

SQL Server

7.0

孙炳旭 陈 飞 梁茂盛 等编著

使用
管理
指南



 机械工业出版社
China Machine Press

SQL Server 7.0 使用管理指南

孙炳旭 陈飞 梁茂盛 等编著



机械工业出版社

JS181/12

本书由浅入深地介绍了从数据库基础到 SQL Server 7.0 高级管理等各方面的知识, 主要内容包括, 有关关系数据库的基本知识、基本 SQL 语句的使用、SQL Server 7.0 的安裝配置、基本操作、权限管理、事务日志管理、性能监控、备份恢复、数据仓库等, 是在数据库和 SQL Server 方面急需入门的读者的一本很好的教材, 也是 SQL Server 7.0 管理开发人员具有较高价值的参考书。

图书在版编目 (CIP) 数据

SQL Server 7.0 使用管理指南/孙炳旭等编编. —北京: 机械工业出版社, 2000.4
ISBN 7-111-07954-X

I. S… II. 孙… III. 关系数据库. —数据库管理系统, SQL Server 7.0 IV. TP311.138

中国版本图书馆 CIP 数据核字 (2000) 第 04733 号

机械工业出版社 (北京市百万庄大街 22 号 邮政编码 100037)

责任编辑: 张秀恩 牛新国 封面设计: 姚毅

责任印制: 何全君

三河市宏达印刷厂印刷 • 新华书店北京发行所发行

2000 年 3 月第 1 版第 1 次印刷

787mm×1092mm 1/16

印张 • 20.75 496 千字

0001—5 000 册

定价: 33.00 元

凡购本书, 如有缺页、倒页、脱页, 由本社发行部调换

本书购书热线电话 (010) 68993821、68326677-2527

Http://www.machineinfo.gov.cn/book/

前 言

欢迎使用本书!这是一本关于微软(Microsoft)公司出品的关系数据库管理系统 SQL Server 7.0 的操作、维护和管理书,主要面向在数据库方面的入门读者以及 SQL Server 7.0 的管理、开发人员。

SQL Server 7.0 是 SQL Server 6.5 的升级版本,在兼容 SQL Server 6.5 的同时加入了很多新的特性,如:

- 增加了部分新的 Transact-SQL 语句
- 新的安全认证模式
- UNICODE 支持
- 全文索引
- 支持 Windows95/98
- 数据仓库
- Microsoft SQL Server OLAP 服务
- PivotTable 服务

这些新的特性在本书中都做了不同程度的介绍。

为了适合不同程度的读者使用本书,在介绍 SQL Server 7.0 的新特性的同时,从数据库的有关基础知识到 SQL Server 的使用及管理都进行了介绍与分析,使得本书各部分既独立成章,又不乏联系;既强调了 SQL Server 的基本使用与管理,又突出了 SQL Server 7.0 所提供的新的特性,所以不同的读者可以根据自己的需要选择参考,而不必通览全书。

本书共分 16 章,第 1、4、5 章介绍了关系数据库基础知识及在 SQL Server 7.0 中的一些对应操作,第 2、3 章介绍了 SQL Server 7.0 的新特性、管理工具及其安装与配置,第 6 章介绍 SQL 语句的使用,第 7 至 14 章介绍了 SQL Server 7.0 的各方面的管理和维护原理和操作,第 15、16 章介绍了如何实现对 SQL Server 7.0 的性能监控、优化以及故障排除等。

本书在介绍一般理论和操作方法的同时,还侧重介绍了作者在使用过程中的一些实践经验和心得体会,具有较高的参考价值。

由于作者的水平有限,再加上时间等条件的限制,本书中不可避免地存在着一些不足和遗漏的地方,我们衷心地希望您能够对本书提出宝贵意见,更希望您在使用 SQL Server 7.0 的同时,把自己的使用心得能够反馈给我们,互相探讨,互相交流学习,使得大家能够更好地了解和使用 SQL Server 7.0,共同进步。

