

项目三

学生信息数据库的创建和管理



学习目标

- (1) 掌握：数据库的创建和管理,数据库的分离和附加。
- (2) 理解：SQL Server 数据库的结构。



学习任务

使用 SQL Server 2022 创建和管理学生信息数据库,并使用 SQL Server 2022 分离和附加该数据库。



知识学习

3.1 SQL Server 数据库的结构



视频讲解

SQL Server 数据库的结构包括数据存储、数据库文件、文件组和数据库对象,数据库的结构主要描述 SQL Server 2022 如何分配数据库空间。

3.1.1 数据存储

SQL Server 有两种存储结构,分别是逻辑存储结构和物理存储结构。逻辑存储结构是指数据库中包含哪些对象,这些对象可以实现什么样的功能,SQL Server 数据库不仅仅只是数据的存储,所有与数据处理操作相关的信息都存储在数据库中;物理存储结构是指数据库文件在磁盘中的存储方式。

3.1.2 数据库的逻辑存储结构

SQL Server 数据库不仅仅是数据的存储,所有与数据处理操作相关的信息都存储在数据库中。实际上,SQL Server 数据库是由表、视图、索引等各种不同的数据库对象组成的,它们分别用来存储特定信息并支持特定功能,构成数据库的逻辑存储结构。SQL Server 包含的对象以及对各对象的简要说明如下。

1. 表

表是 SQL Server 中最重要的数据库对象,表由行和列组成,其定义了具有关联列的行的集合,用来存储和操作数据的逻辑结构。

2. 数据类型

定义列或变量的数据类型,SQL Server 提供了系统数据类型,并允许用户自定义数据类型。

3. 视图

视图也称为虚拟表,是从一个或多个基本表中引出的表,本身不存储实际数据。经过定义的视图,可以进行查询、修改、删除和更新。数据库中只存放视图的定义而不存放视图对应的数据,这些数据存放在导出视图的基本表中。当基本表中的数据发生变化时,根据视图查询出的数据也发生变化。

4. 索引

索引是一种存储结构,能够在无须扫描整个数据表的情况下实现对表中数据的快速访问,索引是关系数据库的内部实现技术,存放于存储文件中。

5. 约束

约束定义了可取的值的规则,约束机制保障了数据库中数据的一致性和完整性。

6. 默认值

为列提供的默认值。

7. 存储过程

存储过程是执行预编译交互式 SQL 语句的集合,是封装了可重用代码的模块或例程。语句集合经过编译后存储在数据库中,能够接收输入参数、输出参数、返回结果和消息等。

8. 触发器

触发器是一种特殊的存储过程的形式,它与表紧密关联,当用户对表或视图中的数据进行修改时,触发器将自动执行。触发器能够实现更为复杂的数据操作,有效保障数据库中数据的完整性和一致性。

3.1.3 数据库的物理存储结构

SQL Server 中的物理存储结构主要有文件、文件组、页和盘区等,主要描述 SQL Server 如何为数据库分配空间。

1. 主数据文件

主数据文件简称主文件,是数据库的起点,指向数据库中的其他文件,包含了数据库的启动信息,用于存储数据。每个数据库都必须有一个主数据文件,其默认扩展名是 .mdf。

2. 次要数据文件

次要数据文件用于辅助主文件存储数据,存储未包含在主文件内的其他数据。某些数据库可能不需要次要数据文件,而有些数据库则需要多个次要数据文件。当数据库非常大时,则可能需要多个次要数据文件;当数据库主文件足够大时,可以容纳所有数据,则不需要次要数据文件。次要数据文件的默认扩展名是 .ndf。

3. 日志文件

日志文件用于保存日后恢复数据库的所有日志信息。每个数据库必须至少有一个日志文件,也可以有多个。日志文件的默认扩展名是 .ldf。

在 SQL Server 2022 中,一个数据库至少包含一个主数据文件和一个日志文件。一般情况下,数据库具有一个主数据文件和一个或多个日志文件,可能还具有次要数据文件。

4. 文件组

文件组是在数据库中组织文件的一种管理机制,它将多个数据文件集成成一个整体,便于管理和分配数据。SQL Server 有两种类型的文件组:主文件组 and 用户定义文件组。

主文件组,包含主数据文件和未明确分配给其他文件组的其他文件。系统表的所有页都分配在主文件组中。

用户定义文件组,是通过在 CREATE DATABASE 或 ALTER DATABASE 语句中使用 FILEGROUP 关键字指定的任何文件组。

在创建数据表时,用户可以指定表到某个文件组,并且通过设置文件组,可以提高数据库的性能。用户可以指定默认文件组,如果用户没有指定默认文件组,则主文件组是默认文件组。



任务实施

3.2 使用 SSMS 创建和管理学生信息数据库

可以用 SQL Server Management Studio 创建和管理数据库。对数据库进行操作主要包括数据库的创建、修改、删除、分离和附加。

3.2.1 使用 SSMS 创建学生信息数据库

在创建数据库时,必须为其确定名称,为每一个文件指定逻辑名、物理名和大小等。具体步骤如下。

(1) 打开 SQL Server Management Studio,连接到 SQL Server 上的数据库引擎。

(2) 展开服务器,右击“数据库”文件夹,在弹出的快捷菜单中选择“新建数据库”命令,如图 3-1 所示。



视频讲解



图 3-1 选择“新建数据库”命令

(3) 打开“新建数据库”窗口,如图 3-2 所示。在“常规”页中,输入学生信息数据库名称“stuinfo”,在“数据库文件”栏中确定数据库文件的逻辑名称、初始大小、自动增长方式、存储位置等。

