

## 第4章 数据检索

### 本章目的

- 会运用查询向导及设计器进行查询
- 掌握数据查询的 SQL 命令及方法
- 建立表之间的关联
- 生成用户订单视图
- 对数据表记录的操作，包括表记录指针的定位，表记录的浏览、修改、删除与恢复、添加与插入
- 表的索引与排序
- SQL 语言

### 本章任务

任务一：使用程序在超市管理系统中查询商品

任务二：查询设计器

任务三：查询向导

任务四：建立超市管理系统的索引

任务五：通过索引查询商品

任务六：组织查询的输出结果

任务七：联接多个表的查询

任务八：定制查询

任务九：生成用户使用的订单视图

任务十：排序

任务十一：更新数据

### 4.1 基础知识

在数据库的应用中，对数据的检索是数据处理中不可缺少的、最常用的。Visual FoxPro 提供了多种较好的数据检索的方法，如文件数据的查询、文件索引及结构化数据查询语言。查询是从数据表中获取所需要的结果，即设定一些过滤条件，并把这些条件存为查询文件，在每次查询数据时，调用该文件并加以执行查询出来的结果，可以加以排序、分类查询出来的结果可存储成多种输出格式。

视图能够从本地或远程表中提取一组记录，使用视图可以处理或更新检索到的记录，即设定条件，可以从几个数据表文件中过滤出所要求的数据，其结果存储成实际的记录数据，可以当作实际的数据表文件来使用，视图中的数据记录更改后，原表中的记录也要随之修改。

结构化查询语言（Structure Query Language，SQL），是目前美国国家标准组织 ANSI 的标

准数据语言, 已有 90 多个数据库管理产品支持 SQL 语言, 已在微型计算机、小型机乃至大型机等各种机型上运行。Visual FoxPro 语言体系包括 3 个部分: 一是传统的 xBase 语言体系; 二是结构化查询语言 (SQL); 三是面向对象程序 (OOP) 设计语言。Visual FoxPro 这 3 部分语言都具有数据查询的功能。对于 xBase 语言和 SQL 语言具有数据查询功能可能大家是没有异议的, 但 Visual FoxPro 中的 OOP 语言应不会有数据查询功能, 因为一般认为在 Visual FoxPro 中数据处理是非面向对象的, 就此可以得出 Visual FoxPro 中的 OOP 语言应不具备数据查询功能。其实不然, 在 Visual FoxPro 中通过 OOP 语言导入当前流行的 RDO 或 ADO 数据处理模型时, 就会得出不同的答案。

## 4.2 任务一 在超市管理系统中查询商品

### 4.2.1 相关知识

查询就是向一个数据库发出检索信息的请求, 从中提取符合特定条件的记录。查询文件即保存实现查询的 Select-SQL 命令的文件。保存查询文件时, 系统自动给出扩展名 .qpr; 查询被运行后, 系统还会生成一个编译后的查询文件, 扩展名为 .qpx。查询结果通过运行查询文件得到一个基于表和视图的动态的数据集合, 结果可以用不同的形式来保存。查询中的数据是只读的, 数据源可以是一张或多张相关的自由表、数据库表、视图。

查询可以实现对数据库中数据的浏览、筛选、排序等操作, 可以为其他数据库提供新的数据表, 可以从单个表中提取有用的数据, 也可以从多个表中提取综合信息。应用中通过运行查询文件, 快速检索存储在表中或通过视图映射到表中的数据, 查找满足条件的记录, 还可以按需要将查找到的记录排序和分组, 将查询结果显示出来, 也可以保存到新建表中或以报表、图表形式输出。也就是说, 查询只能从数据源中检索出数据, 不能更新源数据, 可以将检索出的数据重新组织, 再以多种形式输出。

#### 1. 创建方法

- (1) 用查询向导建立。
- (2) 用查询设计器建立。
- (3) 用 SQL 语言建立。

#### 2. 顺序查询

顺序查询就是按记录的物理顺序进行查询。其特点是查询时不需要为表建立索引, 因为顺序查询操作是从表中的首记录开始直至查询到满足条件的记录为止。如果表中没有满足条件的记录, 系统将记录指针移向文件结束标识位置, 即 EOF() 函数的返回值为 .T。顺序查询操作可以用 LOCATE 和 CONTINUE 命令实现, 也可以用限制对记录和字段访问的菜单实现。

- (1) 顺序查询命令 (LOCATE 和 CONTINUE)。

格式: LOCATE [<范围>][FOR<条件 1>][WHILE<条件 2>]

格式: CONTINUE

功能: 在当前表中按顺序搜索, 以寻找满足指定条件的第一条记录。

说明:

- ① 如命令中缺省<范围>, LOCATE 默认范围是表中的所有 (ALL) 记录。

② 被搜索的表不必建立索引。

③ 若 LOCATE 发现满足条件的记录时，记录指针将定位于该记录上。此时，RECNO() 返回该记录号，FOUND() 函数返回.T.，EOF() 函数返回.F.。

④ 若未找到满足条件的记录，则 RECNO() 返回表中的记录总数加 1，FOUND() 函数返回.F.，而 EOF() 函数返回.T.。

⑤ LOCATE 发现一个满足条件的记录之后，可用 CONTINUE 命令，在表的剩余记录中继续查找满足条件的下一个记录。CONTINUE 命令可以重复使用，直至到达范围的边界或表尾。

⑥ LOCATE 命令和 CONTINUE 命令只能用于当前工作区。若选择了另一工作区后，再返回原来的工作区时，可继续原来的搜索过程。

查找到满足条件的第一条记录后，用 DISPLAY 命令显示该记录的内容，用 CONTINUE 命令将记录指针指向第二条满足条件的记录。也可以通过“显示”菜单的“浏览”命令，观察记录指针的位置和定位记录的内容。

执行 CONTINUE 命令记录指针移位后，重新浏览该表或单击原浏览表，可看到记录指针的变化。

#### (2) 限制对字段和记录的访问。

如果只想查看某些记录，可以设置“数据过滤器”，对浏览窗口中显示的记录进行限制。如果只对表中的某几个字段感兴趣，可以通过“字段筛选”屏蔽其他字段。

##### ① 设置记录筛选条件的命令 (SET FILTER)。

格式：SET FILTER TO [表达式]

功能：指定访问当前表中记录时必须满足的条件。

说明：

- 对于每个打开的表，都可以设置单独的记录筛选条件（过滤器）。表中只有满足表达式指定条件的记录才可以被访问。
- 只有当记录指针在表中移动时，才计算由 SET FILTER 指定的条件。没有“表达式”参数的 SET FILTER TO 命令将关闭当前表的过滤器，使以前指定的条件无效。
- Select-SQL 命令不遵守当前的筛选条件。

##### ② 限制对字段的访问命令 (SET FIELDS)

格式 1：SET FIELDS ON | OFF

格式 2：SET FIELDS TO [[字段名 1] [, 字段名 2...]] | ALL [LIKE 通配符] [EXCEPT 通配符]

功能：指定可以访问表中的哪些字段。

说明：

- ON 指定只能访问出现在字段列表中的字段。OFF（默认值）指定可以访问当前表中的所有字段。
- ALL[LIKE 通配符]，则可以访问与通配符相匹配的字段。ALL[EXCEPT 通配符]，则可以访问不与通配符相匹配的字段。Visual FoxPro 支持的通配符是？和\*。

#### (3) 数据库统计与分类汇总。

##### ① 记录个数统计命令。

格式：COUNT [<范围>] [FOR<条件 1>] [WHILE<条件 2>] [TO <内存变量>]

功能：统计指定范围内满足条件的记录个数，并将结果存于指定的内存变量。

说明：缺省<范围>和<条件>时，系统默认值是 ALL。用 TO 短语将统计结果保存在指定内存变量中。

② 累加求和。

格式：SUM [<范围>][<表达式表>][FOR<条件>][WHILE<条件>][TO<内存变量表>]

说明：

- 缺省<范围>和<条件>短语时，默认值是 ALL。
- 如果缺省<表达式表>，则对数据库中所有数值型字段分别求和。
- 若用 TO 短语指定内存变量，内存变量的个数必须与表达式的个数相同。若不用 TO 短语，仅在屏幕上显示数值型字段求和结果。

③ 计算平均值。

格式：AVERAGE [<范围>][<表达式表>][FOR <条件>][WHILE <条件>][TO<内存变量表>]

说明：

- 缺省<范围>和<条件>短语时，默认值是 ALL。
- 如果缺省<表达式表>，则对数据库中所有数值型字段分别求平均值。
- 若用 TO 短语指定内存变量，内存变量的个数必须与表达式的个数相同。若不用 TO 短语，仅在屏幕上显示数值型字段平均值。

④ 分类汇总命令。

格式：TOTAL ON <字段名> TO <表文件名> [FIELDS <字段名表>][<范围>][FOR <条件 1>][WHILE <条件 2>]

说明：

- 该命令操作的字段名必须先进行排序或索引。新生成的汇总数据存放在指定的<表文件名>中（扩展名为.DBF），其结构与原表相同，但不包括备注字段。在新生成的表文件中，只包含与字段值相同的所有记录中的首记录，该首记录的数值型字段值是同类记录之和。
- 用 FIELDS 短语指定汇总字段，若缺省则对所有数值型字段汇总。范围的默认值是 ALL。
- 如果表文件结构中字段宽度不足以容纳汇总结果，系统将提示“数值溢出（数据丢失）”，并在相应字段中用\*表示。因此，在执行汇总命令之前，应当估计出字段求和大小。若有必要，要先修改表结构，增加相关字段的宽度。

### 3. SQL 语言

结构化查询语言 SQL 已成为关系数据库标准语言，是一种综合的、通用的、功能极强的关系数据库语言，其主要特点如下。

（1）一体化的语言。

结构化查询语言的语言集数据定义（DDL）、数据操纵（DML）、数据管理（DCL）的功能于一体，语言风格统一，可以独立完成数据库的全部操作，包括定义关系模式、录入数据及建立数据库、查询、更新、维护数据、数据库的重新构造、数据库安全性等一系列操作的要求，为数据库应用系统开发者提供了良好的环境。

（2）一种高度非过程化的语言。

（3）以同一种语法结构提供两种使用方式，即命令交互使用和程序方式使用。

(4) 语言简洁, 易学易用。

(5) SQL 语言支持关系数据库 3 级模式的结构。

SQL 数据定义命令包括定义基本表、定义视图及定义索引 3 部分。可用于定义和修改模式(如基本表)、定义外模式(如视图)和内模式(如索引), 也可以删除上述定义。

SQL 命令动词如表 4-1 所示。

表 4-1 命令动词表

| SQL 功能 | 命令动词                 |
|--------|----------------------|
| 数据查询   | SELECT               |
| 数据定义   | CREATE、DROP、ALTER    |
| 数据操纵   | INSERT、UPDATE、DELETE |
| 数据控制   | GRANT、REVOKE         |

#### 4. 用 Select-SQL 命令创建查询

SQL 语句最主要的功能就是查询, 即把数据表中的数据查找出来, 以使用户浏览和使用。SQL 语句中查询表数据使用的是 SELECT 语句。SELECT 语句的基本形式分为 3 部分: 查什么数据、由哪里查数据、查的条件是什么, 即 SELECT-FROM-WHERE。

(1) 基本的 SELECT 语句。

格式: SELECT <字段名 1>[, <字段名 2>...] FROM [数据库名称!]<表或视图>

(2) 唯一性查询。唯一性查询指对于要显示的结果, 若有重复出现时, 则只显示重复行中的一行。这可以通过在基本 SELECT 语句中加入 DISTINCT 关键字的方法实现。若不加此关键字, 则显示全部行。

(3) 带条件的查询。在 SELECT 语句中, 可以通过 WHERE 子句为查询规定条件。

格式: WHERE <条件表达式 1>[AND <条件表达式 2>...]

功能: 查询指定表或视图中满足查询条件的记录。其中<条件表达式>可以是关系表达式, 也可以是逻辑表达式。

(4) 排序查询。通过 ORDER BY 子句可以实现查询结果的排序输出, 允许按一列或多列排序。

格式: ORDER BY <排序选项 1>[ASC|DESC][, <排序选项 2>[ASC|DESC]...]

其中, ASC 表示升序排序(默认方式), DESC 表示降序排序。

(5) 带计算函数的查询。在 SELECT 语句中, 使用 SQL 语言提供一些查询计算函数, 可以增强查询功能。基本的查询计算函数的格式及功能如表 4-2 所示。

表 4-2 计算函数表

| 函数的格式    | 函数功能          |
|----------|---------------|
| COUNT(*) | 计算记录条数        |
| SUM(字段名) | 求字段名所指定字段值的总和 |
| AVG(字段名) | 求字段名所指定字段的平均值 |
| MAX(字段名) | 求字段名所指定字段的最大值 |
| MIN(字段名) | 求字段名所指定字段的最小值 |

(6) 用别名输出列标题 (字段名)。将列名用含义更明确的别名输出, 便于人们理解, 可以通过 AS 关键字实现。

格式: AS <列别名>

(7) 分组查询。通过 GROUP BY 子句可以实现分组查询。

格式: GROUP BY <分组字段名 1>[, <分组字段名 2>...][HAVING <过滤条件>]

其中, <分组字段名>可以是表的字段名、字段函数名或标识列的数值型表达式; [HAVING <过滤条件>]子句进一步限定分组的条件。

说明: 若只有 HAVING 子句而无 GROUP BY 子句, 则 HAVING 子句相当于 WHERE 子句。

本小节主要技能点如下:

- (1) 显示记录命令。
- (2) 暂停命令。
- (3) 显示命令结果的开关命令。
- (4) 显示字段的别名。
- (5) 条件判断。

#### 4.2.2 任务实现

##### 1. 显示记录命令

LIST / DISPLAY 显示当前表中的记录或指定的表达式的值。

格式: LIST|DISPLAY [[FIELDS] <表达式表>] [<范围>] [FOR <条件>] [WHILE <条件>]  
[TO PRINTER [PROMPT]]|TO FILE <文件名>] [OFF]

命令中各子句的含义是:

(1) FIELDS <表达式表>指定要显示的表达式。表达式可直接使用字段名, 也可以是有字段名的表达式, 甚至是不含字段名的任何表达式。

如果省略 FIELDS 命令, 则显示表中所有字段的值。但备注型字段和通用型字段内容不显示, 除非备注型字段和通用型字段明确地包括在<表达式表>中。

(2) 若选定 FOR 子句, 则显示满足所给条件的所有记录。若选定 WHILE 子句, 显示直到条件不成立时为止, 这时后面即使还有满足条件的记录也不再显示。FOR 子句和 WHILE 子句可以同时使用, 同时使用时 WHILE 子句优先。

(3) <范围>、FOR 子句和 WHILE 子句用于决定对哪些记录进行操作。如果有 FOR 子句, 默认的范围为 ALL; 有 WHILE 子句, 默认的范围为 REST。

如果 FOR 子句或 WHILE 子句及范围全省略, 对于 LIST 默认为所有记录, 即取 ALL, 对于 DISPLAY 默认为当前记录, 即取 NEXT 1。此外, 对于 DISPLAY 命令是分屏输出, 对于 LIST 是连续输出。

(4) 选用 OFF 时, 表示只显示记录内容而不显示记录号。若省略该项则同时显示记录号和记录内容。方法:

```
USE d:\supermarket\生产商.dbf EXCLUSIVE
List
```

显示如图 4-1 所示。



图 4-1 命令窗口显示“生产商”表中的数据

## 2. 暂停命令

格式：WAIT[<提示信息>][TO<内存变量>] [WINDOW [AT<行>,<列>]] [NOWAIT]  
[TIMEOUT<数值表达式>]

功能：暂停并显示提示信息，用户按任意键或单击时才继续执行程序。

说明：可选项[TO <内存变量>]，将输入的单个字符存入指定的<内存变量>；若用户是按回车键或单击，<内存变量>的值为空串。若省略所有可选项，屏幕显示“键入任意键继续……”的默认提示信息。

方法：wait

显示如图 4-2 所示。



图 4-2 命令窗口和显示窗口

## 3. 显示命令结果的开关命令

格式：SET TALK ON|OFF

参数：ON 显示 Visual FoxPro 命令执行的状态。OFF 不显示 Visual FoxPro 命令执行的状态。

说明：使用时，ON、OFF 二者之中必须选择一个。

很多 Visual FoxPro 命令执行后，会显示执行后的结果状态，如执行 LOCATE FOR 命令，如找到记录的会显示被找到的记录号，否则会显示“已到文件尾”，但一般在程序中是不需要这些显示的，比如找到记录就直接显示出来，没找到一般用一个对话框给出更清楚的提示，所以在程序一开始往往要将 SET TALK 关闭。

