

目

录

Contents



第一部分 Visual FoxPro 程序设计

第一章 Visual FoxPro 操作基础 1

第一节 初识计算机 2

一、计算机硬件组成 2

二、计算机基本操作 3

第二节 Visual FoxPro 工作环境及

工作方式 4

一、Visual FoxPro 工作环境 4

二、Visual FoxPro 工作方式 5

三、表达式输出命令 6

习题 1 6

第二章 Visual FoxPro 数据及运算 ... 7

第一节 数据类型 8

第二节 常量、变量 9

一、常量 9

二、变量基础知识 10

三、字段变量 11

四、内存变量 11

五、数组变量 12

第三节 函数 13

一、函数基础 13

二、数值函数 13

三、字符函数 15

四、日期和时间函数 18

五、数据类型转换函数 20

第四节 表达式 21

一、算术表达式 21

二、字符表达式 22

三、日期和时间表达式 22

四、关系表达式 24

五、逻辑表达式 25

习题 2 26

第三章 表操作 28

第一节 表的基本操作 29

一、表的基础知识 29

二、表的建立 29

三、表的打开和关闭 30

四、表的复制 30

第二节 表的显示 31

一、表结构显示 31

二、表记录显示 32

第三节 表的维护 34

一、记录指针定位 34

二、测试函数 36

三、表的修改 38

四、表记录的增加 39



五、表记录的删除	40	第五章 面向对象程序设计.....	101
六、表的统计与计算	42	第一节 面向对象基础	102
第四节 多表操作	43	一、面向对象程序设计	102
一、工作区	43	二、对象与类	102
二、工作区的编号	43	第二节 表单基本操作	103
三、工作区的选择	44	一、表单实操步骤	103
第五节 表的排序	44	二、表单的建立和运行	104
第六节 表的索引	45	三、表单属性及方法	104
一、索引文件	45	四、控件操作与布局	105
二、索引的建立	46	第三节 常用表单控件	108
三、索引的种类	46	一、控件分类	108
四、索引的使用	48	二、常用控件属性	108
习题 3	49	第四节 综合实例	111
第四章 结构化程序设计.....	51	习题 5	118
第一节 结构化程序设计基础	52	第六章 数据库.....	120
一、程序基本概念	52	第一节 数据库基础知识	121
二、结构化程序文件的建立、修改和运行	52	一、数据管理技术	121
三、结构化程序的辅助命令	53	二、数据库系统	121
四、交互式输入命令	53	三、数据模型	122
第二节 顺序结构程序	54	四、关系数据库	122
第三节 选择结构程序	56	第二节 数据库基本操作	123
一、双分支选择结构	56	一、项目管理器	123
二、多分支选择结构	59	二、数据库的建立、修改和删除	124
第四节 循环结构程序	63	三、创建表之间的联系	124
一、do while 循环	64	习题 6	124
二、for 循环	80	第七章 结构化查询语言、查询和视图.....	127
三、scan 循环	83	第一节 结构化查询语言	128
第五节 循环的嵌套	84	一、数据查询	128
一、语句格式	84	二、数据定义	129
二、执行过程	85	三、数据操纵	130
三、使用说明	85		
四、例题	86		
习题 4	91		



第二节 查询和视图	131	二、视图	132
一、查询	132	习题 7	133



第二部分 计算机基础

第八章 计算机基础知识	135	第十章 计算机网络与信息安全	152
第一节 计算机产生和发展	136	第一节 多媒体技术和数据通信	
一、计算机的产生和发展	136	技术基础	153
二、计算机的分类和应用	136	一、多媒体技术	153
第二节 计算机系统的组成	137	二、数据通信技术基础	154
一、硬件系统	138	第二节 计算机网络与应用	154
二、软件系统	139	一、计算机网络定义和功能	154
三、计算机语言	139	二、计算机网络分类	155
第三节 常用进位计数制及计算机		三、计算机网络分层	155
中的信息存储	139	四、计算机网络协议和计算机	
一、常用进位计数制	139	地址	155
二、进位计数制之间的转换	140	五、计算机网络应用	156
三、计算机中信息存储和信息		六、网络新技术	158
编码	144	第三节 计算机安全与病毒	158
习题 8	145	一、计算机安全	158
第九章 操作系统	147	二、计算机病毒	159
第一节 操作系统概念、功能及		习题 10	159
分类	148	第十一章 软件技术基础	162
一、操作系统概念和功能	148	第一节 数据结构	163
二、操作系统分类	148	一、线性表	163
第二节 Windows 7 基本概念及基本		二、栈与队列	163
操作	149	三、树与二叉树	163
一、Windows 7 操作系统中的基本		第二节 算法	164
概念和操作	149	一、算法基础	164
二、Windows 7 操作系统基本		二、算法评定	165
操作	150	三、排序方法	165
习题 9	150	第三节 软件工程基础知识	166



一、软件危机和软件维护	166	2020 年黑龙江省二级 VFP 真题	183
二、软件测试	167	2020 年黑龙江省二级 VFP 答案及	
习题 11	167	解析	189
附录.....	169	2019 年黑龙江省二级 VFP 真题	195
附表 1 Visual FoxPro 常用函数	169	2019 年黑龙江省二级 VFP 答案及	
附表 2 表单常用控件及属性	170	解析	201
2021 年黑龙江省二级 VFP 真题	172	参考文献.....	207
2021 年黑龙江省二级 VFP 真题答案			
及解析	178		

