

ORACLE 数据库系统

——基础与工具

李文生 冯凤娟 霍宇红 施海舟 等编著

人民邮电出版社

TP3

内 容 简 介

本书集中详细地介绍了ORACLE数据库的基础与工具, 主要内容包括: ORACLE数据库查询语言SQL*Plus3.1, 过程化SQL语言PL/SQL及其应用, 与C语言的接口工具Pro*C, 最新表格生成工具ORACLE*FORMS4.5, 报表生成工具ORACLE*REPORTS2.5, 图形生成工具ORACLE*GRAPHICS2.5, 以及各种开发工具之间的集成方法等。

本书不仅可以作为ORACLE数据库的教材, 也可以作为ORACLE数据库应用系统开发人员的详细参考资料, 还可以作为ORACLE数据库管理与维护人员的参考书。

ORACLE数据库系统——基础与工具

- ◆ 编 著 李文生 冯凤娟 霍宇红 施海舟 等
责任编辑 刘兴航
- ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
北京朝阳隆昌印刷厂印刷
新华书店总店北京发行所经销
- ◆ 开本: 787×1092 1/16
印张: 52
字数: 1309千字 1999年3月第1版
印数: 1-3000册 1999年3月北京第1次印刷
ISBN 7-115-07619-7/TP·988

定价: 86.00元

目 录

第一章 SQL*PLUS.....	1
第一节 SQL*PLUS 概述	1
一、 启动和退出 SQL*Plus	1
二、 SQL*Plus 环境下, 可以执行的命令	2
三、 SQL*Plus 的限制	4
四、 SQL 缓冲区及 SQL 命令的编辑	5
五、 本章使用的基表和视图	7
六、 语法格式定义中所用符号说明	8
第二节 SQL 语言概述	8
一、 SQL 语言的特点	9
二、 SQL 命令中使用的运算符、函数、表达式和条件	10
三、 SQL 语言命令的分类	27
第三节 数据定义语言命令	28
一、 创建、更新和删除操作对象的命令	31
二、 授权和收回特权的命令	48
第四节 数据操纵语言命令	58
一、 插入命令	58
二、 查询命令	60
三、 修改命令	74
四、 删除命令	77
第五节 事务控制命令	78
一、 提交事务命令(COMMIT)	78
二、 回退事务命令(ROLLBACK)	80
三、 建立保留点命令(SAVEPOINT)	81
四、 创建事务命令(SET TRANSACTION)	81
第六节 对话控制命令	83
一、 ALTER SESSION	83
二、 SET ROLE	85
第七节 系统控制命令	86
一、 ALTER SYSTEM	86
第八节 SQL*PLUS 命令	89
一、 求助命令	89

二、 SQL*Plus 环境设置命令	90
三、 显示 SQL*Plus 系统变量值的命令	93
四、 SQL 缓冲区编辑命令	94
五、 文件操作命令	94
六、 变量的定义和删除	104
七、 格式化查询结果的命令	111
八、 数据库的存取	123
九、 其它的 SQL*Plus 命令	126
第二章 PL/SQL	130
第一节 PL/SQL 概述	130
一、 PL/SQL 的体系结构	130
二、 PL/SQL 的模块结构	131
三、 PL/SQL 的优点	138
第二节 PL/SQL 的语言基础	138
一、 字符集	139
二、 词法单位	139
三、 数据类型	143
四、 标识符的作用域	147
五、 变量的赋值	148
六、 表达式	149
七、 内部函数	153
八、 PL/SQL 表	155
九、 记录	158
第三节 控制结构	161
一、 IF 语句	161
二、 LOOP 和 EXIT 语句	163
三、 GOTO 语句	167
四、 NULL 语句	169
第四节 与 ORACLE 的交互	169
一、 SQL 支持	169
二、 国家语言支持	171
三、 远程访问	172
四、 游标	173
五、 事务处理	180
六、 数据库触发器	185
第五节 错误处理	186
一、 预定义的例外	186