期待您的宝贵意见。

最后,衷心地希望本书能够对您在学习使用 SQL Server 7.0 的道路上有所帮助。

作者

2000 年 2 月于北京

目 录

前言

第 1 章 关系式数据库管理系统	1
1.1 数据库管理系统	1
1.1.1 数据的独立性	1
1.1.2 数据完整性	1
1.1.3 安全恢复与备份	2
1.1.4 查询优化	3
1.1.5 并发控制	3
1.2 关系式数据库管理系统	3
1.2.1 表	3
1.2.2 其他数据库对象	5
1.3 客户端/服务器方式	6
1.3.1 理解客户端/服务器模型	6
1.3.2 客户端/服务器数据库体系结构	7
1.3.3 客户端与服务器端的典型作用	7
1.3.4 SQL Server 的设计方式	8
1.4 SQL Server	9
第 2 章 SQL Server 7.0 新特性和管理工具	10
2.1 体系结构上的增强	10
2.1.1 数据库和文件	10
2.1.2 页面、行和列	11
2.1.3 索引	11
2.1.4 text 和 image 数据	11
2.1.5 行级锁定	11
2.1.6 事务日志	11
2.1.7 tempdb 数据库	12
2.1.8 查询处理器	12
2.1.9 分布式查询	12
2.2 服务器功能的增强	12
2.2.1 配置选项	13
2.2.2 备份、恢复、DBCC 和块拷贝	13
2.2.3 并行的查询执行	13
2.2.4 安全性	13
2.2.5 存储过程	14
2.2.6 一致性和标准遵循性	14

2.2.7 复制.....	14
2.2.8 数据仓库.....	14
2.2.9 OLAP 服务.....	15
2.2.10 PivotTable 服务.....	15
2.2.11 Micorsoft Proxy Server (代理服务器).....	15
2.2.12 SQL Server Enterprise Manager.....	15
2.2.13 SQL Server Agent.....	16
2.2.14 SQL Server Service Manager.....	16
2.3 在开发方面的增强.....	16
2.3.1 English Query.....	17
2.3.2 Transact-SQL.....	17
2.3.3 OLE DB.....	18
2.3.4 ODBC.....	18
2.4 应用程序类型.....	19
2.4.1 数据仓库应用程序.....	19
2.4.2 复制应用程序.....	19
2.4.3 Internet 和 Intranet 应用程序.....	20
2.4.4 分布式应用程序.....	20
2.4.5 管理工具.....	20
2.4.6 原型、跟踪分析和调试.....	20
2.4.7 其他程序设计接口.....	20
2.5 SQL Server 7.0 管理工具.....	21
2.5.1 Microsoft SQL Enterprise Manager.....	21
2.5.2 SQL Server Query Analyzer (查询分析器).....	26
2.5.3 SQL Server Client Network Utility.....	27
2.5.4 SQL Server Service Manager.....	28
2.5.5 SQL Server Profiler.....	29
2.5.6 SQL Server Performance Monitor.....	30
2.5.7 SQL Server Books Online.....	30
2.5.8 Microsoft DTC Administrative Console.....	31
2.5.9 SQL Server Network Utility.....	32
第 3 章 SQL Server 7.0 的安装与配置.....	33
3.1 概述.....	33
3.2 硬件与软件的需求.....	33
3.2.1 硬件需求.....	33
3.2.2 网络要求.....	33
3.2.3 软件需求.....	34
3.2.4 其他.....	34
3.3 SQL Server 7.0 的安装.....	35

3.4	SQL Server 7.0 的环境.....	50
3.4.1	目录及文件内容.....	50
3.4.2	自动创建的数据库.....	50
3.5	启动、暂停与停止 SQL Server.....	51
3.6	注册 SQL Server.....	55
3.7	SQL Server 7.0 的配置.....	56
3.8	SQL Server 7.0 的卸载.....	57
3.9	SQL Server 7.0 的自动升级与数据库移植.....	57
3.10	客户端管理程序的安装与卸载.....	60
3.11	SQL Server 7.0 性能.....	62
第 4 章	数据库管理.....	64
4.1	系统数据库与数据.....	64
4.2	数据存储结构.....	66
4.3	物理数据库文件和文件组.....	67
4.3.1	文件.....	68
4.3.2	文件组.....	68
4.4	管理数据库.....	69
4.4.1	创建 SQL Server 数据库.....	69
4.4.2	设置 SQL Server 数据库选项.....	73
4.4.3	修改数据库.....	76
4.4.4	删除 SQL Server 数据库.....	77
第 5 章	数据库对象.....	78
5.1	数据库对象.....	78
5.1.1	数据类型.....	78
5.1.2	约束.....	79
5.1.3	规则.....	85
5.1.4	默认.....	86
5.1.5	触发器.....	87
5.1.6	索引.....	88
5.1.7	全文目录与索引.....	90
5.1.8	Diagram.....	90
5.1.9	存储过程.....	97
5.2	表的规划.....	103
5.2.1	规划表.....	103
5.2.2	建立表.....	103
5.2.3	删除表.....	104
5.2.4	重新命名表.....	104
5.2.5	更改表的结构.....	104
5.2.6	处理表.....	104

