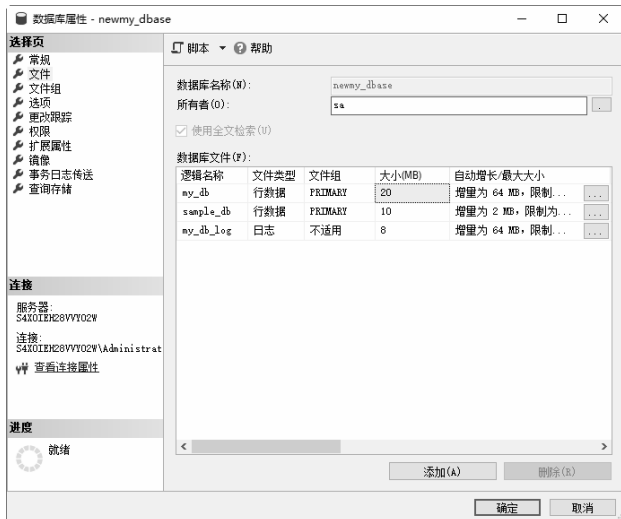


图 3-43 “数据库属性”窗口

3. 修改数据库的初始大小

(1) 选择需要修改的数据库，右击，在弹出的快捷菜单中选择“属性”命令，打开“数据库属性”窗口，选择“文件”选项卡，如图 3-44 所示。

(2) 单击 my_db 行的初始大小列下的文本框，重新输入一个新值，这里输入“25”，单击“确定”按钮，即可完成数据文件大小的修改，如图 3-45 所示。

提示：也可以单击旁边的两个小箭头按钮，增大或者减小值，修改完成之后，读者可以重新打开 newmy_dbase 数据库的属性窗口，查看修改结果。

4. 修改数据库的最大容量

(1) 选择需要增加数据库容量的数据库，这里选择 newmy_dbase 数据库，然后打开“数据库属性”窗口，选择左侧的“文件”选项卡，在 my_db 行中，单击“自动增长”列下面“...”按钮，如图 3-46 所示。

