

第一章 ORACLE 关系数据库系统概述

数据库技术是计算机领域的一个重要分支,是人们有效地进行数据存储、共享和处理的工具。时至今日,数据库技术的发展已近三十年了。早期的数据库管理系统(DBMS)大多是层次型和网状型的,1970年IBM公司的E.F.Codd提出了关系数据模型,引进了规范化理论,为关系数据库的全面发展奠定了理论基础。此后,关系数据库技术无论在理论上还是实践上都取得了巨大进展。

和层次型、网状型的数据库相比,关系数据库具有简洁明了的数据结构,采用非过程化的用户语言,数据的独立性高且具备坚实的理论基础。因此,关系数据库技术已成为当今数据库技术的主流。

在关系数据库中,数据采用“关系”,即二维表的组织方式,一个表可以存放数据本身,也可以存放数据之间的关系。一个数据库可由若干个表组成,用户对数据库的操作也就是对表的操作。

ORACLE 系统是美国 ORACLE 公司推出的一个关系数据库管理系统。从 1979 年开始,ORACLE 系统经过十多年的发展,其功能和性能得到了不断的完善,已成为当今功能最齐全、最受欢迎的关系数据库管理系统之一。

1.1 ORACLE 系统的特点

作为一个关系数据库管理系统,ORACLE 以其卓越的性能为广大用户所接受,以下是 ORACLE 系统的几个显著特点。

1. 高性能的 RDBMS

ORACLE 系统的核心是 ORACLE RDBMS,它采用 ANSI 标

准的 SQL 语言作为核心部分,并在实现中引入多种优化技术,从而使 ORACLE RDBMS 成为一个功能完善、使用方便的高性能的关系数据库管理系统。

2. 硬件独立性

ORACLE 系统不依赖特定的硬件环境,它可以运行在多种类型的计算机上,其中包括:IBM 微型机,各种 VAX 机,IBM 大型机和中型机等等。由于它的硬件独立性,用户可在不同机型上以相同的方法使用 ORACLE,用户应用程序的移植也变得非常容易。

3. 兼容性

由于 ORACLE 系统采用 SQL 语言并独立于硬件,从而使它很容易实现与其它系统的兼容。目前,ORACLE RDBMS 可与 IBM 的 SQL/DS 和 DB2 完全兼容。

4. 完善的开发环境

ORACLE 除为程序员提供了两种编程接口:预编译程序接口和子程序调用接口以外,还提供多种完善的软件开发工具,它们为各种应用开发建立了有效的支持环境。这些工具包括:应用生成器、报表生成、菜单管理、电子表格等等。

5. 网络通信及分布操作

ORACLE 系统具有网络通信能力,并适用于多种网络协议,它使在相同或不同机型上的用户可以共享数据,用户可以非常方便地进行分布操作和远程存取。ORACLE 系统的 SQL * STAR 产品成功地实现了一个分布式数据库系统。

1.2 ORACLE 系统的组成

ORACLE 系统的核心是 ORACLE RDBMS,并包含一系列的软件工具和实用程序。图 1.1 是其产品结构。

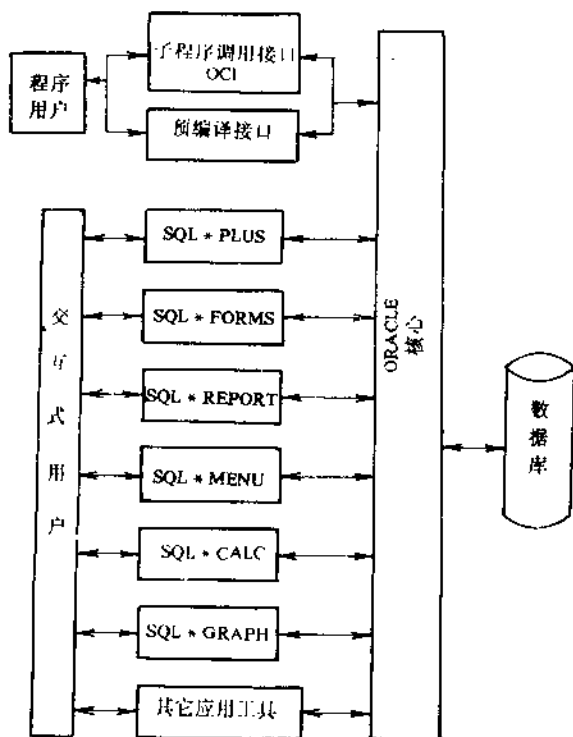


图 1.1 ORACLE 产品结构

ORACLE 产品包括"EASY"产品、"SQL"产品和"PRO"产品，其中"SQL"产品是 ORACLE 的主要产品。ORACLE 产品与 ORACLE RDBMS 一起使用。当然，并不是所有产品对用户来说都是必须的，可以有选择地按照需要和环境进行安装，这也是 ORACLE 系统的优点之一。