5.2.7 视图.....	105
第 6 章 查询和存取 SQL 数据.....	107
6.1 建立基本查询.....	107
6.2 建立多表查询.....	109
6.2.1 利用表间的连接操作.....	109
6.2.2 利用子查询.....	111
6.3 使用汇总查询.....	113
6.3.1 GROUP BY 子句.....	114
6.3.2 HAVING 子句.....	114
6.3.3 ORDER BY 子句.....	114
6.3.4 集合函数.....	115
6.3.5 COMPUTE 子句.....	116
6.3.6 操作符 CUBE 和 ROLLUP.....	116
6.3.7 TOP n 子句.....	118
6.3.8 集合操作符.....	118
6.3.9 CASE 表达式.....	118
6.4 建立并行查询.....	119
6.5 分布式查询.....	119
第 7 章 远程连接与访问.....	121
7.1 概述.....	121
7.2 开放式数据库连接 (ODBC) 接口.....	122
7.3 数据源设置和连接.....	123
7.4 一个通过前端产品访问 SQL Server 的典型例子.....	126
7.4.1 建立连接.....	126
7.4.2 理解 Delphi 的数据库访问操作.....	126
7.4.3 实现 SQL 查询语句.....	128
第 8 章 用户及权限管理.....	129
8.1 基本概念.....	129
8.1.1 认证方法.....	129
8.1.2 认证模式.....	129
8.1.3 数据库用户.....	130
8.1.4 角色.....	130
8.1.5 对象权限和语句权限.....	131
8.2 用户访问 SQL Server 的数据要通过的四层校验.....	132
8.3 使用 SQL Enterprise Manager 来执行各种操作.....	133
8.3.1 配置 SQL Server 采用 Windows NT 认证模式或 SQL Server & Windows NT 混合认证模式.....	133
8.3.2 添加新的 SQL Server 用户.....	134

8.3.3	浏览 SQL Server 用户	139
8.3.4	修改用户属性、删除用户	139
8.3.5	建立数据库用户	140
8.3.6	创建自定义数据库角色	141
8.3.7	分配语句权限与对象权限	142
第 9 章	使用 SQL Server Agent 服务	144
9.1	SQL Server Agent 的配置	144
9.2	作业管理	147
9.2.1	新建作业	148
9.2.2	查看已存在的作业	153
9.3	管理操作员	153
9.3.1	新建操作员	153
9.3.2	查看操作员	154
9.4	管理告警	155
9.4.1	新建告警	155
9.4.2	查看告警	157
9.5	配置 SQL Mail	157
9.5.1	为 SQL Server 创建一个邮件帐号	158
9.5.2	配置 SQL Server 用一 Windows NT 用户 Windows NT Server 而非 System Account	158
9.5.3	在运行 SQL Server 的 Windows NT 上安装电子邮件客户程序	159
9.5.4	配置 SQL Server 和 SQL Server Agent 的电子邮件信息	159
9.5.5	启动 SQL Mail	159
第 10 章	数据备份与恢复	160
10.1	SQL Server 提供的四种数据库备份方法	160
10.1.1	完全数据库备份	160
10.1.2	差异备份	160
10.1.3	事务日志备份	160
10.1.4	数据库文件备份	160
10.2	常用的备份技术及对应的恢复方法	160
10.2.1	只采用完全数据库备份与差异备份	160
10.2.2	只采用完全数据库备份与事务日志备份	161
10.2.3	综合采用完全数据库备份、差异备份与日志备份	161
10.3	一般备份策略与原则	162
10.4	备份策略范例	162
10.4.1	情况一：数据量较小（少于 500M），且修改较少	162
10.4.2	情况二：数据量较大（大于 2G 以上）， 在白天数据受到频繁修改与添加	162
10.5	使用 Enterprise Manager 来进行数据库备份与恢复	163

