

第四代语言

INFORMIX-4GL

实用教程

石文昌 陈振坤 孙玉方 编著



清华大学出版社

第四代语言 INFORMIX-4GL 实用教程

石文昌 陈振坤 孙玉方 编著

清华大学出版社

(京)新登字 158 号

IS145/18
内容提要

本书介绍第四代语言 INFORMIX-4GL 应用软件快速开发系统的强大功能及其应用的开发方法。本书风格独特,首先使读者迅速掌握该语言的使用方法,然后对各部分内容逐步深入讨论。主要内容包括:INFORMIX-4GL 应用程序的组织结构,关系数据库定义与操作语言,报表打印方法,条件查询设计工具,输入/输出屏幕设计工具,自动菜单管理工具,联机帮助信息管理工具,多窗口界面设计工具等。

本书可作为学习 INFORMIX-4GL 的教材,也可供计算机应用软件开发人员参考。

版权所有,翻印必究。

本书封面贴有清华大学出版社激光防伪标志,无标志者不得销售。

第四代语言

INFORMIX-4GL 实用教程

石文昌 陈振坤·孙玉方 编著

☆

清华大学出版社出版

北京 清华园

北京昌平环球印刷厂印刷

新华书店总店科技发行所发行

☆

开本: 787×1092 1/16 印张: 13.5 字数: 316 千字

1994 年 4 月第 1 版 1994 年 4 月第 1 次印刷

印数: 0001—6000

ISBN 7-302-01435-3/TP·560

定价: 11.60 元

前 言

目前,第四代语言已成为计算机应用软件开发环境的重要组成部分。

第四代语言是一种非过程化的程序设计语言,它的主要用途是用于管理信息系统的快速开发。第四代语言以功能强大的应用软件开发工具为显著特点,大多数第四代语言又都以一个数据库管理系统(DBMS)作为其不可分割的组成部分。故此,我们可以这样理解,第四代语言就是由一组应用软件开发工具和一个 DBMS 组合成的一个集合体。

使用第四代语言开发应用软件的突出优点是可以极大地提高应用软件的开发效率、减少应用软件的开发开销、增加应用软件调试和维护的容易程度以及增强应用软件修改的灵活性。

INFORMIX-4GL 是美国 Informix Software 公司研制的运行于 UNIX/XENIX 多用户系统上的第四代语言产品,它同时提供非过程化和过程化的结构,它的主要特点在于多窗口管理、多用户终端上显示信息的彩色处理、多页菜单、联机帮助和“所见即所得”的信息录入与显示屏幕格式,它包含一个关系数据库管理系统,它的数据库语言是 SQL 的变种——RDSQL。

本书以独特的风格介绍 INFORMIX-4GL。组织本书内容的指导思想是首先使读者迅速掌握运用 INFORMIX-4GL 开发应用软件的方法,然后逐步对软件开发工具等各方面的内容展开深入的讨论。

全书共分十章和六个附录。第一章对第四代语言和 INFORMIX-4GL 作简要的介绍。第二章用一个简单的程序实例介绍 INFORMIX-4GL 程序的基本结构和程序开发的基本方法。第三章用一个由几个模块构成的简单实例介绍开发数据库应用程序的基本方法。第四章以一个统分系统应用程序为实例介绍各种软件开发工具的基本用法。第五章全面介绍报表打印的方法。第六章全面介绍屏幕格式设计与信息输入/输出的方法。第七章介绍数据库管理方面的内容。第八章对 INFORMIX-4GL 的程序设计语言作较全面的补充介绍。第九章对 INFORMIX-4GL 的数据库语言 RDSQL 作全面的概括并对数据查询等内容进行较深入的讨论。第十章介绍菜单驱动的程序开发环境。附录一介绍 INFORMIX-4GL 使用的环境变量。附录二介绍 INFORMIX-4GL 的库函数。附录三对信息的显示属性和多窗口管理作进一步的介绍。附录四列出前面章节中没有给出的统分系统的两个程序模块。附录五给出系统用于描述数据库信息的几个二维表。附录六按字母顺序给出各种命令的名称及其在书中出现的页码的对照表。