二、 用户定义的例外.....	187
三、 例外的传播规则.....	190
四、 重新提出例外.....	191
五、 例外的处理.....	192
第六节 执行环境.....	194
一、 SQL*Plus 环境.....	195
二、 ORACLE 预编译环境.....	196
三、 OCI 环境.....	197
第三章 ORACLE*FORMS 4.5	198
第一节 ORACLE*FORMS 4.5 的基本概念.....	198
一、 什么是 FORMS?	198
二、 FORM 的结构.....	199
三、 FORM 与数据库的关系	199
第二节 运行 FORM	200
一、 启动 FORM 应用.....	200
二、 运行界面的屏幕结构.....	201
三、 FORM 运行界面中的导航方法	203
四、 FORM 应用的工作模式.....	204
五、 FORM 运行时的各种操作方法	205
六、 缺省菜单和相应的功能键.....	213
七、 FORM 运行时的参数.....	214
八、 FORM 运行的顺序	217
第三节 ORACLE FORMS 工具集.....	218
一、 工具集的组成.....	218
二、 设计部件(Designer Tool).....	219
三、 设计部件的启动、数据库连接、退出.....	238
四、 模块的创建和打开.....	240
第四节 设计基本 FORMS	243
一、 设计过程.....	243
二、 定义主从块.....	254
三、 屏幕布局器(Layout)的使用.....	259
第五节 改善操作接口	263
一、 文本项(Text Item).....	263
二、 值列表(LOV).....	269
三、 编辑器(Editor).....	275
四、 其它可输入项.....	276
五、 非输入项.....	284

六、 属性的继承.....	291
七、 其它对象.....	294
第六节 窗口和画布(WINDOWS 和 CANVAS).....	296
一、 对象导航器视图.....	296
二、 窗口(Window).....	297
三、 Canvas-View(画布视图).....	299
第七节 触发子	304
一、 触发子概念.....	304
二、 触发子特征.....	305
三、 创建触发子及触发子属性.....	310
四、 触发子编程.....	314
第八节 触发了实例	323
一、 合法性校验.....	324
二、 给输入项增加功能.....	327
三、 给非输入项增加功能.....	329
四、 导航触发子.....	331
五、 事务处理触发子.....	333
六、 查询触发子.....	337
七、 鼠标触发子.....	340
八、 显示信息处理与报警器.....	341
第九节 设计菜单模块	344
一、 菜单的概念和结构.....	344
二、 定义菜单模块.....	347
三、 保存和生存菜单模块.....	350
四、 将菜单模块连到 Form 模块.....	351
第十节 关于用户出口和多个 FORM 应用.....	351
一、 Form 集成的概念.....	352
二、 Form 间的连接.....	353
三、 启动一个新 Form.....	353
四、 定义多个 Form 功能.....	355
第十一节 代码和实体的共享	361
一、 共享实体(对象).....	361
二、 共享 PL/SQL 代码.....	365
第四章 PRO*C	368
第一节 概述	368
一、 Pro*C 的优点.....	368
二、 Pro*C 程序的结构.....	368

三、 Pro*c 程序的处理过程.....	370
四、 Pro*C 编程基本知识.....	371
第二节 程序首部.....	376
一、 DECLARE 节.....	376
二、 SQL 通讯区(SQLCA).....	379
三、 INCLUDE 语句.....	380
四、 与 ORACLE 连接.....	381
第三节 程序体.....	383
一、 可用的 SQL 命令.....	384
二、 主变量的使用.....	384
三、 指示变量的使用.....	385
四、 VARCHAR 变量的使用.....	386
第四节 数据更新操作.....	387
一、 INSERT 语句.....	387
二、 UPDATE 语句.....	388
三、 DELETE 语句.....	389
第五节 数据查询.....	390
一、 嵌入式 SELECT 语句的语法格式.....	390
二、 单行查询.....	391
三、 多行查询.....	391
四、 游标的使用.....	392
第六节 事务管理.....	394
一、 COMMIT.....	395
二、 SAVEPOINT.....	395
三、 ROLLBACK.....	396
四、 SET TRANSACTION.....	397
五、 对数据的锁定.....	397
第七节 错误检测.....	400
一、 使用 SQLCA.....	400
二、 使用 WHENEVER 语句.....	404
三、 使用 ORACA.....	407
第八节 动态 SQL 语句.....	412
一、 动态语句方式 1.....	413
二、 动态语句方式 2.....	416
三、 动态语句方式 3.....	419
四、 动态语句方式 4.....	421
第九节 使用主数组.....	427