10.5.1	创建备份设备	163
10.5.2	执行立即备份	163
10.5.3	安排自动备份	167
10.5.4	恢复数据库	170
10.5.5	恢复 master 数据库	176
第 11 章	数据交换	178
11.1	数据转换服务 DTS	178
11.1.1	概述	178
11.1.2	使用 Enterprise Manager 与脚本编程来实现 DTS 的一个例子	179
11.2	可移动数据库	185
11.2.1	创建可移动数据库	185
11.2.2	配置可移动数据库	186
11.2.3	安装可移动数据库	187
11.3	BCP	187
11.3.1	BCP 的工作方式	188
11.3.2	BCP 的运行选项	188
11.3.3	格式文件	189
11.3.4	/h 参数	192
11.3.5	并行加载	193
11.3.6	加载过程中的事务提交	193
11.3.7	改善加载性能	194
11.4	BULK INSERT	194
11.5	数据同步复制	195
11.5.1	出版商、订阅者模型	195
11.5.2	拉出和推出订阅	196
11.5.3	匿名订阅	196
11.5.4	同步	196
11.5.5	三种同步复制方式	196
11.5.6	使用 Enterprise Manager 配置数据同步复制	196
第 12 章	事务日志管理	204
12.1	与事务日志相关的几个概念	204
12.1.1	事务	204
12.1.2	事务日志	206
12.1.3	前写事务日志	206
12.1.4	数据缓冲区页面的刷新	207
12.1.5	缓冲区修订页	207
12.1.6	活动事务	207
12.1.7	检查点 (CHECK POINT)	207
12.1.8	事务日志的活动区	208

12.2 截短事务日志	208
第 13 章 数据仓库	210
13.1 概述	210
13.1.1 什么是数据仓库	210
13.1.2 数据仓库的特征	210
13.1.3 数据粒度 (Data Granularity)	211
13.1.4 数据站场 (Data Marts)	212
13.2 设计与创建数据仓库	213
13.3 数据仓库的使用	214
13.4 数据仓库与 OLTP	214
13.4.1 OLTP	214
13.4.2 数据仓库与 OLTP 的比较	215
13.5 在线分析处理 (OLAP)	216
13.5.1 功能简介	216
13.5.2 体系结构	217
13.5.3 安装	218
13.5.4 工具	225
13.5.5 OLAP Manager 的使用	232
13.6 English Query	234
13.6.1 安装与卸载	234
13.6.2 主要概念	238
13.6.3 English Query 的使用	238
第 14 章 服务器与客户端管理	242
14.1 服务器管理	242
14.1.1 注册服务器	242
14.1.2 网络连接管理	247
14.1.3 配置远程服务器	248
14.1.4 配置链接服务器	249
14.2 客户端管理	251
14.2.1 网络连接管理	251
14.2.2 配置 ODBC 数据源	252
14.3 使用 Makepipe 和 Readpipe 进行网络连接测试	252
第 15 章 SQL Server 的监控与优化	254
15.1 监控的目的	254
15.2 监控 SQL Server 的工具	254
15.2.1 使用 Profiler	254
15.2.2 使用 NT 事件查看器	261
15.2.3 使用 SQL Server 性能监视器 (Performance Monitor)	263

15.3 监控错误日志	266
15.4 SQL Server 的优化	268
15.4.1 存储过程	268
15.4.2 tempdb 数据库	268
15.4.3 RAID	269
15.4.4 内存管理	269
15.4.5 其他	270
第 16 章 SQL Server 故障排除	271
16.1 SQL Server 信息资源及监测方法	271
16.2 SQL Server 错误信息与分类	272
16.3 使用 DBCC (DataBase Consistency Checker)	273
16.4 对于一些实际情况的判断与处理	275
附录 A 关键词	277
附录 B 常用 Transact SQL 语句快速参考	294

第1章 关系式数据库管理系统

1.1 数据库管理系统

数据库管理系统（DataBase Management System, DBMS）的任务就是管理存储在数据库中的数据，同时还要提供各种工具让用户创建数据库、检索及修改数据等。

数据库管理系统主要完成以下工作：

- 维护数据库中的数据、表等之间的关系。
- 维护数据的独立性和完整性。
- 保持数据表示的一些规则（如表的结构定义）数据的安全。
- 在系统故障时，能够有方法把数据恢复到出现故障之前的一个正常状态。
- 为用户提供操作和管理的用户界面，包括交互式的查询语言等。
- 能够提供一定的查询优化。

数据库类似于数据文件，主要用来存放数据。并且和数据文件一样，数据库并不是把数据很直接、很简单地呈现给用户，用户必须运行一些相关的应用程序，按照一定的格式，读取数据库中的数据，显示给用户。

