

高等学校规划教材
国家精品课程系列教材

数据库原理与实践

(Access 版)

董卫军 邢为民 索琦 编著
耿国华 主审

电子工业出版社
Publishing House of Electronics Industry
北京 · BEIJING

内 容 简 介

本书是国家精品课程“计算机基础”系列课程“数据库原理与技术”的主教材，全书由数据库基础理论篇和 Access 应用实践篇两大部分组成，第一部分为数据库基础理论，介绍数据库组织、管理和使用的一般知识，包括数据库基础、数据库设计和数据库安全 3 个章节；第二部分为 Access 应用实践，主要介绍数据库原理在 Access 上的实现，内容包括 Access 简介、创建数据库、查询数据、窗体的使用、数据报表、数据访问页、宏、VBA 和模块、综合实例 9 个章节。

本书基础理论部分与 Access 应用实践部分相辅相成，既照顾到理论基础的坚实，又强调技术实践的应用。同时，在编写时兼顾了计算机等级考试的要求。为方便教学，本书还配有电子课件，任课教师可以登录华信教育资源网（www.hxedu.com.cn）免费注册下载。

本书可作为高等院校计算机基础课程，以及相关专业数据库技术课程的教材，也可作为全国计算机等级考试二级 Access 的培训或自学教材。

未经许可，不得以任何方式复制或抄袭本书之部分或全部内容。

版权所有，侵权必究。

图书在版编目(CIP)数据

数据库原理与实践：Access 版 / 董卫军，邢为民，索琦编著. —北京：电子工业出版社，2011.6

高等学校规划教材

ISBN 978-7-121-13385-5

I. ①数… II. ①董…②邢…③索… III. ①关系数据库—数据库管理系统，Access—高等学校—教材
IV. ①TP311.138

中国版本图书馆 CIP 数据核字(2011)第 074940 号

策划编辑：索蓉霞

责任编辑：索蓉霞 特约编辑：李 涛

印 刷：

装 订：

出版发行：电子工业出版社

北京市海淀区万寿路 173 信箱 邮编：100036

开 本：787×1092 1/16 印张：15 字数：384 千字

印 次：2011 年 6 月第 1 次印刷

定 价：29.00 元

凡所购买电子工业出版社图书有缺损问题，请向购买书店调换。若书店售缺，请与本社发行部联系，联系及邮购电话：(010)88254888。

质量投诉请发邮件至 zlt@phei.com.cn，盗版侵权举报请发邮件至 dbqq@phei.com.cn。

服务热线：(010)88258888。

前 言

数据库技术是计算机科学技术发展的重要内容，是构成信息系统的重要基础。数据库技术已广泛应用于各类管理信息系统（MIS）、决策支持系统（DDS）、计算机辅助软件工程（CASE）等领域。同时，数据技术与网络通信技术、多媒体技术等新技术的结合又促进了数据库技术的进一步发展。在信息已作为战略资源的今天，建设以数据库为核心的各类信息系统和应用系统，对提高企业效益、改善部门管理具有重要意义。因此，学习和掌握数据库技术的基本知识和基本技能已成为大学生必备的素质要求。

本书是国家精品课程“计算机基础”系列课程“数据库原理与技术”的主教材，教材从培养学生分析问题和解决问题的能力入手，立足于“以理论为基础，以实例为引导，以应用为目的”，采用理论结合实例的模式努力把知识点融入到案例中，做到通俗易懂，循序渐进，满足应用型人才培养的特点和需求。在体系结构安排上，尽可能地将理论、知识点、实例相结合，力求在内容选定、教学方式和方法、对学生的学习方法的指导等方面全面地贯彻素质教育理念。

本教材共 12 章，分为数据库基础理论篇和 Access 应用实践篇两部分。

数据库基础理论篇主要介绍数据库组织、管理和使用的一般知识，包括数据库基础、数据库设计和数据库安全 3 个章节，主要包含数据库系统、数据模型、关系运算、数据库设计、关系规范化、数据库安全等方面的知识，重点介绍了从基本原理到系统设计再到系统安全三环节的技术和方法，使读者能够对数据库有一个从外到里，由浅入深的理解。

Access 应用实践篇从实用性出发，以 Access 为基础介绍了 Access 数据库程序设计所涉及的基本概念、数据表设计方法和程序设计方法。主要内容包括：Access 简介、创建数据库、查询数据、窗体的使用、数据报表、数据访问页、宏、VBA 和模块、综合案例 9 个章节。通过引例，循序渐进地介绍了数据库的设计、建立与使用方法，能够让读者在很短的时间内掌握 Access 数据库设计，实现信息的有效管理。

本书基础理论部分与 Access 应用实践部分相辅相成，既照顾到理论基础的坚实，又强调技术实践的应用。同时，在编写时兼顾了计算机等级考试的要求。为方便教学，本书还配有电子课件，任课教师可以登录华信教育资源网（www.hxedu.com.cn）免费注册下载。

本书由多年从事计算机教学的一线教师编写，董卫军编写第 1~3 章和第 10~11 章，邢为民编写第 4~7 章；索琦编写第 8~9 章和第 12 章；全书由董卫军统稿，西北大学耿国华教授主审。感谢教学团队成员的帮助，感谢西北大学教务处多年来的支持。由于水平有限，书中难免有不妥之处，恳请指正。

编 者

于西安·西北大学

目 录

数据库基础理论篇

第 1 章 数据库基础	1	1.6.1 三级模式	22
1.1 数据库基本概念	1	1.6.2 两种映射	23
1.1.1 数据与数据处理	1	1.7 访问数据库的主要过程	24
1.1.2 数据库	2	1.8 关系数据库	24
1.1.3 数据库管理系统	4	1.8.1 基本概念	24
1.1.4 数据库系统的构成	4	1.8.2 关系数据库的体系结构	25
1.2 数据管理的发展	6	1.8.3 关系模型的完整性规则	27
1.2.1 人工管理阶段	6	1.8.4 关系代数	29
1.2.2 文件系统阶段	6	习题 1	31
1.2.3 数据库系统阶段	6	第 2 章 数据库设计	36
1.2.4 分布式数据库系统阶段	7	2.1 工程化设计思想	36
1.2.5 数据管理新技术	7	2.1.1 软件工程	36
1.3 常用数据库管理系统	10	2.1.2 软件生命周期	36
1.3.1 IBM 的 DB2	10	2.2 数据库设计概述	37
1.3.2 Oracle	11	2.3 数据库设计的基本步骤	37
1.3.3 Informix	12	2.3.1 规划	38
1.3.4 Sybase	12	2.3.2 需求分析	39
1.3.5 SQL Server	14	2.3.3 概念结构设计	40
1.3.6 PostgreSQL	14	2.3.4 逻辑结构设计	41
1.3.7 MySQL	15	2.3.5 物理设计	44
1.3.8 Access 数据库	15	2.3.6 系统实施	45
1.3.9 FoxPro 数据库	16	2.3.7 运行和维护	45
1.4 客观世界的表示	16	2.4 关系模式的规范化	46
1.4.1 现实世界	17	2.4.1 函数依赖和键	47
1.4.2 概念世界	17	2.4.2 关系模式的范式	48
1.4.3 数据世界	19	2.4.3 模式分解	49
1.5 数据模型	19	习题 2	52
1.5.1 数据模型的概念	19	第 3 章 数据库安全	56
1.5.2 数据模型组成要素	20	3.1 数据库安全	56
1.5.3 逻辑模型的基本类型	20	3.1.1 系统安全的主要风险	56
1.6 数据库的体系结构	22	3.1.2 数据库安全的特征	57

3.2 数据库的安全层次	58	3.3.1 数据库加密技术	60
3.2.1 网络系统层次安全	59	3.3.2 存取管理技术	61
3.2.2 操作系统层次安全	59	3.3.3 安全审计技术	62
3.2.3 数据库管理系统层次安全	59	3.3.4 备份与恢复	65
3.3 数据库安全技术	60	习题 3	69

Access 应用实践篇

第 4 章 Access 简介	71
4.1 Access 简介	71
4.1.1 Access 产生与发展	71
4.1.2 Access 安全方式	72
4.2 Access 基本概念	73
4.2.1 数据库和数据库对象	73
4.2.2 Access 基本功能	75
4.2.3 Access 操作模式	76
4.3 Access 的启动和退出	76
4.3.1 启动 Access	76
4.3.2 退出 Access	77
习题 4	78

第 5 章 创建数据库	80
5.1 创建数据库	80
5.1.1 创建一个空数据库	80
5.1.2 通过模板建立数据库	81
5.2 创建表	82
5.2.1 用表向导建立表	82
5.2.2 通过设计器创建表	83
5.3 表间关系的创建	87
5.3.1 表的索引	87
5.3.2 表间关系	88
5.3.3 创建关系	89
5.3.4 编辑和删除关系	90
5.4 修改数据库结构	90
5.4.1 对表的操作	90
5.4.2 对字段的操作	91
5.4.3 对表的行与列操作	92
5.5 表中数据的操作	92
5.5.1 普通数据的添加、修改和	

删除	92
5.5.2 图片、声音和影像的输入	93
5.5.3 建立超链接	93
5.5.4 设置数据表格形式	94
5.5.5 查找和替换数据	94
5.5.6 排序与筛选	94
5.6 数据的导入和导出	96
5.6.1 导入和链接	96
5.6.2 链接外部数据库	96
5.6.3 导入外部数据库	97
5.6.4 导出数据	98
习题 5	98

第 6 章 查询数据	102
6.1 查询与表	102
6.2 常见的查询	102
6.2.1 选择查询	102
6.2.2 参数查询	102
6.2.3 交叉表查询	103
6.2.4 操作查询	103
6.2.5 SQL 查询	103
6.3 创建选择查询	104
6.3.1 利用查询设计视图建立	104
6.3.2 利用查询向导建立	107
6.4 创建参数查询	107
6.5 创建交叉表查询	108
6.6 操作查询	109
6.6.1 删除查询	109
6.6.2 更新查询	110
6.6.3 生成表查询	110
6.6.4 追加查询	111
6.7 Access SQL 查询	112

6.7.1	SQL 的特点	112	8.3	使用报表向导建立报表	146
6.7.2	SQL 数据库的体系结构	113	8.4	自动报表与图表向导	147
6.7.3	Access SQL 的特点	113	8.4.1	创建自动报表	147
6.7.4	Access SQL 的数据定义	115	8.4.2	创建图表	148
6.7.5	Access SQL 的数据查询	117	8.5	通过设计视图创建报表	149
6.7.6	Access SQL 的数据更新	120	8.5.1	创建过程	149
6.7.7	建立 SQL 查询	121	8.5.2	报表控件	150
6.8	查询的打开与修改	122	8.5.3	在报表中添加分组	151
6.8.1	打开查询	122	8.5.4	添加计算字段	152
6.8.2	修改查询	122	8.6	修饰报表	153
习题 6		123	8.6.1	添加文字	153
第 7 章	窗体的使用	128	8.6.2	设置内容的显示效果	153
7.1	窗体的功能与构成	128	8.6.3	调整显示对齐方式	153
7.1.1	窗体的功能	128	8.6.4	调整行间距	154
7.1.2	窗体的构成	128	8.7	打印报表	154
7.2	创建窗体	129	8.7.1	页面设置	154
7.2.1	自动窗体创建	129	8.7.2	预览与打印报表	154
7.2.2	向导创建	130	习题 8		155
7.3	窗体设计视图与控件	130	第 9 章	数据访问页	158
7.3.1	窗体的设计视图	130	9.1	数据访问页的基本概念	158
7.3.2	工具箱的使用	131	9.1.1	数据访问页	158
7.3.3	窗体中的常见控件	131	9.1.2	数据访问页视图	158
7.3.4	给对象添加控件	132	9.1.3	数据访问页的组成	159
7.4	创建子窗体	136	9.2	创建数据访问页	159
7.5	通过窗体处理数据	137	9.2.1	自动创建	159
7.5.1	窗体视图工具栏	137	9.2.2	使用向导创建	160
7.5.2	记录导航按钮集	137	9.2.3	使用设计器创建	160
7.5.3	处理数据	138	9.3	美化数据访问页	161
7.6	创建切换面板	138	9.3.1	添加标签	161
7.6.1	切换面板的作用	138	9.3.2	添加命令按钮	162
7.6.2	切换面板的创建	139	9.3.3	添加滚动文字	162
习题 7		140	9.3.4	添加主题和背景	163
第 8 章	数据报表	143	9.3.5	添加 Office 电子表格	164
8.1	报表的功能与类型	143	9.3.6	建立超链接	164
8.1.1	报表的功能	143	习题 9		164
8.1.2	报表类型	143	第 10 章	宏	166
8.2	报表设计的组成	144	10.1	理解宏	166
8.2.1	报表的节	144	10.1.1	宏的分类	166
8.2.2	报表的常见节	145	10.1.2	宏的介绍	166

10.2 宏创建与运行	170	11.6.3 选择结构	197
10.2.1 创建宏	170	11.6.4 循环结构	200
10.2.2 运行宏	175	11.6.5 其他辅助控制语句	202
10.3 宏向 Visual Basic 代码转换	176	11.7 VB 的过程	204
10.3.1 转换窗体或报表中的宏	176	11.7.1 Sub 过程	204
10.3.2 转换全局宏	176	11.7.2 Function 过程	206
习题 10	176	11.7.3 参数的传递	208
第 11 章 VBA 和模块	179	11.7.4 过程和变量的作用域	208
11.1 VBA 简介	179	11.8 面向对象程序设计基础 知识	210
11.1.1 宏与 VBA 的差异	179	11.8.1 几个基本概念	210
11.1.2 VBA 与 VB 的区别	179	11.8.2 VBA 程序设计的基本 过程	211
11.2 VBA 开发环境	180	11.9 应用举例	214
11.2.1 开发环境界面	180	习题 11	219
11.2.2 编写代码	180	第 12 章 综合实例	224
11.2.3 模块的调试	181	12.1 需求分析	224
11.3 VBA 基础	181	12.2 系统设计	224
11.3.1 VB 语言元素	181	12.3 系统实现	226
11.3.2 书写规则	183	12.3.1 创建数据库	226
11.4 Visual Basic 语言基础	184	12.3.2 查询设计	226
11.4.1 数据类型	184	12.3.3 报表设计	227
11.4.2 常量与变量	187	12.3.4 窗体设计	227
11.4.3 表达式与运算符	190	12.4 系统测试及运行	230
11.5 数组	193	习题 12	231
11.5.1 基本概念	193	参考文献	232
11.5.2 一维数组的使用	193		
11.5.3 多维数组的使用	195		
11.6 基本程序设计	196		
11.6.1 程序的基本结构	196		
11.6.2 顺序结构	196		

数据库基础理论篇

第 1 章 数据库基础

数据是计算机处理的对象，因而如何有效地管理数据就是一个重要的问题。20 世纪 50 年代中期以前没有专门用于数据管理的软件。操作系统出现以后，可以通过操作系统管理数据。但由于文件之间缺乏联系，很难解决重复存储和不一致的问题。20 世纪 60 年代末提出了数据库的概念，它能为多个用户、多种应用所共享，具有最小的冗余度，较高的数据独立性。

1.1 数据库基本概念

1.1.1 数据与数据处理

在实际的工作和学习中，每个人都有很多亲戚和朋友，为了保持与他们的联系，我们常常通过一定的方法将他们的姓名、地址、电话等信息都记录在通讯录中（存于一个笔记本中，或存于手机中），这样在查找时就方便了。这个“通讯录”就是一个最简单的“数据库”，而每个人的姓名、地址、电话等信息就是这个数据库中的“数据”。我们可以根据需要随时在这个“数据库”中添加新朋友的个人信息，也可以修改某个人的电话号码“数据”。

1. 数据

数据是事物特性的反映和描述，是符号的集合。数据不仅包括狭义的数值数据，而且包括文字、声音、图形、图像等一切能被计算机接收并处理的符号。数据在空间上的传递称为通信（以信号方式传输），数据在时间上传递称为存储（以文件形式存取）。信息是和数据关系密切的另外一个概念，数据是信息的符号表示（或称为载体），信息则是数据的内涵，是对数据语义的解释。

2. 数据处理

数据处理可定义为对数据的收集、存储、加工、分类、检索、传播等一系列活动。这些活动是数据处理业务的基本环节，是任何数据处理业务中必不可少的共有部分。对于数据处理而言，其基本目的是从大量的、杂乱无章的、难以理解的数据中抽取并推导出对某些特定用户有价值、有意义的数据。数据管理和数据处理有一定的联系，数据管理技术的优劣，将直接影响数据处理的效率。