(2) 弹出“更改 my_db 的自动增长设置”对话框，在“最大文件大小”文本框输入值 350，增加数据库的增长限制，如图 3-47 所示。

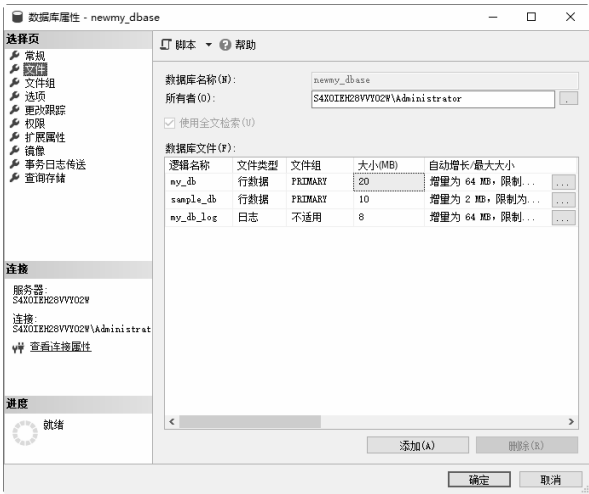


图 3-44 “文件”选项卡



图 3-45 修改初始大小为 25

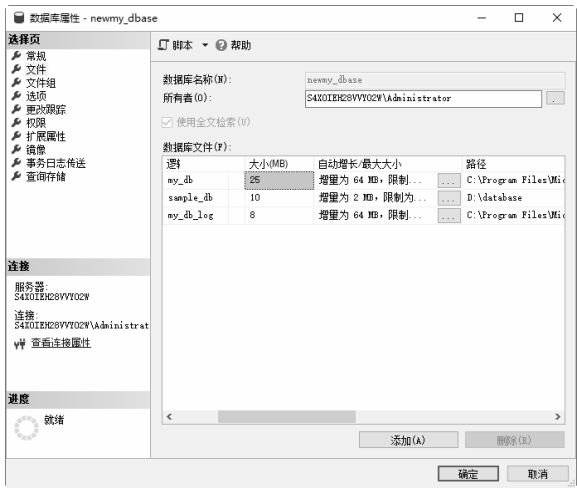


图 3-46 “数据库属性”窗口

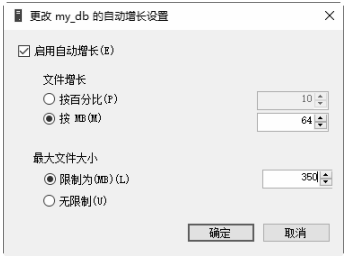


图 3-47 “更改 my_db 的自动增长设置”对话框

(3) 单击“确定”按钮，返回到“数据库属性”窗口，即可看到修改后的结果，单击“确定”按钮完成修改，如图 3-48 所示。

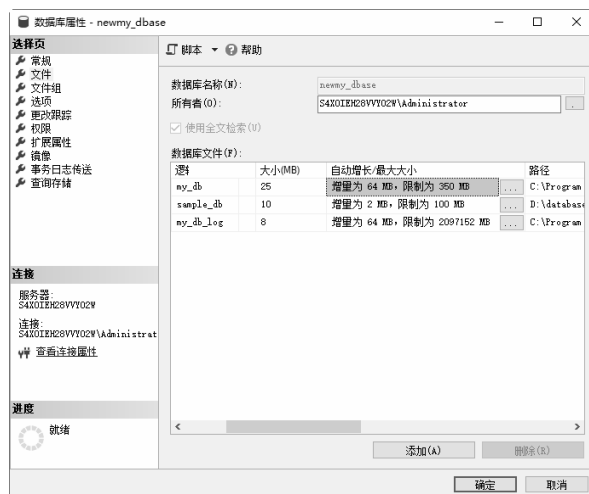


图 3-48 修改最大大小

5. 修改数据库的其他属性

(1) 在“数据库属性”窗口中，选择“文件组”选项，进入“文件组”设置界面，通过单击“添加文件组”按钮，可以对数据库文件组进行添加操作，如图 3-49 所示。

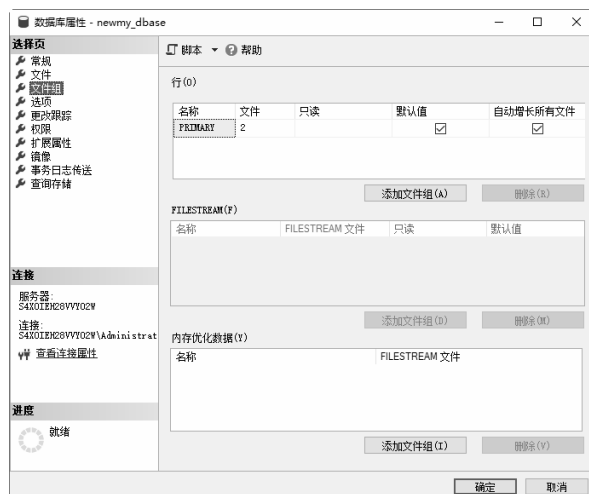


图 3-49 “文件组”设置界面

(2) 选择“选项”选项，在打开的界面中可以对排序规则、恢复模式、兼容性级别等参数进行修改，如图 3-50 所示。