但从某种角度来说，数据库系统的功能要比数据文件强大得多，相关的数据可以使用各种表以及关系组合存放。数据存放在数据库中，还可以使用数据库管理系统软件对其准确性、安全性等进行很好的管理。另外，在使用数据时，还可以根据索引等方法使得数据的查询、统计完成得更简捷。所有这些都是数据文件所无法比拟的。

一般来说，一个数据库至少可以从两个角度来看，对用户来说，数据库在逻辑上就是一个数据的集合；对数据库管理系统来说，数据库则是存放在磁盘上的一系列有联系的字节。

在这里我们还要说明一点，通常所说的“数据库”有时根据不同的应用场合，可以有多种理解，有时可以指保存物理数据库的文件（这和数据文件是两个概念），有时指数据库管理系统本身。不过我们在灵活理解的同时还是要注意这几者之间的差别。

下面将简单介绍一下数据库管理系统中的一些基本概念，在其中将以介绍这些概念在SQL Server 中的应用为主。

1.1.1 数据的独立性

所谓数据的独立性是指数据库应用程序的正常运行，得到正确结果不依赖于数据库中存储数据的结构。这个特点很重要，它使得修改数据库中的数据、更改排序顺序、甚至是部分修改存储结构都不会影响到应用程序的正常执行。

1.1.2 数据完整性

所谓数据完整性就是指数据在逻辑上的一致性和精确性。所谓一致性就是指数据库的独立而又相互关联的各部分数据必须保证相互一致。数据库管理系统（DBMS）的任务之一

就是要发现逻辑上不一致的数据，并拒绝在数据库中存储。举个简单的例子，如 2 月 30 日就是个非法的数据。

一般来说，有关一致性的规则都和数据存放在一起，其控制权在服务器上，而不在客户端上，需要修改时，也要在数据库上进行修改。

本书中所介绍的 SQL Server 充分地保护了数据库的完整性。所有更新操作都是在事务中执行的，每个事务当其状态一致时被完全提交，或者当遇到错误时被全部回滚。如果某服务器出现故障，服务器重新启动后，未完成的任务也将所有 SQL Server 数据库中回滚。SQL Server 利用并发控制来保证在多个用户同时访问时的数据一致性。

数据完整性的类型有四种：

- 实体完整性：也叫做表完整性，它要求表中的所有行都有唯一的标识符，即主键。主键值能否修改或者整个行能否被删除取决于主键和任意外键参照之间所需的完整形级别。
- 域完整性：也叫做列完整性，它指定对某列有效的值的集合，并确定是否允许空值。一般通过使用“有效性检查”来实施完整性。它也可通过限定数据类型、格式（规则和约束）或者允许值的范围（规则、参照和检查约束）来实施。
- 参照完整性：用以确保主键（在被参照表中）和外键（在参照表中）的关系得以维护。
- 用户自定义的完整性。

一般来说数据的完整性大多由服务器进行中央控制，这样的好处是可以把所有的完整性规则和数据一同存储，也就不再需要每个前端的应用程序去单独存储完整性规则，并且在需要对规则进行修改时，只需要在一个地方进行改动就可以了。

1.1.3 安全恢复与备份

数据库管理系统中必须包含有一个子系统来负责硬件及软件错误的恢复工作。恢复子系统必须撤消所有在出错操作之前执行的所有更新操作，以保证在出现错误后相关联的数据仍然是一致的。SQL Server 支持自动恢复和手工恢复两种方式。

数据库管理系统还必须能够保证数据库中存放的数据的安全性，防止一些非法用户的访问及误用。同时还必须对一些用户进行权限方面的限定，比如说，一些用户只能有对数据的读取权限，而不能随意修改或者删除；而一些用户则可以完全控制数据，包括读取及修改等操作。

一般来说，数据库管理系统主要通过用户的帐号管理来完成安全性及授权方面的控制。

所谓备份就是指确定如何创建数据库和事务日志的副本，以及创建副本过程所采用的介质。备份主要是为了防止不同的软硬件错误所造成的数据丢失。SQL Server 提供了静态（指在服务器不接受应用程序的访问请求时进行备份）和动态（指在用户处理数据的过程中进行数据库备份操作）两种备份方式（相应的也就存在两种恢复方式）。SQL Server 对多个数据库分别备份，这种方法使发生故障后的安全恢复变得更加方便一些，同时也增加了数据库的安全性。一般来说，从备份的方法来说，SQL Server 允许采用以下方法：

