

第3章

SQL Server 2019 基础

SQL Server 是 Microsoft 公司推出的典型的关系数据库管理系统,它与 Oracle、MySQL 一起,被称为数据库三巨头。SQL Server 2019 是 2019 年 11 月推出的最新版本,其具有业内领先的性能,为用户的企业数据管理和商业智能应用提供了一个高效、安全、可靠的平台。SQL Server 2019 具有使用方便、伸缩性好、相关软件集成程度高等优点,结合了分析、报表、集成和通告功能,并为结构化数据提供了安全、可靠的存储功能,使用户可以构建和管理用于数据处理的高性能的应用程序。

3.1 SQL Server 2019 简介

3.1.1 SQL Server 的发展历史

SQL Server 是一个典型的关系数据库管理系统,最初由 Microsoft、Sybase 和 Ashton-Tate 三家公司共同开发,于 1988 年推出了 SQL Server 1.0 版本。1992 年,基于 OS/2 操作系统的 SQL Server 4.2 beta 版发布。1993 年,Microsoft 公司推出 Windows NT 操作系统并将数据库产品移植到 Windows NT 上,此后三家公司基本已经分道扬镳。目前,SQL Server 主要指由 Microsoft 公司推出的一系列 SQL Server 版本。表 3.1 分别对这些版本进行了简要介绍。

表 3.1 SQL Server 版本简介

年份	版本	说明
1995	SQL Server 6.0	这是第一个完全由 Microsoft 公司开发的版本。对核心数据库引擎做了重大的改写。性能得以提升,重要的特性得到增强。具备了处理小型电子商务和内联网应用程序的能力,而在花费上却少于其他的同类产品
1996	SQL Server 6.5	该版本满足众多小型商业数据管理的应用需求,也曾风靡一时,与 Oracle 推出的运行于 Windows NT 平台上的 7.1 版本形成直接的竞争。但是,由于受到以前版本在结构上的限制,SQL Server 6.5 在应用中逐步暴露出它的一些缺点
1998	SQL Server 7.0	该版本再一次对核心数据库引擎进行了重大改写,在数据存储和数据库引擎方面发生了根本性的变化,为中小型企业提供了切实可行且廉价的可选方案。该版本易于使用,并提供了对于其他竞争数据库来说需要额外附加的昂贵的重要商业工具(例如分析服务、数据转换服务),因此获得了良好的声誉

续表

年份	版本	说 明
2000	SQL Server 2000	该版本具有更好的可用性和可伸缩性,与相关软件集成程度高,提供了企业级的数据库功能,易于安装和部署。它既可以在 Windows 98 的小型计算机上运行,也支持在 Windows 2000 大型多处理器的服务器等多种平台上使用
2005	SQL Server 2005	该版本对 SQL Server 的许多地方进行了改写。一方面为关系型数据和结构化数据提供了更安全、更可靠的存储功能和更灵活的数据管理功能;另一方面可以有效地执行大规模联机事务处理,可以完成数据仓库和电子商务应用,可以构建和部署经济有效的商业智能解决方案
2008	SQL Server 2008	SQL Server 2008 以处理目前能够采用的许多种不同的数据形式为目的,通过提供新的数据类型和使用语言集成查询(LINQ)。它提供了在一个框架中设置约束的能力,以确保数据库和对象符合定义的标准。并且,当这些对象不符合该标准时,还能够就此进行报告。它是一个全面的数据智能平台
2012	SQL Server 2012	该版本能够顺应云技术发展的需要,全面支持云技术,能够快速实现私有云与公有云之间数据的扩展与应用的迁移,可用于大型联机事务处理、数据仓库和电子商务等方面的数据库平台,为数据存储、数据分析提供基于云技术的解决方案,是一种全新的数据分析处理平台
2014	SQL Server 2014	与其他版本相比,SQL Server 2014 提供了驾驭海量数据的关键技术——in memory 增强技术。该技术能够整合云端的各种数据结构,极大地增强了对云的支持,提供了全新的混合云解决方案,可以实现云备份和灾难恢复,大幅提升数据处理的效率,能够快速处理数以百万条的记录。可以说,SQL Server 2014 为大数据分析提供了一种有效的解决方案
2016	SQL Server 2016	该版本是 Microsoft 数据平台历史上最大的一次跨越性发展,它除了兼容 SQL Server 2014 版本功能以外,还增强了安全性、可用性和灾难恢复功能,是性能最高的数据仓库,提供实时运营分析、大数据简化等功能,再次简化了数据库分析方式
2017	SQL Server 2017	该版本同时面向 Windows、Linux、macOS 以及 Docker 容器,用户可以在 SQL Server 平台上选择开发语言、数据类型、本地开发或云端开发以及操作系统开发等,引入了图数据处理、适应性查询、面向高级分析的 R/Python 集成等功能
2019	SQL Server 2019	该版本附带 Apache Spark 和 Hadoop Distributed File System (HDFS),可实现跨关系、非关系、结构化和非结构化数据进行查询,从而实现所有数据的智能化;通过开源支持,可以灵活选择语言 and 平台;在支持 Kubernetes 的 Linux 容器上或在 Windows 上运行 SQL Server;利用突破性的可扩展性和性能,改善数据库的稳定性并缩短响应时间,而无须更改应用程序;让任务关键型应用程序、数据仓库和数据湖实现高可用性;使用 SQL Server Reporting Services 的企业报告功能在数据中找到问题的答案,并通过随附的 Power BI 报表服务器,使用户可以在任何设备上访问丰富的交互式 Power BI 报表

3.1.2 SQL Server 2019 的版本

根据不同的用户类型和使用需求,Microsoft 公司为 SQL Server 2019 推出了多种不同的版本。用户可以根据自己的实际需求、软/硬件环境、价格水平等来选择安装适当的版本和组件。

1. 企业版(Enterprise)

企业版是功能最强大、最全面的 SQL Server 版本。SQL Server 2019 企业版提供了全面的高端数据中心功能,具有快速的性能、无限的虚拟化和端到端商务智能,为关键任务工作负载和最终用户访问数据提供了高水平的服务。

2. 标准版(Standard)

SQL Server 2019 标准版为部门和小型组织运行其应用程序提供基本的数据管理和商务智能数据库,并支持将常用开发工具用于本地和云部署,以最少的 IT 资源实现有效的数据库管理。

3. Web 版(Web)

对于为从小规模至大规模 Web 资产提供可伸缩性、经济性和可管理性功能的 Web 宿主和 Web VAP 来说,SQL Server Web 版本是一项总拥有成本较低的选择。

4. 开发者版(Developer)

SQL Server 2019 开发者版允许开发人员在 SQL Server 上构建任何类型的应用程序。它包含企业版的所有功能,但有许可限制,只被授权用作开发和测试系统,而不能用作生产服务器。SQL Server Developer 是构建和测试应用程序的人员的理想选择。

5. 精简版(Express)

SQL Server 2019 精简版是入门级的免费数据库,非常适合学习和构建桌面和小型服务器数据驱动应用程序。它是独立软件供应商、开发人员和构建客户端应用程序爱好者的最佳选择。如果需要更高级的数据库特性,可以将 SQL Server Express 无缝地升级到其他更高端的 SQL Server 版本。

3.1.3 SQL Server 2019 的数据库组成

SQL Server 2019 的数据库包括两类:系统数据库和用户数据库。系统数据库是安装后系统自动建立的数据库,存放系统的核心信息,SQL Server 2019 使用这些信息来管理和控制整个数据库服务器系统。它由系统管理,用户只能查看其内容,但不可以进行任何破坏性操作(增加、删除、修改),否则可能导致系统崩溃,需要重新安装。用户数据库是由使用者逐步创建起来的,系统安装之初,没有任何用户数据库,但在一个数据库服务器中,用户可以创建多个数据库。

SQL Server 2019 的系统数据库有下面 4 个。

(1) master: 最重要的系统数据库,记录 SQL Server 系统的所有系统级信息,包括登录账号、角色、权限设置、链接服务器和系统配置信息等。master 还记录了所有其他数据库的关键信息、数据库文件的位置以及 SQL Server 的初始化信息。

(2) model: 是一个模板数据库,存储了可以作为模板的数据库对象和数据,用户在创建数据库时自动调用此数据库中的相关信息。

(3) msdb: 与 SQL Server Agent 代理服务有关的数据库,主要完成定时、预处理等操作,记录有关作业、警报、操作员、调度等信息。

(4) tempdb: 是一个临时数据库。用于存储查询过程中所使用的中间数据或结果。

系统数据库的组成与用户数据库基本相同,包括表、视图、同义词、可编程性、Service Broker、存储、安全性。

3.2 SQL Server 2019 的组件和管理工具

3.2.1 SQL Server 2019 的组件

1. SQL Server 数据库引擎

数据库引擎是 SQL Server 2019 的核心组件,其基本功能是用于存储、处理数据和保证数据安全。数据库引擎提供受控的访问和快速事务处理,以满足企业中要求极高、大量使用数据的应用程序的要求。SQL Server 支持在同一台计算机上最多存在 50 个数据库引擎实例。

2. 大数据群集

从 SQL Server 2019 (15. x) 开始,借助 SQL Server 大数据群集,可部署在 Kubernetes 上运行的 SQL Server、Spark 和 HDFS 容器的可缩放群集。这些组件并行运行以确保可读取、写入和处理 Transact-SQL 或 Spark 中的大数据,这样就可以借助大量大数据轻松合并和分析高价值关系数据。使用 SQL Server 大数据群集可灵活处理大数据,可查询外部数据源,存储通过 SQL Server 管理的 HDFS 中的大数据,或通过群集查询来自多个外部数据源的数据。然后,可以将数据用于人工智能、机器学习和其他分析任务。