(3) 选择“更改跟踪”选项，单击“更改跟踪”右侧的下拉按钮，可以设置是否对数据库启用更改跟踪，如图 3-51 所示。

(4) 选择“权限”选项，在打开的界面中可以对服务器的名称、数据库的名称、用户或角色进行修改，如图 3-52 所示。

(5) 选择“扩展属性”选项，在打开的界面中可以对数据库的排序规则、属性等参数进行设置，如图 3-53 所示。

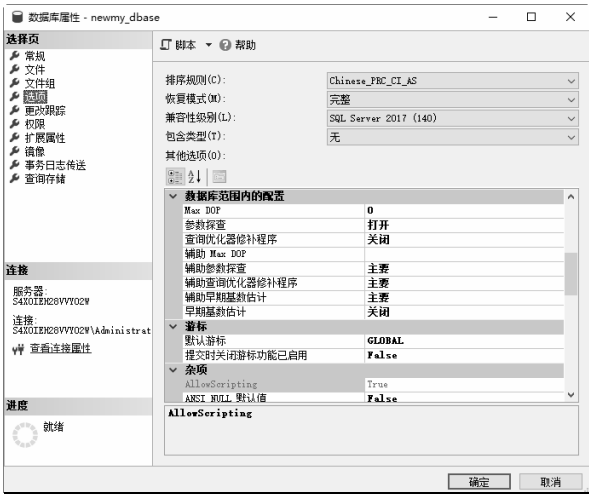


图 3-50 “选项” 设置界面

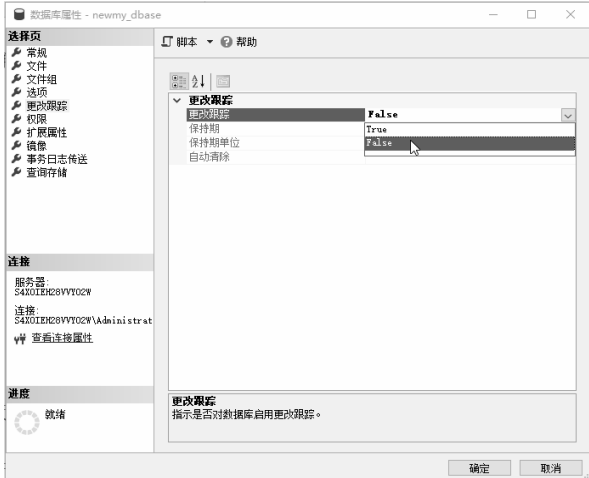


图 3-51 “更改跟踪” 设置界面

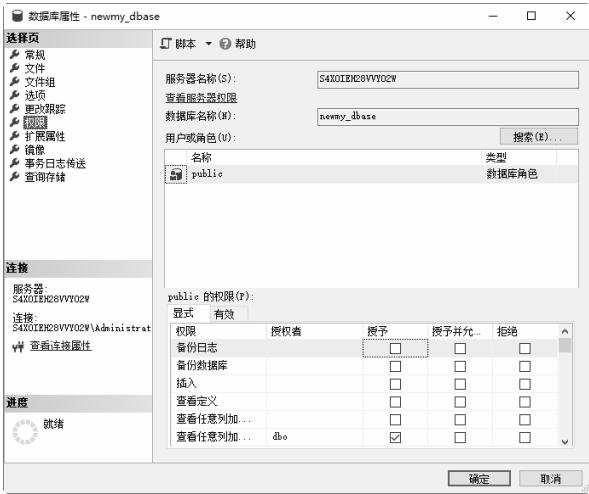


图 3-52 “权限” 设置界面

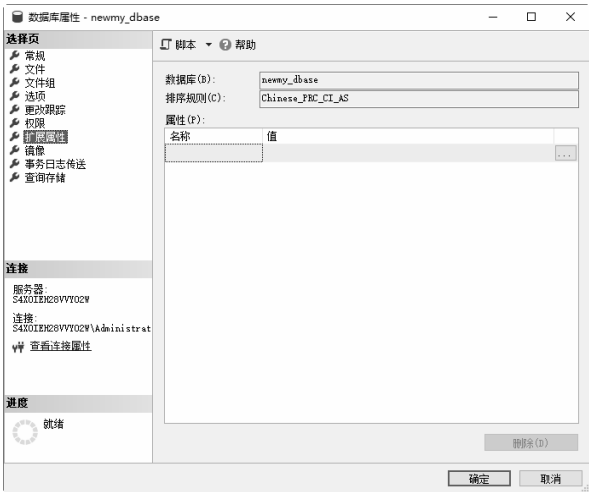


图 3-53 “扩展属性” 设置界面

- (6) 选择“镜像”选项，在打开的界面中可以对数据库镜像进行安全设置，如图 3-54 所示。
- (7) 选择“事务日志传送”选项，在打开的界面中可以设置是否启用将此数据库作为日志传送配置中的主数据库，如图 3-55 所示。
- (8) 选择“查询存储”选项，在打开的界面中可以设置查询存储保留参数、操作模式等选项，如图 3-56 所示。