- 完全备份：将数据库完全复制到备份中。
- 增量备份：只备份自上次完全备份以来数据库中发生的变化。
- 事务日志备份：以增量备份的方式备份事务日志。

- 数据库文件备份：利用复制文件的方法进行备份。

一般来说，完全备份操作简单，但占用空间比较大，备份时间也比较长，而事务日志备份和增量备份相对来说占用的空间则比较小，备份时间也比较短，但都必须以完全备份为基础，并且如果需要使用这两种备份进行恢复时，操作相对来说比较麻烦。

在这里我们要强调一下自动恢复和动态备份，因为这两者是对数据库管理系统的较高要求，系统必须能够在系统失败、操作系统崩溃、电源停电等状况下能够进行自动恢复；同时系统也应该能够在备份介质出现故障的情况下，通过事务日志来完成动态备份。

1.1.4 查询优化

每个数据库管理系统一般都提供一个查询优化器组件，它考虑到查询数据的各种可能的执行策略，并选择其中最有效的一种策略（这个策略也被称为执行计划）来完成用户的查询请求。优化查询需要考虑到许多因素，如表的大小、索引的类型、WHERE 子句的结构等。有关查询优化的具体内容请参考其他相关章节。

1.1.5 并发控制

数据库管理系统是一个多用户系统，也就是说它允许多个用户（或者说用户的应用程序）同时访问数据库，所以每个数据库管理系统必须存在一定的安全控制机制来保证多个并行的应用程序能够正常运行。

1.2 关系式数据库管理系统

数据库组织数据有多种方法，关系数据库是其中最有效的一种。关系数据库系统的概念是 E.F.Codd 在 1970 年首次提出的。它利用关系代数的理论，是数学中的集合论在有效地组织数据上的应用。它以关系数据模型为基础，其中心概念为关系。在关系数据库中，数据按表（在关系理论中称为关系）进行组织。从用户的观点来看，关系数据库中除了表，没有其他成分。表中存在有一到多个列（或者成为字段），零到多行（或者成为记录），行与列的交叉处存放数据值。

要使用数据库中的数据，必须使用由 DBMS 软件定义的一套命令和语句（或者称之为语言）。用于关系数据库的语言有多种，最常用的是 SQL 语言。

SQL 语言，全称 Structured Query Language，即结构化查询语言。在 Microsoft SQL Server 中我们使用 Transact-SQL 语言，它是 SQL 语言的一个子集标准。

SQL 语言是一种面向集合的语言，它可以使用一条语句从一个或者多个表中查询多行，它的另一个重要特性是非过程性。这决定了它可以描述用户想要做些什么，由系统来负责找到解决用户请求的较合适方法，而不必象过程性语言那样，需要用户自己使用语言来写出程序来描述每个任务需要如何完成。

SQL 含有两个子语言，数据定义语言（DDL）和设计修改语言（DML）。DDL 语句用于描述数据库表的模式。DDL 含有 3 个通用的 SQL 语句，CREATE、ALTER 和 DROP。DML 包括操纵数据的所有操作。主要包含有 SELECT、INSERT、DELETE 和 UPDATE。

1.2.1 表

在关系数据库中，表是在数据库中组织数据的基本单位，是最基本的数据库对象。它

可以以一种独立表的形式存在，也可以由多个表以及它们之间的关系组成数据库。表从实质上可以用来表示一个组织中的某些重要的对象类，比如说，某个公司的数据存放在一个数据库中，那么不同的数据就可以根据需要存放在不同的表中，比如雇员信息表、产品销售表等，而这些表可以为独立的表，也可以通过相同的部分信息建立联接，从而可以使以后的查询、统计等工作更加简捷。

