

知识全面的范例操作手册

专业+详尽+实践演练

SQL
核心技术

SQL 语法范例手册

范秀平 尚 武 编著

- 内容安排由浅入深，兼顾深度与广度，适合教与学
- 通过Oracle和SQL Server诠释SQL的函数和功能
- 290个范例+155道习题，关注实践能力的培养
- 所有范例均给出代码、执行步骤与结果屏幕截图

SQL 语法范例手册

1

内容安排由浅入深，兼顾深度与广度，适合教与学

全书共12章，涵盖了SQL语言的方方面面，不仅包括SQL语言的功能与特性、SQL语言的基本语法等基础内容，还包括数据库的创建和管理、数据的查询、数据的更新、视图、SQL触发器和游标等高级应用。

2

290个应用范例，通过Oracle和SQL Server两种语法诠释

全书示例丰富，在每个功能知识点的讲解之后，都配有相关示例操作，介绍如何在Oracle 10g和SQL Server 2000两种语言中编写和运行程序。每个范例都给出代码、执行步骤与结果屏幕截图。

3

习题丰富多样，为读者进一步掌握知识提供平台

全书共计155道习题。习题种类多样，既有关关注实践的上机实验题，也有偏重理论的选择题和填空题，并配有习题解答，便于读者自我检测学习效果。

封面设计：林陶

技术电话：(010) 82896445/46转8407

销售电话：(010) 82896448 82896443

网址：www.khp.com.cn

E-mail：feedback@khp.com.cn

网上购书：www.huachu.com.cn

ISBN 978-7-03-019007-9



9 787030 190079 >

定价：35.00元

SQL
核心技术

SQL 语法范例手册

范秀平 尚 武 编著

科学出版社

北京科海电子出版社
www.khp.com.cn

内 容 提 要

SQL 是一种功能强大的语言, 熟练掌握并灵活应用 SQL 是数据库知识初学者和数据库应用开发人员的必备基本功。本书共 12 章, 详述了 SQL 的功能与特性、SQL 表达式、SQL 函数、创建与管理数据库和表、约束和索引的使用、数据的查询、数据的更新、视图的使用、事务、SQL 触发器和游标、数据库的安全与性能、存储过程等内容。这些内容涵盖了 SQL 的主要知识点。全书示例丰富, 所有功能点分别用 SQL Server 2000 和 Oracle 10g 进行示例演示, 便于读者学习实践。

本书为方便读者掌握 SQL 知识, 每章后都配有习题和上机实验, 以加深对理论内容的理解, 不仅适合大中专院校相关专业的学生学习, 也可作为 SQL 编程人员的参考资料和培训教程。

图书在版编目 (CIP) 数据

SQL 语法范例手册/范秀平, 尚武编著.

—北京: 科学出版社, 2007

ISBN 978-7-03-019007-9

I. S… II. ①范…②尚… III. 关系数据库—数据库管理系统

—教材 IV. TP311.138

中国版本图书馆 CIP 数据核字 (2007) 第 073657 号

责任编辑: 何 武 / 责任校对: 科 海

责任印刷: 科 海 / 封面设计: 林 陶

科 学 出 版 社 出 版

北京东黄城根北街 16 号

邮政编码: 100717

<http://www.sciencep.com>

北京市鑫山源印刷有限公司

科学出版社发行 各地新华书店经销

2007 年 6 月第一版

开本: 16 开

2007 年 6 月第一次印刷

印张: 21.5

印数: 0001-4000

字数: 523 千字

定价: 35.00 元

(如有印装质量问题, 我社负责调换)

前 言

SQL语言是结构化查询语言（Structure Query Language）的缩写，简单易学、风格统一。SQL语言主要包括数据定义、数据控制、数据操纵和数据查询等功能，其中最重要的是数据查询功能。本书侧重于实践操作、语言简洁，书中各知识点的讲解均配有大量可实际运行的示例，供读者边学习边实践，帮助读者快速、全面地掌握SQL的使用方法和技巧。