## 4. 显示字段的别名

将列名用含义更明确的别名输出，便于人们理解，可以通过 AS 关键字实现。

格式：AS <列别名>

例如：SELECT 生产商.厂商名称 as a FROM 生产商

## 5. 条件判断

方法一：locate for 厂商地址="沈阳"  
display

结果如图 4-3 所示。

| 记录号 | 厂商编号 | 厂商名称 | 厂商地址 | 厂商电话   |
|-----|------|------|------|--------|
| 3   | 003  | 稻香村  | 沈阳   | 110032 |

图 4-3 显示结果

在 Visual FoxPro 中使用的标点符号要求均为半角符号，如以上命令窗口中只能使用半角的 " " 而不能使用全角的 “ ”。

方法二：SELECT 厂商地址 FROM 生产商 WHERE 厂商地址="沈阳"

结果显示如图 4-4 所示。



图 4-4 查询结果

6. 查找的重复运行

在命令窗口要重复的语句后面按回车键。

7. 注释

语法：Note 说明

说明：在每行注释的末尾加上一个分号 (;)，表明下一行是该行注释的继续，注释后字体颜色变成绿色。

方法：note 查找 厂商地址="沈阳"的记录

显示如图 4-5 所示。



注释文字

图 4-5 命令窗口

课后练习

1. 显示“供货商”表中的所有记录。
2. 在“销售”表中查找商品编号为 ee 的记录。

3. 给练习 2 加上注释。

## 4.3 任务二 查询设计器

### 4.3.1 相关知识

#### 1. 用查询设计器建立查询

打开查询设计器的 4 种方法如下：

(1) 用 CREATE QUERY 命令打开查询设计器建立查询。例如，输入：CREATE QUERY 名称。

(2) 选择“文件”→“新建”命令，或单击“常用”工具栏上的“新建”按钮，然后选择“查询”并单击“新建文件”按钮，打开查询设计器建立查询。

(3) 在“项目管理器”中选择“数据”选项卡，选取“查询”项，单击“新建”按钮，进入查询设计方式选择对话框后单击“新建查询”，则启动“查询设计器”，如图 4-6 所示。



图 4-6 “查询设计器”窗口

(4) 在“项目管理器”中选择“数据”选项卡，从“查询”项中选择一个已存在的查询文件，再单击“修改”按钮，也启动“查询设计器”。

通过查询设计器，可添加当前数据库中的表或自由表及视图，如图 4-7 所示。

#### 2. 查询设计器中选项卡的设置

(1) “字段”选项卡。“字段”选项卡用于指定要输出的字段。

“函数和表达式”文本框的功能是通过输入一个函数和表达式生成一个虚拟的字段，如图 4-8 所示。



图 4-7 添加表或视图

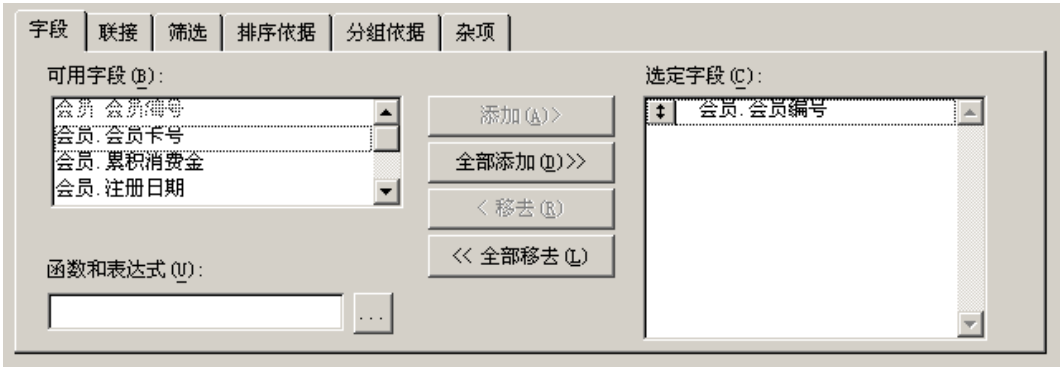


图 4-8 “字段”选项卡

(2)“联接”选项卡。“联接”选项卡是为匹配一个或多个表或视图中的记录指定联接条件，如字段的特定值、表间临时关系的联接条件，如图 4-9 所示。

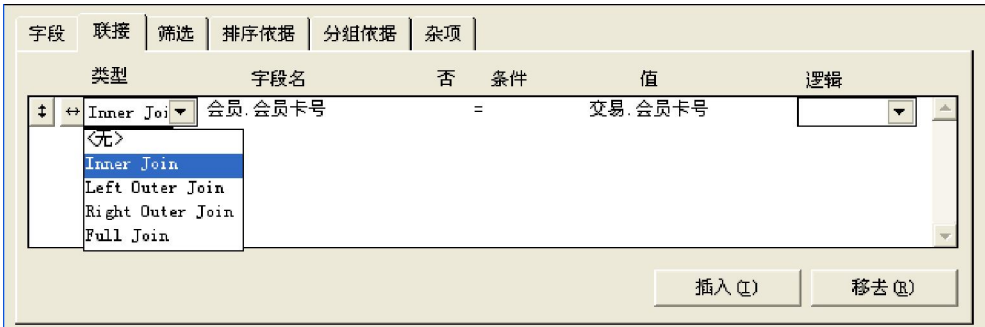


图 4-9 “联接”选项卡

联接类型：

- ① Inner Join: 指定只有满足联接条件的记录包含在输出结果中，此类型是默认的。

② **Right Outer Join:** 指定满足联接条件的记录, 以及联接条件右侧的表中记录 (即使不匹配联接条件) 都包含在输出结果中。

③ **Left Outer Join:** 指定满足联接条件的记录, 以及联接条件左侧的表中记录 (即使不匹配联接条件) 都包含在输出结果中。

④ **Full Join:** 指定所有满足和不满足联接条件的记录都包含在输出结果中。

(3) “筛选”选项卡。“筛选”选项卡用于指定选择记录的条件, 如图 4-10 所示。

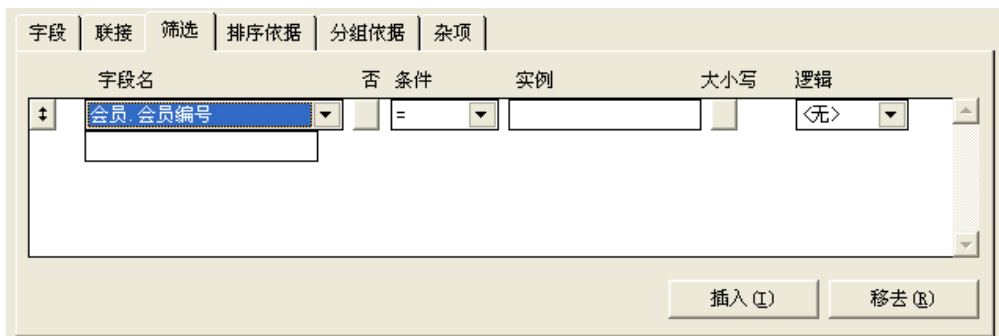


图 4-10 “筛选”选项卡

① 当实例与查询表中字段名相同时, 用引号将其括起来。

② 日期必须使用严格日期格式或用 CTOD 转换函数。

③ 逻辑值的前后必须使用英文句点号 “.”, 如.T。

④ 若想对逻辑操作符的含义取反, 选中“否”下面的按钮。

(4) “排序依据”选项卡。“排序依据”选项卡用来根据选定字段设置查询中检索记录的输出顺序, 如图 4-11 所示。

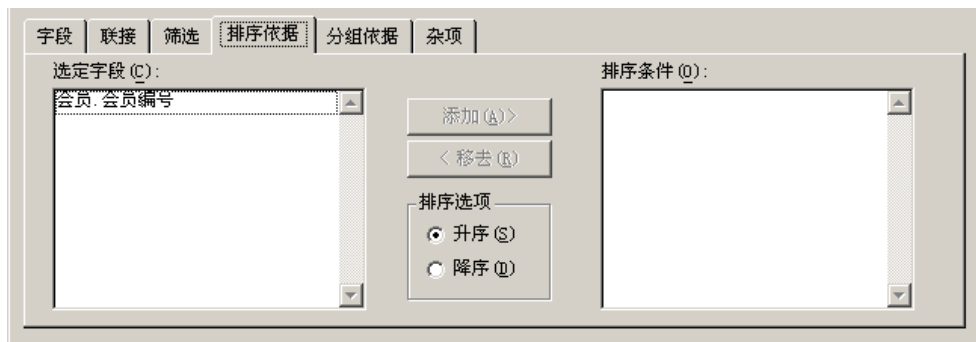


图 4-11 “排序依据”选项卡

(5) “分组依据”选项卡。“分组依据”选项卡用来指定字段、SUM 或 COUNT 之类的合计函数, 或把有相同字段值的记录合并为一组, 实现对视图结果的行进行分组, 可以通过满足条件按钮来进行相关的设置, 如图 4-12 所示。

说明: 在 Visual FoxPro 9 和 Visual FoxPro 8 中, 分组字段所选的字段必须与“字段”选项卡中所选的字段相同, 如图 4-8 所示, 最终的排列次序由字段分组中的字段排列次序决定。在其他的 Visual FoxPro 版本中, 按字段分组不受其他选项影响。

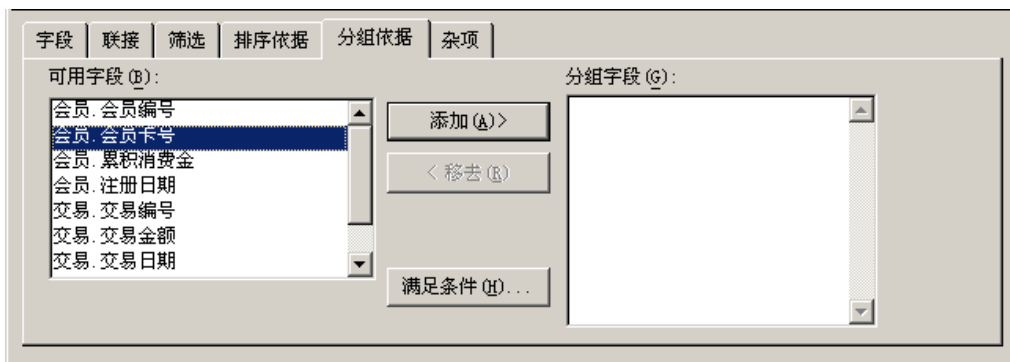


图 4-12 “分组依据”选项卡

(6) “杂项”选项卡，如图 4-13 所示。

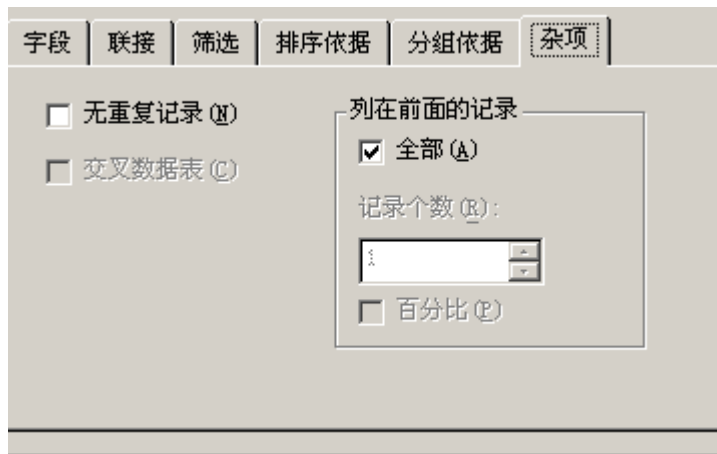


图 4-13 “杂项”选项卡

“无重复记录”：在查询结果中清除重复记录。重复记录是指查询结果的选定字段或表达式的值都相同的记录。

“交叉数据表”：将查询结果以交叉表格传给 Microsoft Graph、报表或表。只有当“选定字段”刚好为 3 项时，才可使用交叉数据表选项。此 3 项代表 X 轴、Y 轴和图形的单元值。此项自动为查询选择“分组依据”和“排序依据”字段。选择后，用户将不能再修改此查询，除非清除此选项。

“列在前面的记录”：并不一定要将满足查询条件的所有记录都包含在输出结果中，可指定只输出一定数目的记录或一定百分比满足条件的记录。“列在前面的记录”区域有以下 3 个选项：

- “全部”：指定满足查询条件的所有记录都包含在查询结果中。
- “记录个数”：该项为一个组合框，可设置一个整数，以决定将多少个选中记录包括在查询结果中。但当选中的下面的“百分比”复选框时，该框中便变成一个百分比框，其中的数表示将满足查询条件的百分之几包括在查询结果中。
- “百分比”：改变“记录个数”组合框中数值的意义，此值必须多于 1% 的记录。

要设计一个查询文件，首先必须明确查询的目的是什么，即想要得到哪些数据，并以什么方式存在；或者想要得到满足某些条件的特定记录，或者想要知道某些记录的字段值组合成的表达式按某一方式输出。明确输出的数据后，就可以开始设计查询了。

本小节主要技能点如下：

- (1) 启动查询设计器。
- (2) 在查询输出中添加字段。
- (3) 选择输出全部字段。
- (4) 显示字段的别名。
- (5) 设置输出字段的次序。
- (6) 选定所需的记录。

#### 4.3.2 任务实现

(1) 启动查询设计器。选择“文件”→“新建”命令，或单击“常用”工具栏上的“新建”按钮，然后选择“查询”并单击“新建文件”图标按钮，打开查询设计器建立查询，如图4-14和图4-15所示。

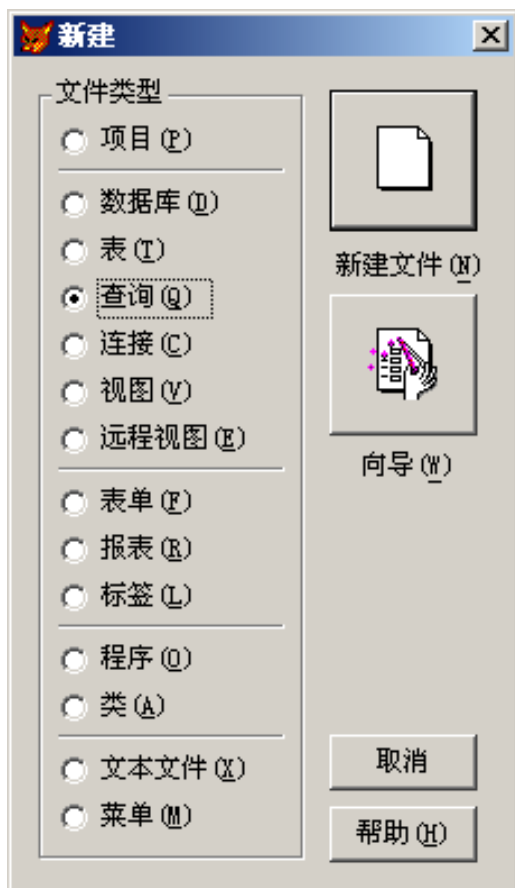


图 4-14 “新建”对话框

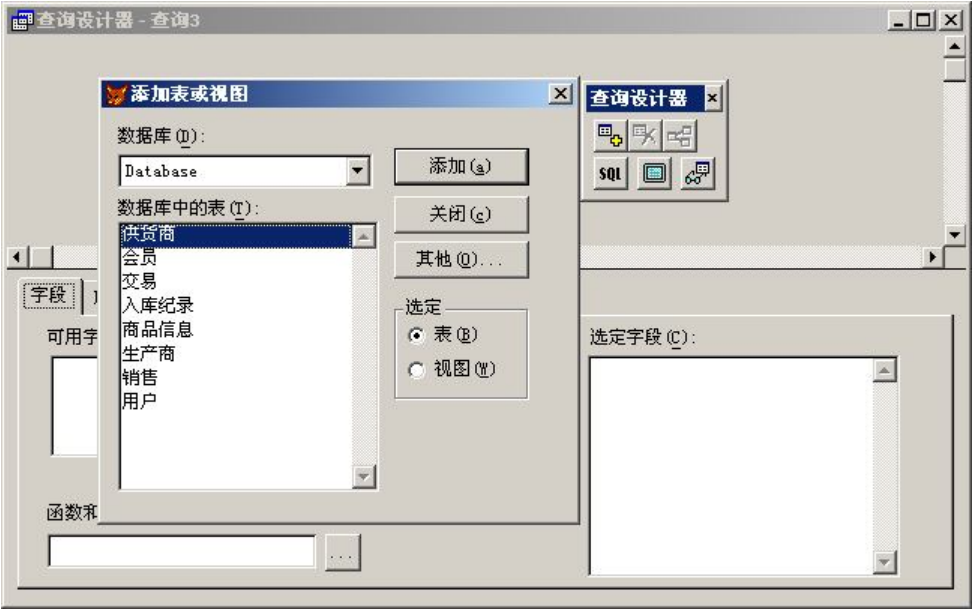


图 4-15 查询设计器

(2) 添加用户表和交易表。在查询输出中添加字段，如图 4-16 所示。



图 4-16 添加字段

(3) 选择输出全部字段。单击 **全部添加(A)>>** 按钮，如图 4-17 所示。

(4) 设置输出字段的次序。选择“排序依据”选项卡中的“升序”单选按钮，如图 4-18 所示。

(5) 选定所需的记录，如图 4-19 所示。

错误



图 4-17 输出全部字段

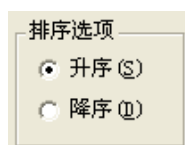


图 4-18 排序选项



图 4-19 选择“用户编号=y002”的记录

在“条件”列表中，除了常用的关系运算符外，还有 Like、Is Null、Between 和 In 运算符，它们的具体含义如下。

**Like:** 在“字段名”框列出的字段值与“实例”框中给出字段值或样本值执行不完全匹配，主要是针对字符类型的。如查询条件为“Student.姓名 Like 王”，则姓名为王自刚、王娅等都满足该条件。

**Is Null:** 指定字段必须包含 NULL 值。

**Between (在中间):** 输出字段的值应大于或等于“实例”框中的最小值，而小于或等于“实例”框中的最大值。在“实例”框中最小值在前，最大值在后，中间以逗号分隔。

**In (在...之中):** 意为输出字段的值必须是“实例”框中给出值中的一个，在“实例”框给出的各值之间以逗号分隔。

## 课后练习

1. 利用查询设计器对“销售”表建立查询。
2. 利用查询设计器对“用户”表建立查询，并且按用户编号排序。

## 4.4 任务三 查询向导

### 4.4.1 相关知识

#### 1. 用查询向导建立查询

创建查询一般包含以下操作：

- (1) 选择要包含在查询中的表和字段。
- (2) 设定表间关联的联接条件（数据跨表时）。
- (3) 设定记录筛选条件。
- (4) 设定查询结果的记录顺序。
- (5) 指定查询结果的输出方式。