由于我们水平有限,书中缺点和错误在所难免,恳请读者不吝赐教。

编 者

一九九三年九月

目 录

第一章 引论	(1)
1.1 第四代语言	(1)
1.2 INFORMIX-4GL	(2)
1.3 INFORMIX-4GL 的基本概念	(3)
1.3.1 数据库	(3)
1.3.2 屏幕格式	(3)
1.3.3 菜单	(5)
1.3.4 报表	(7)
1.3.5 窗口	(8)
习题 1	(9)
第二章 INFORMIX-4GL 程序	(10)
2.1 INFORMIX-4GL 程序结构	(10)
2.1.1 主程序结构	(10)
2.1.2 变量说明	(11)
2.1.3 输入/输出	(12)
2.1.4 赋值和运算	(13)
2.1.5 CASE 开关语句	(15)
2.1.6 IF 条件语句	(16)
2.1.7 FOR 循环语句	(16)
2.1.8 WHILE 循环语句	(17)
2.1.9 函数	(18)
2.2 INFORMIX-4GL 程序的调试运行	(18)
2.2.1 INFORMIX-4GL 程序的编译运行	(18)
2.2.2 INFORMIX-4GL 的程序模块	(19)
习题 2	(20)
第三章 INFORMIX-4GL 数据库应用程序	(22)
3.1 数据库的建立	(22)
3.1.1 建立数据库	(22)
3.1.2 建立表	(23)
3.1.3 建立索引	(23)
3.2 数据库的基本操作	(24)
3.2.1 插入	(27)

3.2.2	查询	(27)
3.2.3	修改	(29)
3.2.4	删除	(31)
3.3	报表的生成	(31)
3.3.1	利用 SELECT 语句处理多个数据行的方法	(33)
3.3.2	报表的生成	(36)
习题 3		(38)
第四章	INFORMIX-4GL 程序开发工具	(40)
4.1	屏幕格式设计工具	(41)
4.1.1	屏幕格式设计的方法	(41)
4.1.2	屏幕格式说明源文件	(42)
4.1.3	屏幕格式的使用	(45)
4.2	菜单管理工具	(57)
4.3	窗口设计工具	(60)
4.4	条件查询设计工具	(62)
4.4.1	INFORMIX-4GL 条件查询的意义	(62)
4.4.2	条件查询程序的编写	(62)
4.4.3	用户给出查询条件的方法	(63)
4.5	屏幕数组管理工具	(64)
4.5.1	INFORMIX-4GL 屏幕数组的意义	(64)
4.5.2	屏幕数组在程序中的运用	(66)
4.5.3	用户对屏幕数组的操纵	(68)
4.6	联机帮助信息管理工具	(69)
习题 4		(71)
第五章	报表格式设计	(73)
5.1	报表格式器的结构	(73)
5.2	在 FORMAT 部分中使用的语句	(75)
5.2.1	默认的报表格式	(75)
5.2.2	PRINT 语句	(76)
5.2.3	SKIP 语句	(81)
5.3	FORMAT 部分中的控制块	(82)
5.3.1	页标题控制块	(82)
5.3.2	行控制块	(83)
5.3.3	组控制块	(83)
5.4	FORMAT 部分中的运算	(85)
5.4.1	算术运算	(85)