全书共分12章，涵盖了SQL语言的方方面面，不仅包括SQL语言的功能与特性、SQL语言的基本语法等基础内容，还包括数据库的创建和管理、数据的查询、数据的更新、视图、SQL触发器和游标等高级应用。

第1章主要介绍了SQL语言一些基本知识，包括SQL的历史、SQL语句的功能和特性、SQL的未来、SQL的基础语法知识等内容。通过这一章的学习，使读者对SQL及其基础语法有一个整体的了解，为以后章节的学习打下基础。

第2章主要介绍了SQL函数，包括聚合函数、字符串函数、日期时间函数、数字函数和转换函数等。通过这一章的学习，读者可以对SQL中的函数有一定的了解。

第3章主要介绍了创建和维护数据库的基本知识，包括数据定义语言概念、如何创建数据库和表，数据库及表的管理、约束和索引的使用等。通过这一章的学习，读者可以应用SQL语言对数据库和表进行相关操作。

第4章主要介绍了数据查询，包括如何使用SELECT子句进行查询、指定要检索的表、如何过滤数据等。通过这一章的学习，读者可以应用SQL查询操作。

第5章主要介绍了数据排序和分组，包括如何使用ORDER BY子句排序、GROUP BY和HAVING子句的使用等内容。

第6章主要介绍了多表连接查询的使用，包括使用连接的原因、如何处理连接、内连接和外连接的使用、自连接和交叉连接的使用、联合查询的应用等。通过这一章的学习，读者可以应用SQL多表连接操作。

第7章主要介绍了子查询的使用，包括相关子查询、子查询的规则、单行子查询和多行子查询的使用、子查询的嵌套等方法。

第8章主要介绍了数据更新，包括如何创建和插入记录，如何删除和更新记录等。通过这一章的学习，读者可以应用SQL语言对数据表中的数据进行简单操作。

第9章主要介绍了视图的应用，包括使用视图的原因、视图的处理、视图的优点和缺点、如何创建和删除视图、如何修改视图、如何通过视图管理数据等。通过这一章读者能更深刻地了解SQL中视图的应用。

第10章主要介绍了事务的处理，包括事务的隔离级别、事务处理命令、事务属性、

如何开始和结束事务、如何使用事务保存点、锁等知识。

第11章主要介绍了SQL触发器和游标,包括如何创建和删除SQL触发器、如何启用和禁用触发器、如何声明游标、如何打开和关闭游标、如何使用游标更新和删除等内容。

第12章主要介绍了数据库的安全与性能,包括SQL安全性、权限的授予和取消、如何提高SQL语句的可读性、如何优化查询、存储过程的使用等。

本书按照由浅入深的原则,对SQL各主要知识点进行了详细的讲解,书中涉及了大量的操作示例,旨在帮助读者加深对所学知识的理解。

本书由范秀平、尚武编著整理,此外,参与本书编辑和修改的还有刘峰、周小杰、徐红、高林宇、施伟伟、张爱华、冯志刚、潘华、金伟、缪辉、戴旭东、许宝建、蔡东军、梁小军、刘小松等同志。在此,编者对以上人员致以诚挚的谢意!