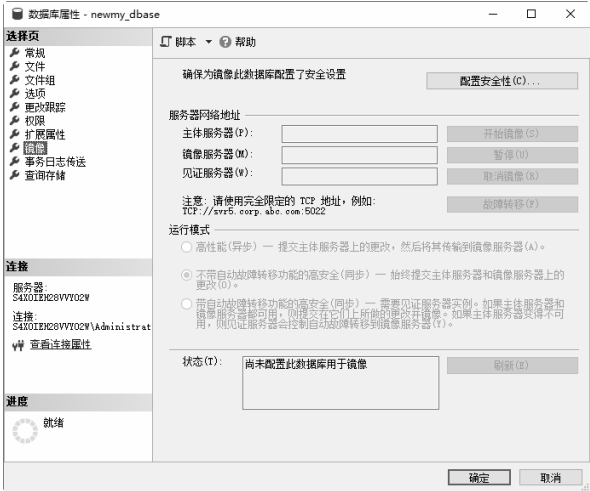


图 3-54 “镜像” 设置界面

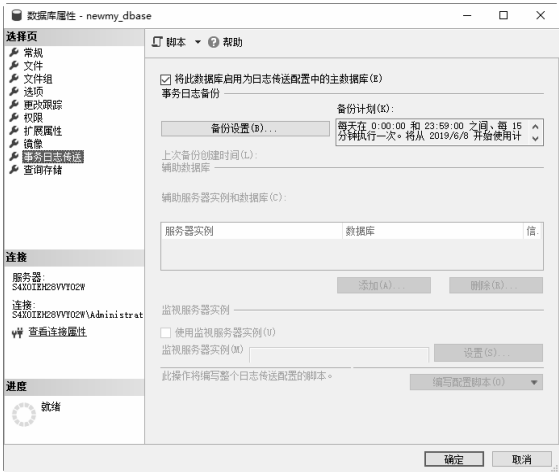


图 3-55 “事务日志传送” 设置界面



图 3-56 “查询存储”设置界面

3.3 删除数据库

当数据库中的所有数据文件都不再需要时，说明整个数据库就没有用了。这时，就可以将该数据库删除。删除之后，相应的数据库文件及其数据都会被删除，且不可恢复。

3.3.1 一行语句删除数据库

删除数据库的语句非常简单，使用 DROP DATABASE 语句就可以删除，具体的语法格式如下。

```
DROP DATABASE database_name [, ...n];
```

主要参数介绍如下。

- database_name: 要删除数据库的名称。
- [, ...n]: 表示可以有多个数据库名称，多个名称之间用逗号分隔。

实例 13: 删除名称为 mydb 的数据库。

在“查询编辑器”窗口中输入以下语句。

```
DROP DATABASE mydb;
```

单击“执行”按钮，mydb 数据库将被删除，如图 3-57 所示。

注意：使用 DROP DATABASE 删除数据库不会出现确认信息，因此使用这种方法时要小心谨慎。此外，不能删除系统数据库，否则会导致 SQL Server 2017 服务器无法使用。

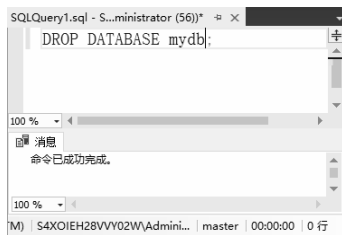


图 3-57 删除数据库

3.3.2 以图形向导方式删除数据库

在 SQL Server Management Studio 中，可以使用图形向导方式删除数据库，这种方法最简单，也最直观。具体删除过程可以分为如下几步。

(1) 登录到 SQL Server 2017 数据库之中，在“对象资源管理器”窗口中选中需要删除的数据库，右击，在弹出的快捷菜单中选择“删除”命令或直接按下 Delete 键，如图 3-58 所示。

(2) 打开“删除对象”窗口，用来确认删除的目标数据库对象，在该窗口中也可以选择是否要“删除数据库备份和还原历史记录信息”和“关闭现有连接”，单击“确定”按钮，即可将数据库删除，如图 3-59 所示。