(4) 若要添加数据文件或日志文件,可单击“新建数据库”窗口下方的“添加”按钮,输入相应的信息。



图 3-2 “新建数据库”窗口

(5) 若要添加文件组,选择“文件组”页,如图 3-3 所示,单击“添加文件组”按钮,输入文件组名称。

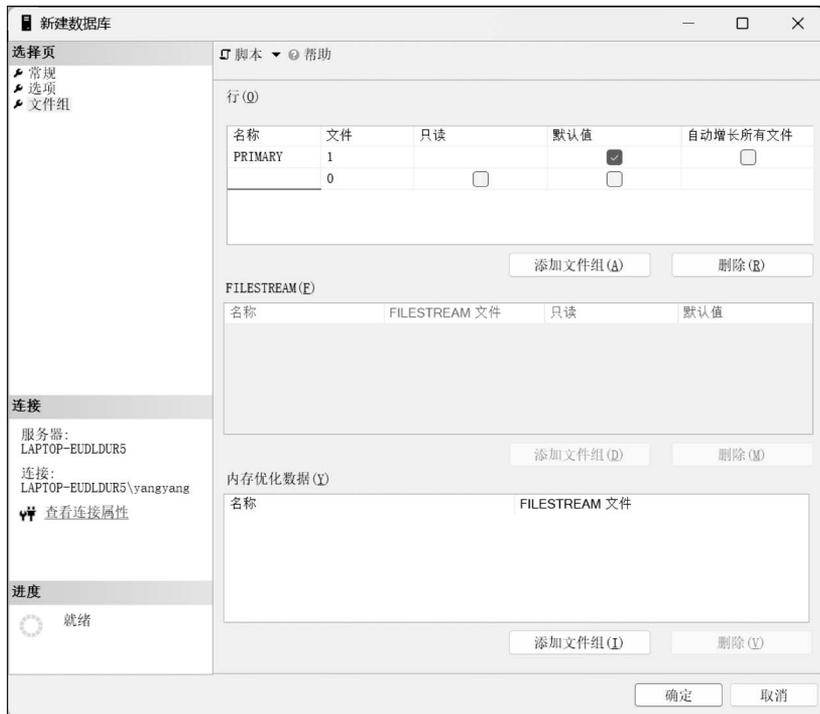


图 3-3 “新建数据库”窗口中的“文件组”页

(6) 单击“确定”按钮,完成学生信息数据库 stuserinfo 的创建。

3.2.2 使用 SSMS 修改和删除学生信息数据库

当数据库创建后,数据文件名和日志文件名不能修改,有时需要对数据库其他选项进行修改,如增加或删除数据文件和日志文件,修改数据文件和日志文件的大小、增长方式,修改数据库选项等。

随着数据库系统的长时间使用之后,运行效率逐渐下降,有一些数据库不再需要使用,或者其已被移到其他数据库或服务器上时,可以删除这些数据库,数据库删除之后,文件及其数据都被删除,及时释放所占的资源 and 空间。

【例 3.1】 在已创建好的 stuserinfo 数据库中,将其主文件的初始大小修改为 10MB,主文件增长方式修改为按百分比增长,每次增长 5%,最大可增长到 200MB;并向该数据库中添加数据文件 stufodata,其属性取默认值;再向该数据库添加一个名为 stufogroup 的文件组,设置其为只读。

操作步骤如下。

(1) 打开 SQL Server Management Studio,连接到 SQL Server 上的数据库引擎。

(2) 展开服务器中的“数据库”。

(3) 选择要修改的数据库 stuserinfo,右击,在弹出的快捷菜单中选择“属性”命令。

(4) 打开“数据库属性-stuserinfo”窗口,在“文件”页中,在主文件的“初始大小”文本框内输入 10,单击主文件“自动增长”栏后的按钮,弹出“更改 stuserinfo 的自动增长设置”对话框,如图 3-4 所示,将其增长方式设置为按百分比增长,每次增长 5%,限制文件增长到 200MB,单击“确定”按钮。

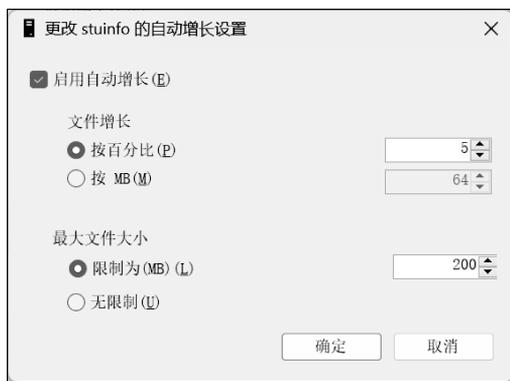


图 3-4 “更改 stuserinfo 的自动增长设置”对话框

(5) 返回“数据库属性-stuserinfo”窗口,单击“文件”页右下方的“添加”按钮,数据库文件下方增加一行文件项,如图 3-5 所示,在该文件项的“逻辑名称”文本框中输入 stufodata,其他属性不变。

(6) 单击“文件组”页右下方的“添加文件组”按钮,文件组下方增加一行文件组项,如图 3-6 所示,在该文件组项的“名称”文本框中输入 stufogroup,勾选“只读”复选框,单击“确定”按钮,完成数据库的修改。