本书内容翔实、结构紧凑、条理清晰。但由于作者水平有限,书中难免会有疏漏和不足之处,恳请读者批评指正。

编 者

2007年2月

目 录

第1章 结构化查询语言	1		
1.1 SQL概述	1		
1.1.1 SQL和SQL标准简史	2		
1.1.2 SQL的功能和特性	3		
1.1.3 SQL语句的结构	4		
1.1.4 SQL的未来	4		
1.2 常量	5		
1.2.1 数字常量	5		
1.2.2 字符串常量	5		
1.2.3 时间和日期常量	6		
1.2.4 符号常量	6		
1.3 数据类型	6		
1.3.1 数字数据类型	7		
1.3.2 字符串数据类型	8		
1.3.3 日期时间数据类型	8		
1.4 SQL表达式	9		
1.5 SQL控制语句	10		
1.5.1 语句的分类	10		
1.5.2 语句的规则	11		
1.6 丢失数据	12		
1.7 上机实验	13		
1.7.1 实验内容	13		
1.7.2 实验指导	13		
1.8 练习题	14		
1.9 习题答案	15		
第2章 SQL函数	16		
2.1 聚合函数	16		
2.1.1 COUNT()函数	17		
2.1.2 SUM()与AVG()函数	17		
2.1.3 MAX()与MIN()函数	17		
2.2 字符串函数	18		
2.3 日期时间函数	19		
2.4 数学函数	20		
2.5 转换函数	22		
2.5.1 CONVERT()函数	23		
2.5.2 TO_CHAR()函数	23		
2.5.3 TO_NUMBER()函数	23		
2.6 上机实验	24		
2.6.1 实验内容	24		
2.6.2 实验指导	24		
2.7 练习题	27		
2.8 习题答案	28		
第3章 创建和维护数据库与表	29		
3.1 SQL数据库概述	29		
3.2 创建数据库	30		
3.3 创建表	33		
3.4 名称	35		
3.4.1 表名和列名	35		
3.4.2 别名	35		
3.5 管理数据库	35		
3.5.1 修改数据库	36		
3.5.2 删除数据库	38		
3.5.3 修改数据表	39		
3.5.4 删除数据表	43		
3.6 规则和默认	44		
3.7 约束	45		
3.7.1 CHECK约束	45		
3.7.2 UNIQUE约束	47		
3.7.3 NOT NULL约束	49		

3.7.4 PRIMARY KEY约束.....	50	第5章 数据排序与数据分组	114
3.7.5 禁用约束.....	51	5.1 数据排序	114
3.7.6 删除约束.....	52	5.1.1 ORDER BY语法格式	114
3.8 索引	52	5.1.2 升序排序和降序排序	116
3.8.1 创建索引.....	53	5.1.3 使用表达式排序	117
3.8.2 索引的分类.....	56	5.1.4 多级排序.....	120
3.8.3 查看索引信息.....	59	5.1.5 排序中的空值.....	122
3.8.4 删除索引.....	60	5.2 数据分组	122
3.9 上机实验	61	5.2.1 GROUP BY子句.....	122
3.9.1 实验内容.....	61	5.2.2 带有WHERE子句的GROUP BY.....	124
3.9.2 实验指导.....	62	5.2.3 多次分组.....	124
3.10 练习题	63	5.2.4 GROUP BY和ORDER BY	126
3.11 习题答案	64	5.2.5 HAVING子句.....	127
第4章 简单查询.....	65	5.3 上机实验	130
4.1 SELECT语句概述.....	65	5.3.1 实验内容.....	130
4.2 使用SELECT子句查询数据.....	67	5.3.2 实验指导.....	130
4.2.1 查询所有的列.....	67	5.4 练习题	132
4.2.2 查询指定的列.....	69	5.5 习题答案	133
4.2.3 查询中使用数字和表达式.....	71	第6章 连接查询	134
4.2.4 查询中使用别名.....	75	6.1 使用连接的原因.....	134
4.2.5 查询列中使用distinct关键字	77	6.2 连接语法	135
4.2.6 查询中使用函数.....	78	6.2.1 传统的连接语法	135
4.3 使用FROM子句指定要检索的表	85	6.2.2 SQL连接语法	137
4.4 使用WHERE子句进行过滤	86	6.3 连接的处理.....	139
4.4.1 比较运算符.....	87	6.3.1 笛卡尔积.....	140
4.4.2 逻辑运算符.....	91	6.3.2 连接中运算符的使用	141
4.4.3 BETWEEN子句.....	96	6.3.3 多表连接.....	142
4.4.4 IN子句	98	6.3.4 连接结果的处理	143
4.4.5 LIKE子句.....	101	6.4 自连接	145
4.4.6 空值的处理.....	106	6.5 内连接	146
4.5 上机实验	109	6.5.1 内连接的分类.....	146
4.5.1 实验内容.....	109	6.5.2 等值连接.....	147
4.5.2 实验指导.....	109	6.5.3 不等连接.....	148
4.6 练习题	112	6.5.4 自然连接.....	149
4.7 习题答案	113	6.6 外连接	151