版权所有 请勿转载

第一部分

Visual FoxPro 程序设计

第一章

Visual FoxPro

操作基础

学习目标 >

- ① 了解计算机的硬件组成和基本操作
- ② 掌握 Visual FoxPro 的工作环境及工作方式
- ③ 掌握表达式输出命令的格式及用法



第一节 初识计算机

考 试 要 点

1. Visual FoxPro 硬件组成。
2. Visual FoxPro 文件和文件夹基本操作。

一、计算机硬件组成

台式计算机一般由主机、显示器、键盘和鼠标组成，如图 1-1 所示。



图 1-1 台式计算机

键盘是计算机输入/输出设备（即 I/O 设备）中最基本也是最重要的输入设备。台式计算机的键盘一般为 108 键键盘，分为五个大区，如图 1-2 所示。



图 1-2 键盘分区



键盘是最主要的输入设备，要想提高输入速度，必须掌握正确的指法。指法示意如图 1-3 所示。

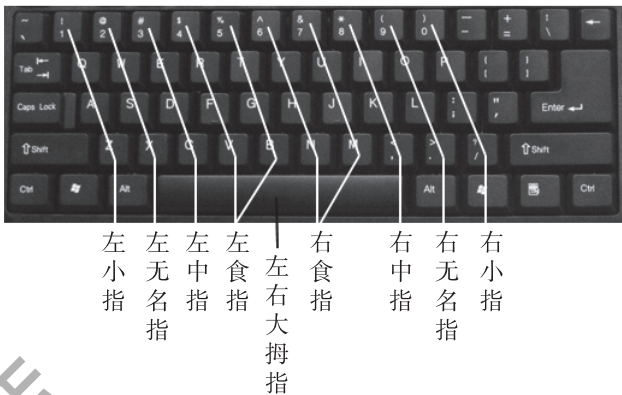


图 1-3 计算机键盘指法图

二、计算机基本操作

1. 计算机的开关机

- (1) 开机：先开“显示器”，后开“主机”。
- (2) 关机：先关“主机”，再关“显示器”。即先单击 Windows 界面左下角“任务栏”上的“开始”按钮后，再单击“关机”按钮，最后关闭“显示器”。

**注意：**
常规情况下，关机时不能直接按主机箱的“电源”按钮进行关机操作。

2. 文件及文件夹操作

- (1) 文件及文件夹的选择，操作方法见表 1-1。

表 1-1 文件及文件夹的选择操作方法

序号	操作	方法
1	单个文件或文件夹的选择	鼠标单击文件或文件夹
2	连续文件或文件夹的选择	方法 1：鼠标左键框选 方法 2：选中第一个文件或文件夹，按住 Shift 键，选中最后一个文件或文件夹
3	不连续文件或文件夹的选择	按住 Ctrl 键，单个选择文件或文件夹
4	全选文件夹中的文件	框选或按 Ctrl+A 键

- (2) 文件及文件夹的复制和移动。
复制操作步骤：选择→复制→粘贴。



移动操作步骤：选择→剪切→粘贴。

复制操作方法：①按快捷键 Ctrl+C。②在选中的文件或文件夹上右击，选择右键快捷菜单中的“复制”选项。

剪切操作方法：①按快捷键 Ctrl+X。②在选中的文件或文件夹上右击，选择右键快捷菜单中的“剪切”选项。

粘贴操作方法：①按快捷键 Ctrl+V。②在目标文件夹上右击，选择右键快捷菜单中的“粘贴”选项。

(3) 文件及文件夹的删除。

删除操作步骤：选择→删除。

删除方法：①按 Delete 键。②在选中的文件或文件夹上右击，选择右键快捷菜单中的“删除”选项。

彻底删除方法：①按快捷键 Shift+Delete。②在选中的文件或文件夹上右击，按住“Shift”键，选择右键快捷菜单中的“删除”选项。

第二节 Visual FoxPro 工作环境及工作方式

考(试)要(点)

1. 退出 Visual FoxPro 所用命令。
2. Visual FoxPro 工具栏操作。
3. Visual FoxPro 命令窗口和主窗口的功能。
4. Visual FoxPro 命令执行过程和书写规则。
5. Visual FoxPro 清除主窗口内容命令。
6. Visual FoxPro 的三种工作方式。
7. 表达式输出命令 ? 和 ?? 的区别及同时输出多个表达式的方法。

一、Visual FoxPro 工作环境

1. Visual FoxPro 的启动

方法 1：单击 Windows 界面左下角的“开始”按钮，选择“所有程序”→“Microsoft Visual FoxPro 6.0”→“Microsoft Visual FoxPro 6.0”选项（注意，这是计算机等级考试要