注意：每次删除时，只能删除一个数据库。而且，并不是所有的数据库在任何时候都可以被删除，删除数据库必须满足以下条件。



图 3-58 选择“删除”命令

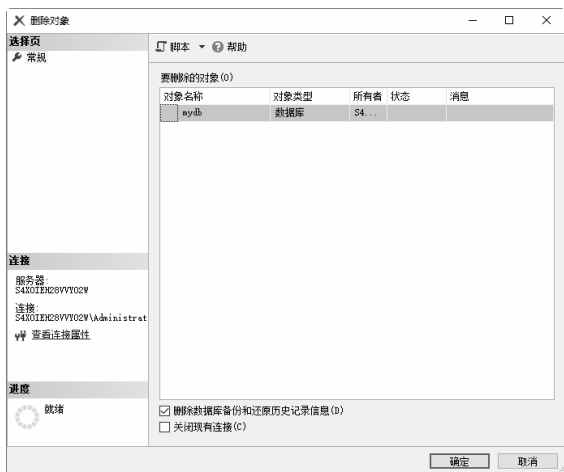


图 3-59 “删除对象”窗口

- 如果数据库涉及日志传送操作，在删除数据库之前必须取消日志传送操作。
- 如果要删除为事务复制发布的数据库，或删除为合并复制发布或订阅的数据库，必须首先从数据库中删除备份。如果数据库已经损坏，不能删除备份，可以先将数据库设置为脱机状态，然后再删除。
- 如果数据库上存在数据库快照，必须首先删除数据库快照，然后再执行删除数据库操作。

总之，只有处于正常状态下的数据库，才能被删除。当数据库处于正在使用、正在恢复、数据库包含用于复制的对象时，都不能被删除。

3.4 查看数据库信息

SQL Server 中可以使用多种方式查看数据库信息，查看系统中全部数据库信息、数据库中的数据文件、数据库的空间使用情况等。

3.4.1 查看系统中的全部数据库

使用存储过程 `sp_helpdb` 可以查看系统中的全部数据库信息，保存系统自带的数据库和用户自定义数据库。

实例 14：使用存储过程 `sp_helpdb` 查看全部数据库。

在“查询编辑器”窗口中输入以下语句。

```
sp_helpdb;
```

单击“执行”按钮，即可在“结果”窗格中显示当前系统中存在的全部数据库信息，其中 `newmy_dbase` 为自定义数据库，其余为系统数据库，如图 3-60 所示。

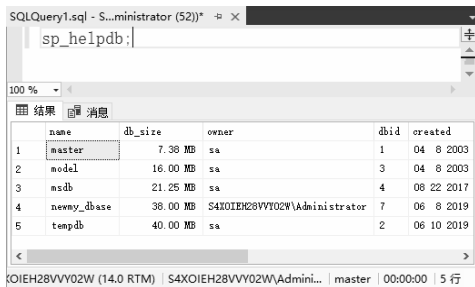


图 3-60 查询全部数据库信息

3.4.2 查看数据库中的文件信息

使用存储过程 `sp_helpdb` 可以查看数据库中的文件信息，具体的方法就是在存储过程 `sp_helpdb` 后面添加上数据库的名称就可以了。

实例 15：使用存储过程 `sp_helpdb` 查看数据库的文件。

在“查询编辑器”窗口中输入以下语句。

```
sp_helpdb newmy_dbase;
```

单击“执行”按钮，即可在“结果”窗格中显示当前数据库中存在的文件信息，该数据库包括两个



微视频

数据文件与一个日志文件，如图 3-61 所示。

3.4.3 查看数据库的空间使用情况

查看数据库的空间使用情况，可以更好地利用数据的空间。查看数据库空间的使用情况可以使用存储过程 `sp_spaceused` 来查看。

实例 16：使用存储过程 `sp_spaceused` 查看数据库空间的使用情况。

在“查询编辑器”窗口中输入以下语句。

```
use newmy_dbase
exec sp_spaceused;
```

单击“执行”按钮，即可在“结果”窗格中显示当前数据库空间的使用情况，可以得到数据库 `newmy_dbase` 中数据的大小（`database_size`）、未分配的空间（`unallocated space`）和数据使用的容量（`data`）等信息，如图 3-62 所示。