1.1.2 数据库

1. 数据库的概念

数据库是存放数据的仓库，是对现实世界有用信息的抽取、加工和处理，并按一定格式长期存储在计算机内的、有组织的、可共享的数据集合。数据库的概念包括两层意思：

(1) 数据库是能够合理保管数据的“仓库”，用户在该“仓库”中存放要管理的事务数据，“数据”和“库”两个概念相结合构成数据库；

(2) 数据库包含数据管理的新方法和新技术，通过数据库可以方便地组织数据、维护数据、控制数据和利用数据。

2. 数据库的特点

相对于传统的人工管理和文件技术，数据库具有如下特点。

(1) 数据结构化

数据库中的数据是从全局观点出发建立的，按一定的数据模型进行组织、描述和存储。其结构基于数据间的自然联系，数据面向全组织，不针对特定应用，具有整体的结构化特征。

(2) 数据独立性

数据独立性是指数据的逻辑组织方式和物理存储方式与用户的应用程序相对独立。数据库的数据独立性包括物理独立性和逻辑独立性两个方面。

数据的物理独立性是指当数据的物理存储改变时，应用程序不用改变。换言之，用户的应用程序与数据库中的数据是相互独立的。数据在数据库中的存储形式是由 DBMS 管理的，应用程序要处理的只是数据的逻辑结构。

数据的逻辑独立性是指当数据的逻辑结构改变时，用户应用程序不用改变。也就是说，用户的应用程序与数据库的逻辑结构是相互独立的。

通过数据独立性可以将数据的定义和描述从应用程序中分离出来。数据的存取由 DBMS 管理，用户不必考虑存取路径等细节，从而简化了应用程序的编制，大大减少了应用程序的维护和修改工作量。

(3) 数据冗余低

数据的冗余度是指数据重复的程度。数据库系统从整体角度描述数据，数据可以被多个应用共享。这样可以节约存储空间、减少存取时间，而且可以避免数据之间的不相容和不一致。但还需要必要的冗余，必要的冗余可保持数据间的联系。

(4) 统一的数据管理和控制

数据库对系统中的用户来说是共享资源。计算机的共享一般是并发的，即多个用户可以同时存取数据库中的数据，甚至可以同时存取数据库中同一个数据。因此，数据库管理系统必须提供以下几方面的数据控制保护功能。

① 数据的安全性保护

数据的安全性保护可以防止不合法的使用所造成的数据泄密和破坏，使每个用户只能按规定对某种数据以某些方式进行使用和处理。例如，可以使用身份鉴别、检查口令或其他手段来检查用户的合法性，只有合法用户才能进入数据库系统。

② 数据的完整性控制

数据的完整性指数据的正确性、有效性和相容性。完整性检查可以保证数据库中的数据在输入和修改过程中始终符合原来的定义和规定，在有效的范围内保证数据之间满足一定的

关系。例如，月份是1~12之间的正整数，性别是“男”或“女”，成绩是大于等于0且小于等于100的整数，学生的学号是唯一的。

③ 数据库恢复

计算机系统的硬件故障、软件故障、操作员的失误以及人为的攻击和破坏，都会影响数据库中数据的正确性，甚至会造成数据库部分或全部数据丢失。数据库恢复机制能够及时发现和修复故障，从而防止数据被破坏。

④ 并发控制

当多个用户的并发进程同时存取、修改数据库时，可能会发生由于相互干扰而导致结果错误的情况，并使数据库完整性遭到破坏。因此，必须对多用户的并发操作加以控制和协调。

3. 数据的组织级别

数据库中数据的组织一般可以分为四级：数据项、记录、文件和数据库，它们之间的关系如图1.1所示。

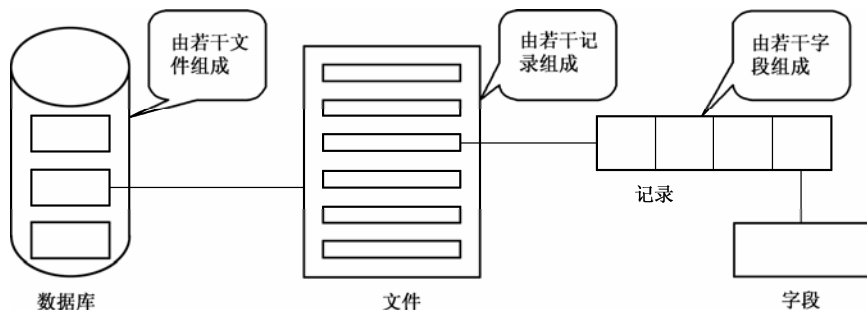


图 1.1 数据组织级别

（1）数据项

数据项是定义数据的最小单位，也叫元素、基本项、字段等。数据项与现实世界实体的属性相对应，每个数据项都有一个名称，称为数据项名。数据项的值可以是数值、字母、字母数字、汉字等形式。数据项的取值有一定的范围，称为域，域以外的任何值对该数据项都是无意义的。

（2）记录

记录由若干相关联的数据项组成，是处理和存储信息的基本单位，是关于一个实体的数据总和。构成记录的数据项表示实体的若干属性。记录有“型”和“值”的区别，“型”是同类记录的框架，它定义记录的组成；而“值”是记录反映的实体的内容。为了唯一标识每个记录，就必须有记录标识符（也叫关键字）。能唯一标识记录的关键字称为主关键字，其他标识记录的关键字称为辅关键字。

（3）文件

文件是一给定类型的记录的全部具体值的集合，文件用文件名称标识。由于数据库文件可以看成是具有相同性质的记录的集合，因而数据库文件具有以下特性：

- ① 文件的记录格式相同，长度相等；
- ② 不同的行是不同的记录，因而具有不同的内容；
- ③ 不同的列表示不同的字段，同一列中的数据性质（属性）相同；
- ④ 每一行各列的内容是不能分割的，但行的顺序和列的顺序不影响文件内容的表达。

（4）数据库

数据库是比文件更大的数据组织形式，数据库是具有特定联系的数据的集合，也可以看

成是具有特定联系的多种类型的记录的集合。数据库的内部构造是文件的集合，这些文件之间存在某种联系，不能孤立存在。

1.1.3 数据库管理系统

数据库管理系统 (Database Management System, DBMS) 是位于用户与操作系统 (OS) 之间的一层数据管理软件，是数据库系统的核心。它为用户或应用程序提供访问数据库的方法，包括数据库的建立、查询、更新以及各种数据控制。DBMS 总是基于某种数据模型 (层次型、网状型、关系型和面向对象型等)。DBMS 具有以下基本功能。

1. 对象定义功能

DBMS 通过提供数据定义语言 (Data Definition Language, DDL) 实现对数据库中数据对象的定义，如对外模式、模式和内模式加以描述和定义；数据库完整性的定义；安全保密的定义，如用户口令、级别、存取权限；存取路径的定义，如索引的定义。

2. 数据操纵功能

DBMS 提供数据操纵语言 (Data Manipulation Language, DML) 实现对数据库中数据的一些基本操作，如检索、插入、修改、删除和排序等。DML 一般有两类：一类是嵌入主语言的 DML，如嵌入到 C++ 或 PowerBuilder 等高级语言 (称为宿主语言) 中；另一类是非嵌入式语言 (包括交互式命令语言和结构化语言)，它的语法简单，可以独立使用，由单独的解释或编译系统来执行，所以一般称为自主型或自含型的 DML。

3. 数据库控制功能

DBMS 提供的数据库控制语言 (Data Control Language, DCL) 保证数据库操作都在统一的管理下协同工作以确保事务处理的正常运行，同时保证数据库的正确性、安全性、有效性和多用户对数据的并发使用以及发生故障后的系统恢复等。数据库控制功能包括并发控制和存取控制，如安全性检查、完整性约束条件检查、数据库内部 (如索引) 和数据字典的自动维护、缓冲区大小的设置等。

4. 数据组织、存储和管理

数据组织和存储的基本目标是提高存储空间利用率和方便存取，提供多种存取方法 (如索引查找、Hash 查找、顺序查找等) 提高存取效率。DBMS 要分类组织、存储和管理各种数据，包括数据字典、用户数据、存取路径等。要确定以何种文件结构和存取方式在存储级上组织这些数据，如何实现数据之间的联系。

1.1.4 数据库系统的构成

数据库系统 DBS (Database System) 主要由硬件、数据库、软件 and 用户 4 部分构成，其基本结构如图 1.2 所示。

1. 硬件

硬件是数据库依存的物理设备，运行数据库系统的计算机硬件需要足够大的内存以存放系统软件，需要足够多的联机直接存取设备 (如大容量磁盘) 存储庞大的数据，需要足够多的脱机存储介质 (如磁盘、光盘、磁带等) 以存放备份数据，需要较高的通道能力，以提高数据传送速率，并且要求系统互联，以实现数据共享。

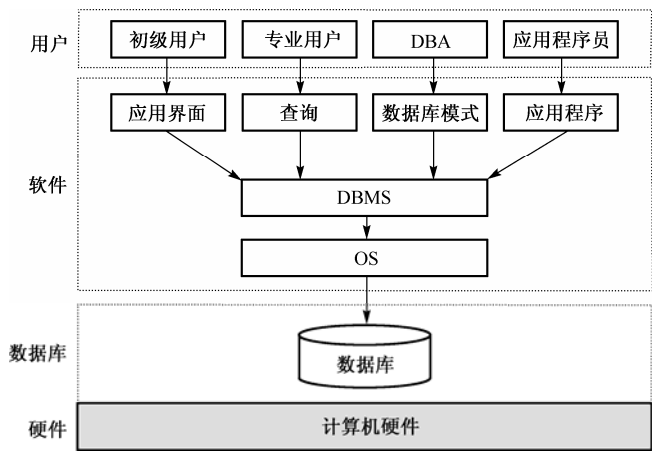


图 1.2 DBS 的基本构成

2. 数据库

数据库是数据库系统的核心和管理对象，是存储在一起相互有联系的数据集合。数据库中的数据是集成的、共享的、最小冗余的，能为多种应用服务。数据按照数据模型所提供的形式框架存放在数据库中。

3. 软件

数据库软件主要是指数据库管理系统 DBMS，DBMS 是为数据库存取、维护和管理而配置的软件，它是数据库系统的核心组成部分，DBMS 在操作系统支持下工作。数据库软件除了 DBMS 之外，还要有数据库应用系统，数据库应用系统是为特定应用而开发的数据库应用软件。数据库管理系统为数据的定义、存储、查询和修改提供支持，而数据库应用系统是对数据库中的数据进行处理和加工的软件，它面向特定应用。如管理信息系统、决策支持系统和办公自动化等都属于数据库应用系统。

4. 用户

数据库系统中存在一组参与分析、设计、管理、维护和使用数据库的人员，他们在数据库系统的开发、维护和应用中起着重要的作用。分析、设计、管理和使用数据库系统的人员主要是：数据库管理员、系统分析员、应用程序员和最终用户。

(1) 数据库管理员

数据库管理员负责全面管理和控制数据库系统，主要职责包括：设计与定义数据库系统，帮助最终用户使用数据库系统，监督与控制数据库系统的使用和运行，改进和重组数据库系统，调整数据库系统的性能，转储与恢复数据库，重构数据库。

(2) 系统分析员

系统分析员是数据库系统建设期的主要参与人员，负责应用系统的需求分析和规范说明，确定系统的基本功能和数据库结构，设计应用程序和硬软件配置并组织整个系统的开发。

(3) 应用程序员

应用程序员根据系统的功能需求负责设计和编写应用系统的程序模块，并参与对程序模块的测试。

(4) 最终用户

数据库系统的最终用户是有不同层次的，不同层次的用户其需求的信息以及获得信息的

方式也是不同的。一般可将最终用户分为操作层、管理层和决策层。他们通过应用系统的用户接口使用数据库。

1.2 数据管理的发展

数据管理包括数据组织、分类、编码、存储、检索和维护。随着硬件、软件技术的发展及计算机应用范围的扩大，数据管理经历了四个阶段。

1.2.1 人工管理阶段

20 世纪 50 年代中期以前，计算机主要用于科学计算。计算机的软硬件均不完善，硬件方面只有卡片、纸带、磁带等，没有可以直接访问、直接存取的外部存取设备。软件方面还没有操作系统，也没有专门管理数据的软件，数据由程序自行携带，数据与程序不能独立，数据不能长期保存，如图 1.3 所示。

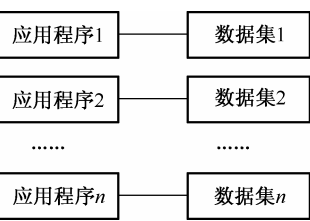


图 1.3 人工管理数据

在人工管理阶段，程序员在程序中不仅要规定数据的逻辑结构，还要设计其物理结构，包括存储结构、存取方法、输入输出方式等。当数据的物理组织或存储设备改变时，用户程序就必须重新编制。由于数据的组织是面向应用的，不同的计算程序之间不能共享数据，使得不同的应用之间存在大量重复数据。

1.2.2 文件系统阶段

20 世纪 50 年代中期至 60 年代中期，由于计算机大容量存储设备（如硬盘）的出现，推动了软件技术的发展，而操作系统的出现标志着数据管理步入一个新的阶段。在文件系统阶段，数据以文件为单位存储在外存，且由操作系统统一管理。操作系统为用户使用文件提供了友好界面。在这个阶段，程序与数据有了一定的独立性，程序与数据分开，有了程序文件与数据文件的区别。数据文件可以长期保存在外存上多次存取，进行诸如查询、修改、插入、删除等操作。文件的逻辑结构与物理结构脱钩，程序和数据分离，使数据与程序有了一定的独立性。用户的程序与数据可分别存放在外存储器上，各个应用程序可以共享一组数据，实现了以文件为单位的数据共享，如图 1.4 所示。

但在文件系统阶段，由于数据的组织仍然是面向程序的，所以存在大量的数据冗余。而且数据的逻辑结构不能方便地修改和扩充，数据逻辑结构的微小改变都会影响到应用程序。由于文件之间互相独立，因而它们不能反映现实世界中事物之间的联系，操作系统不负责维护文件之间的联系信息。如果文件之间有内容上的联系，只能由应用程序去处理。

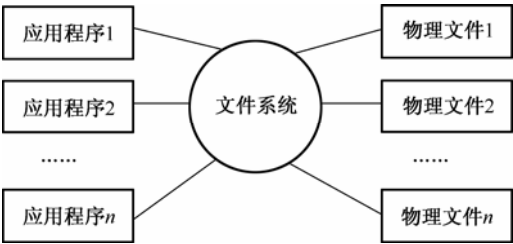


图 1.4 文件系统

1.2.3 数据库系统阶段

20 世纪 60 年代后，随着计算机在数据管理领域的普遍应用，人们对数据管理技术提出了更高的要求。要求以数据为中心组织数据，减少数据冗余，提供更高的数据共享能力，同

时要求程序和数据具有较高的独立性，以降低应用程序研制与维护的费用。数据库技术正是在这样一个应用需求的基础上发展起来的。

在文件系统阶段，人们在信息处理中关注的中心问题是系统功能的设计，因此程序设计占主导地位；而在数据库方式下，数据开始占据了中心位置，数据的结构设计成为信息系统的首要问题。数据库不仅包括数据本身，而且包括数据之间的联系。为了让多种应用程序并发地使用数据库中的共享数据，必须使数据与程序具有较高的独立性，这就需要一个软件系统对数据实行专门管理，提供安全性和完整性等统一控制，方便用户以交互命令或程序方式对数据库进行操作。为数据库的建立、使用和维护而配置的软件就是数据库管理系统 DBMS，如图1.5所示。

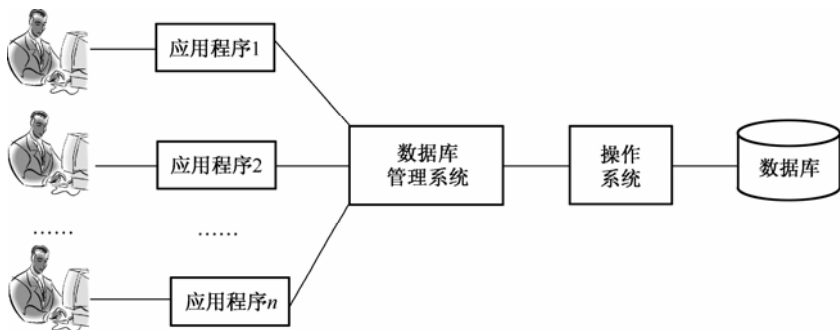


图 1.5 数据库系统阶段

1.2.4 分布式数据库系统阶段

随着数据库应用的不断发展，规模的不断扩大，集中式系统也显现了它的不足之处。如大型 DBS 的设计和操作都比较复杂，系统显得不灵活并且安全性也较差。因此，采用将数据分散的方法，把数据库分成多个，建立多台计算机上，这种系统称为分布式系统。