5.4.2	聚合函数	(86)
5.4.3	组函数	(86)
5.5	报表格式器内外的配合操作	(87)
	习题 5	(91)
第六章	屏幕格式及其输入/输出	(93)
6.1	屏幕格式说明文件	(93)
6.1.1	屏幕格式说明文件的结构	(93)
6.1.2	默认屏幕格式的建立	(95)
6.2	显示域的属性	(96)
6.2.1	显示域的属性	(96)
6.2.2	属性表 sysvolatt 和 sysvolval	(99)
6.3	屏幕记录和屏幕数组	(101)
6.3.1	屏幕记录	(101)
6.3.2	屏幕数组	(102)
6.4	从屏幕格式输入数据	(102)
6.4.1	INPUT FROM 语句	(102)
6.4.2	INPUT BY NAME 语句	(103)
6.4.3	INPUT ARRAY 语句	(103)
6.5	在屏幕格式上显示数据	(103)
6.5.1	DISPLAY TO 语句	(103)
6.5.2	DISPLAY BY NAME 语句	(104)
6.5.3	DISPLAY ARRAY 语句	(104)
6.6	INPUT ARRAY 语句中的子句	(104)
	习题 6	(108)
第七章	数据库管理	(110)
7.1	数据库的访问权限	(110)
7.1.1	数据库的访问权限	(110)
7.1.2	表的访问权限	(111)
7.2	数据库结构的修改	(113)
7.2.1	数据库结构的修改	(113)
7.2.2	表结构的修改	(114)
7.3	数据库视图	(115)
7.3.1	视图的定义和使用	(115)
7.3.2	有关视图的说明	(117)
7.4	数据库的完整性	(118)
7.4.1	事务处理	(118)

7.4.2	数据库的恢复	(121)
7.4.3	表的恢复	(122)
7.5	数据库的加锁	(123)
7.5.1	行加锁	(123)
7.5.2	表加锁	(124)
7.5.3	数据库加锁	(125)
习题 7	(125)
第八章	程序设计	(127)
8.1	数据结构	(127)
8.1.1	标识符	(127)
8.1.2	常量	(127)
8.1.3	变量的范围	(128)
8.1.4	多维数组	(128)
8.1.5	布尔表达式	(129)
8.2	语句	(131)
8.2.1	EXIT PROGRAM 和 RETURN 语句	(131)
8.2.2	INITIALIZE 语句	(132)
8.2.3	GOTO 语句	(132)
8.2.4	OPTIONS 语句.....	(132)
8.2.5	SCROLL 语句	(134)
8.3	内部函数	(135)
8.4	C 语言函数的调用	(137)
8.4.1	C 语言函数的编写	(137)
8.4.2	C 语言函数的登记	(139)
8.4.3	C 语言程序的编译	(139)
8.4.4	调用 C 语言函数的 INFORMIX-4GL 程序的运行	(140)
8.5	中断处理	(140)
8.5.1	中断捕捉	(140)
8.5.2	输入过程中的中断处理	(140)
习题 8	(141)
第九章	RDSQL 语言	(144)
9.1	概貌	(144)
9.1.1	标识符与数据类型	(144)
9.1.2	语句	(144)
9.2	数据查询	(147)
9.2.1	算术运算和聚合函数	(147)

9.2.2	关联名、显示标号及临时表	(148)
9.2.3	连接与合并	(151)
9.2.4	分组	(154)
9.2.5	NULL 值	(154)
9.2.6	内部函数和隐藏列	(155)
9.3	动态可变 RDSQL 语句的构造与执行	(156)
9.3.1	RDSQL 语句的预定义	(156)
9.3.2	待定信息的表示方法	(157)
9.3.3	不能预定义的语句	(158)
9.3.4	预定义语句的执行	(158)
习题 9	(160)
第十章	INFORMIX-4GL 程序开发环境	(162)
10.1	概貌(INFORMIX-4GL 菜单)	(162)
10.2	程序模块开发过程(MODULE 菜单).....	(163)
10.2.1	Modify 选择项	(163)
10.2.2	New 选择项	(164)
10.2.3	Compile 选择项.....	(164)
10.2.4	Program_Compile 选择项	(165)
10.2.5	Run 选择项	(165)
10.2.6	Debug 选择项	(165)
10.3	屏幕格式开发过程(FORM 菜单)	(165)
10.3.1	Modify 选择项	(165)
10.3.2	Generate 选择项	(166)
10.3.3	New 选择项	(167)
10.3.4	Compile 选择项.....	(167)
10.4	应用程序开发过程(PROGRAM 菜单)	(167)
10.4.1	New 选择项	(167)
10.4.2	Modify 选择项	(169)
10.4.3	Compile 选择项.....	(169)
10.4.4	Planned_Compile 选择项.....	(169)
10.4.5	Run 选择项	(169)
10.4.6	Debug 选择项	(170)
10.4.7	Undefine 选择项	(170)
附录一	INFORMIX-4GL 环境变量	(171)
附录二	INFORMIX-4GL 库函数	(172)
附录三	窗口及信息显示.....	(176)