#### 2. 用向导生成查询

- (1) 打开项目文件，如 supermarket.pjx，如图 4-20 所示。

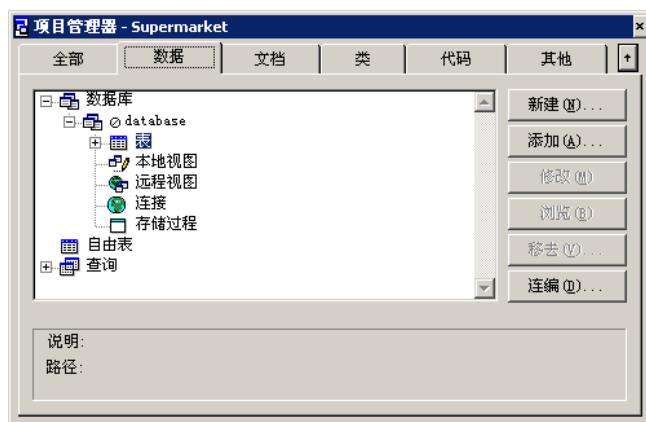


图 4-20 打开项目文件

- (2) 选择“文件”→“新建”命令。
- (3) 选定“查询”单选按钮，单击“向导”图标按钮，如图4-14所示。
- (4) 选择“查询向导”，单击“确定”按钮。
- (5) 字段选取。选定数据库和表，选取查询中需要的字段，如图4-21所示。



图 4-21 选择字段

- (6) 筛选记录，如图4-22所示。



图 4-22 “筛选记录”对话框

- (7) 排序记录，如图4-23所示。



图 4-23 “排序记录”对话框

(8) 限制记录，如图 4-24 所示。



图 4-24 “限制记录”对话框

(9) 单击“完成”按钮。系统将生成查询文件，如查询文件命名为“查询 1”，则在磁盘上会产生文件“查询 1.qpr”。

本小节主要技能点如下：

- (1) 查询的设计过程。
- (2) 查询向导。
- (3) 查询的实质内容。

4.4.2 任务实现

1. 查询的设计过程

用查询设计器设计“商品信息”表的查询。

(1) 选择菜单“文件”→“新建”命令，弹出“新建”对话框，选择“查询”单选按钮，如图 4-25 所示。

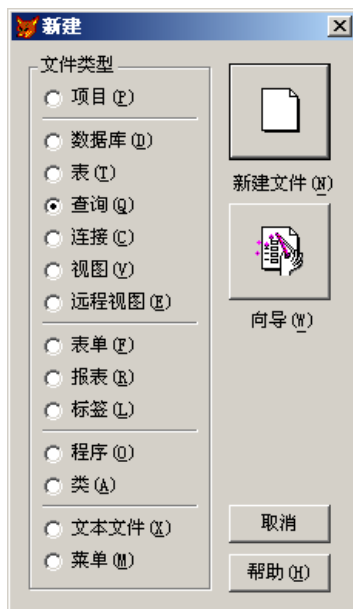


图 4-25 “新建”对话框

- (2) 单击“新建文件”图标按钮，弹出“查询设计器”及“添加表或视图”对话框。  
(3) 选择“商品信息”表进行添加，如图 4-26 所示。

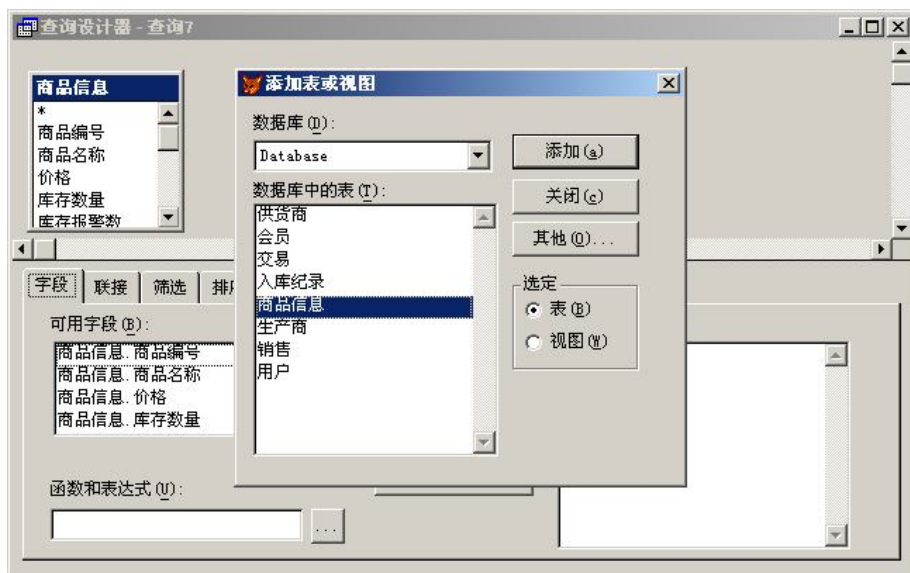


图 4-26 选择“商品信息”表后的设计视图

- (4) 字段选取。选取查询中需要的字段，如图 4-27 所示。



图 4-27 字段选取

(5) 为查询选择需要显示的字段，如图 4-28 所示。



图 4-28 为查询选择需要显示的字段

(6) 保存。原表如图 4-29 所示。运行结果如图 4-30 所示。

2. 查询向导

用查询向导实现对商品信息表的查询，

- (1) 选择菜单命令“文件”→“新建”，在弹出的“新建”对话框中选择“查询”单选按钮，单击“向导”图标按钮中的“查询向导”。
- (2) 单击“查询向导”，弹出“向导选取”对话框，如图 4-31 所示。
- (3) 选择所需要的向导，单击“确定”按钮，弹出“字段选取”对话框，如图 4-32 所示。

| 商品编号 | 商品名称 | 价格         | 库存数量      | 库存报警数     | 计划进货数     | 条形码  | 促销价格      |
|------|------|------------|-----------|-----------|-----------|------|-----------|
| m001 | 音箱   | 30.00000   | 1.00000   | 1.00000   | 1.00000   | 1001 | 1.00000   |
| m002 | 小蛋糕  | 10.00000   | 1.00000   | 12.00000  | 5.00000   | 1002 | 112.00000 |
| m003 | 生日蛋糕 | 10.00000   | 22.00000  | 22.00000  | 12.00000  | 1003 | 112.00000 |
| m005 | 电脑   | 1000.00000 | 100.00000 | 100.00000 | 120.00000 | 1004 | 500.00000 |
| m008 | 手表   | 330.00000  | 330.00000 | 110.00000 | 130.00000 | 1005 | 330.00000 |
| m007 | 石英钟  | 200.00000  | 10.00000  | 10.00000  | 10.00000  | 1006 | 200.00000 |

图 4-29 原商品信息表

| 商品编号 | 商品名称 | 价格         | 库存数量      | 条形码  | 促销起日期    |  |
|------|------|------------|-----------|------|----------|--|
| m001 | 音箱   | 30.00000   | 1.00000   | 1001 | 08/08/08 |  |
| m002 | 小蛋糕  | 10.00000   | 1.00000   | 1002 | 08/08/08 |  |
| m003 | 生日蛋糕 | 10.00000   | 22.00000  | 1003 | 08/08/08 |  |
| m005 | 电脑   | 1000.00000 | 100.00000 | 1004 | 08/08/08 |  |
| m008 | 手表   | 330.00000  | 330.00000 | 1005 | 08/08/08 |  |
| m007 | 石英钟  | 200.00000  | 10.00000  | 1006 | 08/08/08 |  |

图 4-30 运行结果

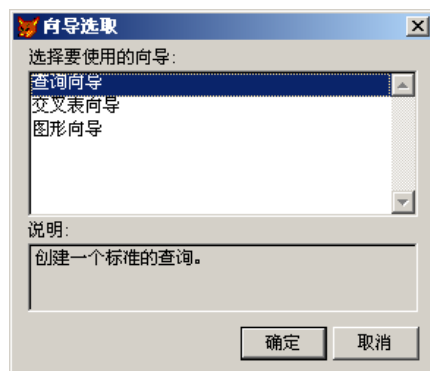


图 4-31 “向导选取”对话框



图 4-32 “查询向导”对话框

(4) 字段选取。选定数据库和表，选取查询中需要的字段，如图 4-33 所示。



图 4-33 “字段选取”对话框

(5) 单击“下一步”按钮，弹出“筛选记录”对话框，如图 4-34 所示。



图 4-34 “筛选记录”对话框

(6) 单击“下一步”按钮，弹出“排序记录”对话框，如图 4-35 所示。

(7) 单击“完成”按钮，保存，如图 4-36 所示。

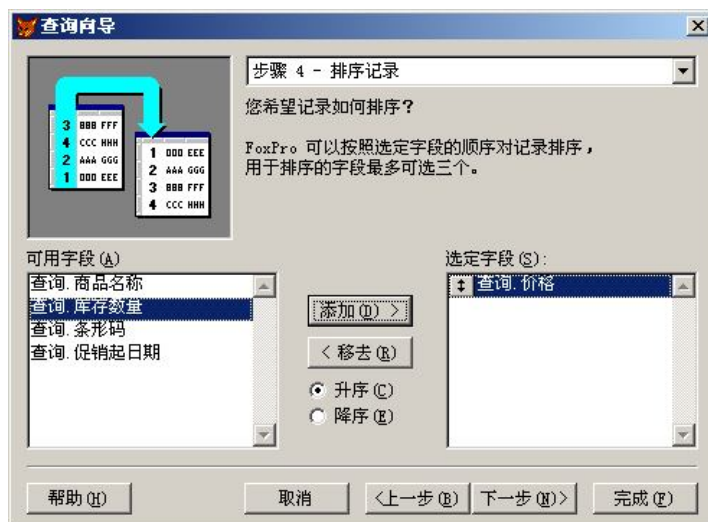


图 4-35 表的排序

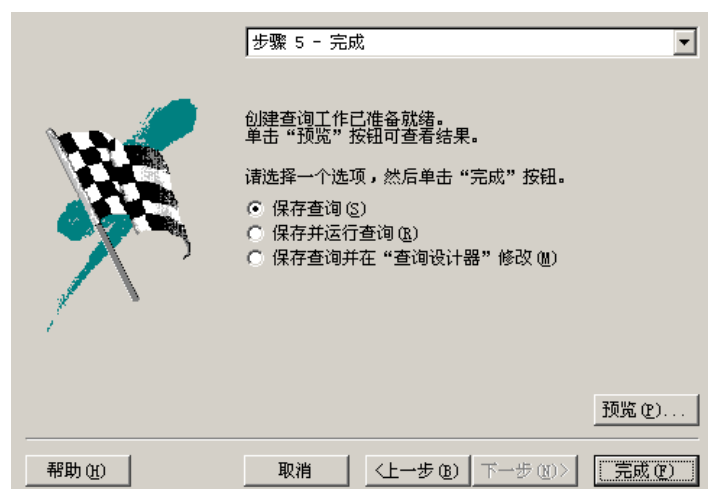


图 4-36 完成查询

## 课后练习

利用查询向导，对用户表建立查询，查找出用户名=“gg”的信息。

## 4.5 任务四 建立超市管理系统索引

### 4.5.1 相关知识

索引是按某一关键字表达式排序建立的文件，为了加快数据表记录的检索、显示、查询的打印速度，需要对数据表记录进行重新组织。索引如同书的目录，通过目录，读者可以快速找到所

需要的内容,索引可以使用户快速找到表中的数据。实现这个方法有两种:排序和索引。

(1) 排序 (Sort): 创建一个新的数据表 (.dbf) 文件,具有原表文件的全部或部分字段和记录,记录按某种要求重新排列顺序,与原来的表文件是相对独立的。

(2) 索引 (Index): 创建一个索引 (.cdx) 文件,该文件包含排序关键字和记录号字段,不能脱离原表独立使用。通过该文件可以方便地实现对数据表的逻辑排序。索引分为单索引和复合索引。复合索引又分为结构复合索引和非结构复合索引。

### 1. 索引的类型

(1) 主索引。主索引能够唯一地确定数据表中一条记录的字段,或字段表达式。就是说,该字段或字段组合表达式的值,在数据表的所有记录中不能出现重复,是唯一的。每一个数据表只能有一个主索引,自由表不能创建主索引。

(2) 候选索引。候选索引关键字的值也是唯一的,但每个表可以有多个候选索引。

(3) 普通索引。索引关键字段的值不要求具有唯一性,可以重复。表记录排序时,把关键字段值相同的记录排列在一起,并按自然顺序的先后排列,一个表中可以创建多个普通索引。

(4) 唯一索引。索引关键字段的值不要求具有唯一性,可以重复。只将相同关键字值的第一条记录编入索引中。

### 2. 索引文件的类型

Visual FoxPro 中常见的是复合索引 (.CDX) 文件,包括:结构复合索引文件——与表文件同名;独立复合索引文件——任意文件名。

#### (1) 结构复合索引文件。

① 一个数据表只能有一个结构复合索引文件与数据表文件同名,扩展名为 .CDX,包含多个索引项——索引标识 (Tag)。

② 打开数据表时,自动打开;关闭数据表时,自动关闭。更新数据表中的数据时,索引文件中的相关数据自动更新,但依附于数据表,不能独立使用和操作。

③ 只有在结构复合索引文件中,才能创建主索引。

④ 复合索引文件被打开时,其中的索引标识同时被打开但没有主标识。

#### (2) 独立复合索引文件。

① 一个表用 Index 命令可以创建多个独立复合索引文件。

② 索引文件名由用户在创建时指定,扩展名为 .CDX。

③ 需要单独打开,允许单独关闭。

### 3. 索引文件的创建

Visual FoxPro 允许创建两种类型的索引文件,即单索引文件和复合索引文件。在单索引文件中包含一个索引项,默认扩展名为 .IDX。如果有多个索引项存放在同一索引文件中,这种索引文件称为复合索引文件,默认扩展名为 .CDX。如果在建立复合索引文件时默认索引文件名,系统将自动取表文件名作为索引文件名,而其扩展名仍为 .CDX。此类复合索引文件称为结构复合索引文件;否则为非结构复合索引文件。