求必须掌握的方法)。

方法 2: 双击桌面上的 Microsoft Visual FoxPro 6.0 快捷图标。

2. Visual FoxPro 的退出

方法 1: 单击 Visual FoxPro 6.0 窗口右上角的“关闭”按钮。

方法 2: 在命令窗口中输入“quit”命令后, 按回车键(“Enter”键)执行该命令。

3. Visual FoxPro 的工作界面

Visual FoxPro 的工作界面如图 1-4 所示。

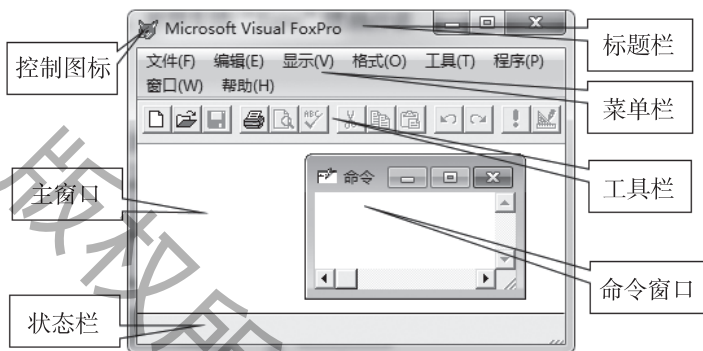


图 1-4 Visual FoxPro 工作界面

(1) 工具栏显示/隐藏操作方法: 在“显示”→“工具栏”菜单项中选择或清除相应的任务栏, 然后单击“确定”按钮(这是计算机等级考试要求必须掌握的方法)。

在 Visual FoxPro 中, 用户可以根据需要自定义工具栏, 用户自定义工具栏可以删除, 但系统自带的工具栏不能够删除。

(2) 命令窗口: 其功能为根据用户的需要输入命令, 输入命令后按回车键执行该命令。命令窗口可以隐藏, 也可以修改大小和位置。

(3) 主窗口: 其功能为显示命令或程序的执行结果。

(4) 清除主窗口内容方法: 在命令窗口输入“clear”命令, 按回车键。

4. 命令书写规则

(1) 命令中的标点符号必须在英文半角状态下进行输入。

(2) 一行只能写一条命令。

(3) 命令窗口输入命令必须按回车键才能执行。

(4) 命令中的命令动词为英文时, 可以用前 4 个或 4 个以上英文字母缩写。

(5) 如果一条命令太长, 在一行内写不下, 可以使用续行符号, Visual FoxPro 的续行符为分号(;)。

二、Visual FoxPro 工作方式

Visual FoxPro 有三种工作方式: 菜单方式、命令方式和程序方式。



三、表达式输出命令

表达式输出命令有以下两种。

格式 1:

?[< 表达式表 >]

功能: 先换行, 后输出。

格式 2:

??[< 表达式表 >]

功能: 不换行, 同行直接输出。

区别: ? 换行输出; ?? 不换行, 同行输出。

当输出多个表达式时, 表达式之间用英文逗号隔开。表达式输出结果均显示在主窗口中。

习题 1

一、填空题

1. 在 Visual FoxPro 中, 要退出 Visual FoxPro 系统环境, 可以使用命令_____后按键盘上的_____键。
2. 在 Visual FoxPro 中, 如果一行命令太长, 在一行内写不下, 可以使用续行符, 续行符为_____。
3. 在 Visual FoxPro 中, 清除主窗口的命令是_____。
4. Visual FoxPro 有三种工作方式, 即命令方式、_____方式和程序方式。

二、判断题

- () 1. 表达式输出命令有两种, 其中使用“??”命令输出结果时, 光标会不换行。
- () 2. 在 Visual FoxPro 中, 命令窗口输入的命令必须按回车键才能执行。
- () 3. “?”和“??”命令功能一样, 结果都是显示在下一行的开头。
- () 4. 运行 Visual FoxPro 的命令文件时, 只需要在命令窗口中输入命令文件名, 回车就可以运行。

第二章

Visual FoxPro

数据及运算

学习目标 >

- ① 掌握 Visual FoxPro 中常用的数据类型
- ② 掌握 Visual FoxPro 中常量和变量的类型及其用法
- ③ 掌握函数的功能、参数数据类型、函数值数据类型
- ④ 掌握表达式运算符、运算数数据类型、返回值数据类型



第一节 数据类型

考试要点

VF 常用数据类型的英文字母缩写及存储宽度。

Visual FoxPro 中常用的数据类型有 8 种：字符型、数值型、货币型、日期型、日期时间型、逻辑型、备注型、通用型，具体见表 2-1。

表 2-1 VF 常用数据类型

数据类型	字母表示	存储宽度	备注
字符型	C	不定长	1 个中文字符占 2 个字节 1 个西文字符占 1 个字节 例如：姓名、籍贯
数值型	N	不定长	例如：入学成绩
货币型	Y	8	例如：工资
日期型	D	8	例如：出生日期
日期时间型	T	8	例如：存储员工上下班打卡时间
逻辑型	L	1	例如：少数民族否
备注型	M	4	1. 只能用于表 2. 备注型数据存储在表同名的备注文件中，备注文件扩展名为 fpt 例如：学生 .dbf 中备注文件名为“学生 .fpt” 3. 备注文件可以不与表文件同时使用 例如：简历
通用型	G	4	只能用于表 例如：照片



第二节 常量、变量

考试要点

1. 常量、变量的定义和区别
2. 常量表示方法
3. 变量命名规则
4. 内存变量赋值及用法
5. 数组变量定义、赋值、数组元素个数、数据类型及使用注意事项

一、常量

定义：在操作或程序运行过程中其值不变的量。

1. 字符型常量

- (1) 必须用定界符括起来。
- (2) 定界符有三种：""、'' 和 []。
- (3) 定界符里面不能包含相同的定界符。

例如，[.T.]、"123"、' 计算机 '、[That's right!] 是字符型常量。

注意：

[[大庆]] 不是正确的字符型常量表达方式。

2. 数值型常量

- (1) 由数字、小数点和正负号组成。
- (2) 数值型常量有以下两种。

小数形式：如 3.14、-2.56。

指数形式：如，3e-4，3e-4 等价于 3×10^{-4} 。

注意：

3/4 不是数值型常量。



3. 逻辑型常量

- (1) 逻辑型常量包含逻辑真和逻辑假。
- (2) 逻辑真: .T. .Y. .t. .y.。
- (3) 逻辑假: .F. .N. .f. .n.。



注意:

字母前后的小圆点 (.) 不能省略。

4. 货币型常量

数值型数据前面加上一个“\$”。

例:

```
$3.1416
```

5. 日期型常量

- (1) 放在一对花括号 ({}) 中, 包括年、月、日 3 部分。
- (2) 采用系统严格的日期格式: {^YYYY-MM-DD}。
- (3) 分隔符有三种格式, 即 “-” “/” “.”。

例:

```
{^2021-10-01}
```



注意:

严格日期格式的年份前面必须有 ^。

6. 日期时间型常量

在日期型常量的日期后加时间。

例:

```
{^2021-10-01 9:10:40}
```

二、变量基础知识

1. 定义

变量指在操作或程序运行过程中可以改变其值的数据对象。

2. 分类

变量分为字段变量、内存变量、数组变量和系统变量。

3. 三要素

变量的三要素: 变量名、变量类型、变量值。



4. 变量的命名规则

- (1) 由字母、汉字、下划线和数字组成（即变量名中如果有标点符号只能是下划线）。
- (2) 不能以数字开头。
- (3) 长度为 1~128 个字符。

▲ 注意:

内存变量与字段变量都是变量，命名必须符合变量命名规则。

三、字段变量

字段变量就是表中的字段，一个表有多少条记录，该表的每一字段就有多少个值。

▲ 注意:

可以取 VF 中的任意数据类型。

四、内存变量

内存变量必须先赋值后使用。

1. 内存变量的赋值

- (1) “=” 方法。
- 格式: < 内存变量 > = < 表达式 >。

▲ 注意:

“=” 左边只能有一个变量名 (即 “=” 方式一次只能给一个变量赋值)。

例:

```
x=100
```

说明:

- ① $a+1=a$ 不是合法的赋值语句。
 - ② $x==val("5+9")$ 不是合法的赋值语句。
- (2) store 方法。
- 格式: store < 表达式 > to < 内存变量表 >。

▲ 注意:

“store 方法” 一次可以给多个变量赋相同的值，变量名之间用英文逗号 (,) 隔开。



例:

```
store 0 to s, i
```

2. 内存变量与字段变量重名时的引用

- (1) 当内存变量和字段变量重名时, 优先处理字段变量。
- (2) 引用内存变量需要在相应变量名前加 “M->” 或 “M.”。

例 2-1 现有两个变量, 一个是字段变量 “姓名”, 值为 “张华”; 另一个是内存变量 “姓名”, 值为 “王敏”, 则

```
? 姓名      && 输出张华  
?m. 姓名    && 输出王敏
```

3. 内存变量注意事项

- (1) 在 Visual FoxPro 中, 以 .mem 为扩展名的文件是内存变量文件。
- (2) 一个内存变量经过多次赋值后, 仅有一个值。
- (3) 清除内存变量的命令: clear memory。
- (4) 退出 VFP 会清除所有的内存变量, 字段变量不会清除。

五、数组变量

数组必须先定义后使用。

1. 数组定义

- (1) dimension <数组名> (<下标上界 1>[, <下标上界 2>])。
- (2) declare <数组名> (<下标上界 1>[, <下标上界 2>])。

例:

```
dimension a(3),b(4,5)  && 定义一维数组 a 和二维数组 b
```

2. 数组变量注意事项

- (1) VF 只能定义一维和二维数组, 下标从 1 开始。
- (2) 每个数组元素可以当作一个内存变量来使用。
- (3) 数组中数组元素的个数如下。

dimension 数组名 (m): 一维数组, m 个数组元素。

dimension 数组名 (m,n): 二维数组, m × n 个数组元素。

例:

```
dimension a(3,5),b(3),ab(3,4)&& 元素个数分别为: 15、3、12
```

- (4) 在没有给数组元素赋值前, 数组元素有初值, 逻辑型, 初始值为 .F. 。



(5) 为数组名赋值相当于为数组中每个数组元素赋相同的值。

例 2-2 定义一个一维数组 a, 给数组中的每个数组元素赋值 3。

dimension a(3)	&& 定义一维数组 a, 有 3 个数组元素 a(1)、a(2)、a(3)
a=3	&& 给数组元素赋相同的值, 即 a(1)、a(2)、a(3) 的值均为 3

(6) 数组中数组元素的数据类型可以不同。

例 2-3 定义一个二维数组 b, 给数组中数组元素赋予不同数据类型的值。

dimension b(4,5)	&& 定义二维数组 b, 有 20 个数组元素
b(1,1)=100	&&b(1,1) 为数值型数据
b(2,3)="祖国伟大"	&&b(2,3) 为字符型数据

第三节 函数

考点要点

1. 函数的调用格式。
2. 函数的功能、参数个数、参数数据类型、函数返回值数据类型。
3. 函数的用法。

一、函数基础

函数调用格式: 函数名 ([参数表])。



注意:

Visual FoxPro 提供了用户自定义函数的功能。

二、数值函数

说明: 以下函数介绍中出现的 N 均表示数值型数据。

1. 求绝对值函数

格式: abs(N)。

功能: 求 N 的绝对值。

函数返回值数据类型: N。



例：

```
?abs(-3.14) && 返回值为：3.14
```

2. 求平方根函数

格式：sqrt(N)。

功能：求 N 的算术平方根，其中 N 不能为负数（即 $N \geq 0$ ）。

函数返回值数据类型：N。

例：

```
?sqrt(4) && 返回值为：2
```

3. 取整函数

格式：int(N)。

功能：求 N 的整数部分，不四舍五入。

函数返回值数据类型：N。

例：

```
?int(78.965) && 返回值为：78
```

4. 求余函数

格式：mod(N1,N2)。

功能：求 N1 除以 N2 的余数。

函数返回值数据类型：N。



注意：

mod(N1,N2) 与 $N1 \% N2$ 功能一致。

例 2-4 分别用求余函数和 % 来计算 12 除以 3 的余数。

```
?mod(12,3) && 返回值为：0
```

```
?12%3 && 返回值为：0
```