附录四 统分系统的其余程序模块.....	(178)
附录五 数据库信息描述表.....	(184)
习题参考答案	(188)
参考文献.....	(204)

第一章 引 论

本章阐明第四代语言的定义,概述第四代语言产品 INFORMIX-4GL 的内容及其各种软件开发工具的强大功能。

1.1 第四代语言

第四代语言(4GL)是应管理信息系统(MIS)的管理人员和程序开发人员渴望得到能用于快速地开发大型应用系统的应用开发工具的要求而诞生的。

从广义上说,第四代语言是一种非过程化的程序设计语言,使用这种语言,系统开发程序员和一些非程序员的用户能够快速而有效地建立起满足需要的应用系统。第四代语言比第三代语言简单,而且更接近于自然语言。

第三代语言(3GL),如 COBOL、C、PASCAL、BASIC 等,属于过程化语言,它们要求程序员为每一个应用任务写出如何完成该任务的一系列明确的过程。第四代语言则不然,它主要要求开发人员定义出所要完成的是什么样的任务,而不必耗费精力去考虑完成任务的具体过程。

就开发一个具有菜单操作界面的应用程序而言,当选用第三代语言时,就要考虑显示菜单和处理用户输入数据的各个细节。具体地说,就是要指明怎样把菜单标题和菜单选择项显示出来,怎样控制光标在菜单选择项之间的移动和定位,怎样用诸如 IF 和 CASE 这样的条件语句根据用户输入的数据确定应该执行的操作。

而当选用第四代语言时,设计菜单的工作就可以归结到对菜单的定义上。也就是说,只需要说明菜单的标题是什么,菜单包含哪些选择项,以及每个菜单选择项对应到什么样的操作。而把显示菜单和处理用户输入数据的任务留给语言的内部过程去完成。

一个典型的第四代语言是由一些经常使用的子程序所组成的库构成的。一个典型的第四代语言程序则是由一系列调用这些预先定义好的子程序的文本或格式化文件构成的。所以,严格地说,第四代语言不是真正的语言,它是一个由一系列非过程化的应用开发工具组成的集合。

第三代语言是与应用无关的通用程序设计语言,第四代语言不同,它是专为数据库应用系统的开发而设计的,与数据库管理系统(DBMS)有着紧密的联系。正因为这样,第四代语言能够预言应用程序所要完成的一些任务,从而把完成这些任务所要执行的操作设计成内部过程,供应用程序开发者开发程序时使用。第四代语言采用了面向对象的一些程序设计思想,把构成应用系统的一些较大的成分作为对象来处理,使得应用程序开发者能够直接对这些对象进行操作,进而从如何实现这些对象的烦琐工作中解脱了出来。例如,一个菜单、一个屏幕格式或一个窗口,都可以作为一个对象来看待,在第四代语言中,应用程序开发者可以把它们作为一个整体来访问,不再象在第三代语言中那样,需要一步一步

地把它们构造出来。

可见,第三代语言在一个比较低的层次上和计算机打交道,适用于硬件层次的程序设计环境。第三代语言程序员需要熟悉他所面对的计算环境的各个方面,利用第三代语言所提供的功能强大的程序流程控制结构,一步一步地按照自己的意愿设计出完成一个应用任务的各个过程。这样,程序员完成一个应用程序设计任务需要付出较大的劳动,但他能够灵活地使程序按照自己的意愿去执行。