一、主数组的声明.....	427
二、使用主数组的使用.....	427
三、使用指示数组.....	431
四、FOR 子句的使用.....	432
五、WHERE 子句的使用.....	432
六、模仿 CURRENT OF 子句.....	433
第十节 嵌入 PL/SQL 程序.....	434
一、嵌入 PL/SQL 块.....	434
二、使用主变量.....	435
三、使用指示变量.....	436
四、使用主数组.....	437
五、存储子程序.....	439
第十一节 预编译器的使用.....	441
一、预编译命令.....	441
二、选项的作用域.....	443
三、输入选项.....	444
四、条件预编译.....	444
五、独立预编译.....	445
六、编译和连接.....	446
第五章 ORACLE*REPORTS 2.5.....	447
第一节 基本概念.....	447
一、ORACLE*Reports 2.5 的概念、功能和特点.....	447
二、ORACLE*Reports 2.5 的组成.....	449
三、设计 Report 的一般过程.....	457
第二节 对象导航器.....	459
一、结构.....	459
二、模块的建立和存贮.....	465
三、报表的对象.....	468
第三节 数据模板.....	478
一、查询.....	479
二、组.....	486
三、列.....	491
四、建立连接(Data link).....	500
五、参数.....	504
第四节 布局编辑器.....	511
一、页结构和报表的组成.....	511
二、缺省布局.....	512

三、 布局编辑器的工具条和菜单.....	518
四、 布局对象.....	522
五、 参数表(Parameter Form).....	561
第五节 设计报表	566
一、 建立缺省报表.....	566
二、 建立分组报表.....	576
三、 在报表中加入图文文本及画线.....	589
四、 多媒体报表.....	594
五、 报表类型的组合.....	601
六、 各种矩阵报表.....	606
第六节 ORACLE*REPORTS 触发子	628
一、 PL/SQL 的概述.....	628
二、 外部 PL/SQL 库.....	635
三、 ORACLE*Reports 触发子.....	637
四、 应用实例.....	648
第七节 SRW 包	657
一、 SRW 包.....	657
二、 SRW 包的应用实例.....	683
第六章 ORACLE*GRAPHICS 2.5.....	689
第一节 ORACLE*GRAPHICS 2.5 简介	689
一、 ORACLE*Graphics 的特点.....	689
二、 ORACLE*Graphics 的对象.....	690
三、 图表的类型.....	691
四、 创建图表的步骤.....	692
五、 存储管理.....	693
六、 运行方法.....	695
七、 产品部件.....	695
第二节 ORACLE*GRAPHICS 的设计界面	697
一、 启动和关闭.....	697
二、 构成 ORACLE*Graphics 设计界面的窗口.....	698
三、 对象导航器.....	699
四、 屏幕布局编辑器.....	700
五、 设计菜单.....	703
六、 环境参数设置.....	707
第三节 创建图表应用	710
一、 创建图表的过程.....	710
二、 图表的组成和属性.....	714

三、 图表框架的属性.....	719
四、 坐标轴的属性.....	724
五、 图表元素的属性.....	730
六、 创建主从关系的图表.....	732
七、 域模板的属性.....	736
八、 图表属性共享.....	742
九、 显示层.....	742
第四节 ORACLE*GRAPHICS 触发子.....	743
一、 PL/SQL 概述.....	743
二、 ORACLE*Graphics 的内置数据类型.....	745
三、 ORACLE*Graphics 的内置全局变量.....	748
四、 ORACLE*Graphics 的内部子程序.....	748
五、 触发子.....	763
六、 调试 PL/SQL 代码.....	777
第七章 工具的集成.....	779
第一节 ORACLE FORMS 和其它工具的集成.....	779
一、 集成的一般方法.....	779
二、 从 ORACLE FORMS 中调用 ORACLE REPORTS.....	781
三、 从 ORACLE FORMS 中调用 ORACLE BOOK.....	784
四、 从 ORACLE FORMS 中调用 ORACLE GRAPHICS.....	784
五、 从 C 语言中调用 ORACLE FORMS.....	788
六、 OG 包.....	789
第二节 从 ORACLE GRAPHICS 中调用其它工具.....	793
一、 调用的一般方法.....	793
二、 调用 Oracle Forms.....	796
三、 调用 Oracle Reports.....	797
四、 调用 Oracle BOOK.....	798
第三节 与非 ORACLE 数据库连接.....	798
一、 开放式网关.....	799
二、 事务型触发子器.....	802
三、 Oracle OCA 和 ODBC.....	816

第一章 SQL*Plus

这一章，我们以 SQL*Plus3.1 版本为例来介绍 SQL 语言以及用于生成报表的 SQL*Plus 命令。

第一节 SQL*Plus 概述

SQL*Plus 是 ORACLE 数据库软件的主要产品，是 ORACLE 系统提供的一种用户接口。用户可以在 SQL*Plus 环境下使用 SQL 命令交互式地访问数据库，也可以使用 SQL*Plus 命令生成报表。