分布式系统是物理上分散逻辑上集中的数据库系统，系统中的数据分布存放在计算机网络的不同场地中，每个场地都有自治处理（即独立处理）能力并能完成局部应用，而每个场地也参与（至少一种）全局应用，程序通过网络通信子系统执行全局应用，如图1.6所示。

分布式系统是用通信网络连接起来的结点（也称为场地）的集合，每个结点都是拥有集中式数据库的计算机系统。

分布式数据库系统主要有三个特点。

① 数据库的数据物理上分布在各个场地，但逻辑上是一个整体。

② 每个场地既可以执行局部应用（访问本地 DB），也可以执行全局应用（访问异地 DB）。

③ 各地的计算机由数据通信网络相联系。本地计算机不能单独胜任的处理任务，可以通过通信网络取得其他 DB 和计算机的支持。

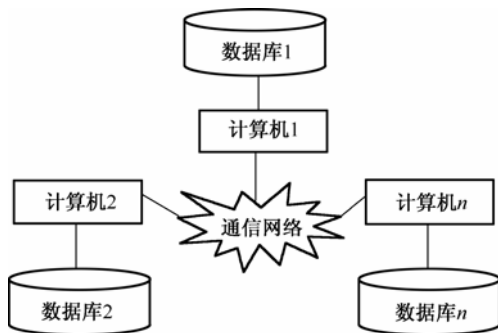


图 1.6 分布式数据库

1.2.5 数据管理新技术

从 20 世纪 60 年代末开始，数据库系统已从第一代的层次数据库、网状数据库，第二代的关系数据库系统，发展到第三代以面向对象模型为主要特征的数据库系统。随着用户应用

需求的提高、硬件技术和互联网技术的发展,促进了数据库技术与网络通信技术、人工智能技术、面向对象程序设计技术、并行计算技术等相互渗透,互相结合,形成了数据库新技术。

1. 面向对象数据库

关系型数据库系统虽然技术很成熟,但其局限性也显而易见:它能很好地处理所谓的“表格型数据”,却对越来越多的复杂类型的数据无能为力。20世纪90年代以后,人们把大量的精力花在研究“面向对象的数据库系统(Object Oriented Database)”或简称“OO数据库系统”上。

面向对象数据库系统主要有以下两个特点:

- ① 对象数据模型能完整地描述现实世界的数据结构,能表达数据间嵌套、递归联系;
- ② 具有面向对象技术的封装性和继承性的特点,提高了软件的可重用性。

面向对象数据库的主要设计思想是用新型数据库系统来取代现有的数据库系统。这对许多已经运用传统数据库系统多年并积累了大量工作数据的客户来说,无法承受新旧数据转换带来的巨大工作量及巨额开支。另外,面向对象的关系型数据库系统所使用的查询语言极其复杂。所以,虽经数年发展,面向对象的关系型数据库系统的市场发展并不理想。

2. 多媒体数据库

多媒体是多种媒体形式的有机集成,如数字、字符、文本、图形、图像、声音、视频等。其中数字、字符等称为格式化数据,文本、图形、图像、声音、视频等称为非格式化数据,非格式化数据具有数据量大、处理复杂等特点。多媒体数据库系统(简称MDBS)是结合数据库技术和多媒体技术,能够有效实现对格式化和非格式化的多媒体数据进行存储、管理和操纵等功能的数据库系统。较于传统的数据库,多媒体数据库具有以下特征。

(1) 与传统数据库的差异大

多媒体数据库虽然在理论和技术上对传统数据库有很多继承,但其处理的数据对象、数据类型、数据结构、应用对象、处理方式都与传统数据库有较大差异,因此不能简单认为多媒体数据库只是对传统数据库的一种简单扩充或者试图用传统技术来处理多媒体数据。

(2) 处理对象的复杂性

多媒体数据库存储和处理的是现实世界中的复杂对象,不仅要处理包括数字、字符等格式化数据,还要处理图像、音频、视频等非格式化数据。

(3) 媒体间的独立性

多媒体数据库面临的数据有单一媒体数据和复合媒体数据(多种单一媒体数据的结合)。多媒体数据库从实用性的要求出发,强调多媒体数据库的用户可最大限度地忽略各媒体间差异,从而实现对多媒体数据的管理和操作。

3. 主动数据库

主动数据库(Active Database, ADB)是相对于传统数据库的被动性而言的。传统的数据库系统只能根据用户或应用程序的服务请求对数据库进行存储、检索等操作,而不能根据发生的事件或数据库的状态主动作出反应。主动数据库系统(ADBMS)具有主动提供服务功能,并且能以一种统一的机制实现各种主动服务的数据库系统。

一个主动数据库系统在某一事件发生时,引发数据库管理系统去检测数据库当前状态,若满足指定条件,则触发规定执行的动作,我们称之为ECA规则。一个主动数据库系统可表示为: $ADBS = DBS + EB + EM$ 。

其中,DBS代表传统数据库系统,用来存储、操作、维护和管理数据;EB代表ECA规

则库,用来存储 ECA 规则,每条规则指明在何种事件发生时,根据给定条件应主动执行什么动作;EM 代表事件监测器,一旦检测到某事件发生就主动触发系统,按照 EB 中指定的规则执行相应的动作。

一个主动数据库系统应该具有以下功能:

① 主动数据库系统应该提供传统数据库系统的所有功能,且不能因为增加了主动性功能而使数据库的性能受到明显影响。

② 主动数据库系统必须给用户和应用提供关于主动特性的说明,且该说明应该为数据库的永久性部分。

③ 主动数据库系统必须能有效地实现主动特性,且能与系统的其他部分有效地集成在一起,包括查询、事务处理、并发控制和权限管理等。

④ 主动数据库系统应能够提供与传统数据库系统类似的数据库设计和调试工具。

4. 数据仓库

数据仓库(Data Warehouse, DW)是指为了满足中高层管理人员预测和决策分析的需要,在传统数据库的基础上产生能够满足预测和决策分析需要的数据环境。数据仓库是面向主题的、整合的、稳定的,并且时变地收集数据以支持管理决策的一种数据结构形式。数据仓库的概念包含以下四方面的含义。

① 数据仓库是面向主题的:与传统数据库面向应用进行数据组织的特点相对应,数据仓库中的数据是面向主题进行组织的。主题是一个抽象的概念,对企业信息系统中的数据在较高层次上进行抽象综合归类并进行分析利用。在逻辑意义上,它是相应企业中某一宏观分析领域所涉及的分析对象。

② 数据仓库是集成的:数据仓库的数据主要用于分析,其数据的最大特点在于它不局限于某个具体的操作数据,而是对细节数据的归纳和整理。数据仓库中的综合数据不能从原有数据库系统中直接得到,而需从其中抽取。因此,数据在进入数据仓库之前,必须进行加工与集成,处理原始数据中的所有歧义和矛盾,如单位不统一、字段的同名异义、异名同义等,将原始数据结构进行从面向应用到面向主题的转变。这是数据仓库建设中最关键、最复杂的一步。

③ 数据仓库是稳定的:数据仓库数据反映的是一段相当长的时间内历史数据的内容,是不同时间内数据快照(来自数据库的一个表或表的子集的最新拷贝)的集合,以及基于这些快照进行统计、综合和重组的导出数据,而不是联机处理的数据,不进行实时更新。

④ 数据仓库随时间变化:数据仓库的数据稳定性是针对应用来说的,即用户进行分析处理时不能进行数据更新操作。但并不是说,在数据从集成输入到数据仓库中开始到最终被删除的整个数据生存周期之中所有数据都永久不变。

当前的数据仓库系统中,直接面向用户的部分前端工具主要有两类:联机分析处理(OLAP)的分析查询型工具和数据挖掘(DM)的挖掘型工具。

(1) 联机分析处理(OLAP)

OLAP 的显著特征是能提供数据的多维概念视图,使最终用户从多角度、多侧面、多层次地考察数据库中的数据,从而深入地理解包含在数据库中的信息和内涵,多维数据分析是决策的主要内容。目前,OLAP 工具可分为两大类:基于多维数据库的 MOLAP 和基于关系数据库的 ROLAP。MOLAP 利用专有的多维数据库存储 OLAP 分析所需的数据,数据以多维方式存储并以多维视图方式显示。ROLAP 则利用关系表模拟多维数据,将分析的结果经多维处理转化为多维视图展示给用户。

(2) 数据挖掘 (MD)

数据挖掘 (Data Mining, DM) 也称为数据库中的知识发现, 是指从大量数据中挖掘出隐含的、先前未知的、对决策有潜在作用的知识和规则的过程。它主要以人工智能、机器学习、统计学等技术为基础, 高度自动化地分析企业原有数据, 作出归纳性推理, 从中挖掘出潜在的模式, 预测客户行为, 帮助企业决策者调整市场策略, 减少风险, 实现科学决策。

1.3 常用数据库管理系统

1.3.1 IBM的DB2

1. DB2 简介

DB2 是IBM公司研制的一种 关系型数据库系统, 主要应用于大型应用系统。IBM 在 1983 年发布了DATABASE 2 (DB2) for MVS, DB2 是成熟的商业数据库, 在各个行业拥有众多的客户。目前越来越多的客户改变了原来单一数据库平台的策略而转向在其新应用中使用DB2。国内的主要软件开发商, 如用友、金蝶、东软、中软、亚信、浪潮、北大青鸟等, 都基于DB2 开发其产品和应用。

2. DB2 的主要特点

DB2 具有以下基本特点:

① DB2 具有较好的可伸缩性, 可支持从大型机到单用户环境, 应用于 OS/2、Windows 等平台上。

② DB2 提供了高层次的数据利用性、完整性、安全性、可恢复性, 以及小规模到大规模应用程序的执行能力, 具有与平台无关的基本功能和 SQL 命令。

③ DB2 采用数据分级技术, 能够使大型机数据很方便地下载到局域网数据库服务器中, 使得客户机/服务器用户和基于局域网的应用程序可以访问大型机数据, 并使数据库本地化及远程连接透明化。

④ DB2 具有很好的网络支持能力, 每个子系统可以连接十几万个分布式用户, 可同时激活上千个活动线程, 对大型分布式应用系统尤为适用。

⑤ 它除了可以提供主流的OS/390 和VM操作系统, 以及中等规模的AS/400 系统之外, IBM 还提供了跨平台 (包括基于UNIX 的Linux, HP-UX, SunSolaris, 以及SCO UnixWare; 还有用于个人计算机的OS/2 操作系统, 以及微软的Windows 2000 和其早期的系统) 的DB2 产品。

⑥ DB2 数据库可以通过使用微软的开放数据库连接 (ODBC) 接口, Java数据库连接 (JDBC) 接口, 或者 CORBA接口代理被任何应用程序访问。

3. DB2 9.X

信息技术已进入面向服务架构的 (Service-Oriented Architecture, SOA) 时代。而实现 SOA 的核心是顺畅解决不同应用间数据交换的问题。可扩展标记语言 (eXtensible Markup Language, XML) 以其可扩展性、与平台无关性和层次结构等特性, 成为构建 SOA 不同应用间进行数据交换的主流语言。而如何存储和管理几何量级的 XML 数据、直接支持原生 XML 文档成为 SOA 构建效率和质量的关键。在这种情况下, IBM 公司于 2006 年发布了全面支持 Original XML 的 DB2 9, DB2 9 将传统的高性能、易用性与自描述、灵活的 XML 相结合, 转变成成为交互式、充满活力的数据服务器, 从而将数据库领域带入 XML 时代。

新的 DB2 9.X 提供了与先前版本不同的体系结构。DB2 Universal Database for Linux、UNIX 和 Windows 第一次提供了新的查询语言、新的存储技术、新的索引技术，以及支持 XML 数据及其固有层次结构的其他特性。但是，所有的 DB2 传统数据库管理特性仍然存在，包括对 SQL 和表结构的支持。新的 XML 技术使 DB2 成为“混合型”（即多种结构的）数据库管理系统（DBMS）。

1.3.2 Oracle

1. Oracle简介

Oracle（汉语译为“甲骨文”）是世界领先的信息管理软件开发商，Oracle 的关系数据库是世界第一个支持 SQL 语言的数据库。Oracle 定位于高端工作站，以及作为服务器的小型计算机，整个产品线包括数据库、服务器、企业商务应用程序以及应用程序开发和决策支持工具。Oracle 数据库产品为大多数的大公司和大型网站使用。

2. Oracle的主要特点

Oracle 具有以下基本特点。

（1）Oracle7.X以来引入了共享SQL和多线索服务器体系结构

新机制的引入减少了 Oracle 的资源占用，并增强了 Oracle 的能力，使之在低档软硬件平台上用较少的资源就可以支持更多的用户，而在高档平台上可以支持成百上千个用户。

（2）提供了基于角色（Role）分工的安全保密管理

基于角色（Role）分工的安全保密管理在数据库管理功能、完整性检查、安全性、一致性方面都有良好的表现。

（3）支持大量多媒体数据

Oracle 支持大量多媒体数据如二进制图形、声音、动画及多维数据结构等。

（4）提供了与第三代高级语言的接口软件系列

接口软件系列能在 C、C++ 等主流语言中嵌入 SQL 语句及过程化（PL/SQL）语句，完成对数据库的数据操纵。加上它有许多优秀的前台开发工具，如 POWER BUILD、SQL*FORMS、VISIA BASIC 等，可以快速开发生成基于客户端 PC 平台的应用程序，并具有良好的移植性。

（5）提供了新的分布式数据库能力

可通过网络较方便地读/写远端数据库中的数据，并有对称复制的技术。

3. 主要的Oracle软件

（1）Oracle电子商务套件（Oracle E-Business Suite）

Oracle 电子商务套件是行业中第一个集成的基于互联网的商务应用套件，它将前台与后台运营中的关键业务流程自动化。Oracle 电子商务套件能够使用户在实施业务应用时拥有很高的可选择性与灵活性，它的开放式基础架构与单一数据模型能够使用户在部署套件中的应用软件时拥有多种选择，既可以单独使用，也可以组成业务流。Oracle 电子商务套件既可以用 CD-ROM 这种传统软件形式作为载体，也可以作为一种在线服务形式来提供给用户。

（2）Oracle10g

Oracle10g 是业界第一个完整的、智能化的新一代 Internet 基础架构，Oracle10g 电子商务平台实际上是指 Oracle 数据库 10g、Oracle 应用服务器 10g 和 Oracle 开发工具套件 10g 的完整集成。

Oracle 数据库 10g 是第一套具有无限可伸缩性与高可用性，并可在集群环境中运行商业软件的互联网数据库，具有 400 多个领先的数据库功能，在集群技术、高可用性、商业智能、

安全性、系统管理等方面都实现了新的突破。Oracle 应用服务器 10g 是 J2EE 认证的、最轻、最快、最具伸缩性的应用服务器,可以提供企业门户软件、无线支持、高速缓存、轻量级 J2EE 引擎、商务智能、快速应用开发、应用与业务集成、Web 服务等多种应用开发功能,形成完整的电子商务应用开发和部署环境。

(3) Oracle 11g

2007 年推出数据库 Oracle 11g 是 Oracle 数据库的最新版本。其具有以下基本特性:

① 与无压缩格式下存储数据相比,新的 Oracle 数据压缩技术能够确保以较小的开销节省 1/3 以上的磁盘存储空间。

② 自动诊断知识库 (Automatic Diagnostic Repository, ADR) 是专门针对严重错误的知识库。该知识库基本上能够自动完成一些以往需要由数据库管理员手动完成的操作。

③ 提供了 SPA (SQL 性能分析器)。SPA 是一个整体调整工具,管理员可以通过该工具在数据库上定义和重演 (replay) 一个典型的工作负载,之后管理员可以调节整体参数使数据库尽快达到最佳性能。由于获得了最优的初始参数,数据库管理员仅仅需要给定一个典型的负载,SPA 就会根据历史记录来决定 SQL 的最终设置。

④ 提供 AMM (自动内存管理) 工具。AMM 工具其实就是一种探测机制。实际上,Oracle 11g 有很多随机访问存储池,当存储管理模式探测到某个存储池中已满时,它将整个随机存储器从一个区域分配到其他相对合适的区域。

1.3.3 Informix

Informix 是 IBM 公司推出的关系数据库管理系统 (RDBMS)。作为一个集成解决方案,它被定位为 IBM 在线事务处理 (OLTP) 旗舰级数据服务系统。Informix 是一种大型的数据库管理系统,具有先进的技术、性能与可靠性,广泛的应用于政府、金融保险、邮政电信、制造及零售等重要行业或领域。