第四代语言在一个较高的层次上进行工作,适用于时间要求较高的应用程序开发环境。第四代语言系统开发人员可以不用顾及他所工作的硬件平台的内部结构,从而把主要精力投入到使用语言所提供的开发工具去构筑应用系统的工作中。

和第三代语言相比,第四代语言直接控制硬件操作的功能比较弱,同时,由于是非过程化语言,它的程序流程控制功能也有一定的局限性,也就是说,把它用于诸如条件判断或循环执行某些动作的方面时,会有一些的不方便。这是第四代语言的一个不够理想的地方。如果能把第三代语言的优点融合到第四代语言之中,那将会得到一种更加令人满意的效果。

1.2 INFORMIX-4GL

INFORMIX-4GL 是由美国 Informix 软件公司推出的应用软件快速开发系统,是一个优秀的第四代语言产品。

在 UNIX 市场上众多的第四代语言产品中,INFORMIX-4GL 的突出特点是同时支持过程化的和非过程化的结构,而且,为把第三代语言的子程序嵌套到第四代语言中提供了成功的方法。具体地说,INFORMIX-4GL 为开发关系数据库管理系统提供了一整套非过程化的开发工具,同时,为灵活地控制程序的流程提供了诸如 IF、CASE、FOR 和 WHILE 等的过程化控制语句,不仅如此,它还允许把第三代语言(C 语言等)的子程序嵌套到它的程序结构中。这样,INFORMIX-4GL 把非过程化语言的高速性和简单性与过程化语言的灵活性组合成了一个有机的整体,为应用系统开发人员提供了极大的便利。

INFORMIX-4GL 的主要内容包含如下几个方面:

- (1) 数据库语言;
- (2) 程序设计语言;
- (3) 屏幕格式设计工具;
- (4) 菜单设计工具;
- (5) 报表书写工具;
- (6) 窗口管理工具。

作为数据库语言,INFORMIX-4GL 是建立在 RDSQL 的基础上的。RDSQL 是 Informix 软件公司对 IBM 开发的 SQL 加以扩充和改造后得到的关系数据库结构化查询语言,它符合 SQL 的 ANSI 标准。利用 RDSQL,可以编写出对数据库进行插入、查询、修改和删除等操作的程序。

作为程序设计语言,INFORMIX-4GL 与第三代语言相似,提供赋值、循环和条件分

支等基本语句,提供记录和数组等数据结构,同时,还提供了建立子程序的语句。

INFORMIX-4GL 的各种开发工具为快速有效地开发大型应用系统提供了良好的条件。

1.3 INFORMIX-4GL 的基本概念

1.3.1 数据库

INFORMIX-4GL 是一个关系数据库管理系统,关系数据库管理系统组织数据的结构是二维表。二维表由行和列构成,如图 1.1 所示。

编号	姓名	收入	支出	结余
1	王兵	175.00	35.00	140.00
2	李平	198.50	47.00	151.50
3	丁梅	163.00	50.00	113.00
4	吴芳	208.75	27.05	181.70

图 1.1 二维表

二维表的结构由表中的列确定,一个列确定一个数据项。每个列由一个列名和一个数据类型定义,列名用于对列的引用,数据类型规定列中数据所属的类型。表中的行由数据组成。图 1.1 的表由 5 个列 4 个行构成。各列的列名分别为“编号”、“姓名”、“收入”、“支出”和“结余”,“编号”列的数据类型是整数,“姓名”列的数据类型是字符串,其余 3 个列的数据类型都是小数。

INFORMIX-4GL 中的一个数据库指的是由若干张表组成的一个集合。一个数据库至少要有一张表。

1.3.2 屏幕格式

屏幕格式是在显示数据库数据和向数据库输入数据的过程中,显示在屏幕上的具有相对固定格式的图案。在图案上,为数据库二维表中那些要显示的或要输入的数据的列分配有固定的有特殊意义的区域(称为显示域)。显示数据时,表中某列上的数据总是显示在分配给它的显示域上,输入数据时,表中某列的数据总是通过分配给它的那个显示域输入到表中。图 1.2 所示的就是一个屏幕格式。