应用 int(N) 和 mod(N1,N2) 可以进行数字分离。

例 2-5 用取整和求余函数分离出 456 的个位、十位和百位数。

a=456	&& 内存变量 a 赋值为 456
?a1=int(a/100)	&& a1=4
?a2=mod(int(a/10),10)	&& a2=5
?a3=mod(a,10)	&& a3=6
?a4=int(mod(a,100)/10)	&& a4=5



5. 四舍五入函数

格式: round(N1,N2)。

功能: 将 N1 四舍五入, N2 决定保留小数位数。

函数返回值数据类型: N。

例:

```
?round(3.14,0)  && 返回值为: 3
```

6. π 函数

格式: pi()。

功能: 圆周率近似值。

函数返回值数据类型: N。

例:

```
?pi()  && 返回值为: 3.14
```

7. 求最大值、最小值函数

格式 1: max()。

功能: 求 () 中表达式的最大值, 要求 () 内参数数据类型必须相同。

例:

```
?max(55,-10,4)  && 返回值为: 55
```

格式 2: min()。

功能: 求 () 中表达式的最小值, 要求 () 内参数数据类型必须相同。

例:

```
?min(55,-10,4)  && 返回值为: -10
```

▲ 注意:

- (1) 字母比较大小: a 最小, z 最大。
- (2) 日期比较大小: 日期越靠后越大。
- (3) 汉字比较大小: 按照拼音进行比较。

三、字符函数

说明: 以下函数介绍中出现的 C 均表示字符型数据; N 表示数值型数据; □表示一个空格。



1. 求字符串长度函数

格式: len(C)。

功能: 求字符串 C 的长度, 即字符串 C 中包含字符个数。

函数返回值数据类型: N。

例 2-6 求以下各字符型数据的长度。

?len(" 中国 ")	&& 返回值为: 4
?len("abc")	&& 返回值为: 3
?len("123.4")	&& 返回值为: 5
?len(" ")	&& 返回值为: 0



注意:

一对定界符 (" " ' ' []) 表示空字符串。

2. 求子字符串位置函数

格式: at(C1,C2)。

功能: 求 C1 在 C2 中的起始位置。

函数返回值数据类型: N。

例:

?at("is","that is a book") && 返回值为: 6 (1 个空格是 1 个字符)

3. 取子字符串函数

格式 1: left(C,N)。

功能: 从字符型数据 C 的左边第一个字符开始截取 N 个字符。

函数返回值数据类型: C。

例:

?left(" 伟大祖国 ",4) && 返回值为: 伟大

格式 2: right(C,N)。

功能: 从字符型数据 C 的右边第一个字符开始截取 N 个字符。

函数返回值数据类型: C。

例:

?right(" 伟大祖国 ",4) && 返回值为: 祖国

格式 3: substr(C,N1,N2)。

功能: 从字符型数据 C 指定的位置 N1 开始截取 N2 个字符。



函数返回值数据类型：C。

例 2-7 用 substr() 函数，分离出字符型表达式“中国大庆”中的各个汉字。

?substr("中国大庆",1,2)	&& 返回值为：中
?substr("中国大庆",3,2)	&& 返回值为：国
?substr("中国大庆",5,2)	&& 返回值为：大
?substr("中国大庆",7,2)	&& 返回值为：庆