(1) 使用表设计器。使用表设计器,只能创建结构复合索引文件。

#### ① 单字段索引。

方法:在数据库设计器中打开数据库文件,选定要创建索引的表,并打开表设计器,如图 4-37 所示。

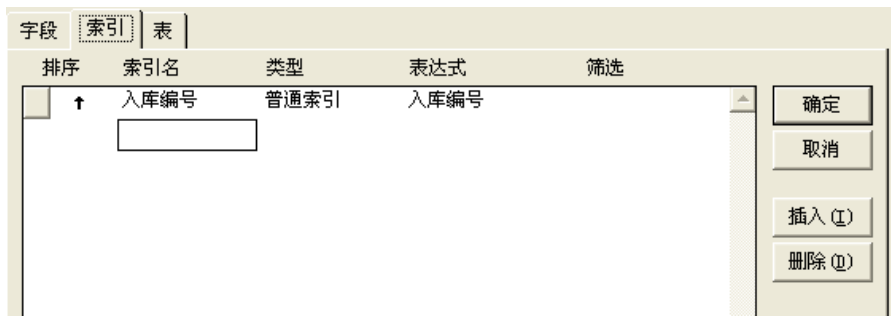


图 4-37 表设计器

选择字段，打开“索引”选项卡，选择排序方式，如图 4-38 所示。这样创建的是普通索引（默认）。



图 4-38 设计索引

创建其他类型索引，切换到“索引”选项卡。  
② 字段表达式索引。多个字段合在一起作为索引关键字，它们必须都变成相同类型——字符型。

方法：在数据库设计器中打开数据库，选定要建索引的表，打开表设计器，选定“索引”选项卡，单击“确定”按钮。

- 说明：
- 索引名：索引标识名，≤10 个字符。
  - 类型：要创建索引标识的类型。
  - 表达式：索引关键字表达式，都转换成 C 型。
  - 筛选：输入条件表达式，选择参加索引的记录。
  - 箭头按钮：在索引名左侧，表示索引顺序：升序、降序。

（2）使用 Index 命令。  
功能：创建结构、独立复合索引文件。不能创建主索引。  
格式：INDEX ON 关键字表达式 TAG 标识名 [OF 独立索引名] [FOR 条件] [ASCENDING|DESCENDING] [ADDITIVE] [UNIQUE |CANDIDATE]

- 说明：
- [ASCENDING|DESCENDING]：升序或降序。
  - [ADDITIVE]：创建独立复合索引时，可以使先前打开的独立索引保持打开。
  - [UNIQUE]：创建唯一索引。
  - [CANDIDATE]：创建结构复合索引中的候选索引。

#### 4. 索引的操作

(1) 打开独立复合索引文件。

① 与数据表同时打开。格式如下：

```
OPEN DATABASE 数据库名  
USE 表名 INDEX 独立复合索引名
```

② 单独打开。格式如下：

```
OPEN DATABASE 数据库名  
USE 表名  
SET INDEX TO 独立复合索引名
```

(2) 关闭独立复合索引文件

① SET INDEX TO

功能：关闭当前工作区所有打开的独立复合索引文件。

② CLOSE INDEXES

功能：关闭所有工作区已打开的独立复合索引文件。

(3) 设置当前索引。

① 命令方式。数据表已经打开，使用命令格式：

```
SET ORDER TO 标识名 [OF 独立复合索引]  
[ASCENDING|DESCENDING]
```

数据表没有打开，使用命令格式：

```
USE 表名 ORDER 标识名 [OF 独立复合索引][ASCENDING|DESCENDING]
```

说明：不论在创建索引项时定义为升序还是降序，可以用选项 ASCENDING 临时指定为升序；用选项 DESCENDING 临时指定为降序。

② 对话框选择方式。

操作步骤：打开数据表，选择菜单“显示”→“浏览”命令。

选择菜单“表”→“属性”命令，打开“工作区属性”对话框。单击“索引顺序”下拉列表框，选择标识项。

单击“确定”按钮，如图 4-39 所示。

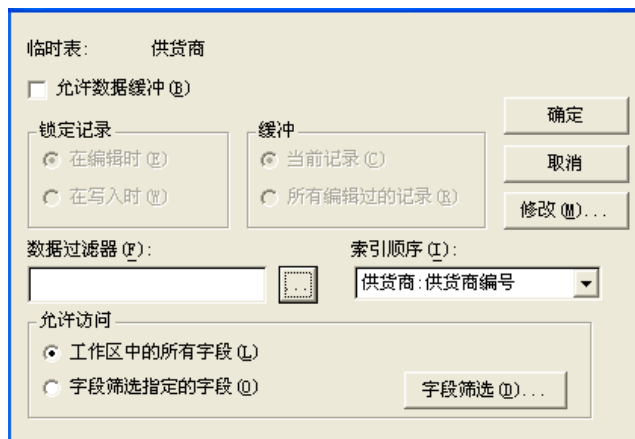


图 4-39 “工作区属性”对话框

### 5. 索引项的删除

(1) 取消当前索引设置。

格式: SET ORDER TO 0            或 SET ORDER TO

功能: 取消主索引标识, 指针指向表文件。

(2) 删除索引标识。

① 删除结构复合索引文件中的索引标识。打开表文件, 打开表设计器, 选择“索引”选项卡, 选定要删除索引标识, 单击“删除”按钮。

② 删除结构、独立复合索引文件中的索引标识。打开数据表, 使用命令:

DELETE TAG ALL|标识1, 标识2    .....

### 6. 重建索引

打开表与索引文件后, 使用命令: REINDEX 即可。

本小节主要技能点如下:

- (1) 建立索引。
- (2) 用多个字段进行排序。
- (3) 对记录进行排序。
- (4) 控制字段中重复值的输入。
- (5) 结构复合索引。
- (6) 索引类型。

## 4.5.2 任务实现

### 1. 建立索引

在数据库设计器中打开数据库文件, 选定要创建索引的表, 并打开表设计器, 选择“索引”选项卡, 建立索引, 如图 4-40 所示。

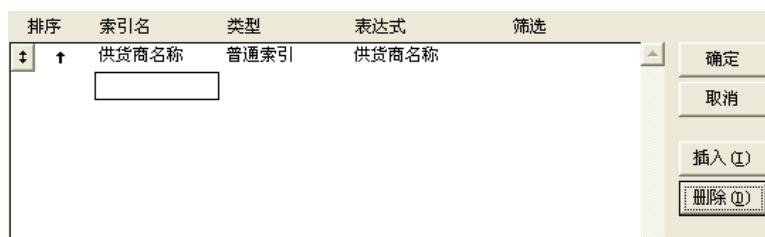


图 4-40 建立的索引

### 2. 用多个字段进行排序

建立索引“供货商编号(升序)”、“供货商名称(降序)”, 如图 4-41 所示, 单击“确定”按钮, 浏览当前表记录, 查看记录显示顺序。在系统主菜单下, 打开“表”菜单, 选择“属性”命令, 进入“工作区属性”对话框。

在“工作区属性”对话框, 打开“索引顺序”下拉列表, 选择索引字段名, 单击“确定”按钮, 表中的数据按所选择索引的次序显示, 如图 4-42 所示。

### 3. 对记录进行排序

操作步骤如下:

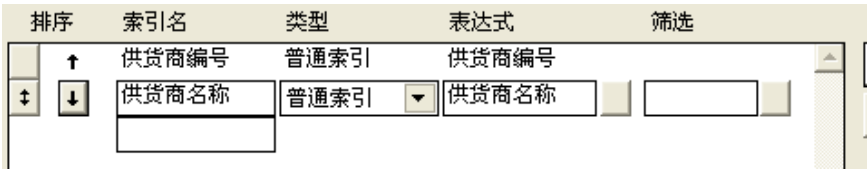


图 4-41 多个字段进行排序



图 4-42 对记录进行排序

- (1) 打开已建好索引的表。
- (2) 单击“浏览”按钮。
- (3) 从“表”菜单中选择“属性”命令。
- (4) 在“索引顺序”下拉列表框中，选择要使用的索引。单击“确定”按钮，如图 4-42 所示。

4. 控制字段中重复

在“索引”选项卡“类型”一栏中，将某一字段设置为“主索引”或“候选索引”就可以控制字段重复值的输入，强制在字段中输入唯一的值，如图 4-43 所示。



图 4-43 设置索引类型控制字段重复

5. 结构复合索引

对供货商.dbf 表建立复合索引，格式如下：

```
INDEX ON 供货商编号 TAG nx
```

6. 索引类型

- (1) 主索引。建立方法：

- ① 打开已建好索引的表。
- ② 选择“浏览”。
- ③ 从“表”菜单中选择“属性”命令。
- ④ 在“类型”栏中，选择要使用的“主索引”。单击“确定”按钮，如图4-44所示。



图 4-44 建立主索引

(2) 候选索引。建立方法：

- ① 打开已建好索引的表。
- ② 选择“浏览”。
- ③ 从“表”菜单中选择“属性”命令。
- ④ 在“类型”栏中，选择要使用的“候选索引”。单击“确定”按钮，如图4-45所示。



图 4-45 建立候选索引

(3) 唯一索引。建立方法：

- ① 打开已建好索引的表。
- ② 选择“浏览”。
- ③ 从“表”菜单中选择“属性”命令。
- ④ 在“类型”栏中，选择要使用的“唯一索引”。单击“确定”按钮，如图4-46所示。



图 4-46 建立唯一索引

(4) 普通索引。建立方法：

- ① 打开已建好索引的表。
- ② 选择“浏览”。
- ③ 从“表”菜单中选择“属性”命令。
- ④ 在“类型”栏中，选择要使用的“普通索引”。单击“确定”按钮，如图4-47所示。



图 4-47 建立普通索引

## 课后练习

1. 对用户表的用户编号字段建立主索引。

2. 对生产商表中的厂商编号字段建立候选索引。

## 4.6 任务五 通过索引查询商品

### 4.6.1 相关知识

索引检索的速度比直接检索的速度快得多。要使用索引检索必须先建立索引文件并打开它。

#### 1. 索引查询命令 (FIND、SEEK)

##### (1) SEEK 命令。

命令: SEEK <索引值>

功能: 索引文件打开的情况下, 在表文件的主控索引中查找与<索引值>的值相匹配的第一条记录。

说明: SEEK 命令可查找字符型、数值型、日期型、逻辑型表达式的值。命令中的表达式必须有定界符, 以确定表达式类型。

##### (2) FIND 命令。

命令: FIND <字符串>/<数值常量>

功能: 在表文件的主控索引中查找定位关键字值与<字符串>或<数值常量>相匹配的第一条记录。

① 如果找到, 则把记录指针指向该记录, 并且将 FOUND()函数的值置为“真”。

② 如果没有找到与其相符的记录, 则将记录指针指向表的末尾, 且将 FOUND()函数的值置为“假”, 将 EOF()函数的值置为“真”。

③ 如果有多个与关键字匹配的记录, 则记录指针定位于其中的第一条记录上, 若查询后继续记录, 可用 SKIP 命令。

④ 执行 SET EXACT OFF 命令后, 用 FIND 命令查找字符串时, 字符串可以是索引表达式值的全部或是从首字符开始的一个子串。

#### 2. 数据的转换

##### (1) 字符型的数据转换数值函数 VAL()。

语法: VAL<字符表达式>

说明: 将以纯数字表示的<字符表达式>或以数字开头的<字符表达式>中的开头数字字符串转换为数值型数据, 精度为小数点后两位, 会四舍五入。如果不是以纯数字表示的或不是以数字开头的字符串, 则转换为 0.00。

##### (2) 数值型数据转换为字符串函数 STR()。

语法: STR(<数值表达式>,[<长度>],[<小数点位数>])

说明: 将<数值表达式>的数值按指定的<长度>和<小数点位数>转换为字符串。<长度>至少要满足整数位的要求; 否则, 转换不成功, 其结果为\*.\*。如果不给定长度, 则默认为 10 位。不给定小数点位数, 则默认为 0 位, 会四舍五入。

##### (3) <字符表达式>转换为日期表达式函数 CTOD()。

语法: CTOD(<字符表达式>)

说明：将<字符表达式>转换为日期表达式。

(4) <日期表达式>转换为字符表达式函数 DTOC()。

语法：DTOC(<日期表达式>,-1)

说明：将<日期表达式>转换为字符表达式。如果选择参数-1，则会删除年、月、日之间的连接符。

(5) 将字符串中所有小写字母转换成大写字母 UPPER 函数。

语法：UPPER(<字符表达式>)

说明：将字符串中所有小写字母转换成大写字母。

(6) 将字符串中所有大写字母转换成小写字母 LOWER 函数。

语法：LOWER(<字符串表达式>)

说明：将字符串中所有大写字母转换成小写字母。

### 3. 常用函数

Visual FoxPro 定义的标准函数十分丰富，共有 200 余个。灵活运用这些函数，不仅可以简化许多运算，而且能够加强 Visual FoxPro 的许多功能。

按函数功能可将函数分为数值型函数、字符型函数、日期和时间函数、数据类型转换函数、字符处理函数、测试函数及宏替换函数。

<函数名>(<参数>)

(1) 数值函数：自变量与函数为数值型。

① 绝对值：ABS(<n1>)

② 取整：INT(<n1>)

③ 平方根：SQRT(<n1>)

④ 最大/小值：MAX/MIN()

⑤ 求余数：MOD(<n1>,<n2>)

(2) 字符函数。字符函数是处理字符型数据的函数，其自变量或函数值中至少有一个是字符型数据。

① 求子串起始位置，整型。

格式：AT(<c1>,<c2>[,n1])

如果串 c2 不包含串 c1，值为零。若有 n1，则找串 c1 在串 c2 中第 n1 次出现的起始位置。

② 取子串，C 型。

取前 n1 个字符：LEFT(<c1>,<n1>)

取后 n1 个字符：RIGHT(<c1>,<n1>)

从 c1 串中第 n1 个字符开始，连续取 n2 个字符：SUBSTR(<c1>,<n1>[,<n2>])

若缺少 n2，则从 c1 串中取第 n1 个字符开始的所有字符。

③ 求串长度，N 型。

格式：LEN(<c1>)

其中汉字为 2 个字符。

④ 删除字符串空格函数，C 型。

删除前面的空格：LTRIM(<c1>)

删除后面的空格: RTRIM | TRIM(<c1>)

删除前面和后面的空格: ALLTRIM (<c1>)

⑤ 生成含 n1 个空格串, C 型。

格式: SPACE (<n1>)

(3) 日期时间函数: 处理日期型或日期时间型数据。

① 系统日期函数, D 型。

格式: DATE( [ <n1>, <n2>, <n3> ] )

功能: 缺省参数时返回当前系统日期; 若有参数则生成一个表示指定日期的值。

② 系统时间函数, C 型。

格式: TIME( [ <n1> ] )

功能: 返回当前系统时间, 时间显示格式为 hh:mm:ss。若选择了 n1, 则不管为何值, 返回的系统时间还包括秒的小数部分, 精确至小数点后两位。

③ 系统日期和时间函数, T 型。

格式: DATETIME([nYear, nMonth, nDay [, nHours [, nMinutes [, nSeconds]]]])

功能: 类似 DATE 函数, 若无参数, 则返回系统的当前日期和时间。若有参数则生成一个表示指定日期和时间的值。

④ 年月日函数: YEAR、MONTH、DAY, N 型。

求年份值: YEAR(<d1>)

求日期值: DAY( <d1> )

求月份值: MONTH(<d1>)

⑤ 星期函数, N 型。

格式: DOW(<d1>)

求 d1 式中星期的数值, 用 1~7 表示星期日至星期六。

⑥ 时分秒函数: HOUR、MINUTE、SEC, N 型。

求小时值: HOUR(<t1>)

求分钟值: MINUTE( <t1> )

求秒钟值: SEC(<t1>)

本小节主要技能点如下:

(1) SEEK 命令。

(2) 数据的转换。

(3) 常用函数。

#### 4.6.2 任务实现

下面用 SEEK 命令查询。先建立索引, 如果已建立索引, 可以直接应用 SEEK 命令。例如:

Use 供货商

Index on 供货商编号 to ghbh

Seek "p001"

DISPLAY

命令窗口如图 4-48 所示, 查询结果如图 4-49 所示。



图 4-48 命令窗口

| 记录号 | 供货商编号 | 供货商名称 | 供货商地址 | 供货商电话    |
|-----|-------|-------|-------|----------|
| 1   | p001  | 超市A   | 大东    | 11122323 |

图 4-49 查询结果

课后练习

利用 SEEK 命令在“生产商”表中查找“产商电话”=110032 的记录。

4.7 任务六 组织查询的输出结果

4.7.1 相关知识

1. 设置排序

排序决定了查询输出结果中记录或行的先后顺序，可以通过“排序依据”选项卡设置查询的排序次序，方法与用查询向导设计查询一样，首先从“选定字段”列表框中选定要使用的字段，并把它们移到“排序条件”列表框中，然后利用“排序选项”框（从中选择“升序”或“降序”单选按钮）设置排序条件。本例中设置商品信息.库存数量，升序为排序依据，如图 4-50 所示。