IBM 对 Informix 和 DB2 都有长远的规划,两个数据库产品互相吸取对方的技术优势。在 2005 年早些时候,IBM 推出了 Informix Dynamic Server (IDS) 第 10 版。目前最新版本的是 IDS11 (v11.5, 代码名为 Cheetah 2)。

Cheetah 2 可支持客户运用 IBM 大型机系统提供的多种信息管理技巧,增强集群服务器环境的业务表现。因此 IDS 是业界第一款非大型机级数据服务器,无论地理位置远近或与备份数据中心站点间距离长短,它都能为集群数据中心提供低成本的持续数据可用性和灾难恢复能力。

IDS 11.5 在原版基础上进行了多处改良,其领先的稳定性和交易性能得到了进一步的提升,可更好地支持用户减少所需的服务器的数量和成本。它允许客户以更少的硬件服务器管理相同数量的数据,因此大大降低了客户对软件许可、管理成本、能源和空间的需求。

基于以上特性,IDS 也成为了众多细分市场上领先的集成数据服务器,这些市场包括零售、电信、政府/公共领域、旅游和娱乐等。全球有 95% 的电信公司均采用 IDS 支持本企业的数据管理。

1.3.4 Sybase

1. Sybase 简介

Sybase 是美国 Sybase 公司研制的一种关系型数据库系统,是一种典型的基于 UNIX 或 Windows NT 平台上客户—服务器环境下的大型数据库系统。Sybase 提供了一套应用程序编程

接口和库，可以与非 Sybase 数据源及服务器集成，允许在多个数据库之间复制数据，适于创建多层应用。

系统具有完备的触发器、存储过程、规则以及完整性定义，支持优化查询，具有较好的数据安全性。Sybase 通常与 SybaseSQLAnywhere 用于客户—服务器环境，前者作为服务器数据库，后者为客户机数据库。该公司研制的开发工具 PowerBuilder 在我国大中型系统中具有广泛的应用。Sybase 主要有三种版本，一是 UNIX 操作系统下运行的版本，二是 Novell Netware 环境下运行的版本，三是 Windows NT 环境下运行的版本。在 UNIX 操作系统下，目前广泛应用的为 Sybase 10 及 Sybase 11 for SCO UNIX。

2. Sybase的主要特点

Sybase 数据库具有以下基本特点。

(1) 基于客户—服务器体系结构

一般的关系数据库都是基于主/从式模型的。在主/从式结构中，所有的应用都运行在一台机器上。用户只是通过终端发命令或简单地查看应用运行的结果。而在客户—服务器结构中，应用被分在了多台机器上运行。一台机器是另一个系统的客户，或是另外一些机器的服务器。这些机器通过局域网或广域网连接起来。

(2) 是真正开放的数据库

由于采用了客户—服务器结构，应用被分在多台机器上运行。更进一步，运行在客户端的应用不必是 Sybase 公司的产品。对于一般的关系数据库，为了让其他语言编写的应用能够访问数据库提供了预编译。Sybase 数据库，不只是简单地提供了预编译，而且公开了应用程序接口 DB-LIB，鼓励第三方编写 DB-LIB 接口。由于开放的客户 DB-LIB 允许在不同的平台使用完全相同的调用，因而使得访问 DB-LIB 的应用程序很容易从一个平台向另一个平台移植。

3. Sybase数据库的组成

Sybase 数据库主要由三部分组成。

(1) 联机的关系数据库管理系统 Sybase SQL Server

Sybase SQL Server 完成数据库管理和维护的软件，起着数据管理、高速缓冲管理、事务管理的作用。

(2) 一组前端工具 Sybase SQL Toolset

Sybase SQL Toolset 支持数据库应用系统的建立与开发。ISQL 是与 SQL Server 进行交互的一种 SQL 句法分析器。ISQL 接收用户发出的 SQL 语言，将其发送给 SQL Server，并将结果以形式化的方式显示在用户的标准输出上。

DWB 是数据工作台，是 Sybase SQL Toolset 的一个主要组成部分，它的作用在于使用户能够设置和管理 SQL Server 上的数据库，并且为用户提供一种对数据库的信息执行添加、更新和检索等操作的简便方法。在 DWB 中能完成 ISQL 的所有功能，而且，DWB 是基于窗口和菜单的，因此操作比 ISQL 简单，是一种方便实用的数据库管理工具。

APT 是 Sybase 客户软件部分的主要产品之一，也是从事实际应用开发的主要环境。APT 工作台是用于建立应用程序的工具集，可以创建从非常简单到非常复杂的应用程序，它主要用于开发基于表格 (Form) 的应用。其用户界面采用窗口和菜单驱动方式，通过一系列的选择完成表格、菜单和处理的开发。

(3) Sybase Open Client/Open Server接口

Sybase Open Client/Open Server 接口可把异构环境下其他厂商的应用软件 and 任何类型的数据连接在一起。通过 Open Client 的 DB-LIB 库, 应用程序可以访问 SQL Server。而通过 Open Server 的 SERVER-LIB, 应用程序可以访问其他数据库管理系统。

1.3.5 SQL Server

1. SQL Server 简介

SQL Server 是由 Microsoft、Sybase 和 Ashton-Tate 三家公司共同开发的一个关系数据库管理系统, 于 1988 年推出了第一个 OS/2 版本。在 Windows NT 推出后, Microsoft 公司将 SQL Server 移植到 Windows NT 操作系统上, 专注于开发推广 SQL Server 的 Windows NT 版本。Sybase 公司则较专注于 SQL Server 在 UNIX 操作系统上的应用。

2. SQL Server 2008

SQL Server 2008 是一个重要的产品版本, 它增加了许多新的特性和关键的改进, 使得它成为迄今为止最强大最全面的 SQL Server 版本。

SQL Server 2008 满足数据爆炸和下一代数据驱动应用程序的需求, 提供了满足需求的解决方案, 可以使用存储和管理多种数据类型, 包括 XML、E-mail、时间/日历、文档等, 同时提供一个丰富的服务集合来与数据交互, 包括搜索、查询、数据分析、报表、数据整合和同步功能等。

1.3.6 PostgreSQL

1. PostgreSQL简介

PostgreSQL 是以美国加州大学伯克利分校计算机系开发的 Postgres4.2 为基础的对象关系型数据库管理系统 (ORDBMS)。PostgreSQL 是目前世界上最先进, 功能最强大的自由数据库管理系统。它是目前支持平台最多的数据库管理系统之一, 所支持的平台多达十几种, 包括不同的系统, 不同的硬件体系。

PostgreSQL 支持大部分 SQL 标准并且提供了许多其他现代特性, 如复杂查询、外键、触发器、视图、事务完整性、多版本并发控制等。同样, PostgreSQL 可以通过增加新的数据类型、函数、操作符、聚集函数、索引方法、过程语言实现扩展。并且, 任何人都可以以任何目的免费使用、修改和分发 PostgreSQL。

2. PostgreSQL的主要特点

PostgreSQL 可以说是最富特色的自由数据库管理系统, 甚至我们也可以说是最强大的自由软件数据库管理系统。PostgreSQL 的特性覆盖了 SQL-2/SQL-92 和 SQL-3/SQL-99, 其主要特点如下:

- ① 它提供了目前世界上最丰富的数据类型的支持, 比如 IP 类型和几何类型等。
- ② 很长时间以来, PostgreSQL 是唯一支持事务、子查询、多版本并行控制系统、数据完整性检查等特性的一种自由软件数据库管理系统。直到最近 Inprise 公司将 InterBase, 以及 SAP 等厂商将各自的专有软件开放为自由软件之后才打破了这个状态。
- ③ PostgreSQL 拥有一支非常活跃的开发队伍, 而且在许多黑客的努力下, PostgreSQL 的质量日益提高。

④ 从技术角度来讲, PostgreSQL 采用经典的 C/S 结构, 支持丰富的客户端接口, 几乎涵盖所有类型的数据库客户端接口。

1.3.7 MySQL

1. MySQL简介

MySQL 是瑞典 MySQLAB 公司开发的一个小型关系型数据库管理系统。由于其体积小、速度快、成本低, 尤其是开放源码这一特点, 而被广泛地应用在 Internet 上的中小型网站中。与其他的大型数据库, 如 Oracle、DB2、SQL Server 等相比, MySQL 规模小、功能有限(MySQL Cluster 的功能和效率都相对较差)。但对于一般的个人使用者和中小型企业来说, MySQL 提供的功能已经绰绰有余, 而且由于 MySQL 是开放源码软件, 因此可以大大降低成本。

目前, Internet 上流行的网站构架方式是 LAMP (Linux+Apache+MySQL+PHP), 即使用 Linux 作为操作系统, Apache 作为 Web 服务器, MySQL 作为数据库, PHP 作为服务器端脚本解释器。由于这四个软件都是自由或开放源码软件, 所以使用 LAMP 方式可以用很低成本建立起一个稳定、免费的网站系统。

2. MySQL的主要特点

MySQL 具有以下基本特点:

- ① 使用C和C++编写, 并使用了多种编译器进行测试, 保证源代码的可移植性;
- ② 支持AIX、FreeBSD、HP-UX、Linux、Mac OS、Novell Netware、OpenBSD、OS/2 Wrap、Solaris、Windows等多种操作系统;
- ③ 为多种编程语言提供了API, 其中包括 C、C++、Python、Java、Perl、PHP、Eiffel、Ruby和Tcl等;
- ④ 支持多线程, 充分利用CPU资源;
- ⑤ 优化的 SQL查询算法, 有效地提高查询速度;
- ⑥ 既可作为一个单独的应用程序用在客户端服务器网络环境中, 也可作为一个库嵌入到其他的软件中提供多语言支持;
- ⑦ 提供 TCP/IP、ODBC 和 JDBC 等多种数据库连接途径;
- ⑧ 提供用于管理、检查、优化数据库操作的管理工具;
- ⑨ 可以处理拥有上千万条记录的大型数据库。

1.3.8 Access数据库

1. Access简介

Access 是微软公司推出的基于 Windows 的桌面关系数据库管理系统 (RDBMS, 即 Relational Database Management System), 是 Office 系列应用软件之一。它提供表、查询、窗体、报表、页、宏、模块 7 种用来建立数据库系统的对象; 提供多种向导、生成器、模板, 将数据存储、数据查询、界面设计、报表生成等操作规范化; 为建立功能完善的数据库管理系统提供了方便, 也使得普通用户不必编写代码, 就可以完成大部分数据管理的任务。

2. Access的优、缺点

(1) Access的优点

① 存储方式单一

Access管理的对象有表、查询、窗体、报表、页、宏和模块,以上对象都存放在后缀为.mdb的数据库文件中,便于用户的操作和管理。

② 面向对象

Access 是一个面向对象的开发工具,利用面向对象的方式将数据库系统中的各种功能对象化,将数据库管理的各种功能封装在各类对象中。通过对象的方法、属性完成数据库的操作和管理,极大地简化了用户的开发工作。

③ 界面友好、易操作

Access 是一个可视化工具,其风格与 Windows 完全一样,直观方便。系统还提供了表生成器、查询生成器、报表设计器以及数据库向导、表向导、查询向导、窗体向导、报表向导等工具,使得操作简便,容易使用和掌握。

④ 集成环境、处理多种数据信息

Access 基于 Windows 操作系统下的集成开发环境,该环境集成了各种向导和生成器工具,极大地提高了开发人员的工作效率,使建立数据库、创建表、设计用户界面、设计数据查询、报表打印等可以方便有序地进行。

⑤ 支持 ODBC

Access 支持 ODBC (开发数据库互连, Open Data Base Connectivity),利用 Access 强大的 DDE (动态数据交换) 和 OLE (对象的连接和嵌入) 特性,可以在一个数据表中嵌入位图、声音、Excel 表格、Word 文档,还可以建立动态的数据库报表和窗体等。

(2) Access的缺点

Access 是小型数据库,有它根本的局限性,在以下几种情况下数据库性能会急剧下降:

- ① 数据库过大。一般地, Access数据库大小达到 50 MB左右的时候,性能会急剧下降;
- ② 网站访问频繁。经常达到 100 人左右的在线人数时,性能会急剧下降;
- ③ 记录数过多。一般记录数达到 10 万条左右的时候,性能会急剧下降。

1.3.9 FoxPro数据库

Visual FoxPro (简称 VFP) 是 Microsoft 公司推出的数据库开发软件,用它来开发数据库,既简单又方便。Visual FoxPro 源于美国 Fox Software 公司推出的数据库产品 FoxBase, FoxPro 是 FoxBase 的加强版,最高版本曾出过 FoxPro 2.6。1992 年, Fox Software 公司被 Microsoft 收购,加以发展,使其可以在 Windows 上运行,并且更名为 Visual FoxPro。

Visual FoxPro 6.0 由微软公司于 1998 年在推出 Windows 98 操作系统时推出。它不仅提供了更多更好的设计器、向导、生成器及新类,并且使得客户—服务器结构数据库应用程序的设计更加方便简捷。Visual FoxPro 目前最新的版本是 9.0。2007 年 3 月 13 日,微软公司向第三方开发者正式确认, Visual FoxPro 9.0 将是微软的最后一款桌面数据库开发工具软件。

1.4 客观世界的数据表示

数据从现实世界到计算机数据库的抽象表示经历了三个阶段,即现实世界、概念世界、数据世界,如图 1.7 所示。

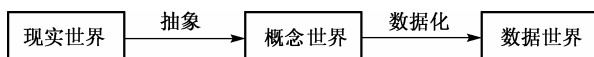


图 1.7 数据抽象三阶段

1.4.1 现实世界

现实世界里的客观事物是我们进行管理的对象，这些对象之间既有区别，也有联系。这种区别和联系取决于事物本身的特性。

1.4.2 概念世界

概念世界是现实世界在人脑中的反映，是对客观事物及其联系的抽象，概念世界也称为信息世界。

1. 概念世界的基本概念

(1) 实体、实体集和实体型

客观存在并可相互区别的事物称为实体。实体可以是具体的人、事、物，也可以是抽象的概念或联系。例如，一个学生、一门课、一个供应商、一个部门、一本书、一位读者等都是实体。

同型实体的集合称为实体集。例如，全体学生就是一个实体集。图书馆的图书也是一个实体集。

具有相同属性的实体必然具有共同的特征和性质。用实体名及其属性名集合来抽象和描述同类实体，称为实体型。例如，学生（学号、姓名、性别、出生年月、系别、入学时间）就是一个实体型。图书（编号、书名、出版社、出版日期、定价）也是一个实体型。

(2) 属性、域和主码

实体所具有的某一特性称为属性。一个实体可以由若干属性来描述。例如，图书实体可以由编号、书名、出版社、出版日期、定价等属性组成。又如，学生实体可以由学号、姓名、性别、出生年月、系别、入学时间等属性组成。如（2010119120、王丽、女、1992-12-28、计算机系、2010），这些属性组合起来体现了一个学生的特征。

属性的取值范围称为该属性的域。例如，职工性别域为（男、女），姓名域为字母字符串集合，年龄域为小于 150 的整数，职工号域为 5 位数字组成的字符串等。

唯一标识实体的属性集称为主码。例如，学生学号是学生实体的主码，职工号是职工实体的主码。

(3) 联系

在现实世界中，事物内部以及事物之间是有联系的，这些联系在信息世界中反映为实体内部的联系和实体之间的联系。实体内部的联系通常是组成实体的各属性之间的联系，两个实体型之间的联系可以分为 3 类。

① 一对一联系（1:1）

如果对于实体集 A 中的每一个实体，实体集 B 至多有一个实体与之联系，反之亦然，则称实体集 A 与实体集 B 具有一对一联系，记为 1:1。

例如，某宾馆只有单人间，每个客房都对应着一个房间号，一个房间号也唯一地对应着一间客房。所以，客房和房间号之间具有一对一联系。再例如，乘客和座位之间存在一对一联系，意味着一个乘客只能坐一个座位，而一个座位只能被一个乘客占有，如图 1.8 所示。

② 一对多联系 ($1:n$)

如果对于实体集 A 中的每一个实体, 实体集 B 中有 n 个实体与之联系 ($n \geq 0$), 反之, 对于实体集 B 中的每一个实体, 实体集 A 中至多有一个实体与之联系, 则称实体集 A 与实体集 B 具有一对多联系, 记为 $1:n$ 。