格式 4: substr(C,N)。

功能：从字符型数据 C 指定的位置 N 开始截取字符，到字符串结尾结束。

函数返回值数据类型：C。

例 2-8 分别计算以下 substr() 函数的函数值。

?substr("中国大庆",3)	&& 返回值为：国大庆
?substr("中国大庆",5)	&& 返回值为：大庆

4. 生成空格函数

格式：space(N)。

功能：生成 N 个空格。

函数返回值数据类型：C。

例：

?space(8) && 返回值为：8 个空格□□□□□□□□

5. 删除字符串前后空格函数

格式 1: alltrim(C)。

功能：删除字符串前后空格。

函数返回值数据类型：C。

例：

?alltrim("□□中国□□") && 返回值为：中国□□

格式 2: ltrim(C)。

功能：删除字符串前导空格。

函数返回值数据类型：C。

例：

?ltrim("□□中国□□") && 返回值为：中国□□

格式 3: rtrim(C)。

功能：删除字符串尾部空格。



函数返回值数据类型：C。

▲ 注意：

trim(C) 等同于 rtrim(C)。

例：

```
?rtrim(" 中国 ")  && 返回值为：中国  
?trim(" 中国 ")  && 返回值为：中国
```

6. 大小写字母转换函数

格式 1：upper(C)。

功能：将字符型数据 C 的小写字母改为大写字母。

函数返回值数据类型：C。

例：

```
?upper("aBc")  && 返回值为：ABC
```

格式 2：lower(C)。

功能：将字符型数据 C 的大写字母改为小写字母。

函数返回值数据类型：C。

例：

```
?lower("aBc")  && 返回值为：abc
```

四、日期和时间函数

说明：以下函数介绍中出现的 D 表示日期型数据；N 表示数值型数据。假设当前系统日期为 2021 年 8 月 15 日，时间为上午 10:18:45。

▲ 注意：

日期和时间函数无参数时，括号不能少。

1. 系统日期函数

格式：date()。

功能：返回当前系统日期。

函数返回值数据类型：D。

例：

```
?date()  && 返回值为：08/15/21
```



2. 系统时间函数

格式: `time()`。

功能: 返回当前系统时间。

函数返回值数据类型: C。

例:

```
?time()    && 返回值为: 10:18:45
```

3. 系统日期时间函数

格式: `datetime()`。

功能: 返回系统日期时间。

函数返回值数据类型: T。

例:

```
?datetime() && 返回值为: 08/15/21 10:18:45 AM
```

4. 求年份函数

格式: `year(D)`。

功能: 返回日期型数据 D 的年份。

函数返回值数据类型: N。

例 2-9 分别计算以下日期的年份。

```
?year(date())    && 返回值为: 2021  
?year({^2003-07-15}) && 返回值为: 2003
```

5. 求月份函数

格式: `month(D)`。

功能: 返回日期型数据 D 的月份。

函数返回值数据类型: N。

例:

```
?month(date()) && 返回值为: 08
```

6. 求天数函数

格式: `day(D)`。

功能: 返回日期型数据 D 的天数。

函数返回值数据类型: N。

例:

```
?day(date())    && 返回值为: 15
```



五、数据类型转换函数

说明：以下函数介绍中出现的 N 表示数值型数据；C 表示字符型数据；D 表示日期型数据。

1. 将数值型数据转换成字符型数据函数

格式 1：str(N)。

功能：将数值型数据 N 转换为字符型数据，默认保留 10 个字符长度，没有小数。

函数返回值数据类型：C。

例：

```
?str(3.14)    && 返回值为：□□□□□□□□ 3
```

格式 2：str(N1,N2)。

功能：将数值型数据 N1 转换为字符型数据，转换后字符串的长度由 N2 决定，没有小数。

函数返回值数据类型：C。

例：

```
?str(3.14,4)  && 返回值为：□□□□ 3
```

格式 3：str(N1,N2,N3)。

功能：将数值型数据 N1 转换为字符型数据，转换后字符串的长度由 N2 决定，保留小数位数由 N3 决定。

函数返回值数据类型：C。

例：

```
?str(3.14,3,1) && 返回值为：□ 3.1
```

2. 将字符型数据转换为数值型数据函数

格式：val(C)。

功能：将字符型数据 C 转换为数值型数据，遇到第一个不是数字、正负号、小数点的字符截止。第一个字符不是数字、正负号、小数点时返回值为 0。

函数返回值数据类型：N。

例 2-10 分别计算以下字符型数据转换为数值型数据的值。

```
?val("-123.45A")    && 返回值为：-123.45  
?val("A12")         && 返回值为：0.00
```