3. 机器学习服务

机器学习服务是 SQL Server 中一项支持使用关系数据运行 Python 和 R 脚本的功能。可以使用开源包和框架,以及 Microsoft Python 包和 R 包进行预测分析和机器学习。使用 SQL Server 机器学习服务,用户可以在数据库中执行 Python 和 R 脚本,还可以使用它来准备和清理数据、执行特征工程以及在数据库中定型、评估和部署机器学习模型。此功能在数据所在的位置运行脚本,无须通过网络将数据传输到其他服务器。

4. 分析服务

分析服务(Analysis Services)是在决策支持和业务分析中使用的分析数据引擎(Vertipaq)。

它为商业智能(BI)数据分析和报告应用程序(如 Power BI、Excel、Reporting Services 报表和其他数据可视化工具)提供企业级语义数据模型功能。SQL Server Analysis Services 安装为本地服务器实例,SQL Server Analysis Services 支持所有兼容级别(取决于版本)、多维模型、数据挖掘和 SharePoint Power Pivot 的表格模型。

5. 报表服务

报表服务(SQL Server Reporting Services,SSRS)提供了一系列本地工具和服务,用于创建、部署和管理移动和分页报表。SSRS 解决方案灵活地将正确信息提供给正确用户。用户可以通过 Web 浏览器、移动设备或电子邮件使用报表。

6. 主数据服务

主数据服务(Master Data Services,MDS)是 SQL Server 2008 R2 开始增加的关键商业智能特性之一。Master Data Services 帮助管理组织的主数据集,可以将数据整理到模型中,创建更新数据的规则,并控制由谁更新数据。通过使用 Excel 以和组织中的其他用户共享主数据集。

7. 集成服务

集成服务(Integration Services)是用于生成企业级数据集成和数据转换解决方案的平台。使用 Integration Services 可解决复杂的业务问题,具体表现为:复制或下载文件、加载数据仓库、清除和挖掘数据以及管理 SQL Server 对象和数据。Integration Services 可以提取和转换来自多种源(如 XML 数据文件、平面文件和关系数据源)的数据,然后将这些数据加载到一个或多个目标。Integration Services 包括一组丰富的内置任务和转换,用于生成包的图形工具和可在其中存储、运行和管理包的 Integration Services 目录数据库。可以使用图形 Integration Services 工具来创建解决方案,而无须编写单行代码。也可以编写广泛的 Integration Services 对象模型,以编程方式创建包,并对自定义任务和其他包对象进行编码。

3.2.2 SQL Server 2019 的管理工具

1. SQL Server Management Studio

SQL Server Management Studio (SSMS) 是自 SQL Server 2005 版本开始增加的组件,是对 SQL Server 2000 查询分析器、企业管理器和分析管理器等工具的集成和扩充,是用于管理任何 SQL 基础结构的集成环境。使用 SSMS,用户可以访问、配置、管理和开发 SQL Server、Azure SQL 数据库和 SQL 数据仓库的所有组件。SSMS 在一个综合实用工具中汇集了大量图形工具和丰富的脚本编辑器,为各种技能水平的开发者和数据库管理员提供对 SQL Server 的访问权限。SSMS 的登录界面和登录后首页如图 3.1 和图 3.2 所示。

2. SQL Server 配置管理器

SQL Server 配置管理器主要用于为 SQL Server 服务、服务器协议、客户端协议和客户端别名提供基本配置管理,如图 3.3 所示。

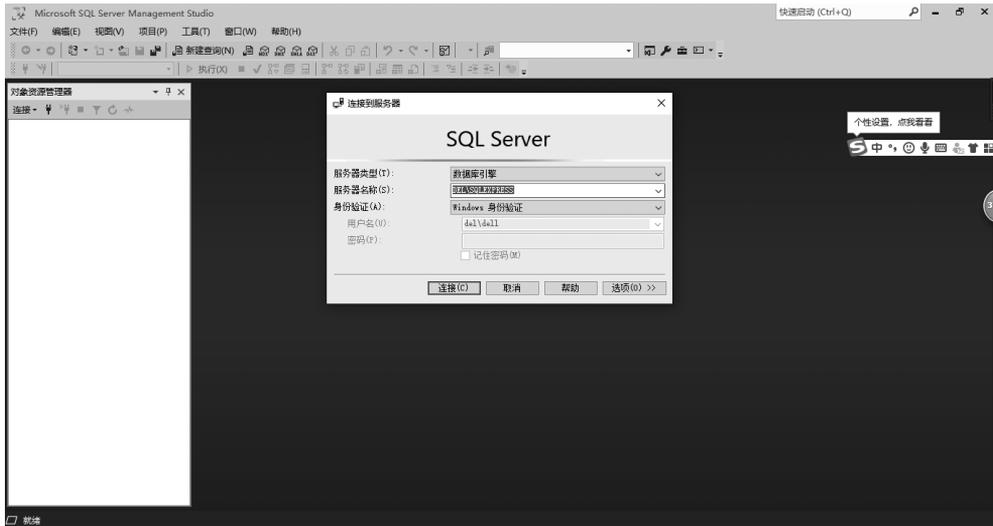


图 3.1 SSMS 登录界面

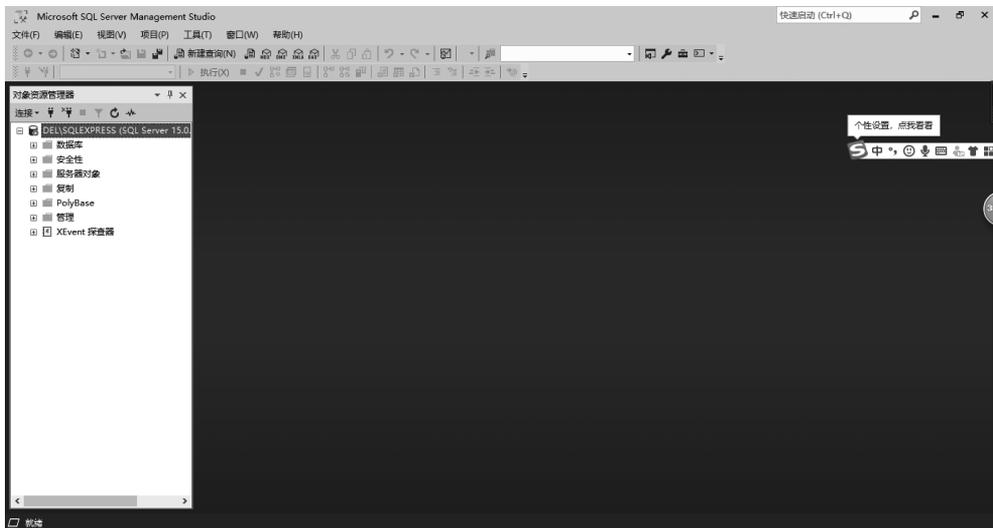


图 3.2 SSMS 登录后首页



图 3.3 SQL Server 配置管理器

3. SQL Server Profiler

SQL Server Profiler 提供了一种图形用户界面,用于创建和管理跟踪并分析和重播跟踪结果。这些事件保存在一个跟踪文件中,稍后诊断问题时,可以对该文件进行分析,或用它来重播一系列特定的步骤。SQL Server Profiler 启动后的界面如图 3.4 所示。



图 3.4 SQL Server Profiler 启动后的界面

4. 数据库引擎优化顾问

数据库引擎优化顾问可以分析工作负荷,并可为该工作负荷推荐可改进服务器性能的物理设计结构。工作负荷可以是计划缓存、SQL Server Profiler 跟踪文件或跟踪表,也可以是 Transact-SQL 脚本。物理设计结构包括索引、索引视图和分区,但其不支持 SQL Server Express。数据库引擎优化顾问启动后的界面如图 3.5 所示。

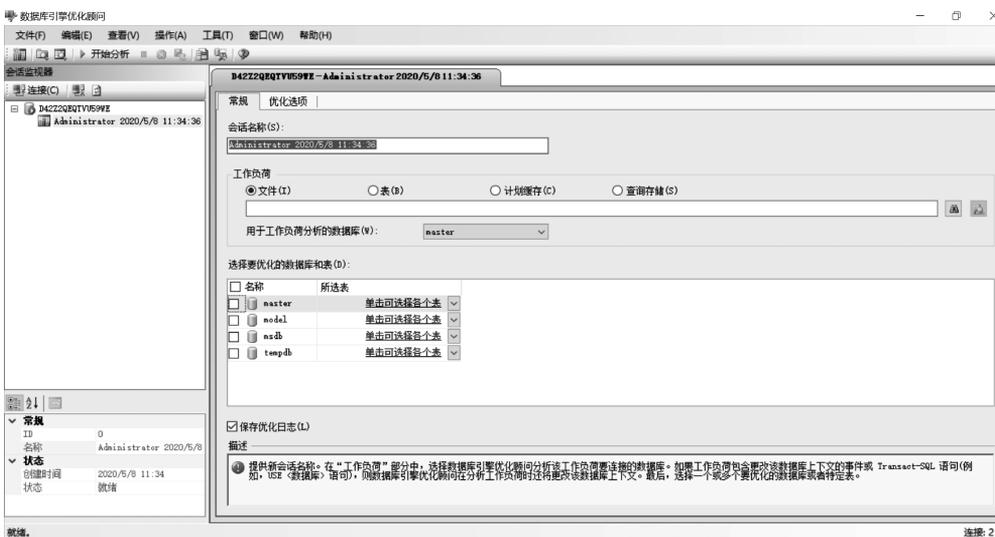


图 3.5 数据库引擎优化顾问启动后的界面

5. SQL Server Data Tools

SQL Server Data Tools(SSDT)是一款新式开发工具,用于生成 SQL Server 关系数据库、Azure SQL 数据库、Analysis Services (AS) 数据模型、Integration Services (IS) 包和 Reporting Services (RS) 报表。SSDT 通过引入跨 Visual Studio 内所有数据库开发阶段的无所不在的声明性模型,为数据库开发带来变革。可以使用 SSDT Transact-SQL 设计功能来生成、调试、维护和重构数据库。用户可以使用数据库项目,或者在内部或外部直接使用所连接的数据库实例。