图 4-50 设置查询排序

如果还需要添加用于排序的字段，可以重复以上的操作步骤。

2. 分组查询结果

所谓分组就是将一组类似的记录压缩成一个结果记录，这样就可以完成基于一组记录的计算。分组在与某些累计功能联合使用时效果最好，如 SUM()、COUNT()、AVG()等，如图 4-51 所示。

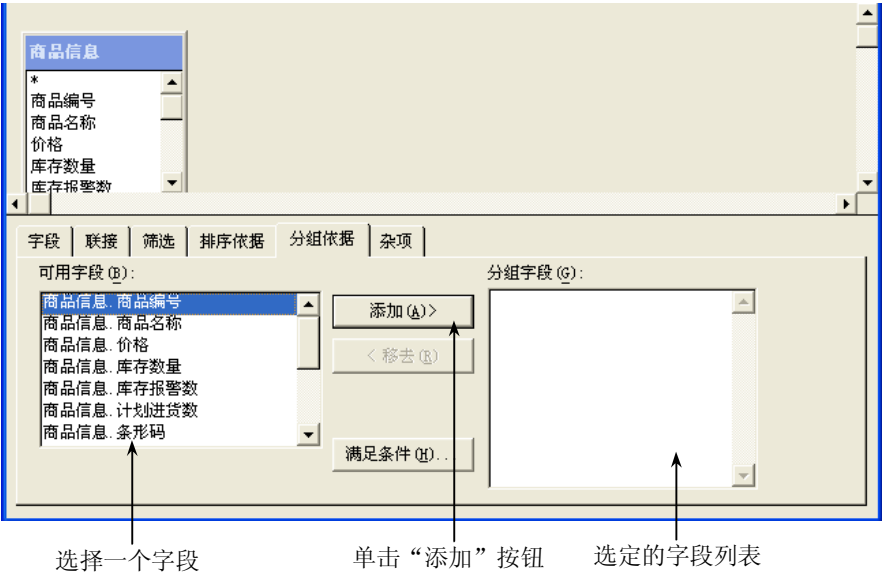


图 4-51 分组设置

3. 分组常用合计函数

命令参数

COUNT(\*)

COUNT(列)

SUM(数值表达式)

AVG(数值表达式)

MIN(表达式)

MAX(表达式)

本小节主要技能点如下：

- (1) 排序查询结果。
- (2) 分组查询结果。
- (3) 选择分组。
- (4) 分组常用合计函数。

说 明

计算表中记录的总数

对一系列中的值计算个数

计算数值表达式的和

计算数值表达式的平均值

求（数值、日期、字符）最小值

求（数值、日期、字符）最大值

4.7.2 任务实现

1. 排序查询结果

排序决定查询输出结果中记录显示的顺序。

例如，对会员表建立排序。

设置方法：选中“排序依据”选项卡，从“选定字段”列表框选中字段，再选择升序或

降序，单击“添加”按钮，如图 4-52 所示。

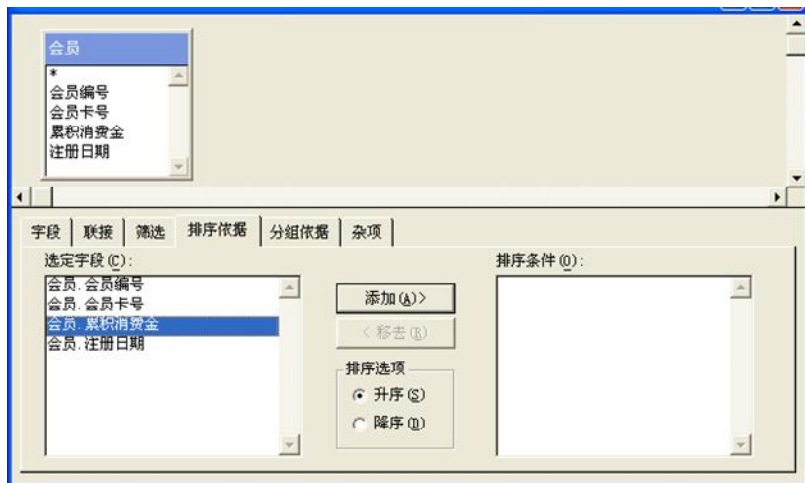


图 4-52 对会员表建立排序

SQL 语句格式：SELECT \* FROM database!会员 GROUP BY 会员.累积消费金

## 2. 分组查询结果

分组是指将一组类似的记录压缩成一个结果记录，目的是为了完成基于该组记录的计算，比如求平均值、总和、统计个数、最大值、最小值等。

例如，对会员表建立分组。设置方法：选择“分组依据”选项卡，从“可用字段”列表框选中字段，单击“添加”按钮，如图 4-53 所示。

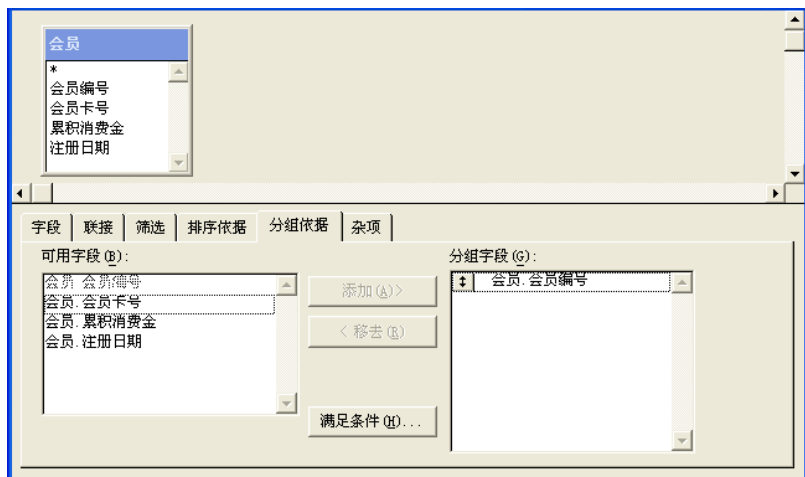


图 4-53 会员表建立分组

SQL 语句格式：SELECT \* FROM database!会员 GROUP BY 会员.会员编号

## 3. 选择分组

对会员表建立选择分组，查找出累积消费金大于 80 的记录。SQL 语句格式：

SELECT \* FROM database!会员 GROUP BY 会员.累积消费金 HAVING 会员.累积消费金 > 80  
查询结果如图 4-54 所示。



| 会员编号 | 会员卡号   | 累积消费金   | 注册日期     |
|------|--------|---------|----------|
| m001 | 080627 | 8000.00 | 02/22/08 |

图 4-54 查询结果

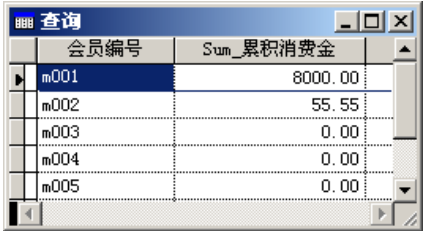
4. 分组常用合计函数

分组是指一组类似的记录压缩成一个结果记录，目的是为了完成基于该组的总和、统计个数以及最大值、最小值等。

对会员表建立选择分组合计，其 SQL 语句格式：

```
SELECT 会员编号, SUM(累积消费金) FROM database!会员 GROUP BY 会员.会员编号
```

查询结果如图 4-55 所示。



| 会员编号 | Sum_累积消费金 |
|------|-----------|
| m001 | 8000.00   |
| m002 | 55.55     |
| m003 | 0.00      |
| m004 | 0.00      |
| m005 | 0.00      |

图 4-55 查询结果

课后练习

- 1. 利用查询设计器对“供货商”表按照供货商编号进行分组。
- 2. 利用 SQL 语言实现对“商品信息”表中的库存数量总计，并且按商品编号分组统计。

4.8 任务七 联接多个表的查询

4.8.1 相关知识

有时需要从多个表中访问数据。在这些表中存在公共字段，可以基于公共字段提取所需数据。在这种情况下，必须将这两个表联接起来以获得需要的结果。根据一个公共字段将关系数据库中的两个或多个表关联起来，称为联接。

1. 联接类型

(1) 内部联接 (Inner Join)：只有两个表的字段都满足联接条件时，才将此记录选入查询结果中。

(2) 左联接 (Left Outer Join)：联接条件左边表中的记录都包含在查询结果中，而右边表中的记录只有满足联接条件时才选入查询结果中。

(3) 右联接 (Right Outer Join)：联接条件右边表中的记录都包含在查询结果中，而左边表中的记录只有满足联接条件时才选入查询结果中。

(4) 完全联接 (Full Join): 无论两个表中的记录是否满足联接条件, 都将全部记录选入查询结果中。

## 2. 联接查询方法

通常有 3 种方法实现多表联接查询: 一是在查询命令中显式地指定联接类型; 二是在查询命令中使用 WHERE 子句; 三是使用嵌套查询。

(1) 指定联接类型实现多表间的联接查询。

格式: SELECT...FROM <表或视图> INNER|LEFT|RIGHT|FULL JOIN <表或视图> ON <联接条件>WHERE...

(2) 用 WHERE 子句实现多表间的联接查询。用 WHERE 子句实现多表间的联接查询时, 无须直接指明联接类型 (隐含是内部联接), 只须把联接条件直接写入 WHERE 子句即可。

(3) 使用嵌套查询实现多表间的联接查询。在 SQL 语言中, 由 SELECT、FROM、WHERE 语句组成一个查询块。嵌套查询就是将第二个查询块放在第一个查询块的 WHERE 条件中, 形成外层 (第一个) 查询包含内层 (第二个) 查询的嵌套查询。外层查询也称为主查询、父查询, 内层查询也称为下层查询、子查询。系统对嵌套查询的处理过程是先做子查询, 在子查询的基础上再做主查询。在子查询中还可以再包含更下层的查询, 从而形成层嵌套查询。

本小节主要技能点如下:

- (1) 在查询中添加表或视图。
- (2) 用联接控制记录的选择。
- (3) 创建表之间的联接。
- (4) 删除联接。
- (5) 修改联接。
- (6) 联接类型。
- (7) 查看 SQL 语句。

## 4.8.2 任务实现

### 1. 在查询中添加表或视图

(1) 先选择菜单命令“文件”→“新建”, 在“新建”对话框中选中“查询”单选按钮, 打开查询设计视图。

(2) 选择添加表, 如图 4-56 所示。



图 4-56 添加表或视图

- (3) 选择“商品信息”表和“生产商”表。
  - 2. 创建表之间的联接
- 对“商品信息”表和“生产商”表建立左联接，如图 4-57 所示。

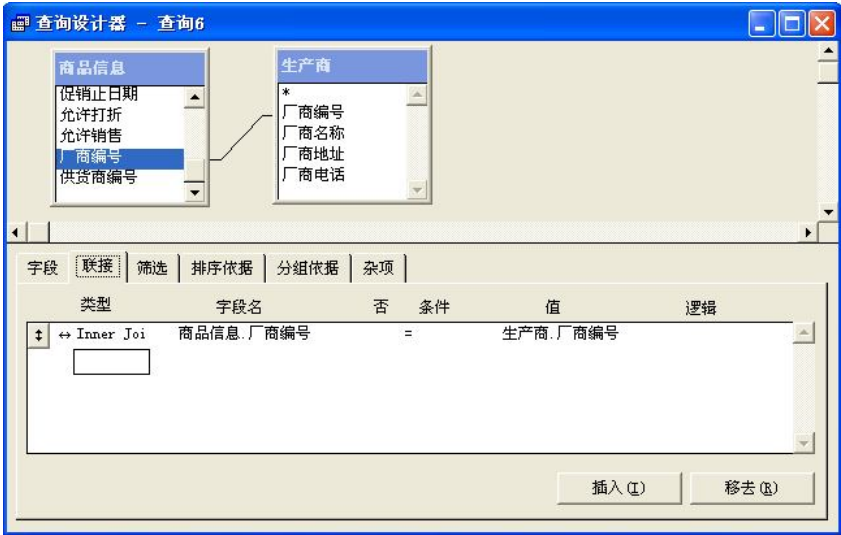


图 4-57 对“商品信息”表和“生产商”表建立左联接

- 3. 删除联接
- 对“商品信息”表和“生产商”表删除联接，单击 **移除(R)** 按钮。
- 4. 修改联接
- 对“商品信息”表和“生产商”表修改联接，如图 4-58 所示。

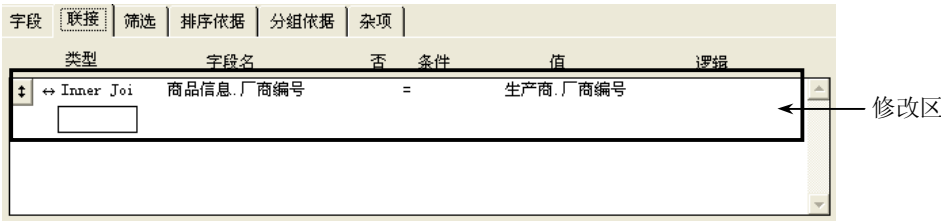


图 4-58 对“商品信息”表和“生产商”表修改联接

- 5. 联接类型
- (1) 对“商品信息”表和“生产商”表建立内部联接 (Inner Join): 在“类型”中选择内部联接，如图 4-59 所示，结果如图 4-60 所示。

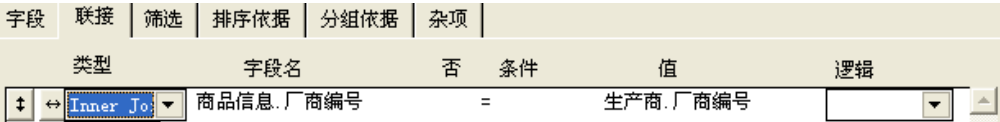


图 4-59 “表商品信息”表和“生产商”表建立内部联接

| 商品编号 | 商品名称 | 价格         | 厂商名称 |
|------|------|------------|------|
| m002 | 小蛋糕  | 10.00000   | 啊美丽卡 |
| m003 | 生日蛋糕 | 10.00000   | 稻香村  |
| m005 | 电脑   | 1000.00000 | 皇威   |
| m006 | 洗衣机  | 2000.00000 | 乐邦   |
| m008 | 手表   | 330.00000  | 乐邦   |
| m001 | 音箱   | 30.00000   | 乐邦   |
| m010 | 依波手表 | 500.00000  | 美的   |
| m007 | 石英钟  | 200.00000  | 美的   |
| m009 | 自行车  | 300.00000  | 三洋   |

图 4-60 用内部联接控制的查询结果

(2) 对“商品信息”表和“生产商”表建立左联接 (Left Outer Join): 在“类型”中选择左联接, 如图 4-61 所示。

| 字段       | 联接        | 筛选 | 排序依据     | 分组依据 | 杂项 |
|----------|-----------|----|----------|------|----|
| 类型       | 字段名       | 否  | 条件       | 值    | 逻辑 |
| Left Out | 商品信息.厂商编号 | =  | 生产商.厂商编号 |      |    |

图 4-61 对“商品信息”表和“生产商”表建立左联接

(3) 对“商品信息”表和“生产商”表建立右联接 (Right Outer Join): 在“类型”中选择右联接, 如图 4-62 所示。

| 字段        | 联接        | 筛选 | 排序依据     | 分组依据 | 杂项 |
|-----------|-----------|----|----------|------|----|
| 类型        | 字段名       | 否  | 条件       | 值    | 逻辑 |
| Right Out | 商品信息.厂商编号 | =  | 生产商.厂商编号 |      |    |

图 4-62 对“商品信息”表和“生产商”表建立右联接

(4) 对“商品信息”表和“生产商”表建立完全联接 (Full Join): 在“类型”中选择完全联接, 如图 4-63 所示。

| 字段        | 联接        | 筛选 | 排序依据     | 分组依据 | 杂项 |
|-----------|-----------|----|----------|------|----|
| 类型        | 字段名       | 否  | 条件       | 值    | 逻辑 |
| Full Join | 商品信息.厂商编号 | =  | 生产商.厂商编号 |      |    |

图 4-63 对“商品信息”表和“生产商”表建立完全联接

#### 6. 用联接控制记录的选择

设计如图 4-64 所示的联接。

| 字段       | 联接        | 筛选 | 排序依据     | 分组依据 | 杂项 |
|----------|-----------|----|----------|------|----|
| 类型       | 字段名       | 否  | 条件       | 值    | 逻辑 |
| Left Out | 商品信息.厂商编号 | =  | 生产商.厂商编号 |      |    |