图 3-5 “数据库属性-stuinfo”窗口中的“文件”页

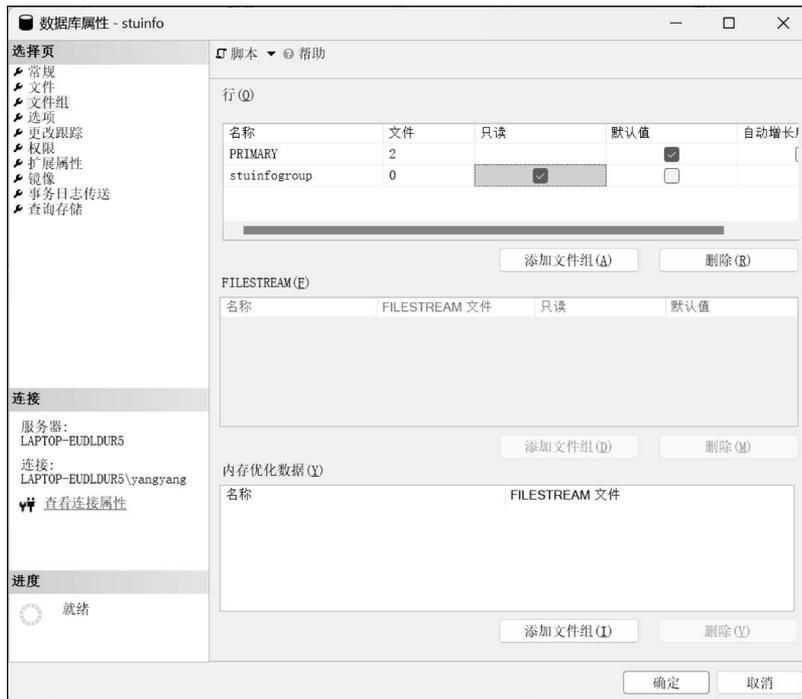


图 3-6 “数据库属性-stuinfo”窗口中的“文件组”页

【例 3.2】 将创建的数据库 stuinfo 的名称修改为 stuinfo1。

操作步骤如下。

- (1) 打开 SQL Server Management Studio, 连接到 SQL Server 上的数据库引擎。
- (2) 展开服务器中的“数据库”文件夹。
- (3) 右击要修改的数据库 stuinfo, 在弹出的快捷菜单中选择“重命名”命令, 如图 3-7 所示。



图 3-7 选择“重命名”命令

- (4) 输入新的数据库名称 stuinfo1, 成功修改数据库的名称。

【例 3.3】 删除 stuinfo1 数据库。

操作步骤如下。

- (1) 打开 SQL Server Management Studio, 连接到 SQL Server 上的数据库引擎。
- (2) 展开服务器中的“数据库”文件夹。
- (3) 右击 stuinfo1 数据库, 在弹出的快捷菜单中选择“删除”命令, 如图 3-8 所示。



图 3-8 选择“删除”命令

- (4) 弹出“删除对象”对话框, 单击“确定”按钮, 删除 stuinfo1 数据库。

重要提示：

修改数据库文件的初始大小时,新指定的空间大小值需大于或等于当前文件初始空间的值;修改数据库后,最好及时备份 master 数据库。删除数据库文件或文件组,选中需删除的文件或文件组,单击窗口右下方的“删除”按钮,单击“确定”按钮后即可删除。但不能删除主文件组(PRIMARY)。

重命名数据库的前提条件是确保没有人使用该数据库,并且将数据库设置为单用户模式。由于数据库创建之后,大多数应用程序可能已经使用该名称,因此,不建议用户重命名已经创建好的数据库。

数据库删除之后,它将被永久删除,将不能再对该数据库进行任何操作。当有用户正在使用某个数据库时,该数据库是不能被删除的。系统数据库是不能删除的。删除数据库后应及时备份 master 数据库。

3.2.3 使用 SSMS 分离和附加学生信息数据库

在实际应用中,需要通过数据库的分离和附加来实现将数据库移到另一台计算机上。分离和附加功能允许在实例和服务器之间移动和复制数据库,也可以在不删除关联数据文件和日志文件的情况下从实例中移走数据库。

【例 3.4】 分离 stuintol 数据库(假设例 3.3 中 stuintol 数据库没有删除)。

操作步骤如下。

- (1) 打开 SQL Server Management Studio,连接到 SQL Server 上的数据库引擎。
- (2) 展开服务器中的“数据库”文件夹。
- (3) 右击 stuintol 数据库,在弹出的快捷菜单中选择“任务”→“分离”命令,如图 3-9 所示。