6. Azure Data Studio

Azure Data Studio 是跨平台的数据库工具,适合在 Windows、macOS 和 Linux 上使用 Microsoft 系列的本地和云数据平台的数据专业人员。

Azure Data Studio 利用内置功能(如多个选项卡窗口、丰富的 SQL 编辑器、IntelliSense、关键字完成、代码片段、代码导航和源代码管理集成(Git))提供一种基于键盘的新式 SQL 编码体验,使日常任务变得更轻松。按需运行 SQL 查询,查看结果并将其保存为文本、JSON 或 Excel 格式。读者可以编辑数据,组织最喜欢的数据库连接,并以熟悉的对象浏览体验浏览数据库对象。

3.3 SQL Server 2019 安装

3.3.1 SQL Server 2019 的安装环境

SQL Server 2019 的安装环境就是安装 SQL Server 2019 对硬件和软件的最低要求,在安装之前首先要了解其对安装环境的要求,下面分别介绍其对硬件和软件的要求。

首先需要注意的是,仅 x64 处理器支持 SQL Server 2019 的安装。x86 处理器不再支持此安装。

1. 硬件要求

硬件配置的高低会直接影响软件运行速度,实际安装的环境都要比最低要求高一些,安装 SQL Server 2019 对硬件的要求如下。

(1) 处理器。x64 处理器,最低要求 1.4GHz,推荐 2.0GHz 或更快。处理器类型:AMD Opteron、AMD Athlon 64、支持 Intel EM64T 的 Intel Xeon,以及支持 EM64T 的 Intel Pentium IV。

(2) 内存。最低要求 Express Edition: 512MB,所有其他版本: 1GB。推荐 Express Edition: 1GB。其他版本: 至少 4GB,并且应随着数据库大小的增加而增加来确保最佳性能。

(3) 硬盘。SQL Server 2019 要求最少 6GB 的可用硬盘空间。磁盘空间要求将随所安装的 SQL Server 2019 组件不同而发生变化。

(4) 监视。SQL Server 2019 要求有 Super-VGA (800×600) 或更高分辨率的显示器。

(5) Internet。使用 Internet 功能需要连接因特网。

2. 软件要求

(1) 操作系统。Windows 10 TH1 1507 或更高版本, Windows Server 2016 或更高版本。

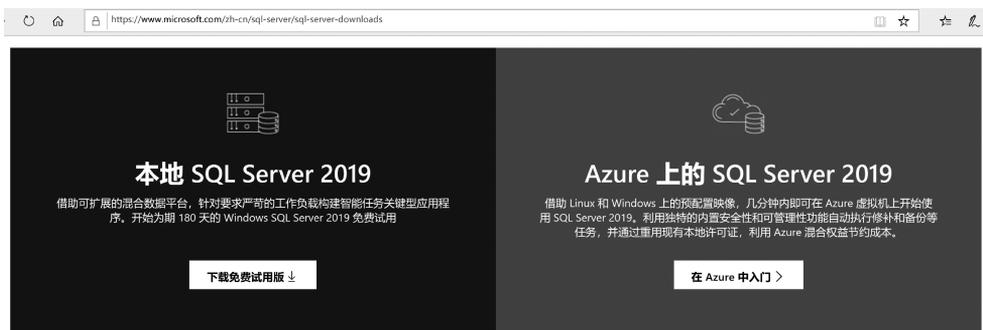
(2) .NET Framework。最低版本操作系统包括最低版本 .NET 框架。

(3) 网络软件。SQL Server 2019 支持的操作系统具有内置网络软件。独立安装项的命名实例和默认实例支持以下网络协议: 共享内存、命名管道和 TCP/IP。

3.3.2 SQL Server 2019 Express 安装

下面介绍 SQL Server 2019 的安装过程。本书使用的操作系统是 Windows 10, 安装的版本为 Express 版。

(1) 下载安装文件。从 Microsoft 官网 <https://www.microsoft.com/zh-cn/sql-server/sql-server-downloads> 下载 SQL Server 2019, 如图 3.6 所示, 可以看到有试用版, 但使用时间受到限制, 只允许免费试用 180 天。还有 Developer 和 Express 两个免费的专用版本可以使用, 这里选择 Express 版。单击“立即下载”按钮, 开始下载文件。



还可以下载免费的专用版本



图 3.6 SQL Server 2019 下载页面

(2) 开始安装。下载完成后, 找到安装文件, 如图 3.7 所示, 双击文件开始安装。



图 3.7 SQL Server 2019 安装文件

(3) 选择安装类型。开始安装后,首先进入选择安装类型界面,如图 3.8 所示,有基本和自定义两种安装类型可以选择。基本安装会按默认配置进行安装;自定义安装则需要根据安装向导,选择要完成的安装内容。若选择基本安装,则进入从步骤(4)开始的基本安装步骤;若选择自定义安装,则进入从步骤(8)开始的自定义安装步骤。



图 3.8 SQL Server 2019 安装——选择安装类型

基本安装

(4) 选择语言和许可条款。选择基本安装后,进入图 3.9 所示的选择语言和许可条款界面,选择语言、了解软件许可条款后,单击“接受”按钮,进入指定安装位置界面。

(5) 指定安装位置。图 3.10 所示界面给出了默认的安装位置,可以通过单击“浏览”按钮指定其他的安装位置,之后单击“安装”按钮。

(6) 下载安装程序包开始安装。如图 3.11 界面所示,先下载安装程序包,然后进行安装,此过程时间稍长。

(7) 完成安装。成功安装后的界面如图 3.12 所示。单击“安装 SSMS”按钮,进入 SSMS 安装步骤。

自定义安装

(8) 选择语言和位置。如图 3.13 所示,选择完语言和位置后,单击“安装”按钮,开始下载安装程序包,如图 3.14 和图 3.15 所示,下载成功后打开 SQL Server 安装中心。

(9) SQL Server 安装中心。如图 3.16 所示,在 SQL Server 安装中心可以选择要安装的软件,这里选择第一个“全新 SQL Server 独立安装或向现有安装添加功能”。