3.4.4 查看数据库和文件的状态

通过查看数据库和文件的状态，可以了解当前数据库是否在线且处于可用状态。

1. 查看数据库的状态

使用 `sys.databases` 可以查看数据库的基本信息以及使用状态。

实例 17：查看数据库 `newmy_dbase` 的状态，在“查询编辑器”窗口中输入以下语句。

```
Select name AS '数据库名',state_desc AS '状态'
FROM sys.databases WHERE name='newmy_dbase';
```

单击“执行”按钮，即可在“结果”窗格中显示当前数据库的状态，该状态为 `ONLINE`，表示该数据库在线且可用，如图 3-63 所示。

2. 查看数据文件的状态

使用 `sys.master_files` 可以查看数据库中数据文件的使用状态。

实例 18：查看数据库 `newmy_dbase` 中 `my_db` 文件的状态，在“查询编辑器”窗口中输入以下语句。

```
Select name AS '数据文件名',state_desc AS '状态'
FROM sys.master_files WHERE name='my_db';
```

单击“执行”按钮，即可在“结果”窗格中显示当前数据文件的状态，该状态为 `ONLINE`，如图 3-64 所示。

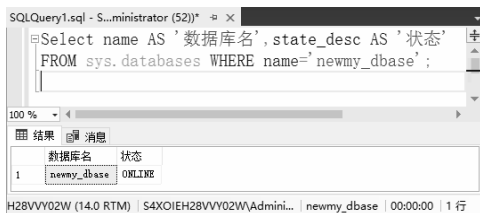


图 3-63 查看数据库的状态

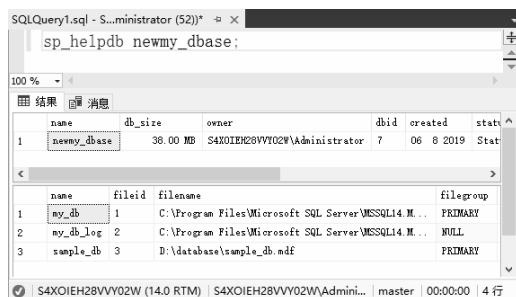


图 3-61 查看数据库中存在的文件信息

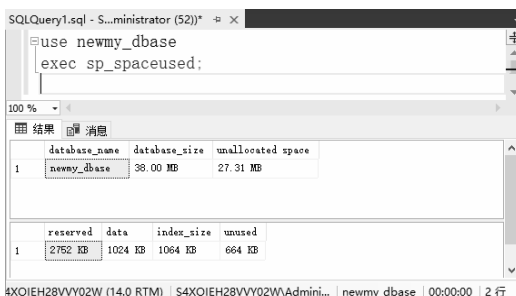


图 3-62 查看数据库的空间使用情况

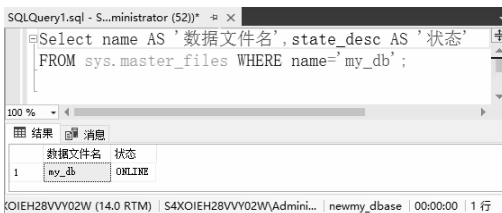


图 3-64 查看当前数据文件的状态

3. 使用函数查询数据库状态

使用 `DATABASEPROPERTYEX()` 函数可以查看数据库的状态。

实例 19：查看 `newmy_dbase` 数据库的状态信息，在“查询编辑器”窗口中输入以下语句。

```
SELECT DATABASEPROPERTYEX('newmy_dbase', 'Status')
AS '数据库状态'
```