图 3-9 选择“任务”→“分离”命令

- (4) 弹出“分离数据库”窗口,如图 3-10 所示。

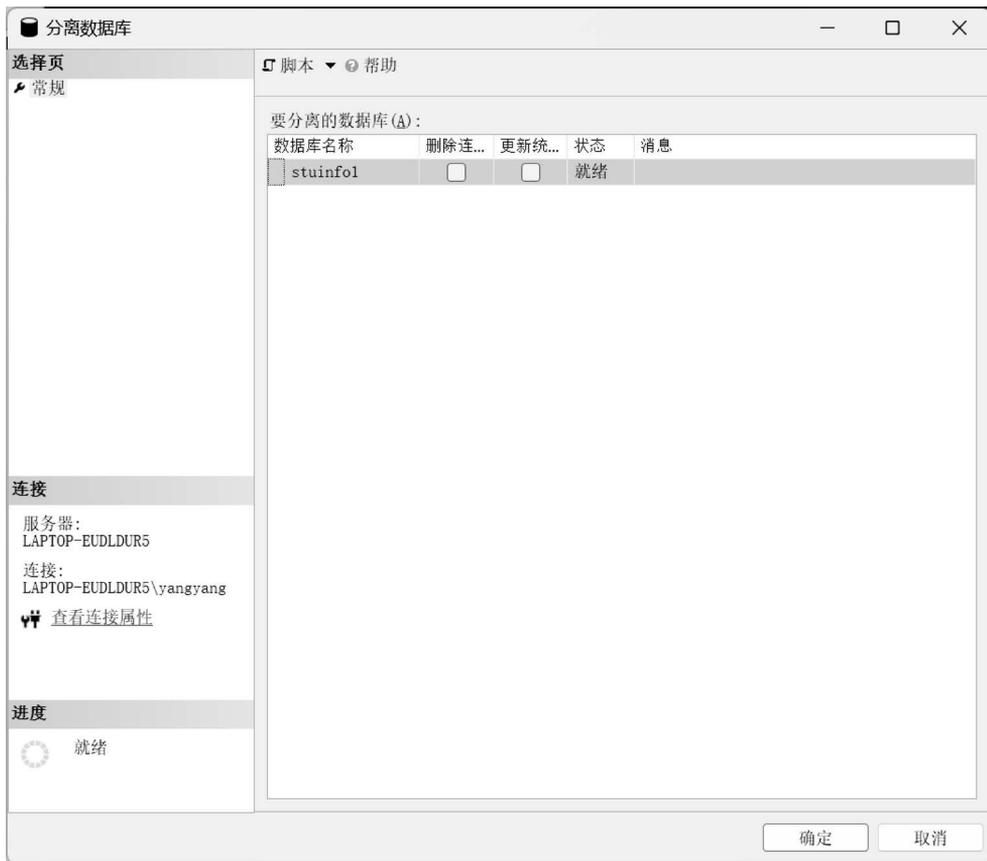


图 3-10 “分离数据库”窗口

(5) 单击“确定”按钮,完成数据库的分离。

【例 3.5】 将例 3.4 分离的 stuiinfo1 数据库附加至本地服务器中。

操作步骤如下。

(1) 打开 SQL Server Management Studio,连接到 SQL Server 上的数据库引擎。

(2) 展开服务器,右击“数据库”文件夹,在弹出的快捷菜单中选择“附加”命令,如图 3-11 所示。



图 3-11 选择“附加”命令

(3) 弹出“附加数据库”窗口,在该窗口中单击“添加”按钮,在弹出的“定位数据库文件”窗口中,如图 3-12 所示,选择要导入的数据库文件 stuinfo.mdf。

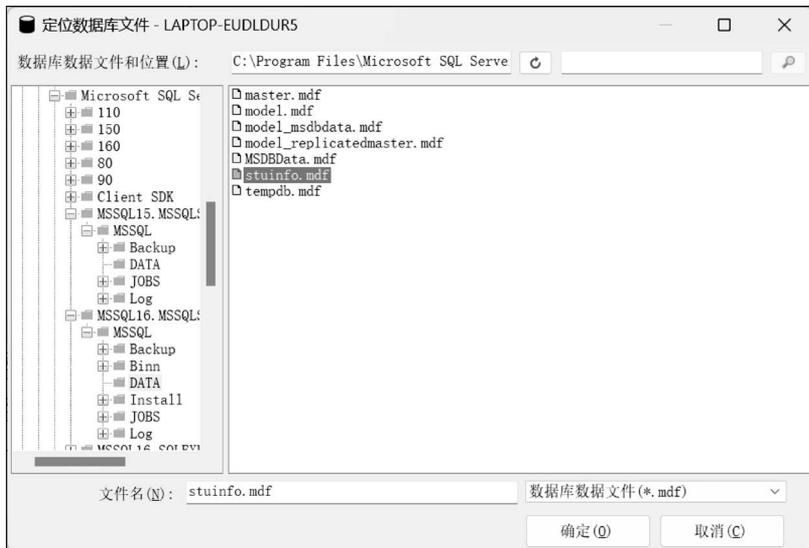


图 3-12 “定位数据库文件”窗口

(4) 在图 3-12 中,单击“确定”按钮,返回“附加数据库”窗口,此时数据库文件已添加进来,如图 3-13 所示。

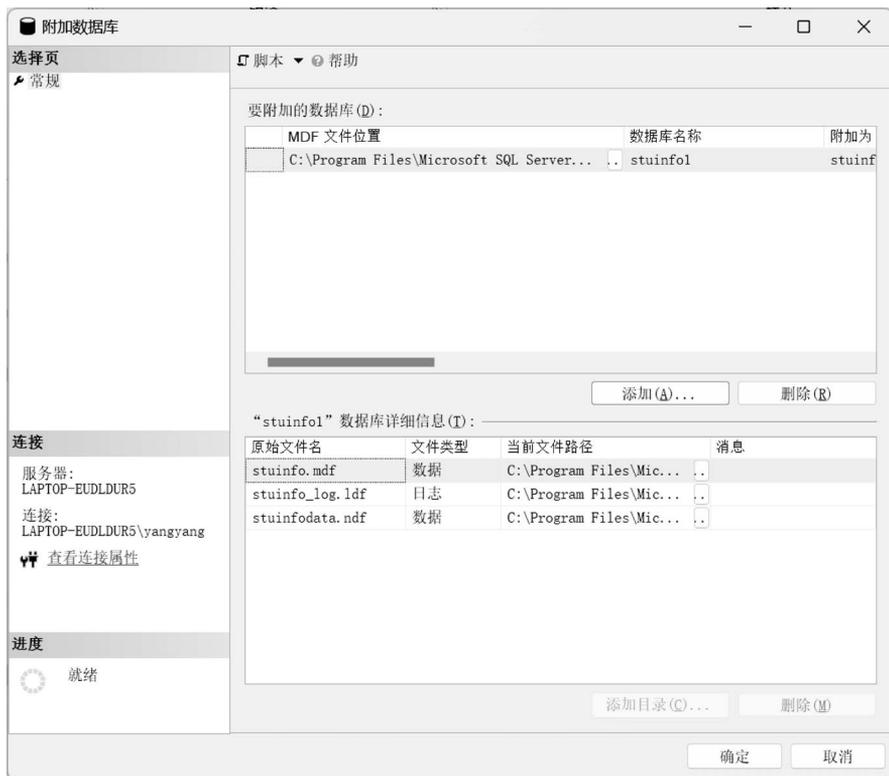


图 3-13 已添加数据库文件的“附加数据库”窗口

(5) 单击“确定”按钮,开始附加 stuinfol 数据库。附加成功后,在“数据库”文件夹下可以找到 stuinfol 数据库。

重要提示:

附加数据库时,所有数据库文件都必须可用。如果任何数据库文件的路径不同于第一次创建数据库或上次附加数据库时的路径,则必须指定文件的当前路径。如果当前数据库中存在与要附加的数据库同名的数据库,附加操作将失败。

3.3 使用 Transact-SQL 语句创建与管理学生信息数据库

除了使用 SQL Server Management Studio 的图形界面方式创建和管理数据库以外,还可以使用 Transact-SQL 语句创建和管理数据库,下面介绍如何使用 Transact-SQL 语句创建和管理数据库。

3.3.1 使用 Transact-SQL 语句创建学生信息数据库

用 Transact-SQL 语句创建数据库使用 CREATE DATABASE 命令来完成,创建前要确保用户具有创建数据库的权限。

语法格式:

```
CREATE DATABASE database_name
    [ON
        [PRIMARY][< filespec >[, ...n]
        [, < filegroup >[, ...n]]
        [LOG ON < filespec >[, ...n]]][;]
其中,
< filespec > ::=
    (NAME = logical_file_name,
    FILENAME = {'os_file_name'|'filestream_path'}
    [, SIZE = size[KB|MB|GB|TB]]
    [, MAXSIZE = {max_size[KB|MB|GB|TB]|UNLIMITED}]
    [, FILEGROWTH = growth_increment[KB|MB|GB|TB| % ]])
< filegroup > ::= FILEGROUP filegroup_name < filespec >[, ...n]
]
[;]
```

说明: Transact-SQL 语言的约定和说明见表 3-1。

表 3-1 Transact-SQL 语言的约定和说明

约 定	用 途
	分隔括号或大括号中的语法项,只能选其一
[]	可选语法项
<>	必选语法项
[, ... n]	前面的项可以重复 n 次,每一项由逗号分隔
[... ..n]	前面的项可以重复 n 次,每一项由空格分隔
[;]	可选的终止符
< label > ::=	语法块的名称
语法中的大写部分	Transact-SQL 语言中的关键语法

创建数据库的语法格式说明如下。

(1) database_name: 新创建数据库的名称。数据库名称在 SQL Server 的实例中必须唯一,并且必须符合标识符规则,长度不可超过 128 个字符。

(2) ON: 指定用来存储数据库的数据文件和文件组。

(3) PRIMARY: 指定主文件。

(4) LOG ON: 指定事务日志文件的明确定义。

(5) NAME = logical_file_name: 指定数据文件或日志文件的逻辑文件名。

(6) FILENAME = 'os_file_name': 指定数据文件或日志文件的物理文件名,即创建文件时由操作系统使用的路径和文件名; FILENAME = 'filestream_path': 对于 FILESTREAM 文件组,FILENAME 指向将存储 FILESTREAM 数据的路径。

(7) SIZE = size: 指定数据文件或日志文件的初始大小。

(8) MAXSIZE = max_size: 指定数据文件或日志文件可增大到的最大大小。

(9) FILEGROWTH = growth_increment: 指定数据文件或日志文件的自动增量。

(10) <filegroup>: 控制数据库文件组的属性。其中,filegroup_name 为文件组的逻辑名称。

【例 3.6】 创建一个名为 stuinfo 的学生信息数据库,其初始大小 2MB,最大大小为 200MB,允许数据库自动增长,增长方式是 5%比例增长;日志文件初始大小为 2MB,最大可增长到 50MB,按 2MB 增长。数据库文件存放位置为“C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA”。

操作步骤如下。

(1) 打开 SQL Server Management Studio,连接到 SQL Server 上的数据库引擎。

(2) 在 SSMS 窗口单击左上方的“新建查询”按钮,新建一个查询窗口,在查询分析器中输入如下 Transact-SQL 语句:

```
CREATE DATABASE stuinfo
ON
(
  NAME = 'stuinfodata',
  FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA\stuinfo.mdf',
  SIZE = 2MB,
  MAXSIZE = 200MB,
  FILEGROWTH = 5% )
LOG ON
(
  NAME = 'stuinfofog',
  FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA\stuinfofog.ldf',
  SIZE = 2MB,
  MAXSIZE = 50MB,
  FILEGROWTH = 2MB
);
```

(3) 输入完毕后,单击 SSMS 窗口上方的“执行”按钮,成功创建 stuinfo 数据库,如图 3-14 所示。

```
CREATE DATABASE stuinfo
ON
(
  NAME='stuinfodata',
  FILENAME='C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA\stuinfo.mdf',
  SIZE=2MB,
  MAXSIZE=200MB,
  FILEGROWTH=5%)
LOG ON
(
  NAME='stuinfolog',
  FILENAME='C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA\stuinfolog.ldf',
  SIZE=2MB,
  MAXSIZE=50MB,
  FILEGROWTH=2MB
);
```

100 %

消息
命令已成功完成。

图 3-14 成功创建 stuinfo 数据库

【例 3.7】 创建一个名为 test 的数据库,它有 3 个数据文件。其中 testdata1 是主文件,初始大小为 10MB,最大大小不限,按 5% 增长; testdata2 是次要数据文件,初始大小为 5MB,最大大小不限,按 10% 增长; testlog 是日志文件,初始大小为 5MB,最大大小为 100MB,按 2MB 增长。数据文件存放位置为“C:\DATA”。