图中的屏幕格式用于显示图 1.1 中那张表中的数据或向该表中输入数据。屏幕格式中划有 5 个显示域,分别标以 a1 至 a5 (这些标记在实际的屏幕格式上不出现)。5 个显示域依次分配给表中的 5 个列。如, a1 区域分配给编号列。显示数据时,编号列上的数据显示到 a1 区域上,输入数据时,从 a1 区域上输入的数据送到编号列上。在这个例子中,结余列的值是收入列值与支出列值之差,不用输入,因此 a5 区域只用于显示数据,不用于输入数据。

编号: [a1] 姓名: [a2]

收入: [a3] 支出: [a4]

结余: [a5]

图 1.2 屏幕格式之一

在 INFORMIX-4GL 屏幕格式上,不要求为一张表中的每个列都分配显示区域,也不要求把显示区域都分配给同一张表中的列。

需要说明一点,INFORMIX-4GL 数据库二维表中的数据在屏幕格式上的输入/输出操作是通过程序变量作为中间媒介实现的。

屏幕格式可以由程序开发人员随意设计。例如,对于图 1.1 中的二维表,也可以设计出如图 1.3 所示的屏幕格式。

编 号	姓 名	收 入	支 出	结 余

图 1.3 屏幕格式之二

图 1.2 和 1.3 的屏幕格式每次可以显示二维表的一行数据,这是简单的屏幕格式。INFORMIX-4GL 还允许程序开发人员设计如图 1.4 所示的较复杂的屏幕格式。

编 号	姓 名	收 入	支 出	结 余
a11	a12	a13	a14	a15
a21	a22	a23	a24	a25
a31	a32	a33	a34	a35

图 1.4 屏幕格式之三

图 1.4 的屏幕格式为其中标出的二维表的每个列分配有 3 个用于输入/输出数据的图形区域,如分配给编号列的图形区域是 a11、a21 和 a31。这个屏幕格式可以同时显示二维表的 3 行数据,每行数据显示在一个图形区域行上,即:第 1 行数据显示在 a11 至 a15 的区域上,第 2 行数据显示在 a21 至 a25 的区域上,第 3 行数据显示在 a31 至 a35 的区域上。图中的 3 个图形区域行构成了 INFORMIX-4GL 屏幕格式中的一个屏幕数组。

INFORMIX-4GL 允许用户控制数据在屏幕数组上滚动。通过数据在屏幕数组上的滚动,可以浏览二维表中满足一定条件的所有数据行。

输入数据的情形与显示数据的情形相似,利用屏幕数组和数据滚动,可以很方便地向二维表输入多行数据。

利用屏幕格式还可以构造条件查询。条件查询就是在检索数据库数据时,由用户给出查找条件。INFORMIX-4GL 允许用户在屏幕格式上动态给出查找条件。以图 1.2 的屏幕格式为例,如果用户希望找出二维表中编号小于 100 且收入大于 150 且姓王的人的数据,他只需在 a1 区域上输入“<100”,在 a2 区域上输入“王*”,在 a3 区域上输入“>150”便可。用户输入查找条件后的屏幕如图 1.5 所示。

在屏幕上发生的所有输入/输出活动,包括屏幕数组数据滚动和条件查询等,都是由 INFORMIX-4GL 的内部过程完成的。程序开发人员所要做的工作主要是为屏幕格式等给出如意的定义,并用几个简单的语句启动有关的输入/输出活动。

1.3.3 菜单

INFORMIX-4GL 为系统开发人员提供现成的菜单显示格式和控制管理机制,它的菜单显示格式呈现在用户面前如图 1.6 所示。

编号: [<100]		姓名: [王 *]	
收入: [>150]		支出: []	
结余: []			