单击“执行”按钮，即可在“结果”窗格中显示当前数据库的状态，该状态为 ONLINE，如图 3-65 所示。

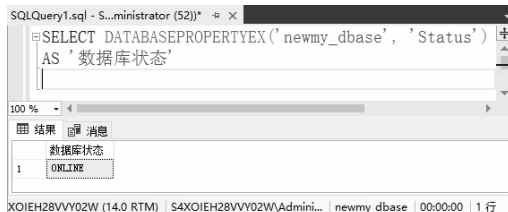


图 3-65 查看数据库状态信息

3.5 课后习题与练习

一、填空题

1. 数据库中主数据文件的扩展名是_____，次数据文件的扩展名是_____，日志文件的扩展名是_____。

答案：.mdf, .ndf, .ldf

2. 数据库通常由_____和_____组成。

答案：数据文件，日志文件

3. 数据库文件的_____状态，表示该数据库当前在线且可用。

答案：ONLINE

4. 删除数据库使用的语句是_____。

答案：DROP DATABASE

5. 如果需要将 test 数据库重命名为“测试数据库”，使用 sp_renamedb 存储过程的实现语句是_____。

答案：sp_renamedb test, 测试数据库;

二、选择题

1. 下面关于数据库的说法错误的是_____。

- A. 一个数据库中至少有一个数据文件，但可以没有日志文件
- B. 一个数据库中至少有一个数据文件和一个日志文件
- C. 一个数据库中可以有多个数据文件
- D. 一个数据库中可以有多个日志文件

答案：A

2. 在创建数据库时，系统会自动将_____系统数据库中的所有用户定义的对象复制到新建的数据库中。

- A. master
- B. model
- C. msdb
- D. tempdb

答案：A

3. 下面关于创建数据库说法正确的是_____。

- A. 创建数据库时文件名必须带有扩展名
- B. 创建数据库时文件名可以不带扩展名
- C. 创建数据库时数据文件可以不带扩展名，日志文件必须带有扩展名
- D. 创建数据库时日志文件可以不带扩展名，数据文件必须带有扩展名

答案: B

4. 如果想要查看数据库 mydb 的状态, 下面_____语句是不正确的。

- A. sp_helpdb mydb;
- B. SELECT state FROM mydb;
- C. SELECT DATABASEPROPERTYEX('mydb', 'Status')
- D. SELECT name,state_desc FROM sys.databases WHERE name='mydb';

答案: B

5. 下面对于修改数据库的描述正确的是_____。

- A. 数据库一旦创建完毕, 不能对其名称进行修改
- B. 在数据库创建完毕后, 不能随意更改数据库的大小
- C. 可以使用存储过程 sp_renamedb 修改数据库的名称
- D. 以上说法都不对

答案: C

三、简答题

1. 简述更改数据库名称的方法。
2. 如何创建包含文件组的数据库?
3. 如何删除数据库中的文件?

3.6 新手疑难问题解答

疑问 1: 在创建包含文件组的数据库时, 为什么要指定自定义文件组, 用户自定义文件组有什么用呢?

解答: 通过指定自定义文件组, 就可以指定在文件组中存放的数据文件了。如果没有自定义文件组, 数据文件会自动地划分到文件组中。换言之, 文件组为数据库管理员管理数据文件提供了方便。

疑问 2: 如果数据库管理员忘记了创建数据库的文件夹, 数据库就找不到了吗?

解答: 肯定是可以找到的。我们可以使用存储过程 sp_helpdb 来查询系统中存在的全部数据库信息, 然后可以从这些数据库列表中找到创建的数据库文件, 包括数据库的名称、数据大小、拥有者、当前状态以及存放的位置。

3.7 实战训练

创建数据库 MyStudent, 并对该数据库进行管理。具体内容如下:

(1) 创建数据库 MyStudent, 其中主要数据文件的逻辑名称为 MyStudentdate, 对应的物理文件是“C:\MyStudentdate.mdf”, 初始大小为 20MB, 最大文件大小为 100MB, 增长幅度为 2%, 日志文件的逻辑名称为 MyStudentlog, 对应的物理文件是“C:\MyStudentlog.ldf”, 初始大小为 5MB, 最大文件大小为 50MB, 增长幅度为 2MB。

(2) 查看 MyStudent 数据库的信息。

(3) 修改数据库 MyStudent 的名称为 MyStudent_01。

(4) 创建文件组 SCOREGROUP, 添加数据文件 Score_data, 对应的物理文件是“C:\MyScore_data.nd”, 将其文件组设置为新创建的 SCOREGROUP 文件组, 将 Score_data 的文件增长幅度设置为“按 10%增长”, 最大文件的大小设置为 100MB。

(5) 删除数据库 MyStudent。