图 4-64 设计联接

7. 查看 SQL 语句

右击，在弹出的快捷菜单中选择“查看 SQL 语句”命令。

- (1) 对表“商品信息”和“生产商”建立内部联接（Inner Join）的 SQL 语句：  
SELECT 商品信息.商品编号, 商品信息.价格, 生产商.厂商名称, 生产商.厂商编号 FROM database!商品信息 INNER JOIN database!生产商 ON 商品信息.厂商编号 = 生产商.厂商编号
- (2) 对表“商品信息”和“生产商”建立左联接（Left Outer Join）的 SQL 语句：  
SELECT 商品信息.商品编号, 商品信息.价格, 生产商.厂商名称, 生产商.厂商编号 FROM database!商品信息 LEFT OUTER JOIN database!生产商 ON 商品信息.厂商编号 = 生产商.厂商编号
- (3) 对表“商品信息”和“生产商”建立右联接（Right Outer Join）的 SQL 语句：  
SELECT 商品信息.商品编号, 商品信息.价格, 生产商.厂商名称, 生产商.厂商编号 FROM database!商品信息 RIGHT OUTER JOIN database!生产商 ON 商品信息.厂商编号 = 生产商.厂商编号
- (4) 对表“商品信息”和“生产商”建立完全联接（Full Join）的 SQL 语句：  
SELECT 商品信息.商品编号, 商品信息.价格, 生产商.厂商名称, 生产商.厂商编号 FROM database!商品信息 FULL JOIN database!生产商 ON 商品信息.厂商编号 = 生产商.厂商编号

课后练习

利用查询设计器为“商品信息”表和“生产商”表建立内部联接。

4.9 任务八 定制查询

4.9.1 相关知识

1. 比较关系运算

关系表达式是由关系运算符将数值表达式或者字符表达式或者日期表达式连接起来的表达式，其返回结果为逻辑值真（.T.）或逻辑值假（.F.）。关系运算表如表 4-3 所示。

表 4-3 关系运算表

| 运算符     | 功能    | 表达式                    | 运算结果 |
|---------|-------|------------------------|------|
| <       | 小于    | 15<4*6                 | .T.  |
| >       | 大于    | 'A'>'I'                | .T.  |
| =       | 等于    | 2+4=3*5                | .F.  |
| <>、#、!= | 不等于   | 5<>-10                 | .T.  |
| <=      | 小于或等于 | 'abc'<='AB'            | .F.  |
| >=      | 大于或等于 | {10-10-02}>={10/01/02} | .T.  |
| ==      | 字符串恒同 | 'abc'== 'abcabc'       | .F.  |

2. 输出查询结果

通常如果不选择查询结果的去向，系统默认将查询检索的结果显示在“浏览”窗口中，

也可以选择其他输出目的地,将查询结果送往指定的地点。查询输出目的可以是浏览窗口、临时表、表、图形、屏幕、报表、标签等。选择结果的去向方法如下:

单击“查询设计器”工具中的“查询去向”按钮,或者从“查询”菜单中选择“查询去向”命令,可看到如图 4-65 所示的“查询去向”对话框。



图 4-65 “查询去向”对话框

“浏览”: 在浏览窗口中显示查询结果,这是查询默认设置。

“临时表”: 将查询结果存储在一个临时只读表中。多次查询的结果可放在不同的表内。该表可用于浏览数据、制作报表等,直到用户关闭它们。

“表”: 将查询结果保存在一个命名的表(.DBF)中,此时的查询结果是真正地存放到磁盘上的,多次查询的结果可以放在不同的表内。

“图形”: 使查询结果可用于 Microsoft Graph (包含在 Visual FoxPro 6.0 中的一个独立的应用程序)中制作图表;

“屏幕”: 在 Visual FoxPro 主窗口或当前活动输出窗口中显示查询结果。

“报表”: 将输出送到一个报表文件(.FRX)中。

“标签”: 将输出送到一个标签文件(.LBX)中。

选定一个去向,按一定的步骤设置一些属性,然后单击“确定”按钮,系统就将按意图放置查询结果。

本小节主要技能点如下:

- (1) 缩小和扩充搜索。
- (2) 比较关系运算。
- (3) 定向输出查询结果。
- (4) 查看查询的 SQL 语句。
- (5) 在查询中删除重复记录。
- (6) 查询一定数目或一定百分比的极值记录。
- (7) 在查询输出中添加表达式。
- (8) 用表达式筛选。
- (9) 查看 SQL 语句。

4.9.2 任务实现

1. 比较关系运算

对“销售”表进行比较关系查询操作，选择销售数量字段大于 50，如图 4-66 所示。



图 4-66 对设计器进行设置

查询结果如图 4-67 所示。

| 销售编号          | 商品编号 | 销售日期     | 销售数量   | 销售金额 |
|---------------|------|----------|--------|------|
| 2008871652    | ee   | 08/07/08 | 123    | 123  |
| 2008827154422 | 44   | 08/27/08 | 55     | 55   |
| 20088281556   | ss   | 08/28/08 | 121123 | 1231 |
| 20088281556   | dd   | 08/28/08 | 222    | 1111 |

图 4-67 查询结果

2. 定向输出查询结果

对“销售”表进行比较关系查询操作结果，定向输出。

选择菜单“查询”→“查询去向”命令，打开“查询去向”对话框，如图 4-68 所示。



图 4-68 “查询去向”对话框

单击“表”图标按钮，填写表名，运行查询之后，结果以表的形式出现，如图 4-69 所示。

3. 查看查询的 SQL 语句

选择菜单“查询”→“查询 SQL”命令。其语句格式：

```
SELECT * FROM database!销售 WHERE 销售.销售数量 > 50 INTO TABLE 关系查询.dbf
```

4. 在查询中删除重复记录

如“销售”表有重复记录，如图 4-70 所示。

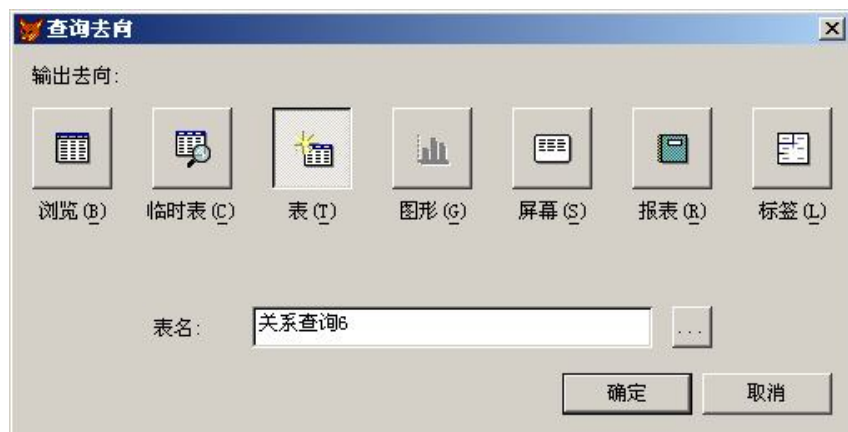


图 4-69 保存查询结果

| 销售编号          | 商品编号 | 销售日期     | 销售数量   | 销售单额 |
|---------------|------|----------|--------|------|
| 2008827154620 | w    | 08/27/08 | 11     | 11   |
| 2008827154620 | ee   | 08/27/08 | 22     | 11   |
| 2008827154647 | ss   | 08/27/08 | 3      | 3    |
| 2008827154647 | ddd  | 08/27/08 | 33     | 1    |
| 2008827154834 | ddd  | 08/27/08 | 1      | 2    |
| 2008827154834 | d2   | 08/27/08 | 22     | 1    |
| 20088281556   | ss   | 08/28/08 | 121123 | 1231 |
| 20088281556   | dd   | 08/28/08 | 222    | 1111 |
| 2008827154620 | ee   | 08/27/08 | 22     | 11   |

重复记录

图 4-70 “销售”表有重复记录

去除重复记录的方法：在 SELECT 后面加入 DISTINCT，SQL 语句格式如下：

```
SELECT DISTINCT 销售.销售编号, 销售.商品编号, 销售.销售数量 FROM database!销售 WHERE
销售.商品编号 = "ee" INTO TABLE 去重复查询记录.dbf
```

#### 5. 查询一定数目或一定百分比的极值记录

(1) 对“销售”表进行比较关系查询一定数目。选择“杂项”选项卡，进行如图 4-71 所示的设置。

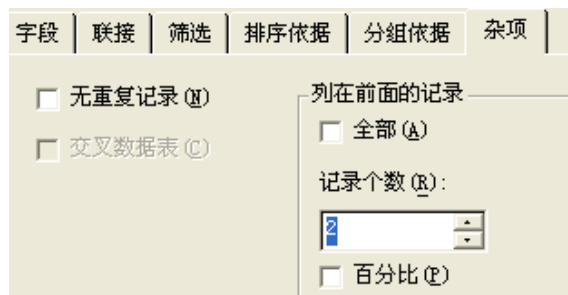


图 4-71 “记录个数”设为 2

查询结果如图 4-72 所示。

(2) 对“销售”表进行比较关系查询设置百分比。选择“杂项”选项卡，进行如图 4-73 所示的设置。

|   | 销售编号          | 商品编号 | 销售日期     | 销售数量 | 销售单额 |
|---|---------------|------|----------|------|------|
| ▶ | 2008826151    | p001 | 08/26/08 | 20   | 0    |
|   | 2008826151350 | p002 | 08/26/08 | 10   | 0    |

图 4-72 查询结果

字段 | 联接 | 筛选 | 排序依据 | 分组依据 | 杂项

☐ 无重复记录 (U)  
☐ 交叉数据表 (C)

列在前面的记录  
☐ 全部 (A)  
百分比 (P):  
2.01  
☒ 百分比 (P)

图 4-73 “杂项”选项卡

查询结果如图 4-74 所示。

|   | 销售编号       | 商品编号 | 销售日期     | 销售数量 | 销售单额 |
|---|------------|------|----------|------|------|
| ▶ | 2008826151 | p001 | 08/26/08 | 20   | 0    |
|   |            |      |          |      |      |

图 4-74 查询结果

6. 在查询输出中添加表达式

对“销售”表查询输出中添加表达式。

(1) 选择“新建”→“设计视图”命令，选择“销售”表。添加所有显示的字段，如图 4-75 所示。

字段 | 联接 | 筛选 | 排序依据 | 分组依据 | 杂项

可用字段 (F):  
销售.销售编号  
销售.商品编号  
销售.销售日期  
销售.销售数量

函数和表达式 (E):

添加 (A)>  
全部添加 (A)>>  
< 移去 (R)  
<< 全部移去 (L)

选定字段 (S):

图 4-75 添加所有显示的字段

(2) 单击“函数和表达式”右侧按钮，弹出“表达式生成器”对话框，如图 4-76 所示。

(3) 在“表达式”列表框中输入：销售数量>50（表示为查询增加字段，用于判断销售数量的值是否大于 50），运行结果如图 4-77 所示。



图 4-76 “表达式生成器”对话框

|  | 销售编号       | 商品编号  | 销售日期     | 销售数量 | 销售单额 | Exp_6 |  |
|--|------------|-------|----------|------|------|-------|--|
|  | xxx        | xxx   | 01/01/08 | 0    | 1222 | F     |  |
|  | 啊啊啊啊       | 21231 | 01/21/11 | 0    | 3441 | F     |  |
|  | 2008871855 | ee    | 08/07/08 | 123  | 123  | T     |  |
|  | 2008871855 | 22    | 08/07/08 | 12   | 22   | F     |  |
|  | 2008881434 | s     | 08/08/08 | 1    | 2    | F     |  |
|  | 2008881434 | 3     | 08/08/08 | 2    | 2    | F     |  |
|  | 2008881439 | 啊     | 08/08/08 | 12   | 12   | F     |  |
|  | 2008881439 |       | 08/08/08 | 0    | 0    | F     |  |

图 4-77 运行结果

7. 用表达式筛选

对“商品信息”表运用表达式筛选，筛选出商品编号字段以 d 开头的记录。

- (1) 选择菜单“查询”→“筛选”命令。
- (2) 在“字段名”下拉列表框中，选择“表达式”，弹出“表达式生成器”对话框。
- (3) 在“表达式生成器”中，输入“LEFT(商品信息.商品编号,1)”，单击“确定”按钮，设置“筛选”选项卡，如图 4-78 所示。

| 字段 | 联接 | 筛选                | 排序依据 | 分组依据 | 杂项     |
|----|----|-------------------|------|------|--------|
|    |    | 字段名               | 否 条件 | 实例   | 大小写 逻辑 |
| ↑  |    | LEFT(商品信息.商品编号,1) | =    | d    | [无]    |

图 4-78 “筛选”选项卡

运行所做的筛选查询，运行结果如图 4-79 所示。

| 商品编号 | 商品名称 | 库存数量     | 价格      |
|------|------|----------|---------|
| ddl  | 自行车  | 11.00000 | 0.20000 |

图 4-79 运行结果

### 8. 查看 SQL 语句

对“商品信息”表运用表达式筛选，筛选出商品编号字段以 d 开头的记录，SQL 语句如下：  
 SELECT 商品信息.商品编号, 商品信息.商品名称, 商品信息.库存数量, 商品信息.价格 FROM  
 database!商品信息 WHERE LEFT(商品信息.商品编号,1) = "d"

## 课后练习

利用查询设计器在“商品信息”表查找“库存数量”=10 的记录，并且查询结果以表的形式输出。

## 4.10 任务九 生成用户使用的订单视图

### 4.10.1 相关知识

视图是一个类似于目录的有关数据的虚拟表或者逻辑表，视图中的数据来源于数据库中的表或者其他视图。它具有普通表的一般性质，可以对它进行浏览、修改和使用。值得一提的是，利用视图修改的结果可以送回数据源进行永久保存。但视图依赖于数据库而存在，在新建视图之前，必须先打开相关数据库。在应用程序设计中，创建自定义并且可更新的数据集合，需要使用视图。视图兼有表和查询的特点：与查询相类似的是，视图可以用来从一个或多个相关联的表或视图中提取有用信息；与表相类似的是，视图可以用来更新其中的信息，并将更新结果永久保存在磁盘上。可以用视图使数据暂时从数据库中分离成为游离数据，以便在主系统之外收集和修改数据。

#### 1. 在开发中使用视图的优点

(1) 视点集中：视图机制能使用户把注意力集中在所关心的数据上，使用户看到的数据结构简单而直截了当。

(2) 简化操作：视图可以把若干张表或视图连接在一起，为用户隐蔽了表与表、表与视图、视图与视图之间的连接操作。

(3) 多角度：视图机制可使不同用户从多角度处理同一数据，当许多不同种类用户使用同一个集成数据库时，这种灵活性显然是很重要的。

(4) 安全性：可针对不同的用户形成不同的视图窗口，使不同的用户了解不同的数据，对数据的安全保密性起到很大作用。

#### 2. 视图和查询之间的区别

(1) 利用查询设计器生成的是.QPR 文件，它是完全独立的，不依赖于任何数据库和表而存在，而视图则依赖于数据库而存在。

(2) 利用查询不能更新数据源，而利用视图可以更新数据源。

(3) 查询的去向有 7 种，而视图只是一个虚拟的表。

### 3. 视图的深入讨论

(1) 视图是基于表或视图建立的，当基础表记录内容发生变化时，视图的内容随之发生更新。

(2) 用视图组织数据时，可组织成任何较易操作的形式。

(3) 用视图还可以组织其他形式的数据，在具体应用中加以研究、组织数据。

(4) 视图设计器中新增字段不能改变列表头，当原有字段通过属性按钮修改列表头后，显示视图结果时列表头为修改后的列表头。

本小节主要技能点如下：

- (1) 使用“视图向导”。
- (2) 启动“视图设计器”。
- (3) 创建本地视图。
- (4) 在视图中使用多个表。
- (5) 在视图中添加表达式。
- (6) 连接远程数据。
- (7) 创建连接。
- (8) 视图的实际内容。