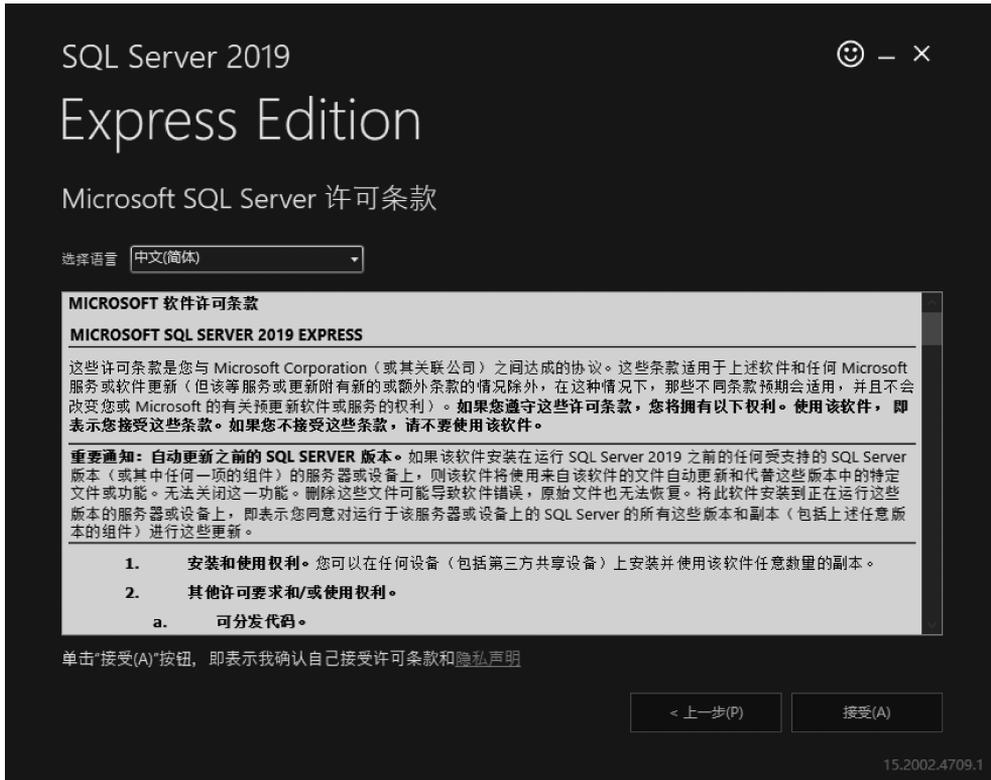


图 3.9 选择语言和许可条款



图 3.10 指定安装位置

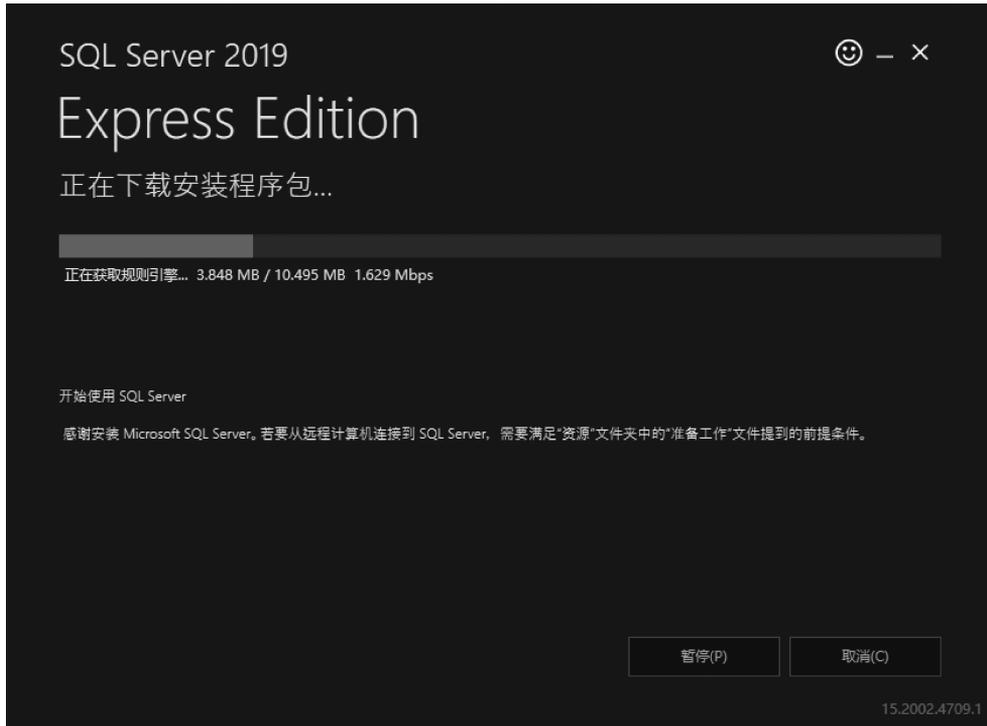


图 3.11 下载安装程序包

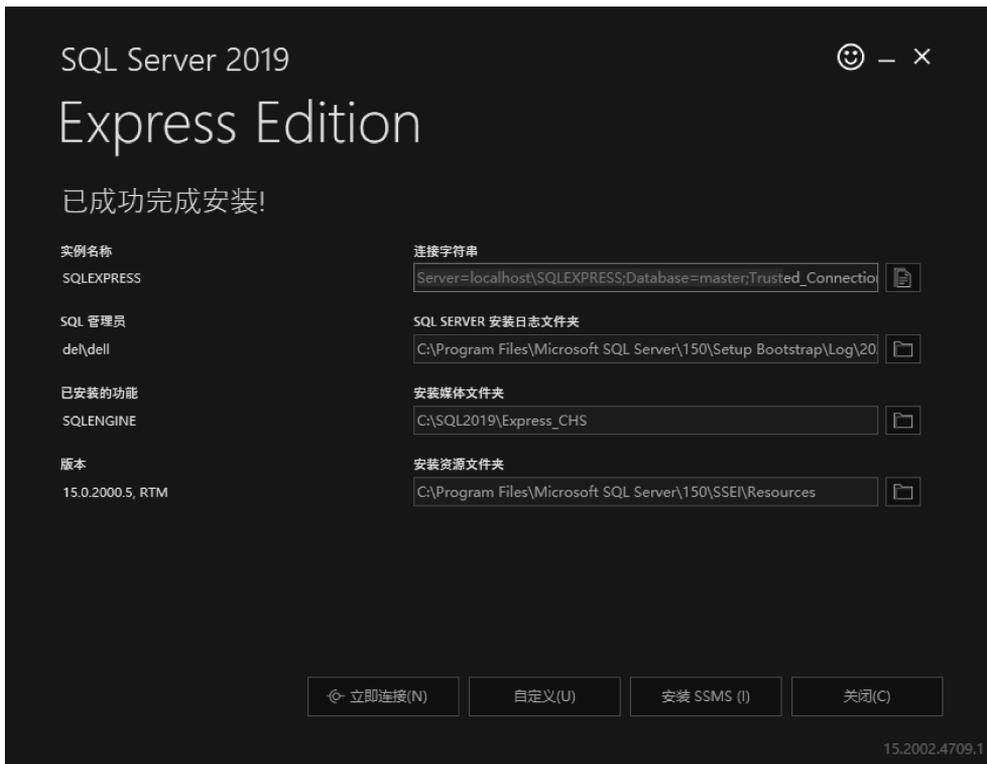


图 3.12 完成安装

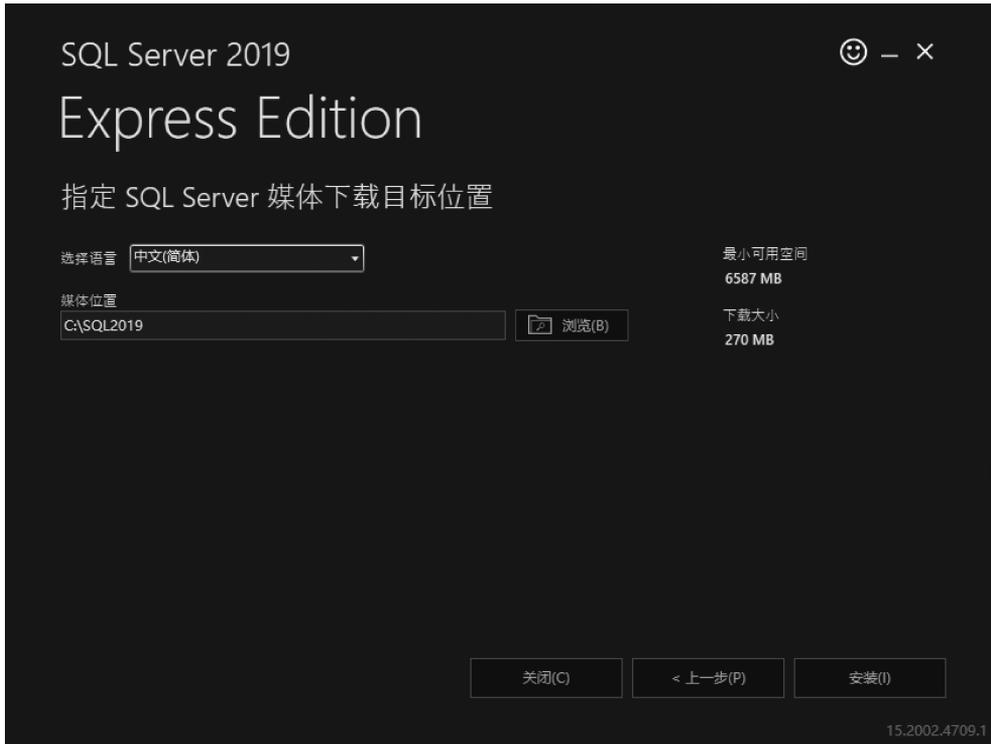


图 3.13 选择语言和位置

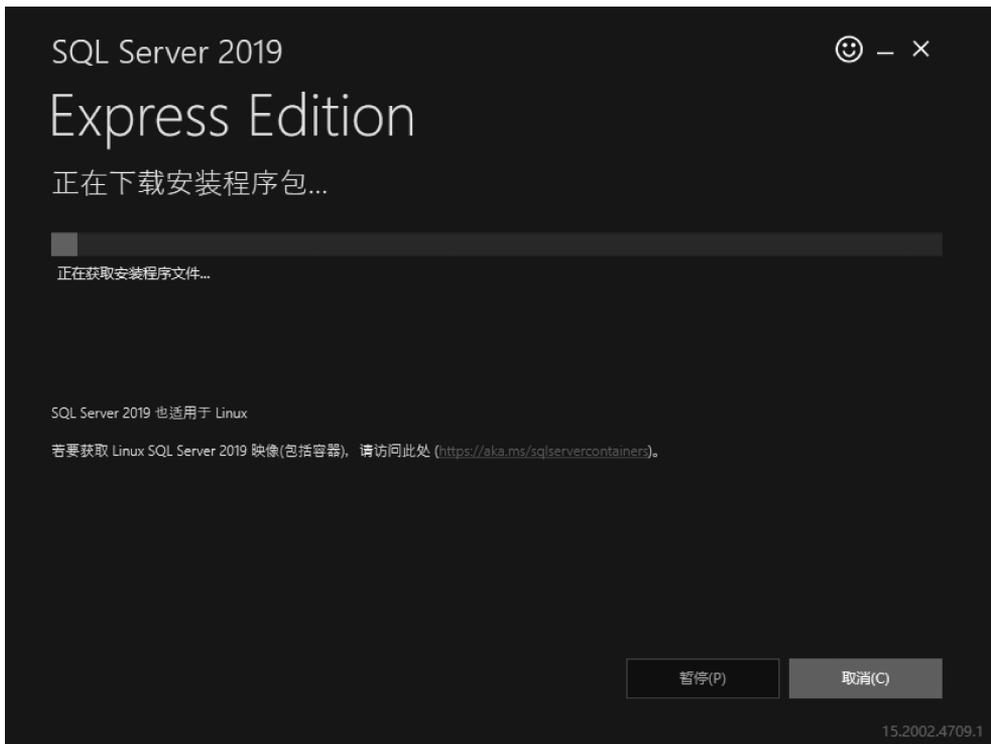


图 3.14 下载安装程序包

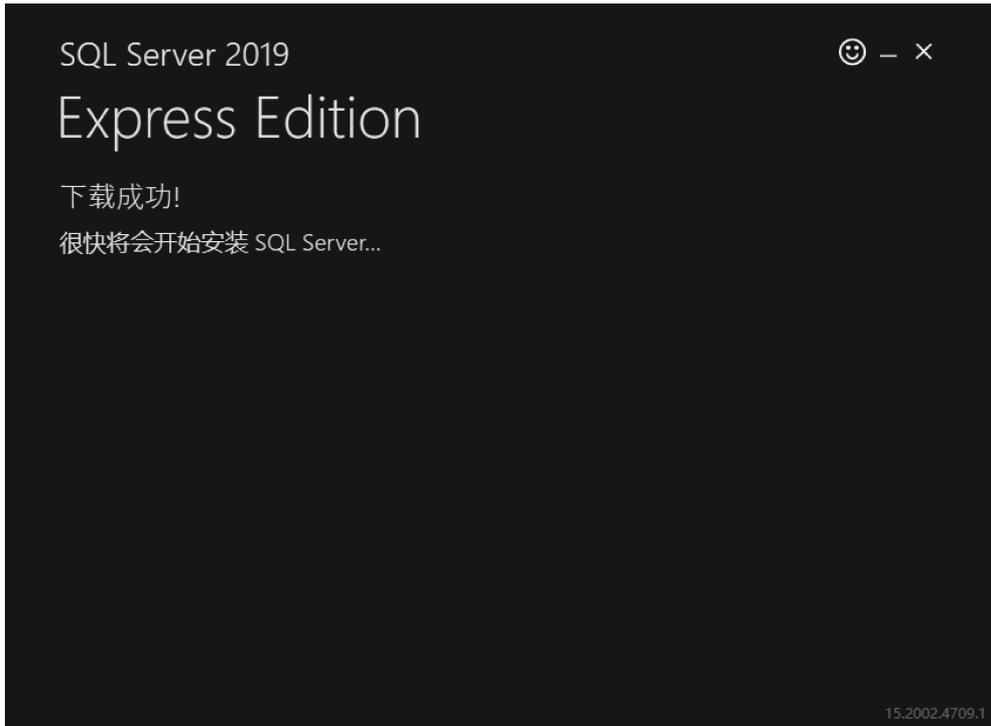


图 3.15 安装包下载成功



图 3.16 “SQL Server 安装中心”界面

(10) 许可条款。开始安装后首先进入许可条款界面,如图 3.17 所示,勾选“我接受许可条款”复选框,然后单击“下一步”按钮,开始安装程序全局规则,其可确定在 SQL Server 安装程序支持文件时可能发生的问题,必须更正所有失败,安装程序才能继续。全部通过则直接进入下一界面——Microsoft 更新。