1. ORACLE RDBMS

ORACLE RDBMS 由一个核心 RDBMS 和一组实用程序所组成。RDBMS 的功能是执行 SQL 操作，完成对数据库的存取和管理，提供安全性、完整性和并发控制；实用程序帮助用户和系统管

理员(DBA)维护、监视和使用数据库,这些实用程序主要包括:

- IOR 实用程序 用于启动、停止及初始化 ORACLE 系统。
- SGI 实用程序 用于估算 ORACLE 所使用的共享内存。
- ODS 实用程序 用于监视用户及 ORACLE 进程。
- AIJ 实用程序 用于登录对数据库进行的修改。当发生磁盘故障时,用它恢复数据库。

• Export/Import 实用程序 用于将 ORACLE 数据卸出到某文件中(或从某文件移到库内),然后利用该文件进行归档或在 ORACLE 数据库与操作系统之间传送数据。

• ODL 实用程序 用于把标准系统文件装入到 ORACLE 数据库中。

2. "EASY"ORACLE 产品

EASY * 产品是全屏幕产品,供初学者使用,用户可根据菜单提示并利用 Help 信息进行操作。目前只有 EASY * SQL。

3. "SQL"ORACLE 产品

SQL 产品是 ORACLE 系列的主要产品,它给用户提供了存取和处理数据库数据的多种途径。该产品系列是对熟悉 ORACLE 系统使用的 SQL 语言和具有软件开发经验的用户提供服务的,这些产品主要包括:

(1)SQL * PLUS

这是 RDBMS 交互式语言接口,提供增强的 SQL 语言环境,利用它可进行数据定义、检索和更新,进行存取控制,生成格式报表,以及调用操作系统命令等多种操作。

(2)SQL * FORMS

这是基于 FORM(表格)的快速应用开发工具。用户通过菜单选择、屏幕定义即可完成应用开发。通过已建立的 FORM,对数据库进行简洁直观的查询、插入、修改和删除等操作,它是支持原型法应用开发的强有力工具。此外,SQL * FORMS 还提供触发器(Trigger)机制,为维护数据完整性,进行数据检验提供了有力的

手段。

(3)SQL * REPORT

这是数据和正文格式化报表处理工具,包括报表生成和格式化处理两部分。用户可以使用 SQL 语言检索数据库,并将查询结果和附加的正文、标题等信息合并,使用包括定界、页号、多格式窗口以及嵌套等功能,在终端或打印机上输出格式化的报表。

(4)SQL * MENU

这是一个动态菜单生成和管理工具,各类应用软件和操作系统命令的执行均可由此提供统一的接口。其功能包括菜单信息维护,菜单文本生成和系统维护。

(5)SQL * CALC

这是一种新型数据库电子表格接口,通过它能够快速存取数据库中的数据,能够象普通电子表格那样进行数据的分析和预测工作。它将 SQL 语言作为普通公式一样看待,可以被拷贝、移动、保存和编辑。另外,SQL * CALC 与 Lotus1-2-3 兼容,具有从 Lotus1-2-3 到 SQL * CALC 的转换接口。

(6)SQL GRAPH

这是电子图形软件,具有交互式图形编辑功能。可以根据检索所得数据生成彩色的圆饼图、直方图和折线图,并且可以改变图形的颜色、形状和位置。SQL * GRAPH 支持多种图形终端、绘图仪和打印机。

(7)SQL * NET

这是分布式体系结构 SQL * STAR 的产品之一。SQL * NET 提供分布式处理功能,这使得数据库应用软件可以在没有驻留 ORACLE 内核的计算机上运行。SQL * NET 可与多种通信协议配合,用户可将 SQL * NET 用在多种网络环境中。