### 4.10.2 任务实现

#### 1. 创建本地视图

利用视图向导为“生产商”表建立本地视图，具体步骤如下：

- (1) 单击“视图向导”图标按钮，如图 4-80 所示。



图 4-80 视图向导对话框

- (2) 进入字段选取对话框，如图 4-81 所示进行设置。

(3) 单击“下一步”按钮，利用视图向导建立本地视图，进入“筛选记录”对话框，如图 4-82 所示。

(4) 单击“下一步”按钮，利用视图向导建立本地视图，进入“排序记录”对话框，如图 4-83 所示。



图 4-81 字段记录对话框



图 4-82 “筛选记录”对话框

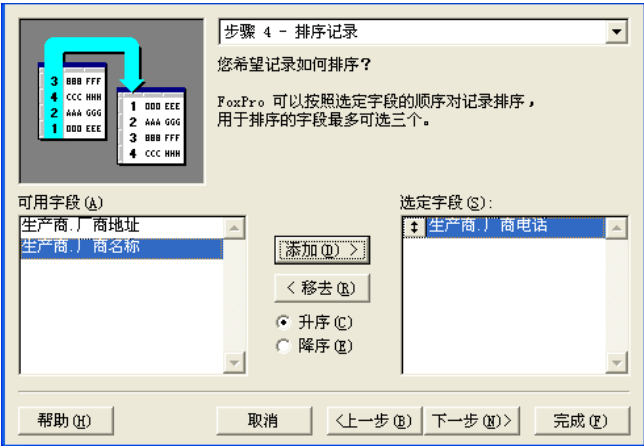


图 4-83 “排序记录”对话框

(5) 单击“下一步”按钮，利用视图向导建立本地视图，出现“限制记录”对话框，如图 4-84 所示。

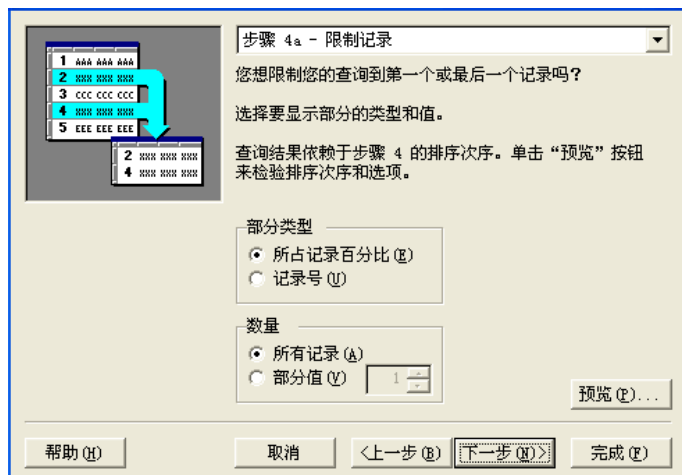


图 4-84 “限制记录”对话框

(6) 单击“下一步”按钮，利用视图向导建立本地视图，单击“完成”按钮，在弹出的“视图名”窗口中输入视图名称，如图 4-85 和图 4-86 所示。



图 4-85 输入视图名称



图 4-86 输入视图名称

## 2. 在视图中使用多个表

(1) 选择“新建”→“新建视图”命令，打开视图设计器中的“添加表或视图”对话框，如图 4-87 所示。



图 4-87 “添加表或视图”对话框

(2) 选择要添加的表或视图，会弹出“联接条件”对话框，如图 4-88 所示。

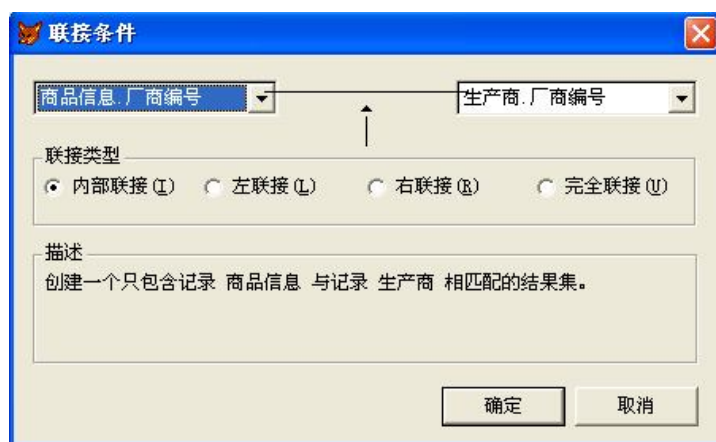


图 4-88 “联接条件”对话框

选择“联接类型”，最后建立多表视图。

## 3. 在视图中添加表达式

对“生产商”表的视图添加表达式，为视图新增加字段。

所谓新字段是在视图设计器中左边选取框中所没有的字段，增加新字段方法是单击函数和表达式按钮，在表达式生成器屏幕的表达式框中输入正确的表达式，单击“确定”按钮，将该表达式添加到选定字段框中，同时新视图为该字段自动取一个新名字。

(1) 选择“新建”→“新建视图”命令，打开视图设计器，添加“生产商”表。

(2) 选择可用字段，运行本视图，得到运行结果，如图 4-89 所示。

| 厂商编号 | 厂商名称 | 厂商地址 |
|------|------|------|
| f001 | 好利来  | 黄姑   |
| f002 | 啊美丽卡 | 大东   |
| f003 | 稻香村  | 沈阳   |
| f005 | 书博尔  | 沈阳   |
| f006 | 格兰仕  | 铁西   |
| f007 | 皇威   | 沈河   |

图 4-89 运行结果

(3) 选择  按钮，弹出“表达式生成器”对话框，如图 4-90 所示。



图 4-90 “表达式生成器”对话框

(4) 在“表达式生成器”中输入“生产商.厂商编号 + "444"”（表示把“生产商.厂商编号”字段连接上字符 444），再单击“确定”按钮，返回视图设计器，单击“添加”按钮，如图 4-91 所示。



图 4-91 添加表达式后的设计视图

(5) 运行结果如图 4-92 所示。

|   | 商编号  | 商名称  | 商地址 | Exp_4    |
|---|------|------|-----|----------|
| ▶ | f001 | 好利来  | 黄姑  | f001 444 |
|   | f002 | 啊美丽卡 | 大东  | f002 444 |
|   | f003 | 稻香村  | 沈阳  | f003 444 |
|   | f005 | 书博尔  | 沈阳  | f005 444 |
|   | f006 | 格兰仕  | 铁西  | f006 444 |
|   | f007 | 皇威   | 沈河  | f007 444 |

图 4-92 运行结果

4. 连接远程数据

建立和 Access 数据库中的表进行连接。

(1) 找到项目 supermarket.pjx，在“项目管理器”中选择“远程视图”，单击“新建”按钮，弹出“选择连接或数据源”对话框，如图 4-93 所示。

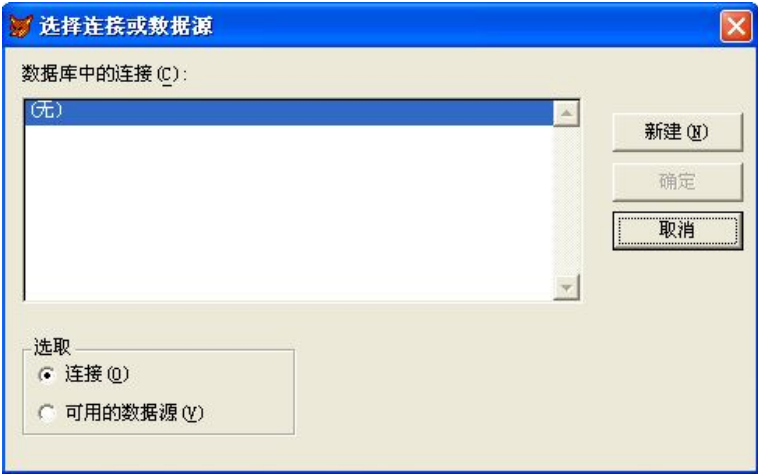


图 4-93 “选择连接或数据源”对话框

(2) 单击“新建”按钮，弹出“连接设计器”对话框，如图 4-94 所示。

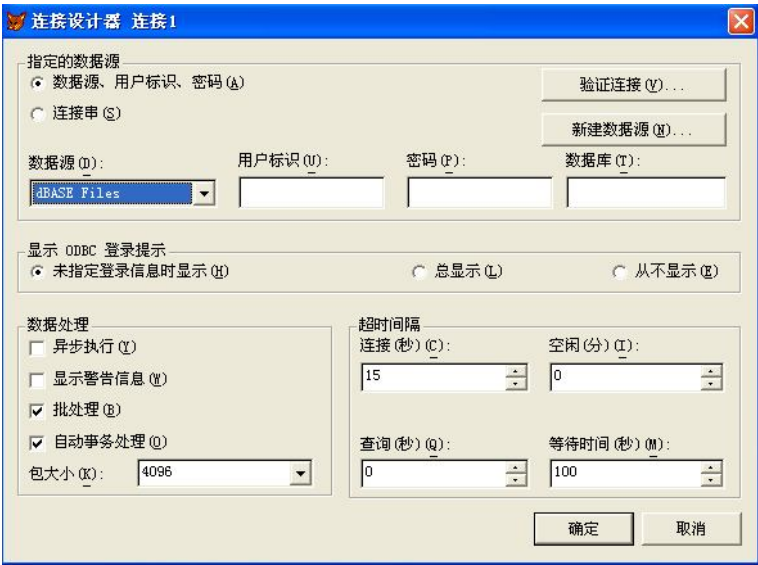


图 4-94 “连接设计器”对话框

(3) 在“数据源”下拉列表框中选择 MS Access Database 选项。单击“新建数据源”按钮，弹出“ODBC 数据源管理器”对话框，选择 MS Access Database 选项，如图 4-95 所示。



图 4-95 “ODBC 数据源管理器”对话框

(4) 双击 MS Access Database 选项，弹出“ODBC Microsoft Access 安装”对话框，如图 4-96 所示。



图 4-96 Access 安装对话框

(5) 单击“选择”按钮，弹出“选择数据库”对话框，如图 4-97 所示。

(6) 选择所需要的数据库，单击“确定”按钮，返回到“ODBC 数据库管理器”对话框，单击“确定”按钮，返回“连接设计器”对话框，单击“验证连接”按钮，弹出验证成功对话框，单击“确定”按钮，会弹出“保存”对话框，输入保存的名称，单击“确定”按钮，进入“视图设计器”窗口，选择表进行添加。

(7) 对选项卡进行设置，最后保存文件。



图 4-97 “选择数据库”对话框

5. 创建连接

（1）打开数据库的项目管理器，选择连接，之后新建连接。弹出“连接设计器”对话框，如图 4-98 所示。



图 4-98 “连接设计器”对话框

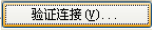
（2）单击  按钮，如果连接成功，会弹出如图 4-99 所示的对话框。



图 4-99 验证连接成功对话框

（3）单击“确定”按钮，然后关闭对话框，会弹出“保存”对话框，如图 4-100 所示。单击“确定”按钮，建立连接。



图 4-100 “保存”对话框

## 课后练习

1. 利用视图向导为“商品信息”表建立视图。
2. 利用视图设计器为“商品信息”表建立视图，并且用表达式输出商品编号+333，形成新字段。
3. 与 MS Access 建立连接。

## 4.11 任务十 排序

### 4.11.1 相关知识

在向新建的表输入数据时，表中记录的顺序是按其输入的先后顺序而定的，因此称之为物理顺序。但有时需要按某种要求重新安排记录的顺序。重新安排记录顺序的方法有两种：一种为排序，从物理上重新组织并生成新的表文件；另一种为索引，它通常按照表中某些关键字的值来建立原数据文件的索引文件，达到在使用中将记录按顺序排列的目的，使用索引的方法排序不从物理上重新组织文件，即不会生成新表。

排序（Sort）就是按表中的字段重新排列记录顺序，按关键字的值排序，从小到大称为升序，从大到小称为降序。排序后生成一个新表，新表的记录按新的物理顺序排列。排序又称为分类。

命令格式：SORT TO <新文件名> ON <字段名 1> [/A/D] [/C][<字段名 2> [/A/D] [/C] ...] [ASCENDING | DESCENDING] [<范围>] [FOR <条件 1>] [WHILE <条件 2>] [FIELDS <字段名表> | FIELDS LIKE 通配字段名 | FIELDS EXCEPT 通配字段名]

功能：将当前数据表按指定的一个或多个字段进行排序，并将排序的结果保存到一个新的数据表中。字段名必须是当前表文件中的，一般为 C、N、L、D、F 型。

#### 1. 单关键字排序

只有一个排序关键字。

说明：/A 表示升序，/D 表示降序，/C 表示字母不分大小写。

/A、/D 不可以连用。都不用时，默认是升序。

排序规则：对 N 型和 F 型，按数值的大小排序；对 C 型，按字符的 ASCII 码值大小排列；对 D 型，按年、月、日的大小顺序排序；对 L 型，按“真”、“假”值的大小排列，即“假”<“真”。

#### 2. 组合关键字排序

有多个排序关键字，之间用“,”号间隔。先按<字段 1>排序，其值完全相同时再按<字段

名 2>排序，依此类推。

3. 选择性排序

(1) 排序范围选择。

[范围]子句，有 4 个选项值（默认值是 All）：

- ① RECORD <N>：指定第 N 个记录。
- ② NEXT <N>：从当前记录开始的连续 N 个记录。
- ③ ALL：表示表文件的所有记录。
- ④ REST：表示从当前记录开始到文件结束的所有记录。

(2) 排序条件选择。

[FOR <条件表达式>]：指定所给范围内满足条件的记录才参加排序。默认值是 All。

(3) 排序字段选择。

[FIELDS <字段名 1，字段名 2，...>]：限制新表中的字段。缺省时是具有原表全部字段。

本小节主要技能点如下：

- (1) 建立索引的命令。
- (2) 用索引对记录排序。
- (3) 排序命令。
- (4) 打开数据库。

4.11.2 任务实现

1. 建立索引的命令

对“商品信息”建立索引。命令如下：

```
USE database! 商品信息
INDEX on 商品编号 TO a
```

2. 用索引对记录排序

操作步骤如下：

- (1) 打开已建好索引的表。
- (2) 选择“浏览”。
- (3) 从“表”菜单中选择“属性”命令。弹出“工作区属性”对话框，如图 4-101 所示。



图 4-101 “工作区属性”对话框

(4) 在“索引顺序”下拉列表框中选择要用的索引，单击“确定”按钮。  
原“商品信息”表如图 4-102 所示。

| 商品编号 | 商品名称 | 价格            | 库存数量      | 库存报警数     | 计划进货数     | 条形码  |
|------|------|---------------|-----------|-----------|-----------|------|
| dd1  | 自行车  | 0.20000       | 11.00000  |           |           |      |
| ff   | 洗衣机  | 11.11000      | 14.01110  |           |           |      |
| gf   | 洗衣机  | 44.44400      | 77.10000  |           |           |      |
| m001 | m001 | 1.00000       | 1.00000   | 1.00000   | 1.00000   | 1001 |
| m002 | m002 | 10.00000      | 1.00000   | 12.00000  | 5.00000   | 1002 |
| m008 | 手表   | 330.00000     | 330.00000 | 110.00000 | 130.00000 | 1005 |
| m003 | m003 | 10.00000      | 22.00000  | 22.00000  | 12.00000  | 1003 |
| m005 | 电脑   | 1000.00000    | 100.00000 | 100.00000 | 120.00000 | 1004 |
| m006 | 洗衣机  | 2000.00000    | 10.00000  | 5.00000   | 4.00000   | 1007 |
| m007 | 飞机   | 9999999.99999 | 10.00000  | 10.00000  | 10.00000  | 1006 |
| m009 | 自行车  | 300.00000     | 10.00000  | 10.00000  | 5.00000   | 1008 |

图 4-102 原“商品信息”表

索引顺序结果如图 4-103 所示。

| 商品编号 | 商品名称 | 价格            | 库存数量      | 库存报警数     | 计划进货数     | 条形码  |
|------|------|---------------|-----------|-----------|-----------|------|
| m001 | m001 | 1.00000       | 1.00000   | 1.00000   | 1.00000   | 1001 |
| m002 | m002 | 10.00000      | 1.00000   | 12.00000  | 5.00000   | 1002 |
| m008 | 手表   | 330.00000     | 330.00000 | 110.00000 | 130.00000 | 1005 |
| m003 | m003 | 10.00000      | 22.00000  | 22.00000  | 12.00000  | 1003 |
| m005 | 电脑   | 1000.00000    | 100.00000 | 100.00000 | 120.00000 | 1004 |
| m006 | 洗衣机  | 2000.00000    | 10.00000  | 5.00000   | 4.00000   | 1007 |
| m007 | 飞机   | 9999999.99999 | 10.00000  | 10.00000  | 10.00000  | 1006 |
| m009 | 自行车  | 300.00000     | 10.00000  | 10.00000  | 5.00000   | 1008 |