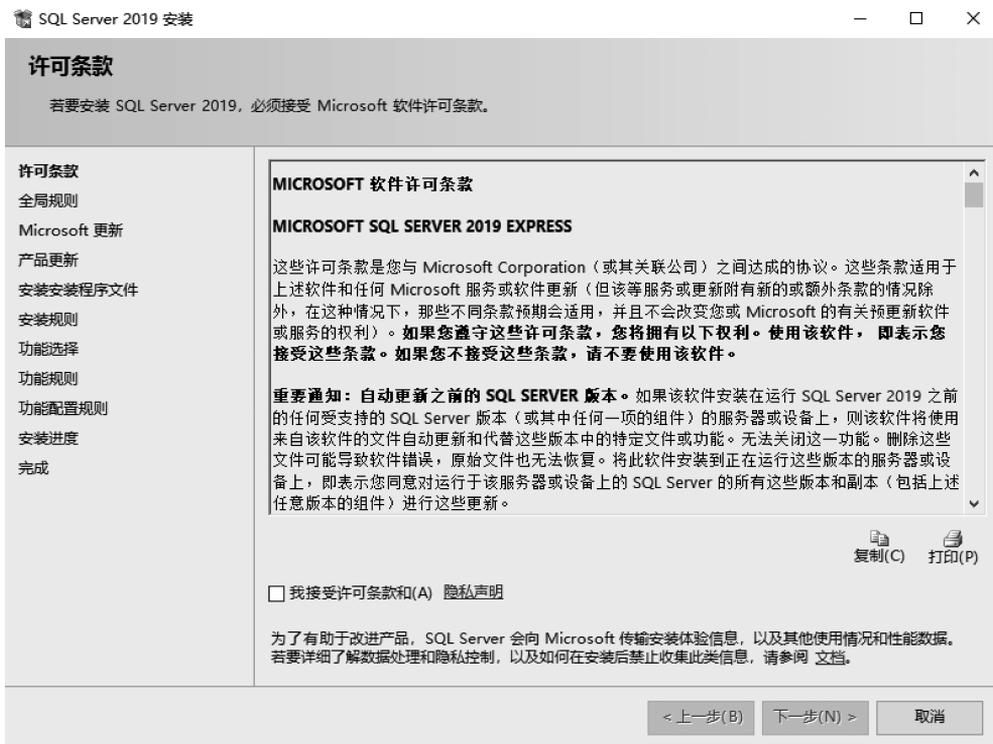


图 3.17 “许可条款”界面

(11) Microsoft 更新。如图 3.18 所示,可以选择勾选“使用 Microsoft 更新检查更新”复选框或者不选,单击“下一步”按钮,进入安装规则界面。

(12) 安装规则。安装规则标识在运行安装程序时可能发生的问题,必须更正所有失败,安装程序才能继续。全部通过的界面如图 3.19 所示,单击“下一步”按钮进入功能选择界面。

(13) 功能选择。如图 3.20 所示,选择自己所需的功能,不建议全选,很多功能暂时用不上。数据库引擎服务和 SQL 复制是必选项,实例根目录也可以选择修改。单击“下一步”按钮,进入服务器实例配置界面。

(14) 实例配置。实例配置可以选择默认实例(MSSQLSERVER),也可以选择命名实例(SQLEXPRESS),如图 3.21 所示,这里选择命名实例,然后单击“下一步”按钮,进入服务器配置界面。

(15) 服务器配置。如图 3.22 所示,直接单击“下一步”按钮,进入数据库引擎配置界面。

(16) 数据库引擎配置。如图 3.23 所示,选择身份验证模式,前面介绍的基本安装中采用的就是 Windows 身份验证模式,此处选择混合模式,为 sa 账户设置密码,然后单击“下一步”按钮,开始安装。