4. "PRO"ORACLE 产品

PRO 产品是为专业程序设计人员提供的高级语言调用接口,这组编程接口允许程序员在高级语言中利用 ORACLE 数据来进

行开发。对不同的操作系统,ORACLE 分别支持不同的语言,主要的有:PRO * C、PRO * COBOL、PRO * FORTRAN 等。

1.3 ORACLE 系统的安装和运行

作为一个示例,本节介绍 ORACLE5.1 版 RDBMS 及其工具软件在 PC-DOS 或 MS-DOS 下进行安装的基本操作,关于其它版本或在其它操作系统下的安装将会有所不同,具体安装时请参阅有关资料。

1. ORACLE RDBMS 的环境要求

- IBM-PC AT 或 100% 的兼容微型计算机。
- 640KB 基本内存 + 至少 896KB 扩展内存。
- 一个软盘驱动器。
- 一个硬盘,用于安装 ORACLE,至少 7.5M 字节的可用空间。
- 彩色或黑白监视器。
- 3.0 以上版本的 PC-DOS 或 MS-DOS。

2. ORACLE 系统安装

微机 5.1 版 ORACLE 的安装由 ORAINST 程序完成,此程序初始化 ORACLE 系统,为运行 ORACLE 作准备,并负责拷贝文件。

整个安装过程分为两个阶段:系统准备阶段和 ORACLE 软件安装阶段。

系统准备阶段为 ORACLE RDBMS 及其工具软件的安装作准备,它将修改或允许用户修改 AUTOEXEC.BAT 自动执行批文件和 CONFIG.SYS 系统设置文件,使得 ANSI.SYS 可访问。准备阶段的这些工作仅在第一次安装 ORACLE 时才进行。ORACLE 软件安装阶段的任务是安装 ORACLE RDBMS 和各种工具软件。

ORACLE 系统的安装在根目录建立一个名为 ORACLE5 的子目录,其目录结构如图 1.2 所示。

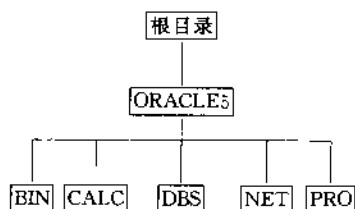


图 1.2 5.1 版 ORACLE 产品目录层次

其中:

- BIN 子目录 包括 ORACLE 及 ORACLE 工具的可执行程序及批文件。
- CALC 子目录 包括 SQL * CALC 使用的文件。
- DBS 子目录 包括存放 ORACLE 数据库的 DBS 文件及若干类型的数据文件和参数文件。
- NET 子目录 包括 SQL * NET 使用的文件。
- PRO 子目录 包括 ORACLE 程序设计语言接口所使用的文件。

下面是微机 5.1 版 ORACLE 系统的安装过程(在安装之前最好将 ORACLE 系统原盘作一全备份)。

(1) 系统准备阶段

1) 设置当前驱动器 ORACLE 系统将被安装在当前驱动器上。例如,若打算将 ORACLE 系统安装在 C 盘上,则应设置 C 为当前驱动器,可在 A)下键入:

A)C:↓ (↓表示 RETURN 键)

C)

2) 运行 ORAINST 将标有"ORACLE INSTALL"的安装盘插入 A 驱动器,然后键入:

C>A:ORAINST↓

3) 确定 ORACLE 主目录 输入你所选择的 ORACLE 主目

录名。如果直接按回车键,则默认 ORACLE5 作为主目录名。

4)确定引导驱动器 ORAINST 要检测 AUTOEXEC. BAT 和 CONFIG. SYS 是否符合要求,因此它要知道哪个驱动器是操作系统引导驱动器。

5)修改 AUTOEXEC. BAT 文件 运行 ORACLE 必须有一个自动执行批文件 AUTOEXEC. BAT,每次启动 DOS 时,都要在根目录下查找这个文件,并自动执行这个文件中的命令。

根据系统情况,对 AUTOEXEC. BAT 文件作如下修改:

①当 AUTOEXEC. BAT 文件不存在时,自动建立之。

②若 AUTOEXEC. BAT 中没有 PATH 命令,则将下面的命令插入到文件中(若不在 C 盘上安装,应将 C 改为相应的驱动器符):

```
PATH C:\; C:\ORACLE5\BIN
```

若已有 PATH 命令,则在此 PATH 命令中加上一条外部路径 C:\ORACLE5\BIN 即可

6)修改 CONFIG. SYS 文件 如果 CONFIG. SYS 文件不存在,则自动建立;如果 CONFIG. SYS 已存在,则需作以下修改:

①若有 FILES 命令,但其值小于 40,,则改为 FILES=40。若没有,则插入 FILES=40。

②若有 BUFFERS 命令,但其值小于 16,则改为 BUFFERS=16。若没有,则插入 BUFFERS=16。

③若有 BREAK 命令,则使其成为 BREAK=ON;若没有,则插入 BREAK=ON。

④若没有引用 ANSI. SYS 的 DEVICE 命令,则插入 DEVICE=C:\ANSI. SYS(此时根目录下应存在 ANSI. SYS 文件,否则应将该文件拷贝到根目录下)。

7)重新启动 在修改了 AUTOEXEC. BAT 和 CONFIG. SYS 文件之后,要求重新启动 DOS,以使这两个修改后的文件中的命令能够得到执行。

(2) ORACLE 软件安装阶段

1) 运行 ORAINST 经过系统准备阶段的操作, 已经为 ORACLE 系统的安装做好了准备, 这时可再次输入 ORAINST 开始安装:

C>ORAINST ↓

2) 选择安装什么类型的 ORACLE 在如下菜单中选择:

(P)rofessional Oracle

(N)etworkstation Oracle

按 P 键安装单用户 ORACLE, 按 N 键安装网络工作站 ORACLE。

3) 安装 ORACLE 工具软件 可以有选择地安装 ORACLE 的各种工具软件。根据屏幕提示顺序在 A 驱动器中插入软盘。

4) 结束 ORACLE 安装 在安装过程中可随时按下“Q”键结束安装过程。

(3) ORACLE 安装完成后工具软件的安装

第一次安装时不必将 ORACLE 的所有工具软件都装上, 但至少应安装 ORACLE RDBMS 部分, 其他工具软件可在以后需要时随时安装。

在安装其它工具软件时, 不必重新开始整个安装过程, 可按如下步骤进行:

① 启动 ORACLE

C>ORACLE ↓

② 启动安装程序

C>ORAINST ↓

以后各步可仿照安装阶段的步骤进行。

3. ORACLE 系统的运行

(1) 启动 ORACLE

平时打开计算机时, ORACLE 系统是驻留在硬盘上的, 要运行 ORACLE 必须首先装入并启动系统。键入命令: -

C>ORACLE ↓

该命令把 ORACLE RDBMS 装入内存,并自动热启动 ORACLE。此命令执行后,就可运行工具软件。例如,执行 SQL * PLUS 命令可键入:

C>SQLPLUS

在用 "QUIT" 命令退出 SQL * PLUS 之后,返回 DOS 提示符状态。此时可继续运行其他工具软件。

若 ORACLE 系统内核没有退出内存,也可用下面的命令进行热启动:

C>IOR W ↓

(2)退出 ORACLE

使用完 ORACLE 之后,应退出 ORACLE 系统,关闭数据库。可以使用如下命令:

C>IOR S ↓

此命令停止 ORACLE 的运行,但并不将其清出内存。若要把 ORACLE RDBMS 从内存中清除,应使用下面的命令:

C>REMORA ALL ↓

第二章 SQL * PLUS 的使用

2.1 SQL * PLUS 简介

SQL * PLUS 是 ORACLE 用来存储、查询、更新、控制和报告数据库中信息的一个交互式工具,是一种增强型的 SQL 语言。它包括标准 SQL 命令和 SQL * PLUS 命令两部分。SQL 命令主要用来对数据库中的数据进行操作,而 SQL * PLUS 命令主要用来控制查询结果的输出形式、提供命令编辑功能和 HELP 信息等。

2.1.1 SQL 语言

SQL (Structure Query Language) 是目前 ANSI 的标准数据语言。它的前身是 1974 年的 SEQUEL (Structure English Query Language) 语言,1976 年经过进一步的修改形成了目前的 SQL。1979 年 ORACLE 公司推出其第一个商品化的关系数据库管理系统——ORACLE 系统就是一个基于 SQL 的 RDBMS,1982 年和 1988 年 IBM 公司相继推出的 SQL/DS 和 DB2 也都采用了 SQL 作为其数据语言。1986 年美国国家标准化组织 ANSI 确认 SQL 作为数据库系统的工业标准。

SQL 语言从它一问世就受到人们的普遍重视和欢迎,现在已有一百多个数据库管理产品支持 SQL 语言,它已能在微机、小型机乃至大型机等各种机型上运行。

1. SQL 语言特点

(1) 非过程化程度高

传统的 FORTRAN、COBOL、C 等语言都是所谓过程化语言,用这种语言编写程序时,在程序中不仅要指出“干什么”,还要清楚地表达“怎么干”。而 SQL 语言是一种非过程化语言,用户只需在