一、启动和退出 SQL*Plus

1. 启动 SQL*Plus

如果 SQL*Plus 是运行在操作系统环境下，则在操作系统提示符后面输入如下命令即可：
SQLPLUS (回车)

这时，系统将显示其版本号、日期和版权信息，并提示输入用户名和口令字(口令字不在屏幕上显示)，如果连续三次输入错误，则拒绝进入 SQL*Plus 环境。也可以使用下面简单的方法启动 SQL*Plus：

SQLPLUS 用户名/口令字(回车)

如果 SQL*Plus 是 Windows 环境下的，则在程序管理器中找到并打开 ORACLE 程序组图标，在此组中找到并打开 SQL*Plus3.1 图标，在系统显示其版本号、日期和版权信息后，会弹出连接对话框，让用户输入用户名和口令字，同样，如果连续三次输入错误，则拒绝进入 SQL*Plus 环境。

如果提供的用户名和口令字是合法的，则进入 SQL*Plus 环境，系统显示 SQL*Plus 命令提示符：

SQL>

这表明 SQL*Plus 已经准备好，可以输入命令了。

2. 退出 SQL*Plus

如果想退出 SQL*Plus 环境, 则在 SQL*Plus 提示符后面输入命令 EXIT 即可:

SQL>EXIT(回车)

这样, 就可以返回到操作系统(如果 SQL*Plus 运行在操作系统环境下)或返回到 ORACLE 程序组了(如果 SQL*Plus 运行在 Windows 环境下)。

二、SQL*Plus 环境下, 可以执行的命令

1. SQL 命令

利用 SQL 语言命令可以操纵数据库中的数据。例如, 可以利用 SELECT 语句查询 emp 表中 20 号部门职工的编号、姓名和工资, 命令如下:

SQL> SELECT empno, ename, sal (回车)

2 FROM emp (回车)

3 WHERE empno=20; (回车)

说明:

- 该命令共有 3 行, 第 1 行、第 2 行输入回车后, 出现下一行的行号, 提示用户输入下一行命令, 第 3 行中的分号(;)表示命令结束, 按回车键后, SQL*Plus 将处理该命令, 并把查询结果显示在屏幕上。之后, 再次出现 SQL 命令提示符以接受新的命令。

- SQL 命令可以输入在一行上, 也可以象上面所示输入在多行上;

- 结束 SQL 命令的方法有三种, 即可以用一个分号(;), 可以在单独一行上输入一个反斜杠(/), 也可以用一个空行来结束命令。但这三种方法是有区别的, 前两种方法首先把 SQL 命令存入 SQL 缓冲区, 然后 SQL*Plus 处理缓冲区中的命令; 而后一种方法只是把命令存入缓冲区, 但并不执行。要想执行缓冲区中的命令, 可以在 SQL 命令提示符后输入反斜杠(/), 或输入命令 RUN, 按回车键后 SQL*Plus 就对该命令进行处理。

- 如果用分号结束命令, 则在分号的同一行上不能输入注释(/* */)。

2. PL/SQL 块

除了 SQL 命令外, 还可以使用 PL/SQL 块来操纵数据库中的数据。有两种方法可以进入 PL/SQL 方式:

(1) DECLARE 或 BEGIN 方式

在 SQL*Plus 命令提示符下输入 DECLARE 或 BEGIN 命令, 就可以进入 PL/SQL 方式, 在这种方式下, 我们可以输入 PL/SQL 块的剩余部分, 所建立的是一个无名的 PL/SQL 块。

(2) CREATE 方式

在 SQL*Plus 命令提示符下输入 CREATE 命令, 进入 PL/SQL 方式, 在这种方式下我们可以建立存储子程序。如:

用 CREATE FUNCTION 可以建立 PL/SQL 函数;

用 CREATE PACKAGE 可以建立包说明;

用 CREATE PACKAGE BODY 可以建立包体;

用 CREATE PROCEDURE 可以建立 PL/SQL 过程;

用 CREATE TRIGGER 可以建立触发器等等。

结束 PL/SQL 方式的方法是: 在单独一行上输入一个句点。

SQL*Plus 把在 SQL 命令提示符下输入的 PL/SQL 块或子程序存入缓冲区中, 当用户在 SQL 命令提示符下发出 RUN 命令或反斜杠 (/) 后, 就可以执行当前子程序。有关 PL/SQL 子程序的信息可以参见本书第二章的内容。