图 3.18 “Microsoft 更新”界面



图 3.19 “安装规则”界面



图 3.20 “功能选择”界面



图 3.21 “实例配置”界面

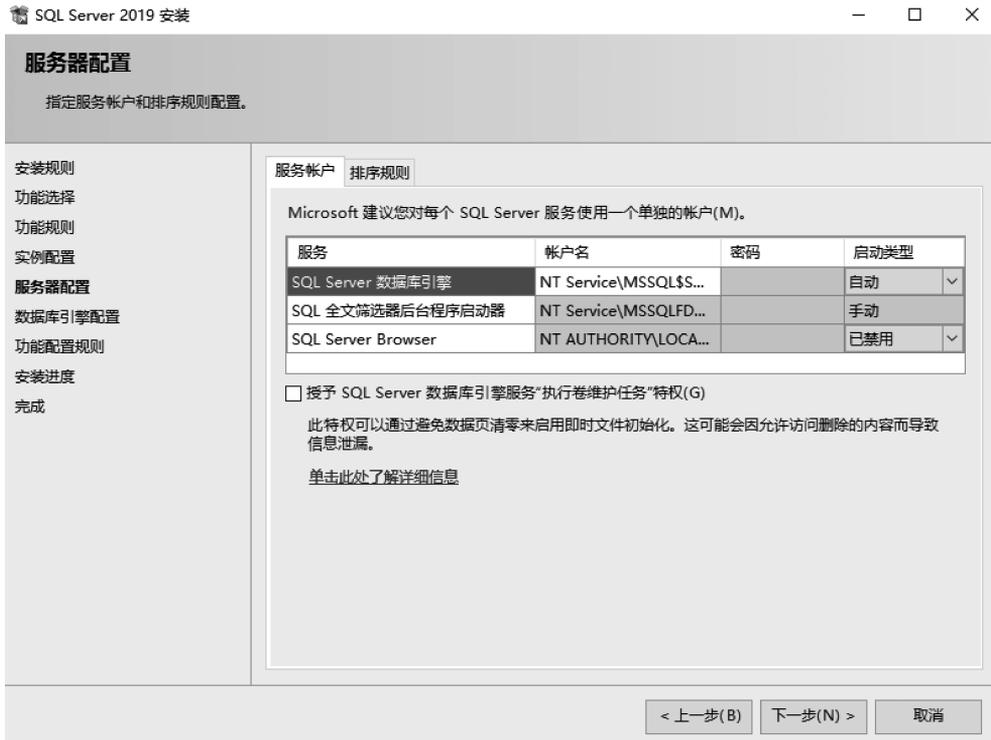


图 3.22 “服务器配置”界面

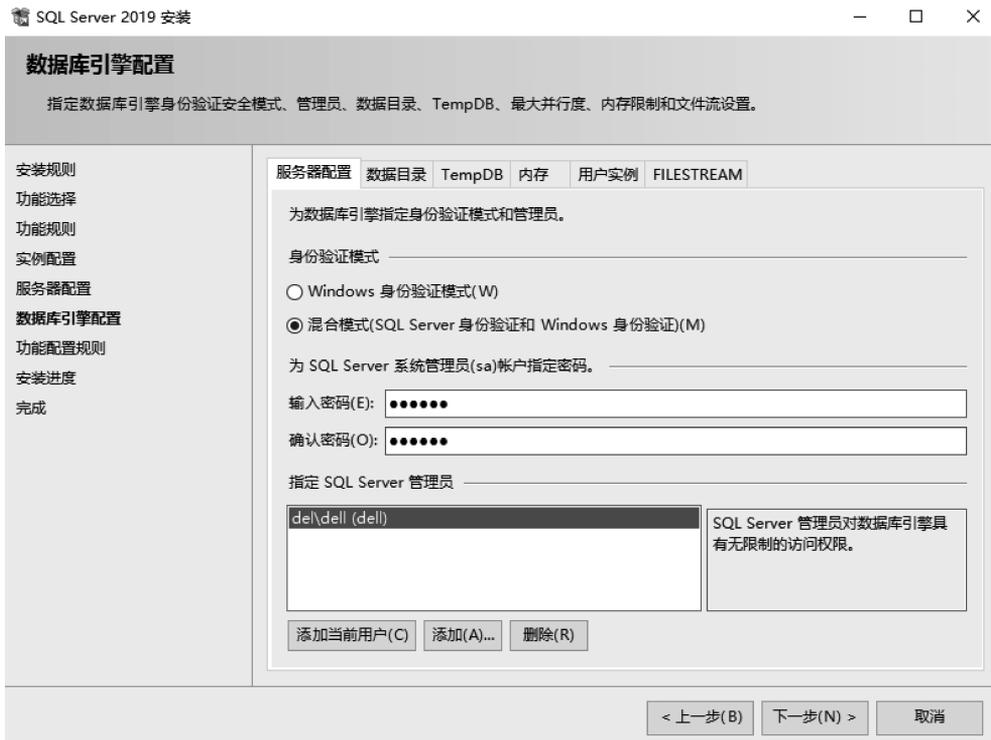


图 3.23 “数据库引擎配置”界面

(17) 安装进度。安装界面如图 3.24 所示。这一过程所需时间相对较长,请耐心等待。

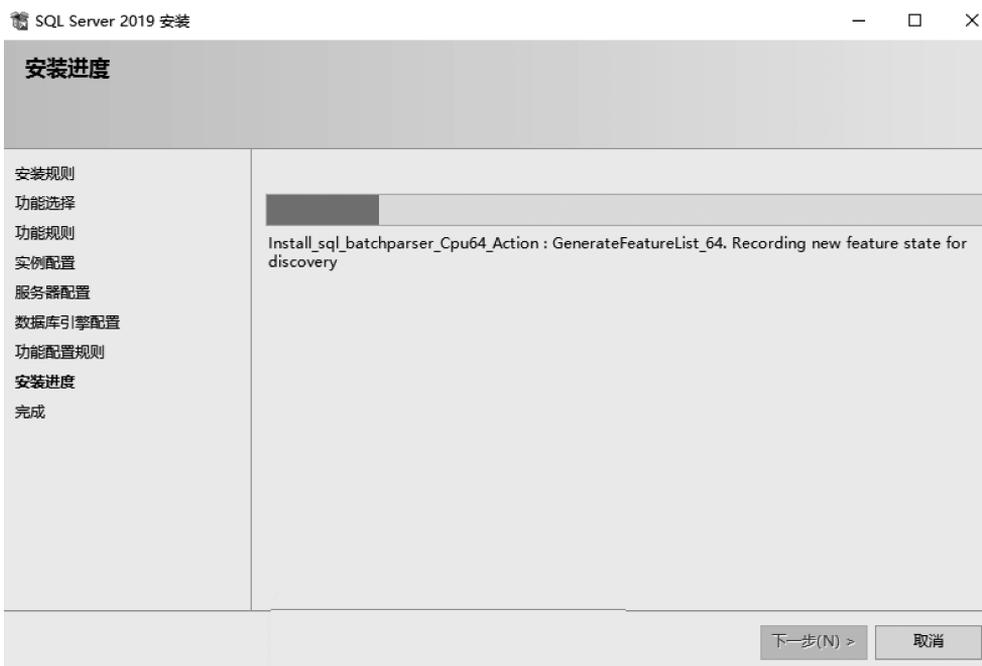


图 3.24 “安装进度”界面

(18) 安装完成。安装成功后界面如图 3.25 所示。



图 3.25 安装进度(已完成)

3.3.3 SQL Server Management Studio(SSMS)的安装

(1) 进入 SSMS 下载页面,如图 3.26 所示,下载 SSMS。

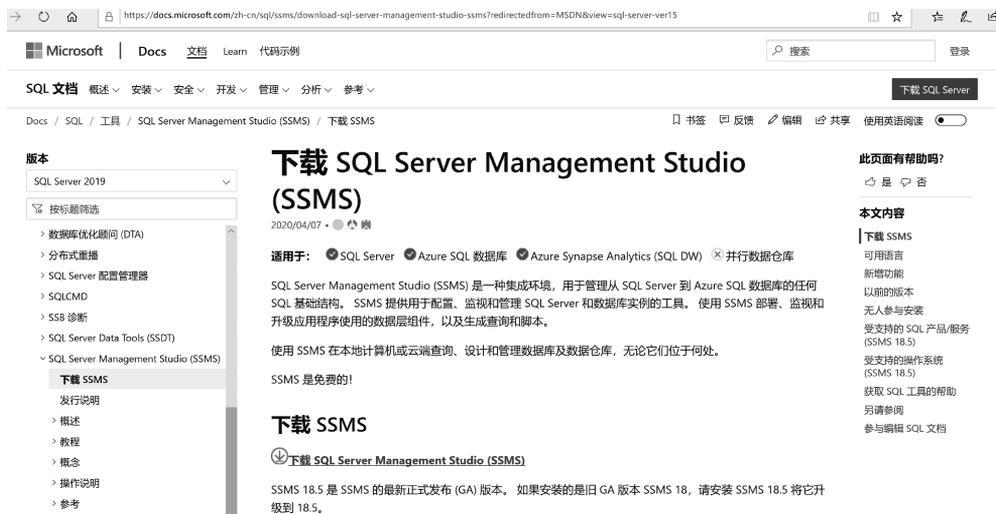


图 3.26 SSMS 下载页面

(2) 下载后,找到 SSMS 安装文件,如图 3.27 所示。



图 3.27 SSMS 安装文件

(3) 双击图 3.27 所示的文件开始 SSMS 安装,如图 3.28 所示。



图 3.28 SSMS 安装——首页

(4) 选择安装位置后,单击“安装”按钮,进入图 3.29 所示界面。



图 3.29 SSMS 安装

(5) 安装完成后的界面如图 3.30 所示。

安装完成后,单击计算机左下角的“开始”可以看到“最近添加”,如图 3.31 所示。



图 3.30 SSMS 安装完成



图 3.31 安装完成后添加的应用

3.4 SQL 和 Transact-SQL 简介

3.4.1 SQL 概述

SQL(Structured Query Language,结构化查询语言)是一种最常用的关系数据库语言,SQL的核心部分和关系代数是等价的,但它还有一些重要的功能已经超越了关系代数的表达能力(如求和与统计功能以及对数据库进行的插入、删除、修改等更新操作),借助于

SQL,人们可以实现数据操纵、定义和控制等功能,SQL也成了关系数据库的标准语言。

1. SQL 的发展

1972年,IBM公司开始研制实验型的关系数据库管理系统 System R,配制的查询语言称为 SQUARE(Specifying Queries As Relational Expression)。

1974年,Boyce 和 Chamberlin 将 SQUARE 修改为 SEQUEL (Structured English QUery Language)语言。这两个语言在本质上是相同的,但后者去掉了一些数学符号,并采用英语单词和结构式的语法规则,看起来很像是英语句子。用户比较欢迎这种形式的语言。后来 SEQUEL 简称为 SQL(Structured Query Language),即“结构化的查询语言”,但 SQL 的发音仍为“sequel”。

1986年10月,美国国家标准委员会(American National Standards Institute,ANSI)公布了 SQL 标准,并发布了 ANSI 文件 X3.135-1986《数据库语言 SQL》。

1987年6月,国际标准化组织(International Standards Organization,ISO)正式将其采纳为国际标准,因此,上述标准被称为 SQL-86。

1989年4月,ISO 提出了具有完整性特征的 SQL,并称之为 SQL-89。SQL-89 标准公布对数据库技术的发展和應用都起到了很大的推动作用。

1992年11月,ISO 又公布了 SQL 的新标准,即 SQL-92。

此后随着新版本 SQL-99、SQL-2000 和 SQL-2003 的相继问世,SQL 语言进一步得到了广泛应用。

2. SQL 语言的特点

(1) 高度非过程化。SQL 语言进行数据操作时,SQL 用户(应用程序或终端用户)只要提出“做什么”,具体怎么做则由系统找出一种合适的方法自动完成。因此用户无须了解存取路径和操作过程,这大大减轻了用户负担,而且有利于提高数据的独立性。