例如, 一个部门中有若干名职工, 而每个职工只能在一个部门工作, 则部门与职工之间具有一对多联系, 如图 1.9 所示。

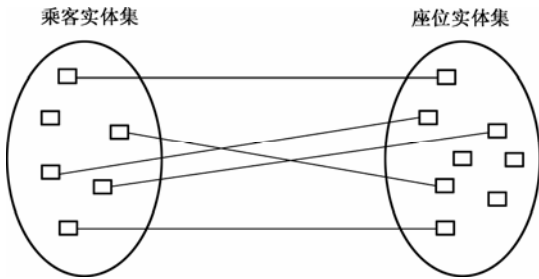


图 1.8 一对一联系

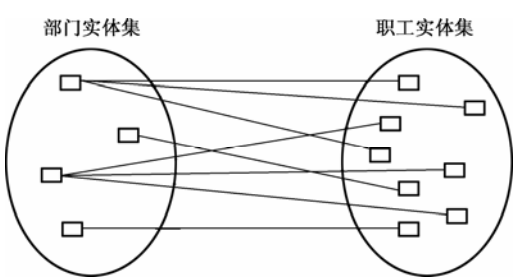


图 1.9 一对多联系

③ 多对多联系 ($m:n$)

如果对于实体集 A 中的每一个实体, 实体集 B 中有 n 个实体与之联系 ($n \geq 0$), 反之, 对于实体集 B 中的每一个实体, 实体集 A 中也有 m 个实体与之联系 ($m \geq 0$), 则称实体集 A 与实体集 B 具有多对多联系, 记为 $m:n$ 。

在选课系统中, 一门课程同时有若干个学生选修, 而一个学生可以同时选修多门课程, 则课程与学生之间具有多对多联系, 如图 1.10 所示。

实际上, 一对一联系是一对多联系的特例, 而一对多联系又是多对多联系的特例。实体集之间的这种一对一、一对多、多对多联系不仅存在于两个实体集之间, 也存在于两个以上的实体集之间。

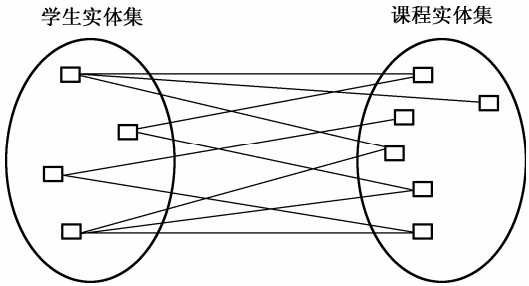


图 1.10 多对多联系

在授课系统中, 对于课程、教师与参考书三个实体型, 如果一门课程可以有若干个教师讲授, 使用若干本参考书, 而每一个教师只讲授一门课程, 每一本参考书只供一门课程使用, 则课程与教师、课程与参考书之间的联系是一对多的。

同一个实体集内的各实体之间也可以存在一对一、一对多、多对多的联系。职工实体集内部有领导与被领导的联系。即某职工为部门领导, 领导若干职工, 而一名职工仅被另外一个职工 (领导) 直接领导, 因此这是一对多联系。

2. 概念世界的表示

概念模型用于信息世界的建模, 与具体的 DBMS 无关。为了把现实世界中的具体事物抽象、组织为某一 DBMS 支持的数据模型, 人们常常首先将现实世界抽象为概念世界, 然后再将概念世界转换为机器世界。实际上, 概念模型是现实世界到机器世界的一个中间层次。概念模型从用户的观点出发, 将管理对象的客观事物及他们之间的联系, 用容易为人所理解的语言或形式表述出来。概念模型应该能够准确、方便地表示概念世界, E-R 图 (实体联系图) 是描述概念模型的主要工具。如图 1.11 所示描述了学生实体集和课程实体集的 E-R 图。

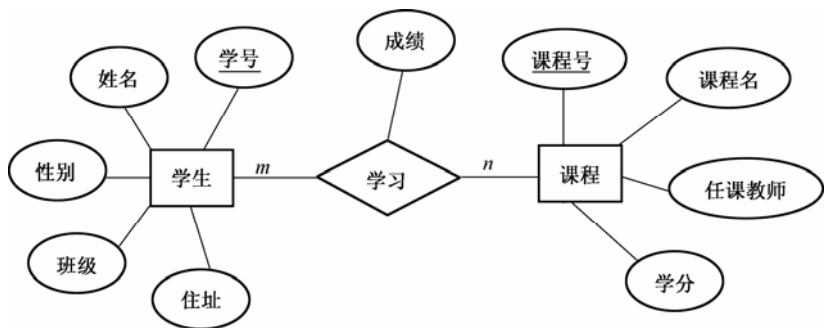


图 1.11 学生和课程的 E-R 图

1.4.3 数据世界

数据世界是信息世界进一步数据化的结果，数据世界主要有以下基本术语。

- ① 数据项。数据项又称字段，是数据库数据中的最小逻辑单位，用来描述实体的属性。
- ② 记录。记录是数据项的集合，一个记录由若干个数据项组成，用来描述实体。
- ③ 文件。文件是一个具有文件名的一组同类记录的集合，用来描述实体集。

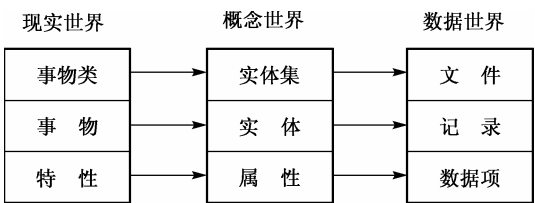


图 1.12 三种世界的概念对应关系

三种世界的概念对应关系如图 1.12 所示。

1.5 数据模型

1.5.1 数据模型的概念

模型是现实世界特征的模拟和抽象。在数据库技术中，数据模型用于表示实体类型及实体类型间联系。数据模型是数据库的基础，是数据库的形式框架，用于描述一组数据的概念和定义，数据模型包含三个层面。

- 1. 概念数据模型
概念数据模型是面向数据库用户现实世界的数据库模型，主要用来描述世界的概念化结构，它使数据库的设计人员在设计的初始阶段，摆脱计算机系统及 DBMS 的具体技术问题，集中精力分析数据以及数据之间的联系。概念数据模型与具体的 DBMS 无关，必须换成逻辑数据模型，才能在 DBMS 中实现。
- 2. 逻辑数据模型
逻辑数据模型是用户从数据库所看到的数据模型，是具体的 DBMS 所支持的数据模型，如网状数据模型、层次数据模型、关系数据模型等。此模型既要面向用户，又要面向系统。
- 3. 物理数据模型
物理数据模型是描述数据在存储介质上组织结构的数据模型，它不但与具体的 DBMS 有关，而且还与操作系统和硬件有关。每一种逻辑数据模型在实现时都有对应的物理数据模型。DBMS 为了保证其独立性与可移植性，大部分物理数据模型的实现工作由系统自动完成，而设计者只设计索引、聚集等特殊结构。

1.5.2 数据模型组成要素

一般地讲,任何一种数据模型都是严格定义的概念集合。这些概念必须能够精确地描述系统的静态特性、动态特性和完整性约束条件。因此,数据模型通常都是由数据结构、数据操作和完整性约束三个要素组成。

1. 数据结构

数据结构研究数据之间的组织形式(数据的逻辑结构)、数据的存储形式(数据的物理结构)以及数据对象的类型等。存储在数据库中的对象类型集合是数据库的组成部分。数据结构用于描述系统的静态特性。在数据库系统中,通常按照其数据结构的类型来命名数据模型。例如层次结构、网状结构、关系结构的数据模型分别命名为层次模型、网状模型和关系模型。

2. 数据操作

数据操作是指对数据库中的各种对象(型)的实例(值)允许执行的操作集合,包括操作及有关操作规则,用于描述系统的动态特性。数据库主要有查询和更新(包括插入、删除、修改)两大类操作。数据模型必须定义这些操作的确切含义、操作符号、操作规则(如优先级),以及实现操作的语言。

3. 数据完整性约束

数据完整性约束是一组完整性规则的集合。完整性规则用于指明给定的数据模型中数据及其联系所具有的制约和储存规则,以保证数据的正确性、有效性和相容性。数据模型应该反映和规定本数据模型应遵守的基本性、通用性和完整性约束。此外,数据模型还应该提供定义完整性约束的机制,以反映所涉及的数据必须遵守的特定语义约束。例如,学生的“年龄”只能取值大于零的值;人员信息中的“性别”只能为“男”或“女”;学生选课信息中的“课程号”的值必须取自学校已经开设课程的课程号等。

1.5.3 逻辑模型的基本类型

数据模型是数据库系统的核心和基础,各种DBMS 软件都是基于某种数据模型的。所以通常也按照数据模型的特点将传统数据库系统分成层次数据库、网状数据库和关系数据库三类。其中,层次模型和网状模型统称为非关系模型,非关系模型的数据库系统在20世纪70年代非常流行,到了20世纪80年代,关系模型的数据库系统以其独特的优点逐渐占据了主导地位,成为数据库系统的主流。

1. 层次模型

层次数据库系统采用层次模型作为数据的组织方式。用树型(层次)结构表示实体类型以及实体间的联系是层次模型的主要特征。层次结构是一棵树。树的结点是记录类型,根结点只有一个,根结点以外的结点只有一个双亲结点,每一个结点可以有多个孩子结点。层次模型如图1.13所示。

层次模型的另一个最基本的特点是,任何一个给定的记录值(也称为实体)只有按照其路径查看时,才能显出它的全部意义,而且所有子记录值均不能够脱离双亲记录值而独立存在。

层次数据库系统的典型代表是IBM公司的推出的IMS(Information Management Systems)信息库管理系统,这是1968年IBM公司推出的第一个大型的商用数据库管理系统,曾在20

世纪 70 年代商业上广泛应用。目前，仍有某些用户在使用。

2. 网状模型

用网状结构表示实体类型及实体之间联系的数据模型称为网状模型。在网状模型中，一个子结点可以有多个父结点，在两个结点之间可以有一种或多种联系。记录之间联系是通过指针实现的，因此数据的联系十分密切。网状模型的数据结构在物理上易于实现，效率较高，但是编写应用程序较复杂，程序员必须熟悉数据库的逻辑结构。网状模型（Network Model）一般使用有向图结构表示实体类型及实体间联系。

网状模型的特点是记录之间联系通过指针实现， $M:N$ 联系也容易实现（一个 $M:N$ 联系可拆成两个 $1:N$ 联系），查询效率较高，网状模型如图 1.14 所示。

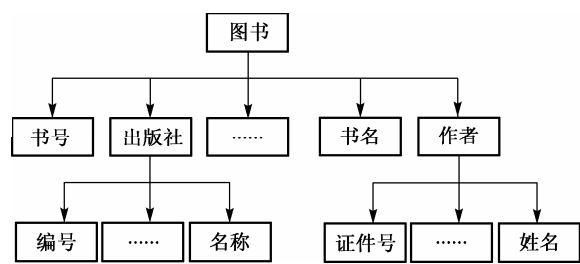


图 1.13 层次模型示意图

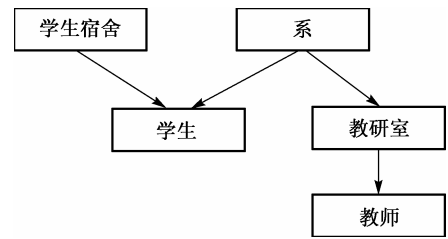


图 1.14 网状模型示意图

网状模型允许结点有多个双亲结点，并且允许两个结点之间有多种联系（复合联系）。网状模型可以更直接地去描述现实世界，层次模型实际上是网状模型的一个特例。网状数据库模型对于层次和非层次结构的事物都能比较自然的模拟，但网状模型的缺点是数据结构复杂且编程复杂。

3. 关系模型

网状数据库和层次数据库已经很好地解决了数据的集中和共享问题，但是在数据独立性和抽象级别上仍有很大欠缺。用户在对这两种数据库进行存取时，仍然需要明确数据的存储结构，指出存取路径。而后来出现的关系数据库较好地解决了这些问题，关系模型是目前最常用的一种数据模型。关系模型比较简单，容易为初学者接受。关系在用户看来是一个表格，记录是表中的行，属性是表中的列。关系模型是数学化的模型，可把表格看成一个集合，因此集合论、数理逻辑等知识可引入到关系模型中来。

关系模型是由若干个关系模式组成的集合。关系模式相当于前面提到的记录类型，它的实例称为关系，每个关系实际上是一张二维表格。

图 1.11 对应的关系模式如表 1.1 所示，表 1.1 中各关系模式对应的关系如表 1.2、表 1.3、表 1.4 所示。

表 1.1 关系模式

模式名	构成
学生	(<u>学号</u> 、姓名、性别、班级、住址)
成绩	(<u>学号</u> 、课程号、成绩)
课程	(<u>课程号</u> 、课程名、任课教师、学分)

表 1.3 课程关系

课程号	课程名	任课教师	学分
1-01	大学英语	马丽	1

表 1.2 学生关系

学号	姓名	性别	班级	住址
2010001119	张瑜婷	女	新闻一班	2-503
2010001120	马云飞	男	新闻一班	7-306
2009001121	周婷婷	女	新闻一班	2-503
.....

表 1.4 成绩关系

学号	课程号	成绩
2010001119	1-01	85

与其他数据模型相比，关系模型突出的优点如下：

- ① 关系模型提供单一的数据结构形式，具有高度的简明性和精确性；
- ② 关系模型的逻辑结构和相应的操作完全独立于数据存储方式，具有高度的数据独立性；
- ③ 关系模型使数据库的研究建立在比较坚实的数学基础上；
- ④ 关系数据库语言与一阶谓词逻辑的固有内在联系，为以关系数据库为基础的推理系统和知识库系统的研究提供了方便。

1.6 数据库的体系结构

数据库系统为了保证数据的逻辑独立性和物理独立性，在体系上采用三级模式和两种映射结构。数据库系统从内到外分为三个层次描述，分别称为内模式、模式和外模式。两种映射是：外模式/模式映射和模式/内模式映射。

1.6.1 三级模式

数据库系统的三级模式如图1.15所示。

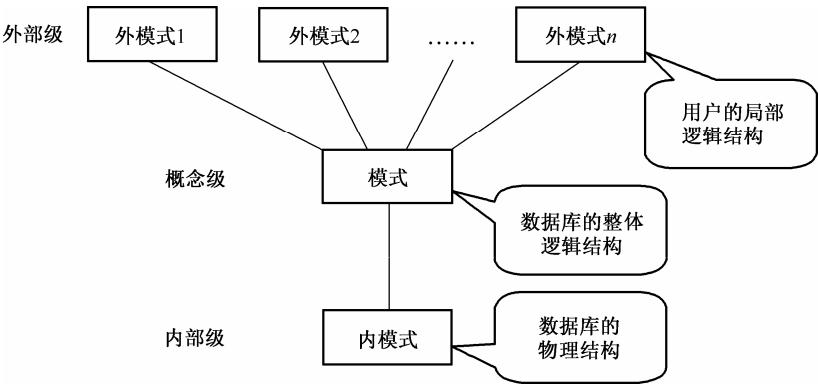


图 1.15 数据库的三级模式

1. 模式

模式也称逻辑模式，是数据中全体数据的逻辑结构和特征描述，是所有用户的公共数据视图。它是数据库系统模式结构的中间层，既不涉及数据的物理存储细节和硬件环境，也与具体的应用程序及其所使用的开发工具无关。

一个数据库只有一个模式。数据库模式以某一种数据模型为基础，统一综合地考虑了所有用户的需求，并将这些需求有机地结合成一个逻辑整体。DBMS 提供描述语言（模式 DDL）

来严格定义模式，定义模式时不仅要定义数据的逻辑结构（包括数据记录由哪些数据项构成，数据项的名字、类型、取值范围等），而且要定义数据之间的联系，定义与数据有关的安全性、完整性要求。

2. 外模式

外模式也称子模式或用户模式，它是数据库用户（包括应用程序员和最终用户）能够看到和使用的局部数据的逻辑结构和特征的描述，是数据库用户的数据视图，是与某一应用有关数据的逻辑表示。外模式通常是模式的子集，一个数据库可以有多个外模式。由于它是各个用户的数据视图，如果不同的用户在应用需求、看待数据的方式、对数据保密的要求等存在差异，则其外模式描述就是不同的。即使对模式中同一数据记录，在外模式中的结构、类型、长度、保密级别等都可以不同。另一方面，同一个外模式也可为某一用户的多个应用系统所用，但一个应用程序只能使用一个外模式。

外模式是保证数据库安全性的一个重要措施。每个用户只能看见和访问所对应的外模式中的数据，数据库中其余数据是不可见的。DBMS 提供子模式描述语言（子模式 DDL）来严格定义子模式。

3. 内模式

内模式也称存储模式，一个数据库只有一个内模式。它是数据物理结构和存储方式的描述，是数据在数据库内部的表示方式。

例如，记录的存储方式是顺序存储，按 B 树结构存储还是按 Hash 方法存储；索引按照什么方式组织；数据是否压缩存储，是否加密存储记录等。DBMS 提供内模式描述语言（内模式 DDL，或者存储模式 DDL）来严格定义内模式。