图 4-103 索引顺序结果

### 3. 排序命令

对“商品信息”表建立排序，命令如下：

```
use database!商品信息
```

```
SORT TO 商品信息1 ON 库存数量 /a
```

原“商品信息”表如图 4-104 所示。

| 商品编号 | 商品名称 | 价格            | 库存数量      | 库存报警数     | 计划进货数     | 条形码  |
|------|------|---------------|-----------|-----------|-----------|------|
| dd1  | 自行车  | 0.20000       | 11.00000  |           |           |      |
| ff   | 洗衣机  | 11.11000      | 14.01110  |           |           |      |
| gf   | 洗衣机  | 44.44400      | 77.10000  |           |           |      |
| m001 | m001 | 1.00000       | 1.00000   | 1.00000   | 1.00000   | 1001 |
| m002 | m002 | 10.00000      | 1.00000   | 12.00000  | 5.00000   | 1002 |
| m008 | 手表   | 330.00000     | 330.00000 | 110.00000 | 130.00000 | 1005 |
| m003 | m003 | 10.00000      | 22.00000  | 22.00000  | 12.00000  | 1003 |
| m005 | 电脑   | 1000.00000    | 100.00000 | 100.00000 | 120.00000 | 1004 |
| m006 | 洗衣机  | 2000.00000    | 10.00000  | 5.00000   | 4.00000   | 1007 |
| m007 | 飞机   | 9999999.99999 | 10.00000  | 10.00000  | 10.00000  | 1006 |
| m009 | 自行车  | 300.00000     | 10.00000  | 10.00000  | 5.00000   | 1008 |

图 4-104 原“商品信息”表

运行结果如图 4-105 所示。

### 4. 打开数据库

打开数据库的命令如下：

```
OPEN database 名称
```

```
OPEN database d:\supermarket\supermarket.pjx
```

| 商品编号 | 商品名称 | 价格            | 库存数量      | 库存报警数     | 计划进货数     | 条形码  |
|------|------|---------------|-----------|-----------|-----------|------|
| m001 | m001 | 1.00000       | 1.00000   | 1.00000   | 1.00000   | 1001 |
| m002 | m002 | 10.00000      | 1.00000   | 12.00000  | 5.00000   | 1002 |
| m007 | 飞机   | 9999999.99999 | 10.00000  | 10.00000  | 10.00000  | 1006 |
| m006 | 洗衣机  | 2000.00000    | 10.00000  | 5.00000   | 4.00000   | 1007 |
| m009 | 自行车  | 300.00000     | 10.00000  | 10.00000  | 5.00000   | 1008 |
| dd1  | 自行车  | 0.20000       | 11.00000  |           |           |      |
| ff   | 洗衣机  | 11.11000      | 14.01110  |           |           |      |
| m003 | m003 | 10.00000      | 22.00000  | 22.00000  | 12.00000  | 1003 |
| gf   | 洗衣机  | 44.44400      | 77.10000  |           |           |      |
| m005 | 电脑   | 1000.00000    | 100.00000 | 100.00000 | 120.00000 | 1004 |
| m008 | 手表   | 330.00000     | 330.00000 | 110.00000 | 130.00000 | 1005 |

图 4-105 排序后的表

## 课后练习

1. 用命令打开 supermarket 数据库。
2. 对“销售”表按照商品编号建立索引。
3. 对“销售”表按照销售数量进行排序。

## 4.12 任务十一 更新数据

### 4.12.1 相关知识

#### 1. 使用视图更新基本表

浏览或运行视图会得到一个临时表，表中数据是从基本表中提取的，可以更改临时表中的数据，并将更新结果回送到基本表中，达到更改数据表的目的。虽然视图是一个“虚表”，但是也可以利用视图更新数据表中的数据。因为视图可以限定数据表中数据的使用范围，因此，也就限定了可更新的数据，数据表中其他的数据就不会被破坏，由此可以提高数据维护的安全性。

(1) 使表可更新。如果希望在表上作的修改能回送到原表中，需要设置“发送 SQL 更新”选项，必须至少设置一个关键字段来使用这个选项。

(2) 设置关键字段。Visual FoxPro 用关键字段来唯一地标识那些已在本地修改过的远程表中的更新记录。

如果选择的表中有一个主关键字段并且已在“字段”选项卡中，则“视图设计器”自动使用表中的该主关键字段作为视图的关键字段。

“钥匙”列：每个要更新的表至少要设置一个关键字，更新过程是通过关键字进行的。

“铅笔”列：用于设置可更新字段，表示该字段可通过视图来更新。

如果已经改变了关键字段，又想把它们恢复到原表中的初始设置，单击“重置关键字”按钮。Visual FoxPro 会检查表并利用这些表中的关键字段。

可用本地或远程视图更新数据。在“视图设计器”的“更新条件”选项卡中可设置更新条件，如图 4-106 所示。

#### 2. 使用 SQL 语言进行更新

SQL 的数据更新命令格式如下：

UPDATE <表名> SET <字段名 1>=<表达式 1> [,<字段名 2>=<表达式 2>] WHERE <条件>

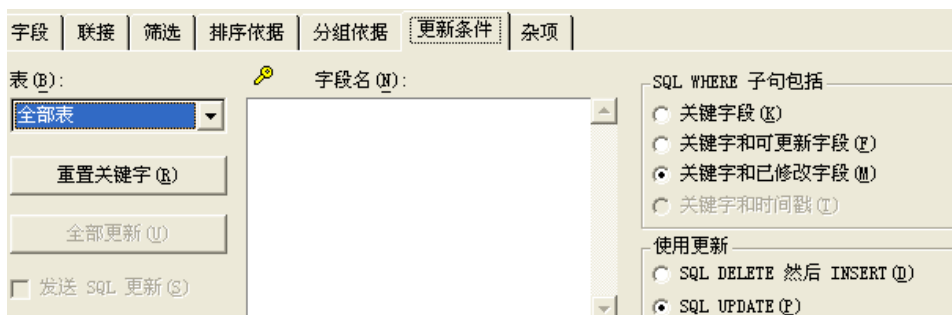


图 4-106 “视图设计器”的“更新条件”选项卡

本小节主要技能点如下：

- (1) 控制字段显示和数据输入。
- (2) 控制更新方法。
- (3) 设置关键字段。
- (4) 更新指定字段。
- (5) 更新所有字段。
- (6) 参数提示。

#### 4.12.2 任务实现

对“生产商”表进行更新。

(1) 控制字段显示和数据输入。在视图设计器中选择“生产商”表，在可用字段中选择需要的字段，如图 4-107 所示。

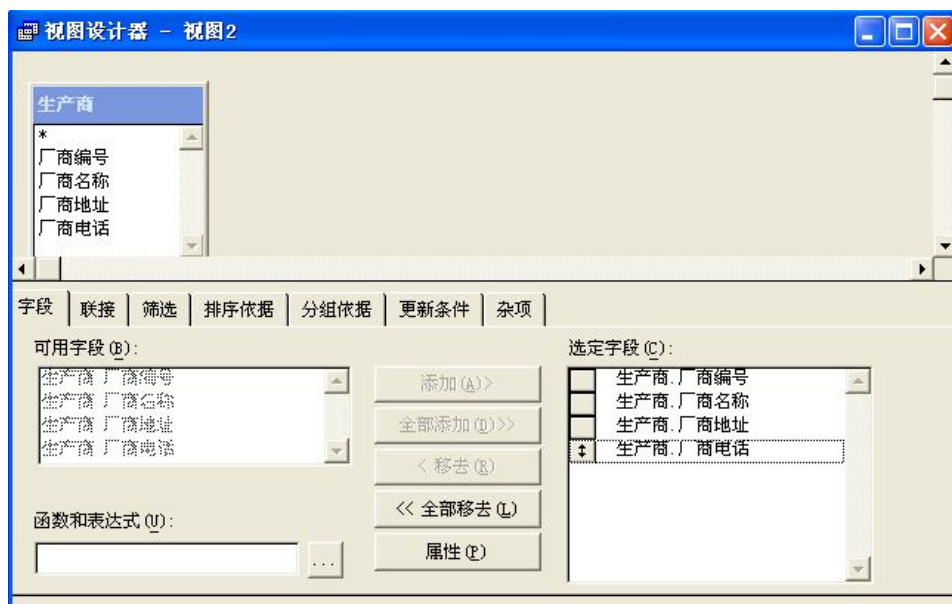


图 4-107 “视图设计器”窗口

(2) 控制更新方法如图 4-108 所示。

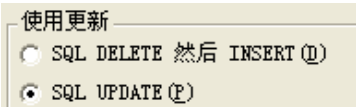


图 4-108 更新方法

(3) 设置关键字段。选择“更新条件”选项卡，设置关键字，如图 4-109 所示。



图 4-109 设置关键字

(4) 更新指定字段，如图 4-110 所示。



图 4-110 设置更新的字段

(5) 更新所有字段，如图 4-111 所示。

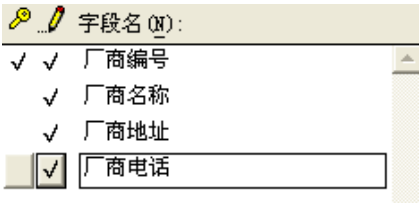


图 4-111 更新所有字段

(6) 参数提示。对“商品信息”表中的字段商品编号进行参数设置。在“实例”文本框中输入一个参数名称，在参数名称的前面加一个“?”，如图 4-112 所示。

| 字段名        | 否                        | 条件 | 实例      | 大小写                      | 逻辑  |
|------------|--------------------------|----|---------|--------------------------|-----|
| 商品信息. 商品编号 | <input type="checkbox"/> | =  | (?商品编号) | <input type="checkbox"/> | <无> |

图 4-112 参数设置

运行结果如图 4-113 所示。

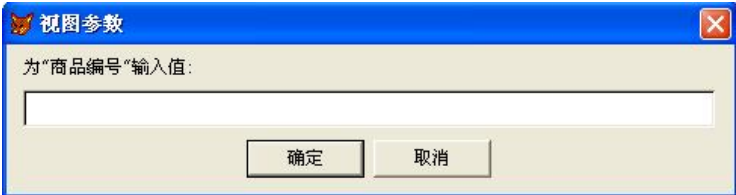


图 4-113 运行结果

在文本框中输入商品信息中的字段商品编号里面的值，如输入 m001，结果如图 4-114 所示。

|   | 商品编号 | 商品名称 | 库存数量    |
|---|------|------|---------|
| ▶ | m001 | m001 | 1.00000 |

图 4-114 输入 m001 的结果

(7) 使用 SQL 语句进行更新。对“商品信息”表进行更新，SQL 语句如下：

```
UPDATE 商品信息 SET 库存数量=库存数量 +100 WHERE 库存数量 <10
```

## 课后练习

1. 对“商品信息”表商品编号字段建立输入查询。
2. 对“商品信息”表中的库存数量进行更新，库存数量小于 200 加上 1000。

## 本章小结

在数据库的应用中，对数据的检索是数据处理中最常用的。Visual FoxPro 提供了多种较好的数据检索的方法、文件数据的查询、文件索引以及结构化数据查询语言，结构化查询语言 SQL 的基本语法。

视图能够从本地或远程表中提取一组记录，使用视图可以处理或更新检索到的记录，即设定条件，可从几个数据表文件中过滤出所要求的数据，其结果存储成实际的记录数据，可以当作实际的数据表文件来使用。

## 习题四

### 一、填空题

1. 结构化查询语言的英文缩写是\_\_\_\_\_。
2. 顺序查找命令\_\_\_\_\_，found()函数找到记录返回\_\_\_\_\_。
3. 在 Visual FoxPro 中，创建索引的命令是\_\_\_\_\_。
4. 在 Visual FoxPro 中，用命令可以创建\_\_\_\_\_索引，但不可以创建\_\_\_\_\_索引。
5. Visual FoxPro 中，视图可以分为\_\_\_\_\_视图和本地视图。
6. 作为查询的数据源，可以是数据库表、\_\_\_\_\_或\_\_\_\_\_。

### 二、选择题

1. 在 SQL Select 查询中，为了使查询结果排序应使用命令（ ）。  
A. ASC                      B. DESC                      C. GROUP BY                D. ORDER BY
2. 关于视图和查询，下列说法正确的是（ ）。  
A. 查询和视图都只能在数据库的建立    B. 查询和视图都不能在数据库的建立  
C. 视图只能在数据库的建立              D. 查询只能在数据库的建立

3. CREATE DATABASE 命令用来建立 ( )。
- A. 表                      B. 查询                      C. 数据库                      D. 视图
4. 在 VFP 数据表中只能有一个 ( )。
- A. 候选索引              B. 主索引                      C. 普通索引                      D. 唯一索引
5. 扩展名为.dbf 的文件是 ( )。
- A. 数据表文件      B. 表单文件                      C. 查询文件                      D. 项目文件

利用下表做出第 6、7 题。

| 设备型号  | 设备名称 | 使用日期     | 设备数量 | 单价        | 使用部门 | 进口 |
|-------|------|----------|------|-----------|------|----|
| W27-1 | 微机   | 01/10/03 | 1    | 143000.00 | 生产一间 | T  |
| W27-2 | 微机   | 02/06/03 | 2    | 98000.00  | 生产一间 | F  |
| C31-1 | 车床   | 03/30/03 | 2    | 138000.00 | 生产二间 | T  |
| C31-2 | 车床   | 04/05/03 | 2    | 97500.00  | 生产二间 | T  |
| M20-1 | 磨床   | 02/10/03 | 3    | 98000.00  | 生产二间 | F  |
| J18-1 | 轿车   | 05/07/03 | 2    | 156000.00 | 办公室  | T  |
| F15-1 | 复印机  | 02/01/03 | 2    | 8600.00   | 办公室  | F  |

6. 从设备表中查询单价大于 100000 元的设备, 并显示设备名称, 正确的命令是 ( )。
- A. SELECT 单价>100000 FROM 设备表 FOR 设备名称
- B. SELECT 设备名称 FROM 设备表 FOR 单价>100000
- C. SELECT 单价>100000 FROM 设备表 WHERE 设备名称
- D. SELECT 设备名称 FROM 设备表 WHERE 单价>100000
7. 利用 SQL 数据更新功能, 自动计算更新每个"设备总金额"字段的字段值, 该字段值等于"单价\*设备数量"的值, 正确命令为 ( )。
- A. UPDATE 设备表 SET 设备总金额=单价\*设备数量
- B. UPDATE 设备表 FOR 设备总金额=单价\*设备数量
- C. UPDATE 设备表 WITH 设备总金额=单价\*设备数量
- D. UPDATE 设备表 WHERE 设备总金额=单价\*设备数量

第 8、9、10 题使用如下的班级表和学生表。

班级表

| 班级号      | 班级名称  | 班级人数 |
|----------|-------|------|
| 20030103 | 计算机一班 | 55   |
| 20030203 | 计算机二班 | 48   |
| 20030303 | 计算机三班 | 50   |

学生表

| 班级号    | 学号   | 姓名 | 性别 | 籍贯 |
|--------|------|----|----|----|
| 200301 | 1001 | 王伟 | 男  | 北京 |
| 200301 | 1002 | 刘红 | 女  | 上海 |
| 200301 | 1003 | 李林 | 女  | 北京 |
| 200302 | 2001 | 张清 | 女  | 上海 |
| 200302 | 2002 | 刘雷 | 男  | 上海 |

8. 有以下 SQL 语句:

```
SELECT 班级名称, 姓名, 性别 FROM 班级表, 学生表;
WHERE 班级表.班级号=学生表1.班级号;
AND 籍贯="上海" AND 性别="女";
ORDER BY 班级名称 DESC
```

执行该语句后，查询结果中共有（     ）条记录，且第一条记录的学生姓名是（     ）。

- A. 1 李林      B. 2 张清      C. 2 刘红      D. 3 张清

9. 有以下 SQL 语句:

```
SELECT 班级名称, 姓名, 性别 FROM 班级表, 学生表;
WHERE 班级表.班级号=学生表.班级号;
AND 姓名 LIKE "刘*";
ORDER BY 班级号
```

该语句的含义是 ( )。

- A. 查找学生表中姓"刘"的学生记录，并根据班级号分组显示学生的班级名称、姓名和性别
- B. 查找学生表中姓"刘"的学生记录，按班级号升序显示学生的班级名称、姓名和性别
- C. 查找学生表中不是姓"刘"的学生记录，按班级号升序显示学生的班级名称、姓名和性别
- D. 语句错误

10. 有以下 SQL 语句:

```
SELECT 班级名称 FROM 班级表 WHERE NOT EXISTS;  
      (SELECT * FROM 学生表 WHERE 班级号=班级表.班级号)
```

执行该语句后，班级名称的字段值是（ ）。

- A. 03 计算机一班                      B. 03 计算机二班  
C. 03 计算机三班                      D. 03 计算机一班和 03 计算机二班

### 三、简答题

1. 查询设计器可以有哪些输出方式？如何解决？
2. 查询与视图的区别是什么？
3. 视图的类型是什么？
4. 查询与视图的扩展名分别是什么？
5. 查询设计器中的“筛选”选项卡的作用是什么？如何设置使用？