(2) 面向集合的操作方式。SQL 语句采用集合操作方式,不仅可以使一条语句从一个或者多个表中查询出一组结果数据(元组的集合),而且一次插入、删除和更新操作的对象也可以是元组的集合。

(3) 语法简单。虽然 SQL 语言功能强大,但是由于设计巧妙,其语法极其简单,完成核心功能只用了 11 个动词,其语法结构接近英语,因此容易学习和使用。

(4) SQL 是关系型数据库的标准语言。无论用户使用哪家公司的产品,SQL 的基本语法都是一样的,有利于各种数据库之间交换数据,有利于程序移植,有利于实现高度的数据独立性,有利于实现标准化。

(5) 可嵌入式的数据库语言。SQL 既可以在交互方式下以命令的形式独立地执行,即用户只需要在终端键盘上直接键入 SQL 命令就可以对数据库进行操作;又可以嵌入高级语言(常用的语言有 C、Java、FORTRAN、Delphi、Visual Basic、PowerBuilder 等)的程序中,以实现数据库数据的存取操作,给程序员设计程序提供了很大的方便。

3. SQL 的基本功能

SQL 的基本功能包括数据操纵、数据定义和数据控制。下面分别对这 3 个功能进行简

要介绍。

(1) 数据操纵功能。

数据操纵功能是通过数据操纵语言(DML)实现对数据库中数据的操纵。DML 包括 4 个基本语句: SELECT,对数据库中的数据进行检索; INSERT,往表中插入数据行; UPDATE,修改已经存在于表中的数据; DELETE,删除表中的数据行。

(2) 数据定义功能。

数据定义功能通过数据定义语言(DDL)实现对数据库中各种数据对象(包括表、视图、索引、存储过程、触发器等)的定义、修改和删除。DDL 包括 3 个基本语句: CREATE,新建数据库对象; ALTER,更新已有数据对象的定义; DROP,删除已经存在的数据对象。

(3) 数据控制功能。

数据控制功能通过数据控制语言(DCL)实现事务管理、数据保护以及数据库的安全性和完整性控制。DCL 包括 4 个基本语句: GRANT,授予权限; REVOKE,收回权限; COMMIT,提交事务; ROLLBACK,回滚事务。

3.4.2 Transact-SQL 简介

Transact-SQL(T-SQL)是 Microsoft 公司在关系型数据库管理系统 SQL Server 中对标准 SQL 的具体实现,是 Microsoft 公司对 SQL 的扩展,具有 SQL 的主要特点,同时增加了变量、运算符、函数、流程控制和注释等语言元素,使得其功能更加强大。T-SQL 增加的内容主要体现在如下 3 方面:

(1) 增加了流程控制语句。SQL 作为一种功能强大的结构化标准查询语言并没有包含流程控制语句,因此,不能单纯使用 SQL 构造出一种最简单的分支程序。T-SQL 在这方面进行了多方面的扩展,增加了语句块、分支判断语句、循环语句等。

(2) 加入了局部变量、全局变量等新概念,用户可以写出更复杂的查询语句。

(3) 增加了新的数据类型,处理能力更强。

1. SQL Server 2019 的数据类型

SQL Server 2019 定义了 33 种标准数据类型,当然用户也可以自己定义数据类型,但很少用到,一般都使用标准数据类型。

(1) 整数型。

整数型包括 int、bigint、smallint、tinyint、bit 5 种类型,它们的区别在于表示数据的范围不同,如表 3.2 所示。

表 3.2 整数型

数据类型	存储空间/字节	说 明
int	4	存储 $-2^{31} \sim (2^{31}-1)$ 的所有正负整数
bigint	8	存储 $-2^{63} \sim (2^{63}-1)$ 的所有正负整数
smallint	2	存储 $-2^{15} \sim (2^{15}-1)$ 的所有正负整数
tinyint	1	存储 $0 \sim 255$ 的所有正整数
bit	1	存储 1、0 或 NULL,非常适合用于开关标记

(2) 实数型。

实数型数据包括精确实数型和近似实数型。精确实数型包括 decimal 和 numeric 两种类型,近似实数型包括 real 和 float 两种类型,如表 3.3 所示。

表 3.3 实数型

数据类型	存储空间/字节	说 明
decimal(m,n)	最多 17	定点型数据类型。可表示 $-10^{38} + 1 \sim 10^{38} - 1$ 的有固定精度和小数位的数值,其中 m 表示总的有效位数,n 表示小数点后的十进制数的位数,即表示精确到小数点后第 n 位。
numeric(m,n)		同 decimal(m,n)
real	4	可精确到第 7 位小数,其范围是 $-3.40 \times 10^{38} \sim 3.40 \times 10^{38}$
float	8	可精确到第 15 位小数,其范围是 $-1.79 \times 10^{308} \sim 1.79 \times 10^{308}$

(3) 字符串型。

字符串型用于存储由英文字母、汉字、数字、特殊符号等组成的字符数据。根据编码方式的不同,字符串数据类型又分为 Unicode 字符串类型和非 Unicode 字符串类型。Unicode 编码方式是对所有字符均采用双字节统一编码,非 Unicode 编码方式是对不同国家或地区采用不同的编码长度,如英文字母使用一字节进行编码,而汉字使用两字节进行编码。常用的字符串数据类型如表 3.4 所示。

表 3.4 字符串型

数据类型	说 明
char(n)	固定长度,非 Unicode 编码,长度为 n,n 的取值范围 1~8000,若输入数据的字符数量小于 n,则系统自动在其后添加空格补齐
varchar(n)	可变长度,非 Unicode 编码,最大长度为 n,n 的取值范围 1~8000,若输入数据的字符数量小于 n,系统不会在其后添加空格补齐
text	存储大量可变长度非 Unicode 编码文本数据,其容量理论上是 $1 \sim (2^{31} - 1)$ 字节
nchar(n)	固定长度,Unicode 编码,长度为 n,n 的取值范围 1~4000,实际占用 2n 字节的存储空间
nvarchar(n)	可变长度,Unicode 编码,最大长度为 n,n 的取值范围 1~4000
ntext	存储大量可变长度 Unicode 编码文本数据,其容量理论上是 $1 \sim (2^{31} - 1)$ 字节

(4) 货币型。

货币型用于存储货币值数据,它固定精确到小数点后 4 位,在输入货币类型数据时,应在其前面加上货币符号,如人民币符号 ¥ 或美元符号 \$,SQL Server 2019 支持两种货币型,如表 3.5 所示。

表 3.5 货币型

数据类型	存储空间/字节	说 明
smallmoney	4	存储范围 $-2^{31} \sim (2^{31} - 1)$
money	8	存储范围 $-2^{63} \sim (2^{63} - 1)$

(5) 日期和时间型。

日期和时间型是用于存储日期和时间的数据类型,SQL Server 2019 支持 6 种日期和时

间型,如表 3.6 所示。

表 3.6 日期和时间型

数据类型	存储空间/字节	说 明
date	3	只存储日期,存储格式为 YYYY-MM-DD,可存储从 0001-01-01~9999-12-31 的日期数据
time(n)	3~5	只存储时间,n 的取值范围 0~7,存储格式为 hh:mm:ss[.nnnnnn],取值范围为 00:00:00.0000000~23:59:59.9999999
datetime	8	存储日期和时间的结合体,存储格式为 YYYY-MM-DD hh:mm:ss[.nnnnnn],可存储 1753-01-01~9999-12-31 的日期和时间数据,精确到千分之三秒
smalldatetime	4	可存储 1900-01-01~2079-06-06 的日期和时间数据,精确到分
datetime2(n)	6~8	可存储 0001-01-01 00:00:00.0000000~9999-12-31 23:59:59.9999999 的日期和时间数据,n 的取值范围 0~7,指定秒的小数位
datetimeoffset(n)	8~10	可存储 0001-01-01 00:00:00.0000000~9999-12-31 23:59:59.9999999 的日期和时间数据,n 的取值范围 0~7,指定秒的小数位。该类型带有时区偏移量,时区偏移量最大为±14 小时,包含了 UTC 偏移量,因此可以合理化不同时区捕捉的时间

(6) 二进制型。

二进制型数据类型有 3 种: binary(n)、varbinary(n)和 image,如表 3.7 所示。

表 3.7 二进制型

数据类型	说 明
binary(n)	用于存储固定长度的二进制数据类型,其中 n 用于设置最大长度,n 的取值范围为 1~8000 字节
varbinary(n)	用于存储可变长度的二进制数据类型,其中 n 用于设置最大长度,n 的取值范围为 1~8000 字节
image	用于存储更大容量可变长度的二进制数据类型,最多可以存储 $2^{31}-1$ 字节,约为 2GB,它既可存储文本格式,也可存储 GIF 等多种格式类型的文件

(7) 其他数据类型。

除了上面介绍的数据类型之外,SQL Server 2019 还支持如表 3.8 所示的 7 种数据类型。

表 3.8 其他数据类型

数据类型	说 明
geography	此类型用于存储诸如 GPS 纬度和经度坐标之类的椭球体(圆形地球)数据
geometry	此类型表示欧几里得(平面)坐标系中的数据
hierarchyid	层次类型,包含对层次结构中位置的引用,占用空间为 1~892B+2B 的额外开销

续表

数据类型	说 明
sql_variant	一种通用数据类型,它可以存储除了 text、ntext、image、timestamp 和它自身以外的其他类型的数据,其最大存储量为 8000B
timestamp	时间戳类型,每次更新时都会自动更新该类型的数据。其作用与邮局的邮戳类似,通常用于证明某项活动(操作)是在某一时刻完成的
uniqueidentifier	全局唯一标识符(GUID),其值可以从 Newsequentialid() 函数获得,这个函数返回的值对所有计算机来说是唯一的
xml	具有 SQL Server 2019 中其他类型的所有功能,还可以添加子树、删除子树和更新标量值等,最多存储 2GB 数据