1.6.2 两种映射

三级模式间存在两种映射：一种是外模式/模式映射，这种映射把用户数据库与概念数据库联系起来；另一种映射是模式/内模式映射，这种映射把概念数据库与物理数据库联系起来。数据库的三级模式和两种映射保证了数据库的数据独立性，数据独立性包括逻辑数据独立性和物理数据独立性。

1. 外模式/模式映射

外模式/模式映射实现了概念模式到内模式之间的相互转换。用户应用程序根据外模式进行数据操作，通过外模式/模式映射，定义和建立某个外模式与模式间的对应关系，将外模式与模式联系起来，当模式发生改变时，只要改变其映射，就可以使外模式保持不变，对应的应用程序也可保持不变，保证了数据与程序的逻辑独立性。

2. 模式/内模式映射

模式/内模式映射实现了外模式到概念模式之间的相互转换。通过模式/内模式映射，定义建立数据的逻辑结构（模式）与存储结构（内模式）间的对应关系，当数据的存储结构发生变化时，只需改变模式/内模式映射，就能保持模式不变，因此应用程序也可以保持不变，保证了数据与程序的物理独立性。

1.7 访问数据库的主要过程

数据库操作的实现是一个复杂的过程，下面以某个应用程序从数据库中读取一条记录为例说明数据库管理系统的工作过程，图1.16描述了访问数据库的主要过程。

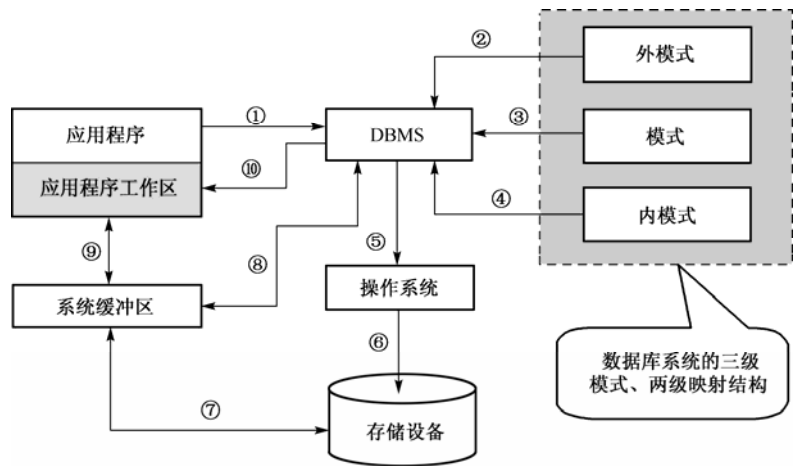


图 1.16 数据库操作的实现过程

一次数据的访问大约需要经过 10 个步骤来完成，不同的 DBMS，其具体实现过程可能会有微小的差别，但其进本原理是相同的，这 10 个步骤的基本功能如下：

- ① 用户发出读取数据的请求，读取时告诉 DBMS 所要读取记录的关键字和模式；
- ② DBMS 收到请求，分析请求的外模式；
- ③ DBMS 调用模式，分析请求，根据外模式/模式映射关系决定读入哪些模式的数据；
- ④ DBMS 根据模式/内模式映射关系将逻辑记录转换为物理记录；
- ⑤ DBMS 向操作系统发出读取数据的请求；
- ⑥ 操作系统启动文件管理功能，对实际的物理存储设备启动读操作；
- ⑦ 操作系统将读取的数据传送到系统缓冲区，同时通知 DBMS 读取成功；
- ⑧ DBMS 根据模式和外模式的结构对缓冲区中的数据进行格式转换，转换为应用程序所需要的格式；
- ⑨ DBMS 将转换后的数据传送到应用程序对应的程序工作区中；
- ⑩ DBMS 向应用程序发出读取成功的消息，应用程序在收到消息后，便可对收到的信息进行下一步的处理。

1.8 关系数据库

关系数据库是当前使用最为普遍的数据库，关系数据库采用关系模型。

1.8.1 基本概念

1. 关系

在关系模型中，基本数据结构被限制为二维表格。因此，数据在用户观点下的逻辑结构就是一张二维表，每一张二维表称为一个关系（relation）。

关系模型中最基本的概念是关系，表 1.5 给出的工资表就是一个关系。

表 1.5 工资表

编号	姓名	基本工资	工龄工资	扣除	实发工资
101	刘安	1520.00	1532.00	545.00	2507.00
102	陈林	1426.00	1524.00	530.50	2420.50
103	赵刚	1388.00	1525.00	540.50	2373.50
201	刘达	1388.00	1515.00	533.00	2370.00
202	陆海	1476.00	1512.00	522.50	2466.50
203	李万	1698.00	1527.00	560.00	2665.00

并非任何一个二维表都是一个关系，只有具备以下特征的二维表才是一个关系：

- ① 表中没有组合的列，也就是说每一列都是不可再分的；
- ② 表中每一列的所有数据都属于同一种类型；
- ③ 表中各列都指定了一个不同的名字；
- ④ 表中没有数据完全相同的行；
- ⑤ 表中行之间顺序位置的调换和列之间位置的调换不影响它们所表示的信息内容。

具有上述性质的二维表称为规范化的二维表。如不作特殊说明，本书中提到的二维表均是规范化的二维表。

2. 元组和属性

表中的行称为元组。一行为一个元组，对应存储文件中的一个记录值。表中的列称为属性，每一列有一个属性名。属性值相当于记录中的数据项或者字段值。

一般来说，属性值构成元组，元组构成关系。即一个关系描述现实世界的对象集，关系中的一个元组描述现实世界中的一个具体对象，它的属性值则描述了这个对象的属性。一个元组由该行全体属性值组成，元组的全体组成了一个关系。

3. 关键字（或码）

在一个关系中，能够唯一地标识一个元组的属性或属性组合称为候选关键字。有时，可能有多个候选关键字，从中选择一个作为主关键字。主关键字在关系中用来作为插入、删除、检索元组的区分标志。如果一个关系中的属性或属性组合并不是该关系的关键字，但它们是另外一个关系的关键字，则称其为该关系的外关键字。

1.8.2 关系数据库的体系结构

关系模型基本上遵循数据库的三级体系结构。在关系模型中，概念模式是关系模式的集合，外模式是关系子模式的集合，内模式是存储模式的集合。

1. 关系模式

关系模式是对关系的描述，它包括模式名、组成该关系的诸属性名、值域名和模式的主键。具体的关系称为实例。图1.17是一个教学模型的实体联系图。

实体类型 *S* 的属性 *SNO*、*SNAME*、*AGE*、*SEX*、*SDEPT* 分别表示学生的学号、姓名、年龄、性别和学生所在系；实体类型 *C* 的属性 *CNO*、*CNAME*、*CDEPT*、*TNAME* 分别表示课程号、课程名、课程所属系和任课教师。*S* 和 *C* 之间有 *M:N* 的联系（一个学生可选多门课程，一门课程可以被多个学生选修），联系类型 *SC* 的成绩属性用 *SCORE* 表示。

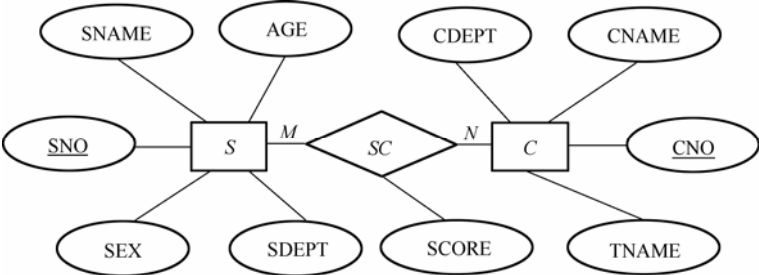


图 1.17 教学模型的实体联系图

转换成的关系模式如下：

关系模式 S (SNO, SNAME, AGE, SEX, SDEPT)

关系模式 C (CNO, CNAME, CDEPT, TNAME)

关系模式 SC (SNO, CNO, SCORE)

表 1.6、表 1.7、表 1.8 是对应的关系。

表 1.6 关系 S

SNO	SNAME	AGE	SEX	SDEPT
2010112001	卢雨轩	18	女	计算机
2010112009	江南	19	男	计算机
2010108093	韩晓云	18	女	新闻
2010102085	刘流	18	男	外语
2010102090	郑重	19	男	外语

表 1.7 关系 C

CNO	CNAME	CDEPT	TNAME
101	组成原理	计算机	张强
103	计算机网络	计算机	周明
201	新闻史	新闻	李莉
202	新闻写作	新闻	范梅
305	英语阅读	外语	杨丽华

表 1.8 关系 SC

SNO	CNO	SCORE
2010112001	101	78
2010112001	103	93
2010112009	101	82
2010112009	103	85
2010108093	201	78
2010108093	202	90
2010102085	305	83

关系模式是用数据定义语言（DDL）定义的。关系模式的定义包括：模式名、属性名、值域名及模式的主键。由于不涉及物理存储方面的描述，因此关系模式仅仅是对数据本身特征的描述。

2. 关系子模式

有时，用户使用的数据不直接来自关系模式中的数据，而是从若干关系模式中抽取满足一定条件的数据。这种结构可用关系子模式实现。关系子模式是用户所需数据的结构的描述，其中包含这些数据来自哪些模式和应满足哪些条件。如果需要用到成绩子模式 G (SNO, SNAME, CNO, SCORE)。可以看出，子模式 G 对应的数据来源于关系 S 和关系 SC ，构造时应满足它们的 SNO 值相等。子模式 G 的构造过程如图 1.18 所示。

子模式定义语言还可以定义用户对数据进行操作的权限，例如，是否允许读取、修改等。由于关系子模式来源于多个关系模式，因此是否允许对子模式的数据进行插入和修改就不一定了。

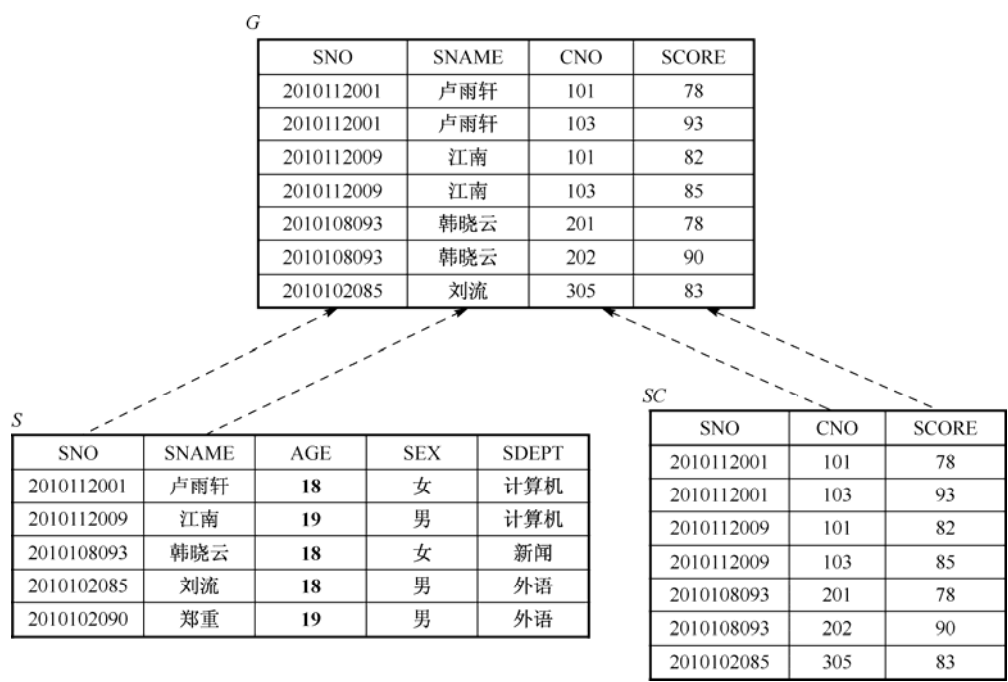


图 1.18 子模式 *G* 的定义

3. 存储模式

存储模式描述了关系是如何在物理存储设备上存储的，关系存储时的基本组织方式是文件。由于关系模式有关键码，因此存储一个关系可以用散列方法或索引方法实现，如果关系中元组数目较少（100 以内），也可以用堆文件的方式实现。此外，还可以对任意的属性集建立辅助索引。

1.8.3 关系模型的完整性规则

关系模型的完整性规则实现对数据的约束。关系模型提供了三类完整性规则：实体完整性规则、参照完整性规则、用户定义完整性规则。其中实体完整性规则和参照完整性规则是关系模型必须满足的完整性约束条件，称为关系完整性规则。

1. 实体完整性规则

实体完整性规则规定关系中元组的主键值不能为空。

如图 1.19 中给出学生表、课程表和成绩表中，学生表的主键是学号，课程表的主键是课程号，成绩表的主键学号和课程号的组合，这三个主键的值在表中应是唯一的且确定的。主键不能取空值（NULL），或是不确定的值，因为空值无法标识表中的一行。为了保证每一个实体有唯一的标识符，主键不能取空值。

2. 参照完整性规则

参照完整性规则的形式定义如下：

如果属性集 *K* 是关系模式 *R1* 的主键，*K* 也是关系模式 *R2* 的外键，那么在 *R2* 的关系中，*K* 的取值只允许两种可能，或者为空值，或者等于 *R1* 关系中某个主键值。这条规则的实质是“不允许引用不存在的实体”。

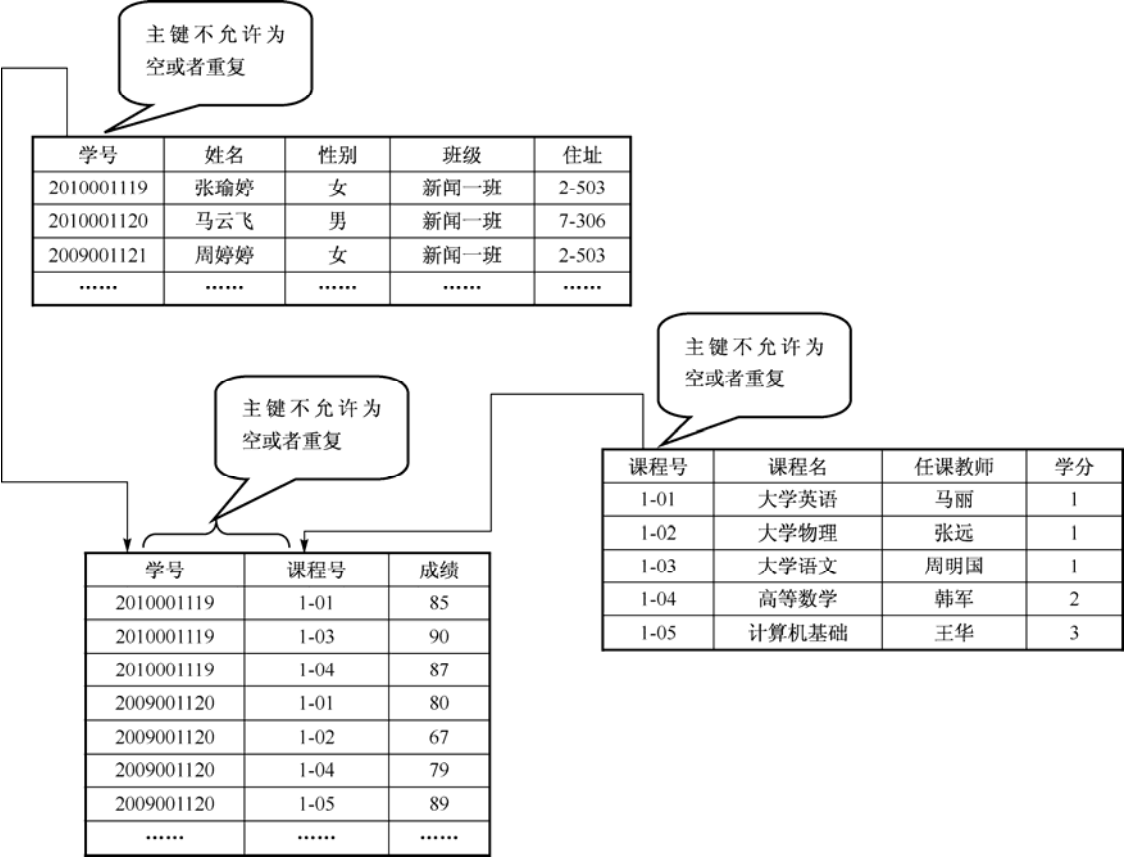


图 1.19 实体完整性和参照完整性

对于参照完整性规则，有三点需要注意。

- ① 外键和相应的主键可以不同名，但要定义在相同值域上。
- ② 当 $R1$ 和 $R2$ 是同一个关系模式时，表示同一个关系中不同元组之间的联系。例如，表示课程之间选修联系的模式 $R(CNO, CNAME, PCNO)$ ，其属性表示课程号、课程名、选修课程的课程号， R 的主键是 CNO ，而 $PCNO$ 就是一个外键，表示 $PCNO$ 值一定要在关系中存在（某个 CNO 值）。
- ③ 外键值是否允许空，应视具体问题而定，若外键是模式主键中的成分，则外键值不允许空，否则允许空。