3. SQL*Plus 命令

可以利用 SQL*Plus 命令来操作 SQL 命令和 PL/SQL 块, 格式化和打印查询结果。有关 SQL*Plus 的命令及其使用可以参见本章第八节的内容。这里我们通过一个简单的例子来说明 SQL*Plus 命令怎样格式化查询结果。

首先在命令行中输入 SQL*Plus 命令:

```
SQL> COLUMN sal HEADING salary FORMAT $99,999.99
```

输入并运行查询命令:

```
SQL> SELECT empno,ename,sal FROM emp WHERE deptno=20;
```

结果如下:

EMPNO	ENAME	SALARY
7369	SMITH	\$800.00
7566	JONES	\$2,975.00
7788	SCOTT	\$3,000.00
7876	ADAMS	\$1,100.00
7902	FORD	\$3,000.00

从上面的结果可以看出, COLUMN 命令已经将 SAL 列格式化, 给它一个新的标题 SALARY, 并且在数据中使用美元符号 (\$), 逗号 (,) 和句点 (.).

注意: SQL*Plus 只将 SQL 命令存入 SQL 缓冲区中, 并不把 SQL*Plus 命令存入缓冲区。

如果一个 SQL*Plus 命令比较长, 可以输入在连续的多行上。为了换行, 可以在一行的末尾输入一个连字符 (-), 然后按回车键, SQL*Plus 就显示一个大于号作为续行的提示符。

如果要结束一个 SQL*Plus 命令, 只需按回车键即可。当然也可以在命令的末尾输入一

个分号。

4. 操作系统命令

在 SQL*Plus 命令提示符下也可以执行主机的操作系统命令。要想运行一个主机操作系统命令，可以在 SQL*Plus 的 HOST 命令后面跟一个主机操作系统命令，例如：

```
SQL> HOST DIR *.SQL
```

当该主机操作系统命令运行结束后，将再次显示 SQL*Plus 命令提示符。

5. 运行 ORACLE*Forms 的格式文件

如果在 SQL*Plus 安装时使用了 RUNFORM 选项，就可以在 SQL*Plus 的命令提示符下运行 ORACLE*Forms 的格式文件，例如，要运行一个文件名为 ORDER 的格式文件，可以发出如下命令：

```
SQL> RUNFORM ORDER
```

6. 求助命令

使用求助命令可以帮助我们获得所需要的信息。例如：

```
SQL> HELP(回车) /* 这条命令执行的结果将显示所有 SQL 和 SQL*Plus 命令 */
```

```
SQL> HELP SELECT(回车) /* 这条命令将显示指定命令的信息 */
```

```
SQL> DESC EMP(回车) /* 显示基表 EMP 中各列的定义 */
```

```
SQL> DESC proc_name(回车) /* 显示该过程的定义 */
```

```
SQL> SHOW ALL(回车) /* 显示当前 SQL*Plus 环境参数的设置情况 */
```

```
SQL> SHOW PAUSE(回车) /* 显示参数 PAUSE 的值 */
```

三、SQL*Plus 的限制

这里给出了使用 SQL*Plus 时的各种限制，见表 1-1。这些值对大多数操作系统都有效，除了 PDP11 和 MC68000。

表 1-1 SQL*Plus 的限制

文件名长度	与系统有关
用户名长度	30 字节
用户变量名长度	30 字节
用户变量值长度	1024 字节
SQL INSERT 命令的 INTO 表中的变量数	50
每个 SQL 命令的变量数	100
命令行长度	2500 字符
通过 SQL*Plus 输入的 LONG 值的长度	LINESIZE 值

续表 1-1

LINESIZE 值	与系统有关
LONGCHUNKSIZE 值(要求 ORACLE7)	MAXDATA 值
MAXDATA 值	与系统有关
输出行大小	与系统有关
变量替代后的行大小	1000 字符
每个 SQL 命令的行数	500(假定每行 80 字符)
每页行数	50000
总行宽	32767 字符, VMS 为 60000 字符
一次取数组的行数	5000
嵌套的命令文件数	5, VMS、CMS 和 UNIX 为 20
页数	99, 999
PL/SQL 错误消息缓冲区	2K (ORACLE7), 512 (ORACLE6)

四、SQL 缓冲区及 SQL 命令的编辑

1. SQL 缓冲区

当输入 SQL 命令或 PL/SQL 块时, SQL*Plus 将其保存在内部缓冲区中, 这个内部缓冲区称为 SQL 缓冲区。SQL 缓冲区中只能存放一条 SQL 命令, 即当输入一条新的 SQL 命令后, SQL*Plus 将清除原来保存的 SQL 命令, 而把新的 SQL 命令存入缓冲区中。SQL 缓冲区中不能保存 SQL*Plus 命令。我们可以编辑和运行存放在 SQL 缓冲区中的命令而不用重新输入该命令。