2. 变量、流程控制和注释

(1) 变量。

变量用于存储临时存放数据,变量中的数据随着程序的运行而变化,变量定义时,必须有名字及数据类型两个属性。变量名用于标识该变量,变量类型确定了该变量存放值的格式、变量的取值范围及允许的运算。SQL Server 中的变量分为全局变量和局部变量两种。

全局变量是以“@@”开始的变量,全局变量是由系统提供且预先声明的变量,用户一般只能查看而不能修改全局变量的值。T-SQL 全局变量作为函数引用。例如,@@ERROR 返回执行的上一个 T-SQL 语句的错误号,@@CONNECTIONS 返回自上次启动 SQL Sever 以来连接或试图连接的次数。

局部变量是以“@”开始的变量,是用户声明的用以保存特定类型的单个数据值的对象,它局部于一个语句批,例如保存运算的中间结果作为循环变量等。

(2) 流程控制。

高级语言的一个重要特性是具有流程控制的能力,流程可以将更多的语句组织在一起,成为一个程序,来完成更为复杂的功能。T-SQL 也引入了一些流程控制,主要包含 BEGIN...END 语句,IF...ELSE 语句,CASE 语句,循环语句和 RETURN 语句。

变量和流程控制语句的具体应用详见第 7 章数据库编程。

(3) 注释。

注释是 T-SQL 程序代码中不被执行的文本,其作用是说明程序各模块的功能、设计思路等,方便程序的阅读、修改和维护。注释的方法有两种:一种是用“--”(两个连续的减号),用于注释一行代码,注释掉不被执行的部分从“--”开始,一直到行末尾结束;另一种是用“/*”开头,用“*/”结尾,用于注释多行代码,注释掉不被执行的部分为两个星号之间的文本。

3. 运算符

运算符是用于执行特定操作的一种符号。SQL Server 2019 中使用的运算符有算术运算符、逻辑运算符、比较运算符、赋值运算符、字符串连接运算符、位运算符等。

算术运算符包括加(+)、减(-)、乘(*)、除(/)和取模(%)5 种,用于实现两个数值型表达式的运算,包括货币型。另外,加(+)、减(-)运算符还可用于日期时间类型的数据的运算。

逻辑运算符用于逻辑判断,返回值为 TRUE 或 FALSE。逻辑运算符包括 AND、OR、NOT、BETWEEN、IN、LIKE、EXISTS、ALL、ANY、SOME 等。其应用会在后面章节中介绍。

比较运算符包括等于(=)、大于(>)、小于(<)、大于或等于(>=)、小于或等于(<=)、不等于(<>或!=)、不大于(!>)、不小于(!<),用于比较两个表达式的关系,几乎可以用于所有类型的表达式(text、ntext 和 image 数据类型除外)。

赋值运算符是等号(=),用于给变量、字段赋值。

字符串连接运算符是加号(+),用于把两个字符串连接起来形成一个新的字符串。表 3.4 中列出的字符串型数据类型的数据都可以用此运算符。

位运算符是实现两个操作数之间按位运算的符号,包括按位逻辑与(&)、按位逻辑或(|)、按位逻辑异或(^)和对一个操作数的按位取非操作(~)。进行位运算的操作数必须是整形数据或二进制数据(image 数据类型除外)。

4. SQL Server 2019 常用系统函数

系统函数是由系统预先编制好的程序代码,可以在任何地方调用。本文列举一些常用的系统函数,方便读者在使用时查阅。

(1) 字符串函数。

字符串函数用于字符串的处理,字符串的索引从 1 开始,常用的字符串函数如表 3.9 所示。

表 3.9 常用的字符串函数

函数表达式	功能说明	示 例
SUBSTRING(表达式,起始,长度)	截取子字符串	SUBSTRING('ABCDE',3,2) 结果为'CD'
LEFT(表达式,长度)	从左边开始截取指定长度的子串	LEFT('ABCDE',3) 结果为'ABC'
RIGHT(表达式,长度)	从右边开始截取指定长度的子串	RIGHT('ABCDE',3) 结果为'CDE'
LTRIM(表达式)	删除字符串左边的空格	LTRIM('ABCD') 结果为'ABCD'
RTRIM(表达式)	删除字符串右边的空格	RTRIM('ABCD') 结果为'ABCD'
UPPER(表达式)	将小写字符转换为大写字符	UPPER('abC12') 结果为'ABC12'
LOWER(表达式)	将大写字符转换为小写字符	LOWER('abC12') 结果为'abc12'
CHARINDEX(表达式 1,表达式 2)	返回表达式 1 在表达式 2 中第一次出现的起始位置,不存在则返回 0	CHARINDEX('ABC','abABCc') 结果为 3
LEN(表达式)	返回字符串长度,右边空格不计入	LEN('ABC') 结果为 4

(2) 日期函数。

这里介绍 7 个常用的日期函数,如表 3.10 所示。

表 3.10 常用的日期函数

函数表达式	功能说明	示 例
GETDATE()	返回当前数据库系统的日期和时间	GETDATE()
YEAR(表达式)	返回表达式的年份值	YEAR(GETDATE())
MONTH(表达式)	返回表达式的月份值	MONTH(GETDATE())
DAY(表达式)	返回表达式的日期值	DAY(GETDATE())
DATEADD(标志,间隔,日期)	返回日期间隔后的日期,标志 YY:年份,MM:月份,DD:日	DATEADD(YY,2,GETDATE())返回系统日期时间两年后的日期时间; DATEADD(MM,2,GETDATE())返回系统日期时间两月后的日期时间
DATEDIFF(标志,日期1,日期2)	返回日期2和日期1之间的时间间隔,标志 YY:年份,MM:月份,DD:日	DATEDIFF(YY,'2020-2-15','2020-5-15')返回值是相差年份为0; DATEDIFF(MM,'2020-2-15','2020-5-15')返回值是相差月份为3
DATEPART(标志,日期)	返回日期在指定标志的整数YY:年份,MM:月份,DD:日	DATEPART(MM,'2020-2-15')返回值为2

(3) 数值函数。

常用的数值函数如表 3.11 所示。

表 3.11 常用的数值函数

函数表达式	功能说明	示 例
ABS(表达式)	返回表达式的绝对值	ABS(-6) 结果为 6
RAND([种子])	返回 0~1 的随机数	RAND(); RAND(1)
ROUND(表达式,精度)	返回表达式指定精度的四舍五入的值	ROUND(12.3456,2) 结果为 12.35
CEILING(表达式)	返回大于或等于表达式的最小整数	CEILING(12.34) 结果为 13
FLOOR(表达式)	返回小于或等于表达式的最大整数	FLOOR(12.34) 结果为 12
SQRT(表达式)	返回表达式的算术平方根	SQRT(25) 结果为 5
POWER(底,指数)	返回底的指数次方	POWER(5,3) 结果为 125

(4) 类型转换函数。

常用的类型转换函数如表 3.12 所示。

表 3.12 常用的类型转换函数

函数表达式	功能说明	示 例
CAST(表达式 AS 数据类型 [(长度)])	将表达式由一种类型转换为另一种类型	CAST(9.5 AS int) 结果为 9; CAST(9.5 AS decimal(6,4)) 结果为 9.5000
CONVERT(数据类型 [(长度)],表达式[,日期样式])	将表达式由一种类型转换为另一种类型 日期样式如: 1: MM/DD/YY 111: YYYY/MM/DD	CONVERT(int,9.5) 结果为 9; CONVERT(varchar(128),GETDATE(),1) 结果为 05/15/20

(5) 聚合函数。

聚合函数是对一组值进行计算,并返回单个值,常用的聚合函数有 5 个,除 COUNT 外,其他的聚合函数在计算时都忽略 NULL 值,如表 3.13 所示。

表 3.13 常用的聚合函数

函数表达式	功能说明	示 例
COUNT([ALL DISTINCT] 表达式)	统计项数值 COUNT(*)返回所有的项数,包括 NULL 值和重复项; COUNT (ALL 表达式)返回非空的项数; COUNT(DISTINCT 表达式)返回 唯一且非空的项数	COUNT(*); COUNT(AGE); COUNT(DISTINCT AGE)
AVG([ALL DISTINCT] 表达式)	计算平均值	AVG(AGE)
SUM([ALL DISTINCT] 表达式)	求和	SUM(GRADE)
MIN([ALL DISTINCT] 表达式)	计算最小值	MIN(AGE)
MAX([ALL DISTINCT] 表达式)	计算最大值	MAX(AGE)

3.5 本章小结

本章首先介绍了 SQL Server 的发展历史、SQL Server 2019 的组件和管理工具、SQL Server 2019 的安装过程,然后介绍了 SQL 的基本功能和特点,最后介绍了 SQL Server 2019 的数据类型、变量、流程控制语句、运算符和常用函数,这些内容为后续章节的学习打下基础。

习题 3

简答题

1. SQL Server 2019 有哪些版本?
2. 简述 SQL Server 2019 的系统数据库构成。
3. 简述 SQL 的基本功能。
4. 简述 T-SQL 全局变量和局部变量的区别。
5. 简述 char(n)和 varchar2(n)两种数据类型的区别。
6. 简述常用的聚合函数及其功能。