6.6.1 外连接的分类.....	151	7.6.4 在INSERT中使用子查询.....	193
6.6.2 左外连接 (LEFT JOIN)	151	7.6.5 在DELETE中使用子查询.....	194
6.6.3 右外连接 (RIGHT JOIN)	152	7.7 上机实验	195
6.6.4 全外连接 (FULL JOIN)	153	7.7.1 实验内容.....	195
6.7 复合连接	154	7.7.2 实验指导	195
6.8 交叉连接	156	7.8 练习题	197
6.9 联合查询	157	7.9 习题答案	197
6.9.1 UNION运算	157	第8章 数据处理	199
6.9.2 INTERSECT运算	162	8.1 数据操纵语言概述.....	199
6.9.3 MINUS运算	163	8.2 插入数据	200
6.10 上机实验	164	8.2.1 使用INSERT...VALUES语句 插入数据.....	200
6.10.1 实验内容.....	164	8.2.2 使用INSERT...SELECT语句 插入数据.....	205
6.10.2 实验指导.....	164	8.3 更新数据	208
6.11 练习题	167	8.3.1 指定更新的列和新值	208
6.12 习题答案	168	8.3.2 嵌套更新	213
第7章 子查询	169	8.4 删除数据	214
7.1 子查询概述	169	8.5 其他用于处理数据的SQL语句	216
7.2 相关子查询	171	8.5.1 MERGE语句	216
7.3 子查询规则	173	8.5.2 TRUNCATE语句.....	217
7.4 单行子查询	174	8.6 上机实验	218
7.4.1 比较子查询.....	174	8.6.1 实验内容.....	218
7.4.2 聚合函数和子查询.....	176	8.6.2 实验指导	218
7.4.3 单行子查询中的GROUP BY 和HAVING子句	178	8.7 练习题	220
7.5 多行子查询	179	8.8 习题答案	221
7.5.1 IN的使用	179	第9章 视图	222
7.5.2 NOT IN的使用	182	9.1 视图概述	222
7.5.3 ANY的使用	183	9.2 使用视图的原因.....	223
7.5.4 ALL的使用	185	9.3 视图的优点和缺点.....	224
7.5.5 交集和差集的查询: EXISTS.....	187	9.4 视图的处理.....	224
7.5.6 NOT EXISTS的使用	190	9.5 创建视图	225
7.6 子查询的嵌套	191	9.5.1 创建简单视图	225
7.6.1 嵌套子查询.....	191	9.5.2 视图中使用列的别名	229
7.6.2 在HAVING中使用子查询	192		
7.6.3 在UPDATE中使用子查询	193		