每个表内部都由列和行组成（列和行，由分别成为字段与记录）。每一列代表着对象的指定的属性，每一行则代表着一个元组，也可以说表示了一个表所代表的对象的一个实例。

下面举一个数据库的简单实例，以便于大家理解数据库以及表中的一些基本概念。

数据库 Student 包含有三个表，Student_Info、Student_Score 和 Student_Book。

下面给出三个表的信息：

表 1-1 Student_Info 表

Number	Name	Age	Class
1	张 小 小	23	99031
2	李 三	22	99031
3	陈 胜	23	99031

表 1-2 Student_Score 表

Number	Phy_Score	Che_Score	Mat_Score
2	90	70	98
3	60	90	80
1	80	60	76

表 Student_Info 表示学生信息，每个学生有如下几方面的信息：

Number 表示学号，在此它是可以唯一标识学生的字段。Name 表示姓名，Age 表示年龄，Class 表示班级。

表 1-3 Student_Book 表

Book_Number	Book_Name	Borrow_Number	Return_Time
TN1234/3	高等数学题解	1	99/10/10
I183.3/3	只有一个人人生	3	99/11/1
TN459	大学物理（下）	2	99/10/23
KN86/6.80	新概念英语（四）	3	99/11/1
J38/9	摄影手册	1	99/10/20

表 Student_Score 表示学生成绩信息，其中包括：

Number 表示学生的学号，Phy_Score 表示物理成绩，Che_Score 表示化学成绩，Mat_Score 表示数学成绩。

表 `Student_Book` 表示学生借阅书籍信息, 其中包括:

`Book_Number` 表示书号, `Book_Name` 表示书名, `Borrow_Number` 表示借阅学生的学号, `Return_Time` 表示书籍应还时间。

生成如上的几张表需要经过几个简单的步骤, 定义表结构、选择各列的数据类型、输入数据。这些工作都可以使用数据库管理系统提供的工具或者支持的语言来完成。

在上面的例子, 我们可以看到, 当我们需要得到的信息无法从单独一张表中获得时 (比如想要了解每个学生的名字及其成绩, 此时就需要 `Student_Info` 和 `Student_Score` 两张表中的数据), 我们可以根据多张表之间的联系, 进行组合查询, 从而获得结果。

从上面的例子中我们可以了解到数据库中的一些特点 (在这里需要声明一下, 这里举的例子为关系数据库的实例, 所以下面的属性严格来说应该说是关系数据库的特点):

- 表中的行或者列是没有固定位置的, 比如说在表 `Student_Info` 中我们把第一行和第二行整行的数据进行交换, 其结果是相同的。
- 表中的每一列都有自己唯一的名字, 但不同表则可以包含有相同的列的名字 (甚至可以表示不同的含义, 取值也不同), 同时, 在不同的表中不同名字的列也可以取值相同, 如表 `Student_Info` 中的 `Number` 列和表 `Student_Book` 中的 `Borrow_Number` 列 (当然在这里我们完全可以命名此列为 `Number`)。
- 表中的每个数据项必须是单值的, 即每行与列的交叉处只能包含有单一的数据。
- 对于每个表中, 至少有一个关键字能够唯一标识某一行数据, 这个关键字可以由一列组成 (如表 `Student_Info` 中的 `Number` 列), 也可以由多个列组合而成。在关系数据库模型中, 这种关键字被称为候补关键字, 如果表中包含有多个关键字, 则数据库设计员可以选择其中之一为主关键字。注意某一行 (或者某几列的组合) 在一个表中为关键字, 而在另一个表则未必也一定是关键字, 如表 `Student_Info` 中列 `Number` 为关键字, 而表 `Student_Book` 中列 `Borrow_Number` 则不是关键字。

1.2.2 其他数据库对象

1. 索引

索引指用来根据一个或者多个列中的有序值来在一个表中搜索行, 可以提高数据访问的性能, 同时还可以用于保证数据行间的唯一性 (即实体完整性)。索引可以分为以下两种: 聚集索引和非聚集索引。当创建索引时, 默认的是非聚集索引。

所谓非聚集索引, 必须对表中的每一行都包括一个索引项。表的物理顺序与索引的顺序并不相同。采用非聚集索引时, 如果增加新的数据, 仍然将增加在表的尾部。

而采用聚集索引的表中数据是按照索引的顺序来存放的, 因而表的物理顺序和索引的顺序是一致的, 并且, 在增加数据时, 也将根据索引而插在相应的位置。每个表只能定义一个聚集索引。

还有一种比较常用的索引是唯一索引, 所谓唯一索引, 就是指不允许表中出现具有相同索引值的两行数据。

可以使用 `CREATE INDEX` 语句来创建索引。

2. 视图

视图是在一个或者多个表中查询数据的一种方式, 可以通过 `SELECT` 语句来实现。使用视图往往可以比较简便地定制到用户所需要的数据, 同时还可以增加数据库的安全性。