图 1.5 条件查询条件的输入

主菜单: <u>U: 统分</u> F. 登记 S. 成绩 L. 名次 N. 名单 R. 新名单 C. 重显 E. 退出
登记选手在本场比赛中的各项得分

图 1.6 菜单

菜单在屏幕上占第 1 和第 2 两个显示行。屏幕的第 1 行显示菜单名和菜单选择项，最左边显示菜单名，菜单名右边显示各个菜单选择项。图 1.6 中菜单的菜单名是“主菜单”，该菜单名的右边是 8 个菜单选择项，光标停在第 1 个菜单选择项上。屏幕的第 2 行显示有关光标照亮的那个菜单选择项的提示信息。

当菜单选择项较多，一行显示不下时，INFORMIX-4GL 把它们分成多页进行显示。当光标停在最右边的菜单选择项上时，按一下右方向键可显示在下一页上的菜单选择项。当光标停在最左边的菜单选择项上时，按一下左方向键可显示在上一页上的菜单选择项。当有下一页菜单选择项存在时，INFORMIX-4GL 在最右边的那个菜单选择项的右方给

出一个省略号(…)。当有上一页菜单选择项存在时,INFORMIX-4GL 在菜单名和最左边那个菜单选择项之间给出一个省略号。

在 INFORMIX-4GL 中,可以很方便地为菜单选择项提供联机帮助信息。在这种情况下,当光标停在某菜单选择项上时,用户只要按一下指定的键(一般是 CONTROL-W),便可得到有关该菜单选择项的联机帮助信息。例如,在图 1.6 的菜单情形中,用户按一下指定的键可以得到如图 1.7 所示的联机帮助信息。

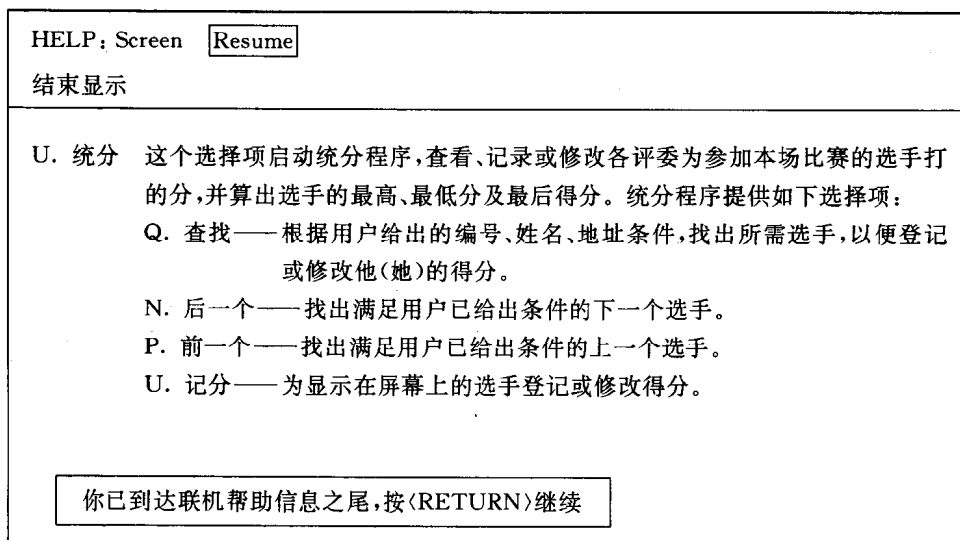


图 1.7 选择项“U. 统分”的联机帮助信息

1.3.4 报表

数据库中的二维表是数据库数据在计算机硬盘上的组织形式,用户是看不见的。在实际应用中,经常要把数据库中的数据以一定方式打印出来,以便用户查阅。以数据库二维表为依据,把数据库中的某类数据按照一定的格式打印出来所得到的结果,就是一份报表。例如,根据图 1.1 的二维表,可以输出如图 1.8 所示的报表。

职工收支情况表

编 号	姓 名	收 入 (元)	支 出 (元)	结 余 (元)
1	王 兵	175.00	35.00	140.00
2	李 平	198.50	47.00	151.50
3	丁 梅	163.00	50.00	113.00
4	吴 芳	208.75	27.05	181.70

图 1.8 报表