9.5.3 视图中使用连接	233	第11章 SQL触发器和SQL游标	270
9.5.4 视图中使用子查询	237	11.1 了解SQL触发器和游标	270
9.6 更改视图	238	11.2 创建和删除SQL触发器	272
9.7 删除视图	239	11.2.1 创建SQL触发器	272
9.8 通过视图管理数据	240	11.2.2 删除SQL触发器	275
9.8.1 通过视图查询数据	241	11.2.3 更改触发器	276
9.8.2 通过视图插入数据	242	11.3 创建INSERT触发器	278
9.8.3 通过视图删除数据	245	11.4 创建Update触发器	278
9.8.4 通过视图更新数据	246	11.5 创建Delete触发器	280
9.9 查询视图信息	247	11.6 触发器的启用和禁用	281
9.10 上机实验	248	11.7 声明游标	282
9.10.1 实验内容	248	11.7.1 SQL-92语法组成	282
9.10.2 实验指导	249	11.7.2 游标声明	283
9.11 练习题	250	11.8 打开和关闭游标	287
9.12 习题答案	251	11.8.1 打开游标	287
第10章 事务	253	11.8.2 关闭游标	288
10.1 事务概述	253	11.9 删除游标	289
10.1.1 并发控制	254	11.10 应用游标	289
10.1.2 数据库恢复	255	11.11 使用游标更新和删除	292
10.2 事务隔离级别	256	11.11.1 使用游标更新	292
10.3 事务处理命令	257	11.11.2 使用游标删除	293
10.4 设置事务属性	257	11.12 上机实验	294
10.5 开始事务	258	11.12.1 实验内容	294
10.6 在事务中使用保存点	260	11.12.2 实验指导	294
10.7 提交事务	262	11.13 练习题	297
10.8 撤消事务	263	11.14 习题答案	298
10.9 锁	264	第12章 数据库的安全与性能	299
10.9.1 锁的分类	264	12.1 理解SQL安全性	299
10.9.2 关于锁的其他信息	265	12.1.1 用户	300
10.9.3 死锁处理	265	12.1.2 数据库元素	301
10.10 上机实验	267	12.1.3 权限	302
10.10.1 实验内容	267	12.2 授予和撤消权限	303
10.10.2 实验指导	267	12.2.1 授予权限	303
10.11 练习题	268	12.2.2 撤消权限	311
10.12 习题答案	269	12.3.3 禁止授权	315

12.3 视图与SQL安全	315	12.7.3 提高系统性能的方法	323
12.4 数据库性能	317	12.8 存储过程	325
12.4.1 影响数据库性能的因素	317	12.8.1 创建存储过程	326
12.4.2 基准测试	318	12.8.2 执行存储过程	328
12.4.3 提高性能的工具	318	12.8.3 删除存储过程	328
12.5 提高SQL语句的可读性	321	12.9 上机实验	329
12.6 在表上创建索引以提高性能	321	12.9.1 实验内容	329
12.7 优化查询	322	12.9.2 实验指导	330
12.7.1 查询优化器	322	12.10 练习题	332
12.7.2 编写高性能的SQL语句	323	12.11 习题答案	333

第 1 章

结构化查询语言

本章导读

SQL (Structured Query Language, 结构化查询语言) 是一个功能强大的数据库语言。SQL语言的主要功能就是同各种数据库建立联系, 进行沟通。按照ANSI (美国国家标准协会) 的规定, SQL被作为关系型数据库管理系统的标准语言。在关系型数据库中的地位就犹如英语在世界上的地位。SQL语句通常用于完成一些数据库的操作任务, 比如在数据库中更新数据, 或者从数据库中检索数据, 为用户灵活地完成数据库任务提供了很大的便利。使用SQL的常见关系数据库管理系统有: Oracle, Sybase, Microsoft SQL Server, Access等等。本章将对SQL进行一些基础性的介绍, 使读者能较好地掌握SQL的基础知识。

学习目标:

- 掌握SQL简史及功能特性
- 了解SQL语句的结构
- 掌握常量的使用
- 掌握SQL的数据类型
- 掌握表达式和运算符的使用
- 了解控制语句的分类和规则

1.1 SQL概述

SQL早已确立起自己作为关系数据库标准语言的地位, 已被众多商用DBMS (DataBase Management System, 数据库管理系统) 产品所采用, 使得它已成为关系数据库领域中一个主流语言。SQL语言简单易学、风格统一, 利用简单的几个英语单词的组合就可以完成所有的功能。它不仅包含数据查询功能, 还包括插入、删除、更新和数据定义功能。SQL的用户可以是应用程序, 也可以是终端用户。

1.1.1 SQL和SQL标准简史

SQL语言是IBM公司在20世纪70年代开发的关系数据库原型SYSTEM R的一部分。

在20世纪70年代初，Dr.E.F.Codd首先提出了关系模型。70年代中期，IBM公司在研制SYSTEM R关系数据库管理系统中研制了SQL语言。1986年美国国家标准协会（American National Standard Institute, ANSI）和国际标准化组织（International Standard Organization, ISO）发布的ANSI/ISO标准正式确认SQL的标准地位。1989年，美国ANSI采纳在ANSI X3.135-1989报告中定义的关系数据库管理系统的SQL标准语言，称为ANSI SQL 89。