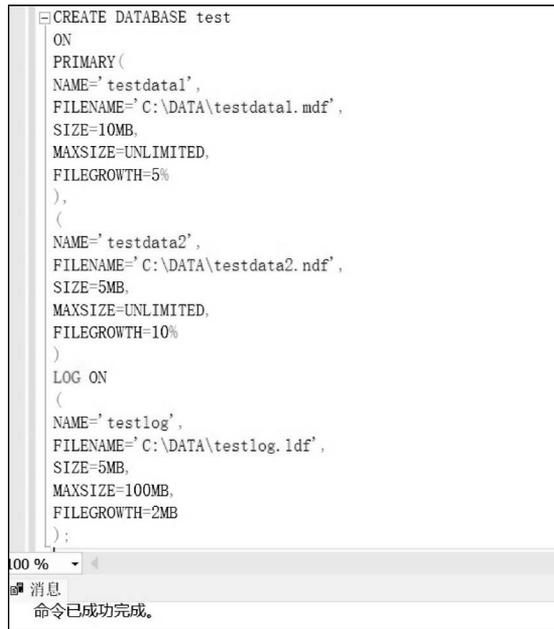
在查询分析器中输入如下 Transact-SQL 语句并执行:

```
CREATE DATABASE test
ON
PRIMARY(
  NAME = 'testdata1',
  FILENAME = 'C:\DATA\testdata1.mdf',
  SIZE = 10MB,
  MAXSIZE = UNLIMITED,
  FILEGROWTH = 5 %
),
(
  NAME = 'testdata2',
  FILENAME = 'C:\DATA\testdata2.ndf',
  SIZE = 5MB,
  MAXSIZE = UNLIMITED,
  FILEGROWTH = 10 %
)
LOG ON
(
  NAME = 'testlog',
  FILENAME = 'C:\DATA\testlog.ldf',
  SIZE = 5MB,
  MAXSIZE = 100MB,
  FILEGROWTH = 2MB
);
```



视频讲解

执行结果如图 3-15 所示。



```
CREATE DATABASE test
ON
PRIMARY (
NAME='testdata1',
FILENAME='C:\DATA\testdata1.mdf',
SIZE=10MB,
MAXSIZE=UNLIMITED,
FILEGROWTH=5%
),
(
NAME='testdata2',
FILENAME='C:\DATA\testdata2.ndf',
SIZE=5MB,
MAXSIZE=UNLIMITED,
FILEGROWTH=10%
)
LOG ON
(
NAME='testlog',
FILENAME='C:\DATA\testlog.ldf',
SIZE=5MB,
MAXSIZE=100MB,
FILEGROWTH=2MB
);
```

消息
命令已成功完成。

图 3-15 成功创建 test 数据库

【例 3.8】 创建一个具有 2 个文件组的数据库 testnew。其中，主文件组包括文件 testnewdata1，初始大小 10MB，最大大小为 100MB，按 10MB 增长；1 个文件组名为 test2group1，包括文件 testnewdata2，文件初始大小为 5MB，最大为 50MB，按 10% 增长。数据文件存放位置为“C:\DATA”。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
CREATE DATABASE testnew
ON
PRIMARY(
NAME = 'testnewdata1',
FILENAME = 'C:\DATA\testnewdata1.mdf',
SIZE = 10MB,
MAXSIZE = 100MB,
FILEGROWTH = 10MB
),
FILEGROUP test2group1
(
NAME = 'testnewdata2',
FILENAME = 'C:\DATA\testnewdata2.ndf',
SIZE = 5MB,
MAXSIZE = 50MB,
FILEGROWTH = 10 %
);
```

执行结果如图 3-16 所示。

```
CREATE DATABASE testnew
ON
PRIMARY (
NAME='testnewdata1',
FILENAME='C:\DATA\testnewdata1.mdf',
SIZE=10MB,
MAXSIZE=100MB,
FILEGROWTH=10MB
),
FILEGROUP test2group1
(
NAME='testnewdata2',
FILENAME='C:\DATA\testnewdata2.ndf',
SIZE=5MB,
MAXSIZE=50MB,
FILEGROWTH=10%
);
```

100 %

消息
命令已成功完成。

图 3-16 成功创建 testnew 数据库

3.3.2 使用 Transact-SQL 语句修改学生信息数据库

使用 ALTER DATABASE 命令可以对数据库进行修改,包括增加或删除数据文件,改变数据文件、日志文件的大小和增长方式,增加或删除日志文件,增加或删除文件组。

语法格式如下:

```
ALTER DATABASE database_name
    ADD FILE <filespec>[,...n]
| ADD LOG FILE <filespec>[,...n]
| REMOVE FILE logical_file_name
| ADD FILEGROUP filegroup_name
| REMOVE FILEGROUP filegroup_name
| MODIFY FILE <filespec>
| MODIFY NAME = new_dbname
| MODIFY FILEGROUP filegroup_name
[;]
```

语法说明如下。

- (1) database_name: 数据库名。
- (2) ADD FILE: 添加数据文件。<filespec>是给出文件的属性。
- (3) ADD LOG FILE: 添加日志文件。<filespec>是给出日志文件的属性。
- (4) REMOVE FILE: 删除数据文件。logical_file_name 是给出删除的数据文件的逻辑文件名。
- (5) ADD FILEGROUP: 添加文件组。
- (6) REMOVE FILEGROUP: 删除文件组。
- (7) MODIFY FILE: 修改数据文件的属性。
- (8) MODIFY NAME: 更改数据库名。new_dbname 是给出新的数据库名。
- (9) MODIFY FILEGROUP: 更改文件组的属性。

【例 3.9】 在例 3.7 中,已创建了 test 数据库,其中 testdata1 主文件,初始大小为 10MB,最大大小不限,按 5%增长。现修改为,其最大大小为 500MB,增长方式按 2MB 增长。

在查询分析器中输入如下 Transact-SQL 语句并执行:

```
ALTER DATABASE test
MODIFY FILE
(
NAME = testdata1,
MAXSIZE = 500MB,
FILEGROWTH = 2MB
);
```

执行结果如图 3-17 所示。

【例 3.10】 先从数据库 test 中删除 testdata2 数据文件,然后再增加次要数据文件 testdata2,要求初始大小为 1MB,最大大小 100MB,按 2%增长。

在查询分析器中输入如下 Transact-SQL 语句并执行:

```
ALTER DATABASE test
REMOVE FILE testdata2
```

在查询分析器中输入如下 Transact-SQL 语句并执行:

```
ALTER DATABASE test
ADD FILE
(
NAME = 'testdata2',
FILENAME = 'C:\DATA\testdata2.ndf',
SIZE = 1MB,
MAXSIZE = 100MB,
FILEGROWTH = 2 %
);
```

执行结果如图 3-18 所示。

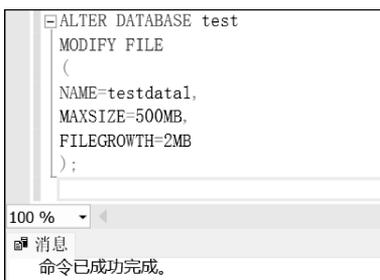


图 3-17 成功修改数据库 test 的数据文件

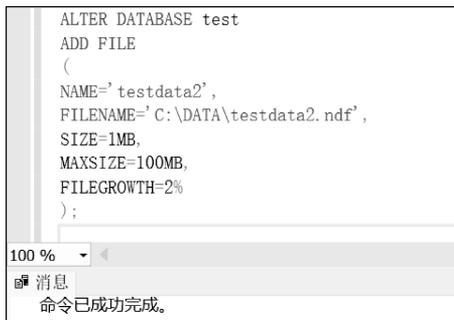


图 3-18 成功增加数据文件

3.3.3 使用 Transact-SQL 语句查看学生信息数据库信息

在 SQL Server 2022 中,可以使用存储过程来查看数据库的属性。

1. 使用 sp_helpdb 查看数据库信息

语法格式：

```
sp_helpdb [database_name][;]
```

语法说明：database_name 是指定的数据库名称，若不给出指定数据库，则显示服务器中所有数据库的信息。

【例 3.11】 查看 testnew 数据库的信息。

在查询分析器中输入如下 Transact-SQL 语句并执行：

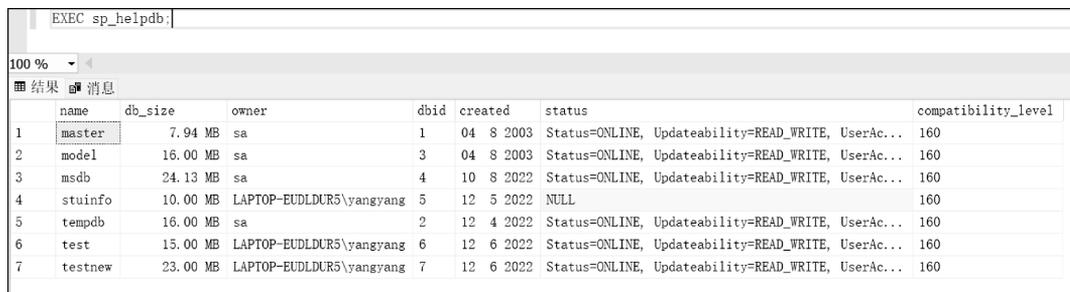
```
EXEC sp_helpdb testnew;
```

【例 3.12】 查看服务器中所有数据库的信息。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
EXEC sp_helpdb;
```

执行结果如图 3-19 所示。



	name	db_size	owner	dbid	created	status	compatibility_level
1	master	7.94 MB	sa	1	04 8 2003	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160
2	model	16.00 MB	sa	3	04 8 2003	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160
3	msdb	24.13 MB	sa	4	10 8 2022	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160
4	stuiinfo	10.00 MB	LAPTOP-EUDDLUR5\yangyang	5	12 5 2022	NULL	160
5	tempdb	16.00 MB	sa	2	12 4 2022	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160
6	test	15.00 MB	LAPTOP-EUDDLUR5\yangyang	6	12 6 2022	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160
7	testnew	23.00 MB	LAPTOP-EUDDLUR5\yangyang	7	12 6 2022	Status=ONLINE, Updateability=READ_WRITE, UserAc...	160

图 3-19 查看所有数据库的信息

2. 使用 sp_databases 查看可以使用的数据库的信息

语法格式：

```
sp_databases[;]
```

语法说明：显示所有可以使用的数据库的名称和大小。

【例 3.13】 查看有哪些数据库可以使用。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
EXEC sp_databases;
```

3. 使用 sp_helpfile 查看数据库文件信息

语法格式：

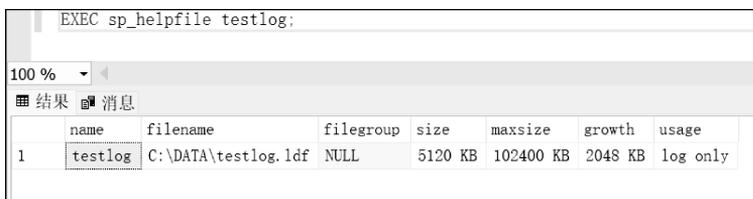
```
sp_helpfile [filename][;]
```

语法说明：显示与当前数据库关联的文件的物理名称及属性。若不指定文件名，则显示数据库的所有文件的信息。

【例 3.14】 查看 test 数据库中的日志文件的信息。
在查询分析器中输入如下 Transact-SQL 语句并执行：

```
EXEC sp_helpfile testlog;
```

执行结果如图 3-20 所示。



	name	filename	filegroup	size	maxsize	growth	usage
1	testlog	C:\DATA\testlog.ldf	NULL	5120 KB	102400 KB	2048 KB	log only

图 3-20 查看 test 数据库中的日志文件的信息

4. 使用 sp_helpfilegroup 查看文件组信息

语法格式：