3. 将字符型数据转换成日期型数据函数

格式：ctod(C)。



功能：将字符型数据转换为日期型数据。

函数返回值数据类型：D。

例：

```
?ctod("08/12/2021")    && 返回值为：08/12/21
```

4. 将日期型数据转换成字符型数据函数

格式：dtoc(D)。

功能：将日期型数据转换为字符型数据。

函数返回值数据类型：C。

例：

```
?dtoc({^2021-04-02})    && 返回值为：04/02/21
```

5. 宏代换函数

格式：&< 字符型内存变量 >。

功能：去掉字符型内存变量最外层的定界符。

函数返回值数据类型：不确定。

例 2-11 求下列程序段的结果。

```
n=8  
m=3  
k="m+n"  
?1+&k                && 返回值为：12
```

➡ 第四节 表 达 式

📝 考 试 要 点

1. 各类表达式的运算符及运算符优先级。
2. 各类表达式功能、用法，以及参数和运算结果的数据类型。

一、算术表达式

1. 定义

算术表达式是用算术运算符将数值型数据连接起来的式子。



2. 运算符

优先级从高到低依次为 ()、** 或 ^ (乘方)、*、/、% (求余)、+、-。

例：

```
?3*4^2-5/10+2^3  && 返回值是 55.5
```

二、字符表达式

说明：以下表达式中 C 表示字符型数据；L 表示逻辑型数据；□表示空格。

1. 定义

字符表达式是用字符运算符将字符型数据连接起来的式子。

2. 运算符

字符运算符包括 +(完全连接运算符)、-(不完全连接运算符)、\$(包含符)。

3. 格式

格式 1: C1+C2。

功能：将 C1 和 C2 完整的连接在一起。

函数返回值数据类型：C。

例：

```
? [665 □□ ]+[870]  && 返回值为：665 □□ 870
```

格式 2: C1-C2。

功能：将 C1 和 C2 连接在一起，若 C1 尾部有空格，则将 C1 尾部的空格放在 C2 的后面。

函数返回值数据类型：C。

例：

```
? [665 □□ ]-[870]  && 返回值为：665870 □□
```

格式 3: C1\$C2。

功能：看 C1 是否包含在 C2 里，如果是，返回 .T.，否则返回 .F.。

函数返回值数据类型：L。

例：

```
? "fh" $ "jfhgs"  && 返回值为 .T.
```

三、日期和时间表达式

说明：以下表达式中的 D 表示日期型数据；N 表示数值型数据；T 表示日期时间型



数据。

1. 定义

日期和时间表达式是指含有日期型或日期时间型数据的表达式。

2. 运算符

日期和时间表达式运算符包括 +、-。

3. 格式

格式 1：日期 +< 天数 >。

功能：若干天后的某个日期。

函数返回值数据类型：D。

例：

```
?{^2021/09/5}+10 && 返回值为：09/15/21
```

格式 2：日期 -< 天数 >。

功能：若干天前的某个日期。

函数返回值数据类型：D。

例：

```
?{^2021/09/15}-10 && 返回值为：09/05/21
```

格式 3：日期 1- 日期 2。

功能：两个日期相差天数。

函数返回值数据类型：N。

例：

```
?{^2021/09/18}-{^2021-09-20} && 返回值为：-2
```

格式 4：日期时间 1- 日期时间 2。

功能：两个日期时间相差的秒数。

函数返回值数据类型：N。

例：

```
?{^2021/09/15 10:33:40}-{^2021/09/10 12:34:20} && 返回值为：424760
```

注意：

两个日期不能相加。



四、关系表达式

1. 定义

关系表达式是用关系运算符将两个同类型的数据连接起来的式子。

2. 运算符

关系运算符包括 >、<、>=、<=、=、==（精确等于，只能用于字符型数据的比较）、!= 或 <> 或 #（不等于）。

3. 格式

格式：<表达式 1><关系运算符><表达式 2>。

功能：两个同类型数据的比较。

函数返回值数据类型：L（即 .T. 或 .F.）。

说明 1：各种类型数据的比较规则如下。

（1）数值型数据和货币型数据按数值大小比较。

例 2-12 分别计算以下关系表达式的计算结果。

?456>68 && 返回值 .T.

? \$879>=\$70 && 返回值 .T.

（2）日期或日期时间型比较，越靠后越大。

例：

?{^2020-9-16}>{^1996-6-9} && 返回值 .T.

（3）逻辑真大于逻辑假。

例：

? .T.>.F. && 返回值 .T.

（4）汉字比较默认按拼音字母进行比较，a 最小，z 最大。

例：

? "助教">"讲师" && 返回值 .T.

（5）精确等于“==”，只应用于字符型数据，只有左右两边完全相同才为真。

例：

? "计算机考试"=="计算机" && 返回值 .F.

（6）非精确等于“=”，当等号右边的字符串与等号左边的字符串的前几个字符相同时，即为真。

例 2-13 分别计算以下关系表达式的计算结果。

? " 计算机考试 "=" 计算机 " && 返回值 .T.
姓名 =" 李奋斗 "
? 姓名 =" 李 " && 返回值 .T.

说明 2：取整函数 int(N) 和求余函数 mod(N1,N2) 在关系表达式中的重点应用。

下面判断一个数能否被另一个数整除（特例：奇偶判断）

（1）判断任意两个数 m 能被 n 整除的条件：

mod(m,n)=0&& 或 int(m/n)=m/n

（2）判断 m 是一个偶数的条件：

mod(m,2)=0&& 或 int(m/2)=m/2

（3）判断 m 是一个奇数的条件：

mod(m,2)=1&& 或 int(m/2)! =m/2 或 mod(m,2)! =0

五、逻辑表达式

1. 定义

逻辑表达式是由逻辑运算符将逻辑型数据连接起来的式子。

2. 逻辑运算符优先级由高到低排列的顺序

其顺序依次为 NOT 或 .NOT. 或 ! → .AND. 或 AND. → .OR. 或 OR。

3. 格式

格式：< 逻辑型数据 1>< 逻辑运算符>< 逻辑型数据 2>。

函数返回值数据类型：L（即 .T. 或 .F.）。

4. 逻辑运算的规则

逻辑运算结果见表 2-2。

表 2-2 逻辑运算结果

表达式	结果	表达式	结果
not .T.	.F.	.T. or .T..	.T.
not .F.	.T.	.T. or .F.	.T.
.T. and .T.	.T.	.F. or .T.	.T.
.T. and .F.	.F.	.F. or .F.	.F.
.F. and .T.	.F.	—	—
.F. and .F.	.F.	—	—



(1) 逻辑非 (not) 表示“不”，是对后面逻辑型数据的否定，不假即真，不真即假。

(2) 逻辑与 (and) 表示“并且”，只有当其左右的逻辑型数据同时为真才是真，否则为假。

(3) 逻辑或 (or) 表示“或者”，只有当其左右的逻辑型数据同时为假结果才是假，否则为真。

例 2-14 分别计算以下关系表达式的计算结果。

设：工资=600, 职称="讲师", 性别="女"

? 工资 > 550 and 职称="讲师" and 性别="女" && 返回值为 .F.

? 工资=600 and 职称="教授" or 性别="女" && 返回值为 .T.

? 工资 > 550 and 职称="助教" or 职称="教授" && 返回值为 .F.

? 性别="男" or not 职称="讲师" && 返回值为 .F.

注意：

函数和五类表达式共存时，运算优先级由高到低依次为函数→数值表达式→字符表达式→日期和时间表达式→关系表达式→逻辑表达式。即函数和表达式共存时，先算函数。

习题 2

一、单选题

- 下列字符型常量的表示中，错误的是 ()。
A. '[x--u]' B. '[' 计算机应用 ']' C. '[' 大庆 ']' D. '65-3'
- 下列可以作为字段名的是 ()。
A. FAT-2 B. 2FAT C. FAT+2 D. FAT_2
- 设有一个字段变量“姓名”，目前值为“王丽”，又有一个内存变量“姓名”，其值为“张小敏”，则命令“? 姓名”的结果为 ()。
A. “张小敏” B. “王丽” C. 王丽 D. 张小敏
- DIMENSION M(9),N(3,5)，数组 M 和 N 的元素个数分别为 ()。
A. 7 21 B. 6 9 C. 9 15 D. 6 5
- 若 D1、D2 是日期型变量，则在 D1+10、D2-10、D1-D2 和 D1+D2 四个表达式中，有效表达式个数为 ()。
A. 4 B. 2 C. 3 D. 1
- 下列表达式中，写法错误的是 ()。