在上述形式定义中，关系模式 $R1$ 的关系称为参照关系、主表或者父表；关系模式 $R2$ 的关系称为依赖关系、副表或者子表。

例如，图 1.19 中学生表和课程表为主表，成绩表为副表，学号是学生表的主键、成绩表的外键，课程号是课程表的主键、成绩表的外键。成绩表中的学号必须是学生表中学号的有效值，成绩表与学生表之间的联系是通过学号实现的。同样，成绩表中的课程号必须是课程表中课程号的有效值，成绩表与课程表之间的联系是通过课程号实现的。

实体完整性规则和参照完整性规则是关系模型必须满足的规则，由系统自动支持。

3. 用户定义的完整性规则

用户自定义完整性规则是针对某一具体数据的约束条件，由应用环境决定。它反映某一

具体应用所涉及的数据必须满足的语义要求。系统应提供定义和检验这类完整性的机制，以使用统一的系统方法处理它们。例如，学生成绩应该大于或等于零，职工的工龄应小于年龄，人的身高不能超过 3 m 等。

1.8.4 关系代数

关系代数是施加于关系上的一组集合代数运算，每个运算都以一个或多个关系作为运算对象，并生成另外一个关系作为运算的结果。关系代数包含两类运算，传统的集合运算和专门的关系运算。

1. 传统的集合运算

传统的集合运算主要包括并、差、交、笛卡儿积运算和除。

(1) 并

设关系 R 和 S 具有相同的模式， R 和 S 的并是由属于 R 或属于 S 的元组构成的集合，记为 $R \cup S$ 。形式定义如下：

$$R \cup S = \{t | t \in R \vee t \in S\}, \quad t \text{ 是元组变量, } R \text{ 和 } S \text{ 的元数相同}$$

(2) 差

设关系 R 和 S 具有相同的模式， R 和 S 的差是由属于 R 但不属于 S 的元组构成的集合，记为 $R - S$ 。形式定义如下：

$$R - S = \{t | t \in R \wedge t \notin S\}, \quad t \text{ 是元组变量, } R \text{ 和 } S \text{ 的元数相同}$$

(3) 交

关系 R 和 S 的交是由既属于 R 又属于 S 的元组构成的集合，记为 $R \cap S$ ，这里要求 R 和 S 定义在相同的模式上。形式定义如下：

$$R \cap S = \{t | t \in R \wedge t \in S\}, \quad t \text{ 是元组变量, } R \text{ 和 } S \text{ 的元数相同}$$

交操作不是一个独立的操作。

(4) 笛卡儿积

设关系 R 和关系 S 的元数分别为 r 和 s 。定义 R 和 S 的笛卡儿积 $R \times S$ 是一个 $(r+s)$ 元的元组集合，每个元组的前 r 个分量（属性值）来自 R 的一个元组，后 s 个分量是 S 的一个元组，记为 $R \times S$ 。形式定义如下：

$$R \times S = \{t | t = \langle t_r, t_s \rangle \wedge t_r \in R \wedge t_s \in S\}$$

若 R 有 n 个元组， S 有 m 个元组，则 $R \times S$ 有 $n \times m$ 个元组。

(5) 除

设关系 R 和 S 的元数分别为 r 和 s （设 $r > s > 0$ ），那么 $R \div S$ 是一个 $(r-s)$ 元的元组的集合。 $R \div S$ 是满足下列条件的最大关系： $R \div S$ 中每个元组 t 与 S 中每个元组 u 组成的新元组 $\langle t, u \rangle$ 必在关系 R 中。形式定义如下：

$$R \div S = \Pi_{1,2,L,r-s}(R) - \Pi_{1,2,L,r-s}((\Pi_{1,2,L,r-s}(R) \times S) - R)$$

例如，有 5 个关系 R 、 S 、 T 、 U 和 V ，如图 1.20 所示，它们的并、差、笛卡儿积和除的运算结果如图 1.21 所示。

R		S		T		U				V	
A	B	A	B	A	B	A	B	C	D	C	D
a	d	d	a	b	b	a	b	c	d	c	d
b	a	b	a	c	d	a	b	e	f	e	f
c	c	d	c			c	a	c	d		

图 1.20 基本关系

R ∪ S		R - S		R × T				U ÷ V	
A	B	A	B	A	B	C	D	A	B
a	d	a	d	a	d	b	b	a	b
b	a	c	c	a	d	c	d		
c	c			b	a	b	b		
d	a			b	a	c	d		
d	c			c	c	b	b		
				c	c	c	d		

图 1.21 运算结果

2. 专门的关系运算

专门的关系运算主要包括选择、投影和连接运算。

(1) 选择

从关系中找出满足给定条件的所有元组称为选择，选择条件是以逻辑表达式给出的，该逻辑表达式的值为真的元组被选取。这是从行的角度进行的运算，即水平方向抽取元组。经过选择运算得到的元组可以形成新的关系，其关系模式不变，但其中元组的数目小于或等于原来的关系中元组的个数，它是原关系的一个子集。

关系 R 关于公式 F 的选择操作作用 $\sigma_F(R)$ 表示，形式定义如下：

$$\sigma_F(R) = \{t \mid t \in R \wedge F(t) = \text{true}\}$$

式中， σ 为选择运算符， $\sigma_F(R)$ 表示从 R 中挑选满足公式 F 为真的所有元组所构成的关系。

例如， $\sigma_{2>3}(R)$ 表示从 R 中挑选第 2 个分量值大于 3 的元组所构成的关系。书写时，为了与属性序号区别起见，常量用引号括起来，而属性序号或属性名直接书写。

(2) 投影

从关系中挑选若干属性组成新的关系称为投影。这是从列的角度进行的运算，相当于对关系进行垂直分解。经过投影运算可以得到一个新关系，其关系所包含的属性个数往往比原关系少，或者属性的排列顺序不同。如果新关系中包含重复元组，则要删除重复元组。

设关系 R 是 k 元关系， R 在其分量 $A_{i_1}, A_{i_2}, \dots, A_{i_m}$ ($m \leq k, i_1, i_2, \dots, i_m$ 为 $1 \sim k$ 间的整数) 上的投影用 $\Pi_{i_1, i_2, \dots, i_m}(R)$ 表示，它是一个 m 元元组集合，形式定义如下：

$$\Pi_{i_1, i_2, \dots, i_m}(R) = \{t \mid t = \langle t_{i_1}, t_{i_2}, \dots, t_{i_m} \rangle \wedge \langle t_1, t_2, \dots, t_k \rangle \in R\}$$

例如， $\Pi_{3, 1}(R)$ 表示关系 R 中取第 1、3 列组成新的关系，新关系中第 1 列为 R 的第 3 列，新关系的第 2 列为 R 的第 1 列。如果 R 的每列标上属性名，那么操作符 Π 的下标处也可以用属性名表示。例如，对于关系 $R(A, B, C)$ ，那么 $\Pi_{C, A}(R)$ 与 $\Pi_{3, 1}(R)$ 是等价的。

例如，有关系 R 、 S 如图 1.22 所示，它们的笛卡儿积、选择和投影运算结果如图 1.23 所示。

关系 R			关系 S		
A	B	C	A	B	C
a	b	C	b	g	a
d	a	F	d	a	f
c	b	d			

图 1.22 关系 R 和 S

$R \times S$						$\Pi \pi_{CA}(R)$		$\sigma_{B=\text{b}}(R)$		
$R.A$	$R.B$	$R.C$	$S.A$	$S.B$	$S.C$	C	A	A	B	C
a	b	c	b	g	a	c	a	a	b	c
a	b	c	d	a	f	f	d	c	b	d
d	a	f	b	g	a	d	c			
d	a	f	d	a	f					
c	b	d	b	g	a					

图 1.23 R 和 S 的笛卡儿积、选择和投影运算

(3) 连接

连接也称为 θ 连接。它是从两个关系的笛卡儿积中选取属性满足一定条件的元组，记作：

$$R \bowtie_{A\theta B} S = \{t_r t_s \mid t_r \in R \wedge t_s \in S \wedge t_r[A] \theta t_s[B]\}$$

式中， A 和 B 分别为 R 和 S 上度数相等且可比的属性组。 θ 是比较运算符。连接运算从 R 和 S 的广义笛卡儿积 $R \times S$ 中选取 (R 关系) 在 A 属性组上的值与 (S 关系) 在 B 属性组上值满足比较关系 θ 的元组。

连接运算中有两种常用的连接，一种是等值连接，另一种是自然连接。 θ 为 “=” 的连接运算称为等值连接，它是从关系 R 与 S 的广义笛卡儿积中选取 A 、 B 属性值相等的那些元组。例如，对于图 1.22 的关系 R 和 S ，它们的连接运算结果如图 1.24 所示。

自然连接是除去重复属性的等值连接，它是连接运算的一个特例，是最常用的连接运算。两个关系 R 和 S 的自然连接的形式定义为：

$$\Pi_{i_1, i_2, \dots, i_m}(R) (\sigma_{R.A_1=S.A_1 \wedge L \wedge R.A_K=S.A_K}(R \times S))$$

其中， i_1, i_2, \dots, i_m 为 R 和 S 的全部属性，但公共属性只出现一次。其具体计算过程如下：

- ① 计算 $R \times S$ ；
- ② 设 R 和 S 的公共属性是 A_1, A_2, \dots, A_K ，挑选 $R \times S$ 中满足 $R.A_1 = S.A_1, R.A_2 = S.A_2, \dots, R.A_K = S.A_K$ 的元组；
- ③ 去掉 $S.A_1, S.A_2, \dots, S.A_K$ 这些列。

例如，图 1.25 表示了关系 R 、 S 的自然连接算结果。

$R \times S$					
[2]=[1]					
$R.A$	$R.B$	$R.C$	$S.A$	$S.B$	$S.C$
a	b	c	b	g	a
c	b	d	b	g	a

图 1.24 等值连接运算

R			S			$R \times S$			
A	M	B	A	M	C	A	M	B	C
a	b	c	a	b	a	a	b	c	a
d	a	f	a	a	k	a	b	c	n
a	b	n	b	a	y	a	b	n	a
			a	b	n	a	b	n	n
			b	b	c				

图 1.25 自然连接运算

由于关系数据库是建立在关系模型基础上的，而选择、投影、连接是作为关系的二维表的三个基本运算，因此，应很好理解、掌握这三种基本运算。

习 题 1

一、填空题

- 1. 数据库管理技术的发展经过人工管理阶段，文件系统阶段和_____。
- 2. 在数据库系统中管理数据的软件称为_____。

3. 现实世界中的事物的每一个特性, 在信息世界中称_____, 在数据世界中称为_____。
4. 数据库系统提供了两个方面的映像功能, 使得数据既具有_____独立性, 又具有_____独立性。
5. 数据库系统的体系结构分为三级: 内部级、概念级和_____级。
6. 数据库的三级模式结构中, 全局数据视图的描述称为_____。
7. 模式/内模式映射为数据库提供了_____独立性。
8. 外模式/模式映射为数据库提供了_____独立性。
9. 构成数据模型的三大要素分别是数据结构、数据操作和_____。
10. 数据库系统中常用的三种数据模型是层次模型、网状模型和_____。
11. 概念模型最常用的表示方法是 E-R 图, 描述实体的特性称为_____。
12. 关系模式的三类完整性约束条件分别是_____、参照完整性约束和_____。
13. 与数据库系统相比较, _____的数据共享性低, 数据独立性低。
14. 数据库系统各类用户对数据库的各种操作请求(数据定义、查询、更新及各种控制)都是由一个复杂的软件来完成的, 这个软件称为_____。
15. 若实体 A 和 B 是多对多的联系, 实体 B 和 C 是 1 对 1 的联系, 则实体 A 和 C 是_____的联系。
16. 关系中的每一行称为一个_____, 每一列称为一个_____。
17. 数据库管理系统下层支持软件是_____, 上层软件是数据库应用_____。
18. 设有学生表 S (学号, 姓名, 班级) 和学生选课表 SC (学号, 课程号, 成绩), 为维护数据一致性, 表 S 与 SC 之间应满足_____完整性约束。
19. _____是 IBM 公司研制的一种 关系型数据库系统, 主要应用于大型应用系统。
20. 甲骨文的关系数据库是世界第一个支持_____的数据库。
21. _____是 IBM 公司出品的关系 数据库管理系统 (RDBMS) 家族。作为一个集成解决方案, 它被定位为作为 IBM 在线事务处理 (OLTP) 旗舰级数据服务系统。
22. _____是美国 Sybase 公司研制的一种关系型数据库系统, 是一种典型的 UNIX 或 Windows NT 平台上客户机-服务器环境下的大型数据库系统。
23. _____是微软公司推出的基于 Windows 的桌面关系数据库管理系统 (RDBMS, Relational Database Management System), 是 Office 系列应用软件之一。
24. 设 D_1 , D_2 和 D_3 域的基数分别为 2, 3 和 4, 则 $D_1 \times D_2 \times D_3$ 的元组数为_____, 每个元组有_____个分量。
25. 设一个关系 A 具有 a_1 个属性和 a_2 个元组, 关系 B 具有 b_1 个属性 2 个元组, 则关系 $A \times B$ 具有_____个属性和_____元组。

二、选择题

1. E-R 模型属于 ()。
A. 概念模型 B. 层次模型 C. 网状模型 D. 关系模型
2. 现实世界中, 事物的一般特性在信息世界中称为 ()。
A. 实体 B. 实体键 C. 属性 D. 关系键
3. 一个供应商可供应多种零件, 而一种零件可由多个供应商供应, 则实体供应商与零件之间的联系是 ()。
A. 一对一 B. 一对多 C. 多对一 D. 多对多
4. 关系数据库管理系统应能实现专门的关系运算包括 ()。
A. 排序、索引和统计 B. 选取、投影和连接
C. 关联、更新和排序 D. 选取、投影和更新

5. 下列哪种运算不属于关系代数的基本运算 ()。
- A. 连接 B. 笛卡儿积 C. 相减 D. 投影
6. 现有如下关系:
- 患者 (患者编号, 患者姓名, 性别, 出生日期, 所在单位)
- 医疗 (患者编号, 患者姓名, 医生编号, 医生姓名, 诊断日期, 诊断结果)
- 其中, 医疗关系中的外码是 ()。
- A. 患者编号 B. 患者姓名
C. 患者编号和患者姓名 D. 医生编号和患者编号
7. 数据的逻辑独立性是指 ()。
- A. 概念模式改变, 外模式和应用程序不变 B. 概念模式改变, 内模式不变
C. 内模式改变, 概念模式不变 D. 内模式改变, 外模式和应用程序不变
8. 在数据库系统的三级模式结构中, 用来描述数据的局部逻辑结构的是 ()。
- A. 子模式 B. 用户模式 C. 模式 D. 存储模式
9. 在关系数据库管理系统中, 创建的视图在数据库三层结构中属于 ()。
- A. 外模式 B. 存储模式 C. 内模式 D. 概念模式
10. 不允许在关系中出现重复记录的约束是通过 ()。
- A. 外键实现 B. 索引实现 C. 主键实现 D. 唯一索引实现
11. 建立基本表结构时, 定义字段的有效性规则是实现数据的 ()。
- A. 实体完整性 B. 参照完整性 C. 域完整性 D. 表完整性
12. 在数据库技术中, 实体联系模型是一种 ()。
- A. 概念数据模型 B. 结构数据模型
C. 物理数据模型 D. 逻辑数据模型
13. 假定学生关系是 $S(S\#, SNAME, SEX, AGE)$, 课程关系是 $C(C\#, CNAME, TEACHER)$, 学生选课关系是 $SC(S\#, C\#, GRADE)$, 要查找选修“COMPUTER”课程的女学生的姓名, 将涉及关系 ()。
- A. S B. SC, C C. S, SC D. S, C, SC
14. 数据是信息的符号表示或称载体; 信息则是数据的内涵, 是数据的 ()。
- A. 语法解释 B. 语义解释 C. 语意说明 D. 用法说明
15. 下列关于关系数据模型的术语中, 术语 () 所表达的概念与二维表中的“行”的概念最接近?
- A. 属性 B. 关系 C. 域 D. 元组
16. 数据库的物理独立性是指数据的 () 和组织方法改变时, 不影响数据库的逻辑结构。
- A. 逻辑结构 B. 存储格式 C. 输出格式 D. 输入格式
17. 关系数据库操作中, 从表中选出满足某种条件的记录的操作称为 ()。
- A. 选择 B. 投影 C. 扫描 D. 连接
18. 使用数据库技术进行人事档案管理是属于计算机的 ()。
- A. 科学计算应用 B. 过程控制应用 C. 数据处理应用 D. 辅助工程应用
19. 在关系理论中, 把二维表表头中的栏目称为 ()。
- A. 数据项 B. 元组 C. 结构名 D. 属性名
20. 如果要改变一个关系中属性的排列顺序, 应使用的关系运算是 ()。
- A. 重建 B. 选取 C. 投影 D. 连接
21. 我们把客观存在并且可以相互区别的事物称为 ()。