SQL语言主要包括数据定义、数据控制、数据操纵和数据查询等功能，其中最重要的是数据查询功能。开始时SQL是为IBM公司的DB2系列数据管理系统而开发的。SQL是一种非过程化语言（非过程化语言的意思就是指与具体过程无关，例如SQL描述了如何对数据进行检索、插入、删除，但它并不说明如何进行这样的操作）。不过，大多数数据库系统对SQL进行了有针对性的扩展使它们成为了过程化语言。现在SQL广泛地被采用，它使全部用户，包括应用程序员、DBA管理员和终端用户受益匪浅，其优点主要有以下几点：

非过程化语言

非关系数据模型的数据操纵语言是面向过程的，若要完成某项请求，必须指定存取路径，而SQL是一个非过程化的语言，它让数据库系统负责决定怎样处理数据的查询问题。当进行数据操作时，只要指出要做什么，而无需指出怎么做，因此用户无需了解存取路径，存取路径的选择以及SQL语句的操作过程由系统自动完成，这样提高了数据的独立性。

综合统一

非关系模型的数据语言分为模式定义语言和数据操纵语言，其缺点是，当要修改模式时，必须停止现有数据库的运行，转储数据，修改模式并编译后再重装数据库，因此很麻烦。SQL集数据定义、数据操纵和数据控制功能于一体，语言风格统一，可独立完成数据库生命周期的所有活动，包括定义关系模式、查询、更新、建立数据库、数据库安全性控制等一系列操作的要求，这就为数据库应用系统开发提供了良好的环境。

可移植性

由于所有主要的关系数据库管理系统都支持SQL语言，用户可将使用SQL的技能从一个RDBMS转到另一个，所以用SQL编写的程序都是可以移植的。

两种使用方式

第一种方式，用户可以在终端键盘上键入SQL命令，对数据库进行操作，称之为自含式语言。第二种方式，将SQL语言嵌入到高级语言程序中，此为嵌入式语言。在这两种不同的使用方式下，SQL语言的语法结构基本上是一致的，它以统一的语法结构提供两种不同的使用方法，为用户提供了极大的灵活性与方便性。

语言简洁，易学易用

SQL语言功能极强，但由于设计巧妙，语言十分简洁，而且SQL语言语法简单。完成数据定义、数据操纵、数据控制的核心功能只用了9个动词：CREATE、DROP、ALTER、SELECT、INSERT、UPDATE、DELETE、GRANT、REVOKE等，因此易于学习，容易使用。

1.1.2 SQL的功能和特性

SQL是应用程序与DBMS进行通信的一种语言和工具，将DBMS的组件联系在一起，可以为用户提供强大的功能，使用户可以方便地进行数据库的管理、数据的操作。通过SQL命令，程序员或数据库管理员（DBA）可以完成以下功能。

- 数据定义：SQL能让用户自己定义所存储数据的结构，以及所存储数据各项之间的关系。
- 数据更新：SQL为用户和应用程序提供了添加、删除、修改等数据更新操作，使用户或应用程序可以向数据库中增加新的数据、删除旧的数据以及修改已有数据，有效地支持了数据库数据的更新。
- 数据查询：SQL使用户或应用程序可以从数据库中按照自己的需要查询数据并组织使用它们，SQL不仅支持简单条件的检索操作，而且支持子查询、查询的嵌套、视图等复杂的查询。
- 保护数据安全：SQL能对用户和应用程序访问数据、添加数据等操作的权限进行限制，以防止未经授权的访问，有效地保护数据库的安全。
- 维持数据完整性：SQL使用户可以定义约束规则，定义的规则将存在于数据库内部，可以防止因数据库更新过程中的意外或系统错误而导致的数据库崩溃。
- 修改数据库结构：SQL使用户或应用程序可以修改数据库的结构。