2. SQL 命令的编辑

对于存放在 SQL 缓冲区中的 SQL 命令或 PL/SQL 块, 可以用主机操作系统提供的编辑程序进行编辑, 也可以使用 SQL*Plus 命令进行编辑。SQL*Plus 的编辑命令见表 1-2。

表 1-2 SQL*Plus 的编辑命令

命令	缩写	用途
APPEND text	A text	将 text 追加到当前行末尾
CHANGE /old/new/	C /old/new/	将当前行中的字符串 old 修改为 new
CHANGE /text	C /text	从当前行中删除 text
CLEAR BUFFER	CL BUFF	删除全部行, 即把缓冲区清空
DEL	(无)	删除当前行
INPUT	I	在当前行下面增加一行或若干行

续表 1-2

INPUT text	I text	在当前行下面增加由 text 组成的一行
LIST	L	列出 SQL 缓冲区中的所有行
LIST n	L n 或 n	列出指定行
LIST *	L *	列出当前行
LIST LAST	L LAST	列出最后一行
LIST m n	L m n	列出从 m 到 n 的所有行

例如, 输入命令:

```
SQL> SELECT empno, ename, sal
2 FROM emp;
```

(1) 列出缓冲区中的命令

```
SQL> L(回车) /* 列出所有行 */
1 SELECT empno, ename, sal
2* FROM emp /* 这里的星号(*)标识当前行 */
```

```
SQL> L1(回车) /* 列出第一行 */
1* SELECT empno, ename, sal
```

(2) 编辑当前行

① 字符串替代

假如要将第一行中的 ename 修改为 job, 首先用 LIST 命令使第一行成为当前行, 然后发出如下命令即可:

```
SQL> C /ename/ job/ (回车)
1* SELECT empno, job, sal
```

② 将正文加到行尾

假如还要列出 comm 列, 首先使第一行成为当前行, 然后发出如下命令即可:

```
SQL> A , comm(回车)
1* SELECT empno, job, sal, comm
```

(3) 增加新行

如果要查看 20 号部门的职工情况, 就需要再增加一个 WHERE 子句加以限定, 首先用 LIST 命令列出缓冲区中的内容, 使第二行成为当前行, 然后再发出如下命令即可:

```
SQL> INPUT(回车) /* 在下一行上出现一行号, 提示用户输入 */
3i WHERE deptno=20(回车)
4i /* 可以继续输入正文, 也可以直接按回车键退出插入操作 */
```

这时再列出缓冲区中的内容如下:

SQL> LIST(回车)

```
1  SELECT empno, job, sal, comm
2  FROM emp
3* WHERE deptno=20
```

(4) 删除行

如果要删除缓冲区中的某一行，首先使该行成为当前行，然后用 DEL 命令即可删除之。如要删除缓冲区中的第三行，可以发出下面的命令：

```
SQL> L 3(回车)
3* WHERE deptno=20
SQL> DEL(回车)
```

这时原来的第三行就被删除，它的下一行将成为当前行(如果有的话)。

如果要删除缓冲区中的所有行，可以使用 CLEAR BUFFER 命令。如：

```
SQL> CLEAR BUFFER(回车)
```

这条命令执行的结果是清除 SQL 缓冲区中的全部内容。

(5) 系统编辑程序

在主机操作系统中，一般都提供一个或几个正文编辑程序供用户使用。用户可以通过 SQL*Plus 命令 EDIT 进入主机操作系统的缺省正文编辑程序而不必退出 SQL*Plus，如：

```
SQL> EDIT(回车)
```

如果用户不想使用系统缺省的编辑程序，而是使用某一个指定的编辑程序的话，可以用 SQL*Plus 的 DEFINE 命令，定义参数 _EDITOR 的值为该编辑程序的名字。例如：

```
SQL> DEFINE _EDITOR = PE3(回车)
```

这样，当用户使用 EDIT 命令时，就进入 PE3 编辑程序。

五、本章使用的基表和视图

这里给出本章例子中所使用的基表 DEPT 和 EMP。

1. DEPT 表

DEPTNO	DNAME	LOC
10	ACCOUNTING	NEW YORK
20	RESEARCH	DALLAS
30	SALES	CHICAGO
40	OPERATIONS	BOSTON