程序中指出要干什么,至于如何干,用户不必在程序中指出,这由系统决定和完成。由于 SQL 语言的这种特点,使得 SQL 语言使用起来相当方便,程序的编写也相当简单。

(2) 用户性能好

SQL 语言在结构上很类似英语,是一种用户性能很好的语言,非常便于学习和掌握。

(3) 语言功能强

SQL 语言具有很强的数据定义和操作能力,丰富和完备的数据处理能力,它结合了关系代数式语言和关系演算式语言的优点,具有很强的功能。

(4) 提供有“视图”数据结构

SQL 语言中用户可对基本关系,即“表”进行操作,也可以对“视图”进行操作。所谓“视图”是指由一个或几个基本关系导出的虚拟关系。视图的概念提高了数据的独立性,有利于数据的安全保密。

2. SQL 语言分类

(1) 数据定义命令

用于定义和修改数据库中的各种对象,这些命令包括:ALTER、COMMENT、CREATE、DROP、RENAME。

(2) 数据操作命令

用于完成检索和更新操作,这些命令包括:SELECT、INSERT、UPDATE、DELETE、VALIDATE。

(3) 数据控制命令

用于控制和管理数据库的使用,这些命令包括:AUDIT、COMMIT、GRANT、LOCK、NOAUDIT、REVOKE、ROLLBACK。

2.1.2 SQL * PLUS 初步

1. SQL * PLUS 的进入和退出

启动 ORACLE RDBMS 之后,在操作系统提示符 C) 下键入

命令 SQLPLUS,并按照系统提示输入用户名和口令:

```
C>SQLPLUS ↓  
Enter username:jfj ↓  
Enter password:jx ↓
```

其中“jfj”为某用户的用户名,“jx”为该用户的口令字。注意,为保密起见,口令是不在屏幕上显示的。另外,用户名和口令应在使用之前由数据库管理员先行建立。

如果用户名和口令字正确,则系统进入 SQL * PLUS,并显示 SQL * PLUS 提示符:

```
SQL>  
此时即可输入 SQL 命令和 SQL * PLUS 命令了。  
退出 SQL * PLUS,只要键入 EXIT 即可:  
SQL>EXIT ↓  
C>
```

2. SQL 和 SQL * PLUS 命令的输入

SQL * PLUS 接受两种类型的命令:SQL 命令和 SQL * PLUS 命令。

在输入 SQL 命令时,应遵循下面的语法规则:

- ①所有 SQL 命令都必须以分号(;)结束。
- ②SQL 命令可以在一行,也可分为多行输入,只要不把一个独立的单词分开。分行时只要在需要分行的地方按回车键即可,系统将在下一行的开始自动出现一个行号。
- ③命令使用大写和小写均可,各单词之间用空格(或制表符)分开。

SQL * PLUS 命令的输入与 SQL 命令略有差别。

SQL * PLUS 命令不必以分号结束,通常应将一条 SQL * PLUS 命令在一行上输入,按回车键表示输入完成。如果此命令一行写不下,在此行的末尾,光标会自动移到下一行,可继续输入,直到完成按回车键即可。

3. SQL * PLUS 编辑命令

(1) SQL 缓冲区

SQL * PLUS 提供了一个内部缓冲区,用于保存当前最近使用的一条 SQL 命令,称为 SQL 缓冲区。

SQL 缓冲区可能包含一条 SQL 语句的若干行,其中最后一行被设置为当前行,用户可以利用 SQL 缓冲区编辑和重新运行上一次的 SQL 命令。

注意 SQL 缓冲区不能保存 SQL * PLUS 命令。

(2) 编辑命令

SQL * PLUS 提供了以下几个用于编辑 SQL 缓冲区的命令。

编辑命令	意义
L(ist)	显示 SQL 缓冲区所有行
L(ist) n	显示缓冲区第 n 行,并置其为当前行
A(ppend) text	将 text 添加在缓冲区当前行后
I(nput) text	将 text 作为一行,插入到缓冲区当前行之后
C(hange)/old/new	对缓冲区当前行,用 new 替换 old
DEL	删除缓冲区当前行
CL(ear) BUFF(er)	清除缓冲区
R(un)	显示并执行缓冲区中的 SQL 命令