SQL是一种易于理解的语言，同时又是综合管理数据的工具。作为现在数据库市场普遍应用的语言，它具有以下一些特性。

- 确定的标准：美国国家标准协会（ANSI）和国际标准化组织（ISO）在1986年制订了SQL的标准，并在1989年、1992年与1999年三次进行了扩展，使得所有厂商都可以按照统一标准实现对SQL的支持，SQL语言在数据库厂家之间具有广泛的适用性。虽然在不同厂家之间SQL语言的实现方式存在某些差异，但是通常情况下无论选择何种数据库平台，SQL语言都保持相同。
- 软件提供商的独立性：所有主流的DBMS软件提供商均提供对SQL的支持，SQL标准的确立使不同的厂商可以独立地进行DBMS软件的设计，查询、报表生成器等数据库工具能在许多不同类型的SQL数据库中使用。
- 各大公司的支持：SQL由IBM研究人员发明，然后得到了Microsoft公司、Oracle公司等数据库市场各大软件公司的支持，保证了SQL今后的发展。
- 数据的多视化：可以通过使用SQL产生不同的报表和视图，将数据库中数据从用户

所需的角度显示出来供用户使用，具有很大的灵活性。同时，SQL的视图功能也能提高数据库的安全性，并且能满足特定用户的需要。

- 程序化数据库访问：SQL语句既能用于交互式访问也能用于程序化访问，这样应用程序就具有很大的灵活性，可以将这两种方式结合起来设计更好的程序。
- 可移植性：基于SQL的数据库产品能在不同计算机上运行，也支持在不同的操作系统上运行，还可以通过网络进行访问和管理。
- 应用程序的支持：在数据库发展的初期，SQL支持基于主机的应用程序；随着计算机技术的发展，客户机/服务器体系结构出现，SQL使每个系统处于最佳工作状态；而Internet和WWW的迅速发展，建立了以SQL作为应用程序和数据库连接的标准；Java的出现也引入了SQL，SQL在最新的编程语言中有了有效的应用。
- 可扩展性和对象：面向对象编程技术的兴起，使数据库市场也面临对象技术的引入，各个SQL数据库厂商正在扩展和提高SQL对对象的支持。

1.1.3 SQL语句的结构

所有的SQL语句均有自己的格式，每条SQL语句由一个谓词开始，谓词描述该语句要产生的动作。因此，SQL语句可以被划分为3个部分：SQL操作、目标和条件。SQL操作和目标是必要的，至于条件，则根据执行的SQL操作决定其是否可选。

1. SQL操作

SQL操作语句有4种基本操作：SELECT、INSERT、UPDATE、DELETE。每一个操作都是SQL语句名。这些操作语句在后面的章节将分别进行介绍。

2. 目标

所有的SQL操作语句都在一个或多个数据库表或视图上进行操作。目标组件的目的是定义那些表或视图。根据使用的语句不同，该组件也不同，例如，SELECT和DELETE语句具有相似的目标结构，而INSERT和UPDATE语句则具有不同的目标结构。

3. 条件

使用WHERE条件语句说明的条件定义了被SELECT、UPDATE或DELETE语句操作的特定行，而它最终是求得每行数据的TRUE或FALSE值，而且该行为控制了该操作是否发生在每一行上。

1.1.4 SQL的未来

以访问二维表数据为主的SQL和XML的结合成了SQL的一个发展方向。SQL和XML结合的第一步是将关系型数据作为XML格式发布。XML发布是合乎逻辑的起点，因为它可以容易地在XML中代表SQL结果集合，因为很多的动态网页都是由SQL查询来提供的。传统的方法要求用程序访问结果集合和用程序构建网页。新方法以完全公布的方式制作动态网页，利用SQL-to-XML查询生成数据的XML表示，并利用XSLT（可扩展样式表语言转换）

将XML融入到HTML中。

最初这些虚拟文档是利用专有的SQL扩展来创建的。现在有了一种叫做SQL/XML的新ISO/ANSI标准,这项标准定义了一种通用的方法。目前,SQL/XML得到了Oracle和DB2的支持。它定义了用于这些产品中的本机XML数据类型的面向XML的运算符。不过,SQL Server现在还不支持XML数据类型或SQL/XML扩展。