```
sp_helpfilegroup [filename][;]
```

语法说明：显示与当前数据库关联的文件组的物理名称及属性。若不指定文件组名，则显示当前数据库的所有文件组的信息。

读者根据该语法格式自行练习。

3.3.4 使用 Transact-SQL 语句重命名学生信息数据库

可以使用 ALTER DATABASE 语句重命名数据库，语法格式如下。

```
ALTER DATABASE database_name  
MODIFY NAME = new_database_name[;]
```

语法格式说明如下。

- (1) database_name：要修改的数据库的名称。
- (2) new_database_name：新数据库名称。

【例 3.15】 将数据库 testnew 的名称修改为 test1。
在查询分析器中输入如下 Transact-SQL 语句并执行：

```
ALTER DATABASE testnew  
MODIFY NAME = test1;
```

执行结果如图 3-21 所示。

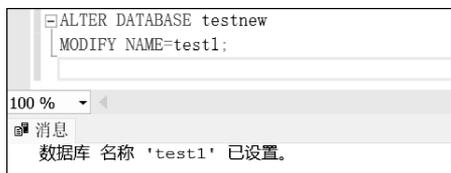


图 3-21 修改数据库名

重要提示：重命名数据库的前提是没有用户使用该数据库，并且该数据库设置为单用户模式。一般情况下，数据库创建后，不要轻易更改其名称，因为数据库名称是许多相关数据库应用程序访问和使用该数据库的基础。

3.3.5 使用 Transact-SQL 语句分离和附加学生信息数据库

1. 用 Transact-SQL 语句分离数据库

可以使用存储过程 `sp_detach_db` 实现数据库的分离。

语法格式：

```
sp_detach_db database_name[ ; ]
```

【例 3.16】 将学生信息数据库 `stuinfo` 从服务器上分离。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
EXEC sp_detach_db stuinfo;
```

在对象资源管理器中，右击“数据库”文件夹，在弹出的快捷菜单中选择“刷新”命令，此时可以看到 `stuinfo` 已被分离。

2. 用 Transact-SQL 语句附加数据库

可以使用 `CREATE DATABASE` 语句中的 `FOR ATTACH` 子句来完成数据库的附加。

语法格式：

```
CREATE DATABASE database_name  
ON (FILENAME = 'os_file_name')  
FOR ATTACH[ ; ]
```

语法说明如下。

(1) `database_name`：即将要附加的数据库的名称。

(2) `'os_file_name'`：主文件的物理文件的名称。

【例 3.17】 附加例 3.16 中分离出去的数据库 `stuinfo`。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
CREATE DATABASE stuinfo  
ON (FILENAME = 'C:\Program Files\Microsoft SQL Server\MSSQL16.SQLEXPRESS\MSSQL\DATA\stuinfo.mdf')  
FOR ATTACH;
```

在对象资源管理器中，右击“数据库”文件夹，在弹出的快捷菜单中选择“刷新”命令，此时可以看到 `stuinfo` 已被附加。

3.3.6 使用 Transact-SQL 删除学生信息数据库

删除数据库使用 `DROP DATABASE` 命令。

语法格式：

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

语法说明：database_name 是要删除的数据库的名称。

【例 3.18】 删除学生信息数据库 stuinfo。

在查询分析器中输入如下 Transact-SQL 语句并执行：

```
DROP DATABASE stuinfo;
```

重要提示：删除数据库要特别小心，因为使用 DROP DATABASE 命令不会出现确认信息。不能删除系统数据库。



单元小结

本项目介绍了 SQL Server 数据库的结构，使用 SSMS 和 Transact-SQL 语句创建和管理数据库。

SQL Server 有两种存储结构，分别是逻辑存储结构和物理存储结构。



单元实训

【实训目的】

- (1) 了解安装 SQL Server 中数据库文件的组成。
- (2) 掌握使用 SQL Server Management Studio 创建和管理数据库。
- (3) 掌握使用 Transact-SQL 语句创建和管理数据库。

【实训内容】

- (1) 使用 SSMS 创建一个图书销售管理数据库 books_sale。
- (2) 使用 SSMS 将 books_sale 数据库的主文件的逻辑名称修改为 books，存储路径修改为 C:\DATA，物理名称修改为 books.mdf，文件初始大小为 10MB，最大大小为 500MB，按 2MB 增长。将日志文件的逻辑名称修改为 books_log，存储路径修改为 C:\DATA，物理名称修改为 books_log.ldf，文件初始大小为 2MB，最大大小为 100MB，按 2% 增长。
- (3) 使用 SSMS 在 books_sale 数据库中添加次要数据文件 booksnew，存储路径为 C:\DATA，物理名称为 booksnew.ndf，其他值均取默认值。
- (4) 使用 SSMS 将 books_sale 数据库名修改为 books_salenew。
- (5) 使用 SSMS 将 books_salenew 数据库删除。
- (6) 使用 Transact-SQL 语句创建一个名为 books_sale 数据库，要求有一个主文件和一个日志文件，存储路径为 C:\DATA，其中主文件的初始大小为 5MB，最大大小为 100MB，按 5% 增长，日志文件的初始大小为 2MB，最大大小为 50MB，按 1MB 增长。
- (7) 使用 Transact-SQL 语句修改 books_sale 数据库，添加数据文件 books_data1.ndf，初始大小为 5MB，添加一个名为 fgroup 的文件组。
- (8) 使用 Transact-SQL 语句查看 books_sale 数据库中所有文件的信息，查看该数据库中文件组的信息。
- (9) 使用 Transact-SQL 语句修改 books_sale 数据库名为 books_salenew。
- (10) 使用 Transact-SQL 语句和 SSMS 分离 books_salenew 数据库，再附加至服务器。
- (11) 使用 Transact-SQL 将 books_salenew 数据库删除。