注意 上面的 SQL * PLUS 命令,可以使用缩写,即只写大写字头,例如:R,L 等。也可以使用全称,即写全整个命令。

4. 帮助信息

SQL * PLUS 提供一个 HELP 命令,使用户可能随时用该命令获取 SQL * PLUS 命令的语法、含义和基本用法。

如果要获取所有有关 SQL 语句和 SQL * PLUS 命令的信息,只需输入:

SQL)HELP ↓

如果只希望获得某一命令的信息,可在 HELP 之后输入该命令或语句的名字。例如,若要获取 SELECT 命令的信息,可以输

入:

SQL>HELP SELECT ↓

2.2 SQL * PLUS 的简单使用

为使初学者对用 SQL 语言能够完成什么工作以及如何完成有一个基本的概念并获得一定的感性认识,本节用几个基本的 SQL 语句,并从用户角度和实际应用的步骤来介绍对一个表进行处理的过程。对本节用到的 SQL 语句将不作详细说明,读者只要能够理解就行,详细完整的介绍将在以后相应章节给出。

1. 样本数据库

为便于讨论,我们虚设了一个样本数据库,它由教员(TEAC)表、学员(STUD)表、课程(COUR)表、成绩(GRAD)表组成。本书以后的章节将围绕这个样本数据库的四个样表(表 2.1、2.2、2.3、2.4)举例。以下是它们的结构和内容。

表 2.1 教员(TEAC)表

TNO (编号)	TNAME (姓名)	TSEX (性别)	TAGE (年龄)	TDATE (工作日期)	TITLE (职称)
T053	张伟	男	54	10-JUN-62	教授
T034	王强	男	36	05-SEP-79	讲师
T036	叶莉	女	48	15-APR-66	副教授
T040	刘志强	男	29	01-SEP-86	讲师
T023	王虹	女	23	07-AUG-92	助教
T024	赵一	男	26	18-MAR-89	讲师
T076	李文华	男	32	26-MAY-82	讲师
T082	宋梅	女	28	06-DEC-86	讲师
T085	张东	男	25	20-SEP-88	助教
T094	孙建	男	34	22-JUN-78	讲师

表 2.2 学员(STUD)表

SNO (学号)	SNAME (姓名)	SSEX (性别)	SAGE (年龄)
92004	陈平	男	22
92005	王波	男	20
92013	张玲	女	20
92014	张东	男	25
92022	李敏	女	19

表 2.3 课程(COUR)表

CNO (课程号)	CNAME (课程名)	HOUR (学时)	CREDIT (学分)
C01	数据结构	80	4
C02	操作系统	60	3
C03	计算机原理	80	4
C04	人工智能	40	2

表 2.4 成绩(GRAD)表

SNO (学号)	CNO (课程号)	GRADE (成绩)
92004	C01	85
92004	C03	94
92005	C02	76
92005	C03	82
92013	C01	79
92013	C02	82
92014	C01	90

2. 表的建立

用户在建立自己的数据库时, 首行要创建表结构, SQL 语言用 CREATE 命令建表。下面的命令用来建立教员表 (TEAC) 的表

结构:

SQL)CREATE TABLE TEAC

2 (TNO CHAR(4)NOT NULL,TNAME CHAR(8) NOT NULL,

3 TSEX CHAR(4),TAGE NUMBER(2),TDATE DATE,TITLE
CHAR(6));

在CREATE TABLE 命令中,需要给表命名(TEAC)和给字段命名(如:TNO,TAGE 等),还需指定字段所含信息的类型 (CHAR 代表字符型,NUMBER 代表数字型,DATE 代表日期型),最后指定每个字段存储的值的最大宽度。例如,职称字段 TITLE CHAR(6)表示:TITLE 是一个字符型字段,其值的最大长度是 6 个字符。(命令的 NOT NULL 表示此列不能含有空值)。

下面给出样本数据库的四个样表的数据结构。

TEAC 表

字段名	TNO	TNAME	TSEX	TAGE	TDATE	TITLE
类型长度	CHAR(4)	CHAR(8)	CHAR(2)	NUMBER(2)	DATE	CHAR(6)

STUD 表

字段名	SNO	SNAME	SSEX	SAGE
类型长度	CHAR(5)	CHAR(8)	CHAR(2)	NUMBER(2)

COUR 表

字段名	CNO	CNAME	HOUR	CREDIT
类型长度	CHAR(3)	CHAR(10)	NUMBER(3)	NUMBER(1)

GRAD 表

字段名	SNO	CNO	GRADE
类型长度	CHAR(5)	CHAR(3)	NUMBER(6,2)