1.2 常 量

常量也称为字面值或标量值,是表示一个特定数据值的符号。常量的格式取决于它所表示的值的数据类型。同其他编程语言一样,SQL也提供了对常量的支持,以方便用户更好更灵活地使用SQL语句。SQL支持数字、字符串、时间和日期、符号常量等4种常量。

1.2.1 数字常量

整数和浮点数类型的数据都可以作为常量使用。整形常量由没有用引号括起来且不含小数点的一串数字表示,例如11111。在Danwei数据库中,如果想列出编号在15以上的员工的名字,可以使用下面的SQL语句:

```
SELECT name
FROM Danwei
WHERE id > 15
```

在SQL中,常量前面也可以加上加号或减号。如+700, -100等。

对于浮点数据类型的常量,通常规定使用符号E,如6.25E5, +6.25E5, 6.05E-7。其中,E代表“乘以10的几次方”。所以,6.25E5就代表常量 6.25×10^5 。

1.2.2 字符串常量

字符串常量括在单引号内并包含字母数字字符(a~z、A~Z和0~9)以及特殊字符,如感叹号(!)、at符(@)和数字号(#)。字符串常量的引入,大大方便了人们使用SQL语句,如查询、添加等操作。例如,在Danwei数据库中,查找名字叫作“王五”的工作人员的具体情况,可以使用下面的SQL语句:

```
SELECT *
FROM Danwei
WHERE name = '王五'
```

SQL标准中规定,字符串常量要包括在单引号中,例如'王五'、'张三'、'李四',如果字符串常量中包括单引号,在其之前再加上一个单引号,以表示其是字符串常量中的一个字符,如:'I don't know'。

注意: 如果单引号中的字符串包含一个嵌入的引号,可以使用两个单引号表示嵌入的单引号。对于嵌入在双引号中的字符串则没有必要这样做。

空字符串用中间没有任何字符的两个单引号来表示。

1.2.3 时间和日期常量

在使用时，也要将时间和日期常量用单引号括起来。例如：'04/15/99'（日期常量），'15:30:20'（时间常量）。如果要在Danwei数据库中查询启动时间早于2006年1月1日的所有工程的信息，可以使用下面的SQL语句：

```
SELECT *
FROM Danwei
WHERE gongcheng < TO_DATE('january 01 2006', MONTH DD YYYY)
```

该例中使用了Oracle数据库系统中的TO_DATE函数，来将一个字符串常量转换为Oracle的内部数据格式。大多数数据库系统都提供了时间和日期的转换函数，以使系统中时间和日期的格式得以统一。

注意：通常时间和日期的使用都必须结合转换函数一起使用，以保证进行操作时，时间和日期的格式是相同的。

1.2.4 符号常量

除了上面3种常量外，SQL语言还包含了许多特殊的符号常量，如CURRENT_DATE、USER、SYSTEM_USER、SESSION_USER等等。这些都是在当前数据库系统中使用得比较多且很有用的符号常量。例如，在Danwei数据库中，查询当前正在进行的项目的信息，可以使用下面的SQL语句：

```
SELECT *
FROM Danwei
WHERE gongcheng < CURRENT_DATE
```

符号常量CURRENT_DATE能生成当前日期的值。

注意：在某些数据库产品中，是通过内嵌函数而不是符号常量来访问系统值，如在Sybase中是通过GETDATE()函数。

1.3 数据类型

如果您曾经使用过某种编程语言，那么一定对数据类型的概念很熟悉。数据类型是数据的一种属性，代表数据所表示信息的类型。

关系数据库提供了很大范围的数据类型供用户使用，包括字符串数据类型、数字数据类型、日期时间数据类型及大型对象等。关系数据库中的字符串数据类型基本上也可以拥有其他类型的数据，但这些数据只是作为字符串来对待的。