- A. "大庆 "+DTOC({^2021/05/01}) B. "大庆 "+"0123"
C. {^2021/04/01}+10 D. .T+.F.
7. 数学表达式 $6 \leq X \leq 15$ 在 Visual FoxPro 中应表示为 ()。
- A. $X \geq 6$ and $X \leq 15$ B. $X \geq 6$ or $X \leq 15$
C. $6 \leq X$ or $X \leq 15$ D. $X \leq 6$ and $15 \leq X$
8. 在 "档案" 表中, 婚否是 L 型字段, 性别是 C 型字段, 若检索 "已婚的讲师", 应使用逻辑表达式 ()。
- A. 婚否=.T. and 职称='讲师' B. 婚否 and 职称=讲师
C. 婚否 or 性别='讲师' D. 已婚 or 性别=讲师

二、填空题

1. 要存储员工上下班时间, 应该选用的数据类型是_____。
2. 已知 $P='1'$, $Q='2'$, $X12='COOL'$, 则表达式 $X \& P \& Q$ 的值为_____。
3. 表达式 $\{^2021/09/18\} - \{^2021/09/20\}$ 的值是_____。
4. Visual FoxPro 的字符表达式中, 字符运算符有_____、_____、_____。
5. 逻辑运算符优先级最低的是_____。

三、判断题

- () 1. dimension 和 declare 都能用来定义一维和二维数组。
- () 2. 4E-8 是一个变量。
- () 3. 数组中要求各个数组元素的数据类型必须相同。
- () 4. mod(N1,N2) 表示求 N1 除以 N2 的整数部分。
- () 5. dtoc() 函数是将字符型转化为日期型。
- () 6. $\{^2021/10/10\} - 1 = \{^2021/10/9\}$ 的结果是逻辑值。

第三章

表 操 作

学习目标 >

- ① 掌握表的建立、打开、修改、显示操作
- ② 掌握记录指针定位及表相关的测试函数
- ③ 掌握排序、索引的基础知识
- ④ 掌握工作区的概念和基础知识



第一节 表的基本操作

考试要点

1. 表的基础知识。
2. 表的建立、打开、关闭、复制命令及操作要点。

一、表的基础知识

表的扩展名：.dbf 或 dbf。

1. 表的分类

(1) 分类依据：表是否属于某一数据库。

(2) 分类。

① 自由表：不属于任何数据库的表。

② 数据库表：属于某一数据库的表。

(3) 自由表和数据库表的转换。自由表和数据库表可以相互转换，当自由表加入数据库后就变成数据库表，从数据库中移出就变成自由表，自由表和数据库表的功能是不一样的，一个表只能属于一个数据库。

2. 表的组成

表结构：字段名、数据类型、字段宽度、是否为空等。

表记录：表中的行数据。

说明：

只有数据库表可以使用长字段名；字段——表中的列，列名即为字段名；记录——表中的行。

二、表的建立

1. 表建立步骤

- (1) 创建表结构。在“表设计器”窗口建立表的结构。
- (2) 输入表记录。



2. 创建表

命令格式：create [< 表文件名 >]。

功能：打开“表设计器”创建表结构。

例 3-1 创建表：学生 .dbf。

```
create 学生          && 创建表 " 学生 .dbf"
```

三、表的打开和关闭

1. 打开表

命令格式：use < 表文件名 >。

▲ 注意：

- (1) 扩展名可写可不写。
- (2) 打开一个新表，原来的表自动关闭。
- (3) 打开表，记录指针默认指向第 1 条记录。

2. 关闭表

命令格式：use

例 3-2 先打开表“学生 .dbf”，然后再关闭此表。

```
use 学生          && 打开表：学生 .dbf  
use              && 关闭当前打开的表：学生 .dbf
```

四、表的复制

1. 复制表

命令格式：copy to < 表文件名 >。

功能：将打开的表文件复制成一个新的表文件（表结构和表记录一起复制到新表中）。

▲ 注意：

新复制的表文件不能与原表文件同名。

2. 复制表结构

命令格式：copy structure to < 表文件名 >。

功能：只复制表结构，即复制的新表中只有表结构，没有表记录，为空表。

例 3-3 打开表“学生 .dbf”，复制表“学生 .dbf”，生成新表“学生 1.dbf”；复制



“学生.dbf”的表结构，生成新表“学生 2.dbf”。

use 学生	&& 打开要复制的表，表名：学生.dbf
copy to 学生 1	&& 复制成新表，新表名：学生 1.dbf
copy structure to 学生 2	&& 仅复制表结构，新表名：学生 2.dbf

第二节 表的显示

考试要点

1. 表结构的显示。
2. 表记录的显示。
3. 范围子句、for 子句及其用法。

一、表结构显示

1. 显示表结构

命令格式 1：list structure。

功能：连续显示表结构（满一屏后直接在下一屏显示）。

命令格式 2：display structure。

功能：分屏显示表结构（满一屏后按任意键，进入下一屏继续显示）。

例 3-4 显示“学生.dbf”的表结构。

use 学生	&& 打开表“学生.dbf”
list structure	&& 显示表“学生.dbf”的表结构
display structure	&& 显示表“学生.dbf”的表结构

2. 表结构中用户实际可用字段宽度与总宽度关系

用 list structure 命令在“主窗口”显示表的结构，用户实际字段宽度 = 总宽度 - 1，如图 3-1 所示。