- A. 关系 B. 属性 C. 实体 D. 连接
22. E-R 图包括的要素是 ()。
- A. 实体和属性 B. 实体之间的关系和属性
C. 实体之间的关系 D. 实体, 实体之间的关系和属性
23. 单个用户使用的数据视图的描述称为 ()。
- A. 外模式 B. 概念模式 C. 内模式 D. 存储模式
24. 数据库物理存储方式的描述称为 ()。
- A. 外模式 B. 内模式 C. 概念模式 D. 逻辑模式
25. 在下面给出的内容中, 不属于 DBA 职责的是 ()。
- A. 定义概念模式 B. 修改模式结构
C. 编写应用程序 D. 编写完整性规则
26. 在数据库三级模式间引入两种映射的主要作用是 ()。
- A. 提高数据与程序的独立性 B. 提高数据与程序的安全性
C. 保持数据与程序的一致性 D. 提高数据与程序的可移植性
27. DB、DBMS 和 DBS 三者之间的关系是 ()。
- A. DB 包括 DBMS 和 DBS B. DBS 包括 DB 和 DBMS
C. DBMS 包括 DB 和 DBS D. 不能相互包括
28. 数据库系统中的数据模型通常由 () 三部分组成。
- A. 数据结构、数据操作和完整性约束 B. 数据定义、数据操作和安全性约束
C. 数据结构、数据管理和数据保护 D. 数据定义、数据管理和运行控制
29. 数据库技术的三级模式中, 数据的全局逻辑结构用 () 来描述。
- A. 子模式 B. 用户模式 C. 模式 D. 存储模式
30. 用户涉及的逻辑结构用 () 描述。
- A. 模式 B. 存储模式 C. 概念模式 D. 子模式
31. 数据库管理系统与操作系统、应用软件的层次关系从核心到外围依次是 ()。
- A. DBMS、OS、应用软件 B. DBMS、应用软件、OS
C. OS、DBMS、应用软件 D. OS、应用软件、DBMS
32. 下列各种模型中不是数据模型的是 ()。
- A. 概念模型 B. 层次模型 C. 网状模型 D. 关系模型
33. 同网状模型和层次模型相比, 关系模型具有很多的特点, 下列各项中 () 不是关系模型所具有的特点?
- A. 概念单一 B. 规范化 C. 直观性好 D. 以二维表格表示
34. 数据库系统依靠 () 支持了数据独立性。
- A. 具有封装机制 B. 模式分级、各级之间有映像机制
C. 定义完整性约束条件 D. DDL 语言和 DML 语言互相独立
35. 关系 R 和 S 进行自然连接时, 要求 R 和 S 含有一个或多个公共 ()。
- A. 元组 B. 行 C. 记录 D. 属性
36. 关系 R 和 S , $R \cap S$ 的运算等价于 ()。
- A. $S - (R - S)$ B. $R - (R - S)$ C. $(R - S) \cup S$ D. $R \cup (R - S)$

第 2 章 数据库设计

数据库设计是数据库应用的基础，其核心任务是针对特定的应用环境，在给定硬件环境、操作系统和数据库管理系统的基础上，创建一个性能良好的数据库模式，建立数据库及应用系统，使之能够有效、合理地采集、存储、操作和管理数据，满足企业或组织中各类用户的应用需求。

2.1 工程化设计思想

1968 年北大西洋公约组织的计算机科学家在前联邦德国召开国际会议，讨论软件危机问题，在这次会议上正式提出并使用了“软件工程”这个名词。

2.1.1 软件工程

软件工程是指导计算机软件开发和维护的工程学科，软件工程通过采用工程的概念、原理、技术和方法来开发与维护软件，把经过时间考验而证明正确的管理技术和当前能够得到的最好的技术方法结合起来。

软件工程的核心理念是把软件看作一个工程产品来处理，把需求计划、可行性研究、工程审核、质量监督等工程化的概念引入到软件生产中。软件工程包括三个要素：方法、工具和过程。方法是完成软件工程项目的手段；工具用来支持软件的开发、管理和文档生成；过程是指软件开发各个环节的控制和管理。通过三要素以期达到工程项目的三个基本目标：进度、经费和质量。

软件工程的目的是提高软件的质量与生产率，最终实现软件的工业化生产。质量是软件需求方最关心的问题，生产率是软件供应方最关心的问题。质量与生产率之间有着内在的联系，高生产率必须以质量合格为前提，好的软件工程方法可以同时提高质量与生产率。软件工程的主要环节包括可行性分析、需求分析、概要设计、详细设计、编码、软件测试、维护、人员管理、项目管理等。

2.1.2 软件生命周期

同任何事物一样，一个软件产品或软件系统要经历孕育、诞生、成长、成熟、衰亡等阶段，一般称为软件生存周期（软件生命周期），即从软件的产生直到消亡的周期，整个软件生存周期可划分为若干阶段，每个阶段有明确的任务。通常，软件生存周期包括可行性分析、需求分析、系统设计（概要设计和详细设计）、编码、测试、维护等活动，如图2.1所示。

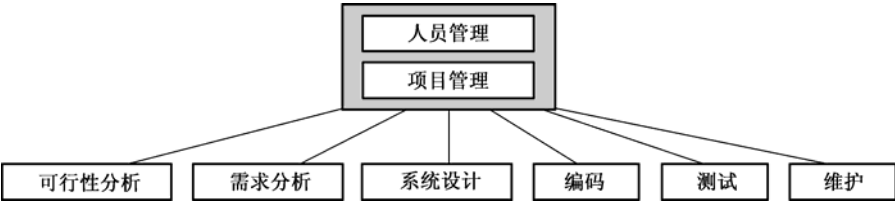


图 2.1 软件工程的主要环节

(1) 可行性分析

确定软件系统开发目标和总的要求，给出它的功能、性能、可靠性以及接口等方面的可能方案，制订完成开发任务的实施计划。

(2) 需求分析

对待开发软件提出的需求进行分析并给出详细定义。编写软件规格说明书及初步的用户手册，提交评审。

(3) 软件设计

系统设计人员和程序设计人员在反复理解软件需求的基础上给出软件结构、模块划分、功能分配以及处理流程。编写概要设计说明书、详细设计说明书和测试计划初稿，提交评审。在系统比较复杂的情况下，设计阶段可分解成概要设计阶段（总体设计）和详细设计阶段。

(4) 软件实现

把软件设计转换为计算机执行的程序代码。即完成程序编码，编写用户手册、操作手册等面向用户的文档，制订单元测试计划。

(5) 软件测试

在设计测试用例基础上，检验软件的各个组成部分，编写测试分析报告。

(6) 运行和维护

将交付的软件投入运行，并在运行中不断维护，根据新提出的需求进行必要的扩充和删改。

2.2 数据库设计概述

数据库设计以具体的 DBMS 为基础，进行数据库的结构特性设计和数据库的行为特性设计。其中数据库的结构特性设计起着关键作用，数据库设计的基本内容如图 2.2 所示。

数据库的结构特性是静态的，在一般情况下不会轻易变动。因此，数据库的结构特性设计又称为静态结构设计。其设计过程是：先将现实世界中的事物和事物之间的联系用 E-R 图表示，再将各个分 E-R 图汇总，得出数据库的概念结构模型，最后将概念结构模型转换为数据库的逻辑结构模型表示。

数据库的行为结构设计包括确定数据库用户的行为和动作。数据库用户的行为和动作是指数据查询和统计、事物处理及表报处理等，这些都需要通过应用程序表达和执行。因此，设计数据库的行为特征要与应用系统的设计结合进行。因为用户的行为是动态的，所以，数据库的行为特性设计也称为数据库的动态设计。其设计过程是，首先将现实世界中的数据及应用情况用数据流图和数据字典表示，然后详细描述其中的数据操作要求，进而得出系统的功能结构和数据库的子模式。

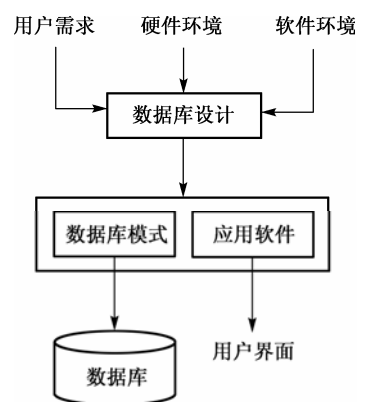


图 2.2 数据库设计的基本内容

2.3 数据库设计的基本步骤

同软件一样，数据库也存在生存期，其生存周期如图 2.3 所示。

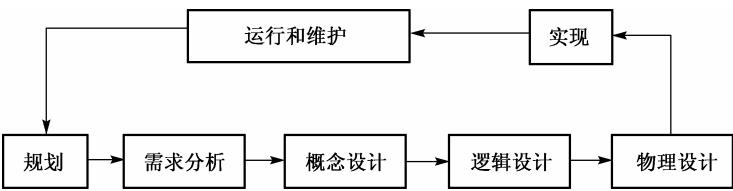


图 2.3 数据库生命周期

根据生命周期思想数据库设计过程可分为规划、需求分析、概念设计、逻辑设计、物理设计、实现、运行和维护 7 个阶段。

2.3.1 规划

系统规划的目的在于确定系统名称、范围，确定系统的目标功能和性能，确定系统所需的资源，估算预期效益和开发成本，确定开发计划和开发进度。主要包括系统调查和可行性分析两个阶段。

1. 系统调查

对企业组织作全面的调查，画出组织层次图，以了解企业的组织结构。

2. 可行性分析

可行性分析是要决定“做还是不做”，是整个项目中的第一步，其最根本的任务是对以后的行动方针提出建议。这一步要作的事情很多，不仅涉及成本，还要涉及人力资源，环境分析，预计回报等很多因素。一般情况，软件领域的可行性分析主要考虑四个要素：经济、技术、社会环境和人。

- (1) 经济可行性
经济可行性分析主要包括：成本分析和收益分析。成本一般包括开发成本、维护成本。收益包括短期收益和长远收益。
- (2) 技术可行性
技术可行性是指参与开发的相关人员的技术水平与软、硬件资源能否满足开发的要求。技术可行性分析可以简单地表述为：是否能做？是否做得好？是否做得快？
- (3) 社会可行性
社会可行性至少包括两种因素：市场与政策。市场又分为未成熟的市场、成熟的市场和将要消亡的市场。
- (4) 人员分析
软件开发是个复杂的过程，需要各种人员的配合，最好是人尽其才。即适合组织、决策的作领导，适合软件开发的作程序员，适合于人际关系的作协调员。

3. 可行性分析的过程

在复查系统规模、目标和研究目前正在所使用系统的基础上导出新系统的高层逻辑模型，然后重新定义问题，导出和评价供选择的解法并推荐行动方针，草拟开发计划，最后编写可行性分析报告并提交审查。

经过可行性分析，如果问题没有可行的解，分析员应该建议停止开发工程，以避免时间、资源、人力和金钱的浪费；如果问题可以解决，分析员应该推荐一个较好的解决方案，并且为工程制定一个初步的计划。

2.3.2 需求分析

需求分析是整个开发过程的基础，也是最困难、最耗费时间的一步。需求分析是否充分与准确将决定在其上构建数据库的速度与质量。需求分析做的不好，会导致整个数据库应用系统开发返工重做。

进行数据库应用软件的开发，首先要准确了解与分析用户需求（包括数据处理）。通过对现实世界的处理对象进行详细调查，收集支持系统目标的数据，确定新系统的功能和处理方法。

1. 基本流程

需求分析以用户调查为基础，通过分析和综合，逐步明确用户对新系统的功能和性能要求，一般采用自顶向下的方法实现。需求分析的一般过程如图2.4所示。

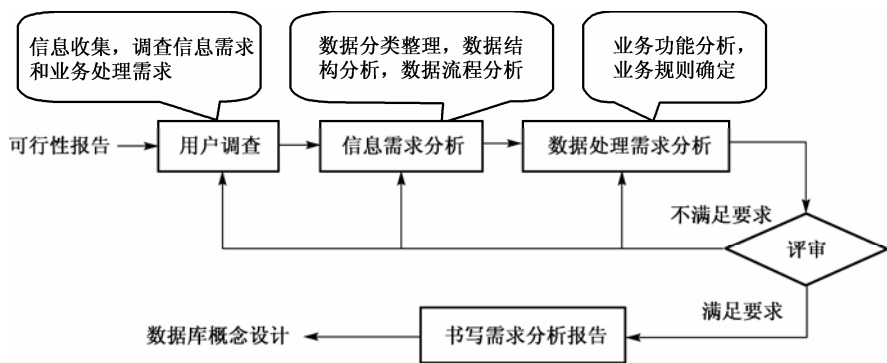


图 2.4 需求分析基本步骤

需求分析一般采用数据流图描述数据处理过程，数据流图的构成元素如图2.5所示。数据流图简称 DFD，它以图形的方式描绘数据在系统中流动和处理的过程，由于它只反映系统必须完成的逻辑功能，所以它是一种功能模型。图2.6是银行取款过程的数据流图，数据流图仅是一个事务处理流程的抽象描述，在数据流图中并没有对数据流进行详细的说明。而数据字典（Data

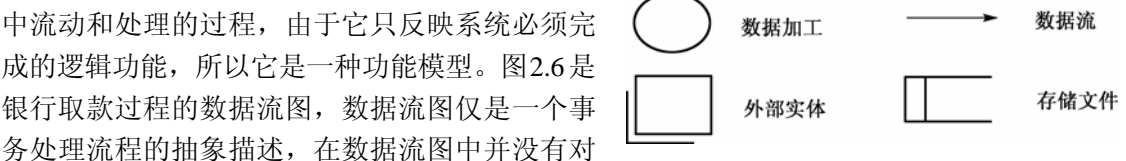


图 2.5 数据流图中的主要图形元素

Dictionary，DD）的作用就是用来定义数据流图中的各个成分的具体含义，它以一种准确的、无二义性的说明方式为系统的分析、设计及维护提供了有关元素一致的定义和详细的描述。它和数据流图共同构成了系统的逻辑模型，是需求规格说明书的主要组成部分。数据词典与数据流图配合，能清楚地表达数据处理的要求。对于在数据流图中每一个被命名的图形元素在数据字典中均加以定义，其内容主要有：名字，编号，描述，定义，其他等。

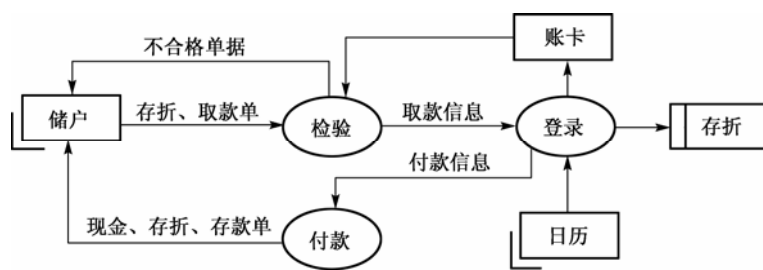


图 2.6 描述银行取款过程的数据流图

2. 阶段成果

需求分析阶段的主要成果包括以下内容：

- ① 分析用户活动，生成业务流程图；
- ② 确定系统范围，生成系统范围图；
- ③ 分析用户活动涉及的数据，生成数据流图；
- ④ 分析系统数据，生成数据字典。

2.3.3 概念结构设计

概念结构设计的主要任务是根据需求分析阶段产生的需求分析报告，产生能够反映信息需求的数据库概念结构，即数据库的逻辑模型。产生的概念模型不依赖于计算机系统和具体的 DBMS，产生概念模型的过程称之为概念设计。概念结构设计是整个数据库设计的关键，它通过对用户需求进行综合、归纳与抽象，形成一个独立于具体 DBMS 的概念模型，一般用 E-R 图表示概念模型。概念设计的主要任务如图2.7所示。

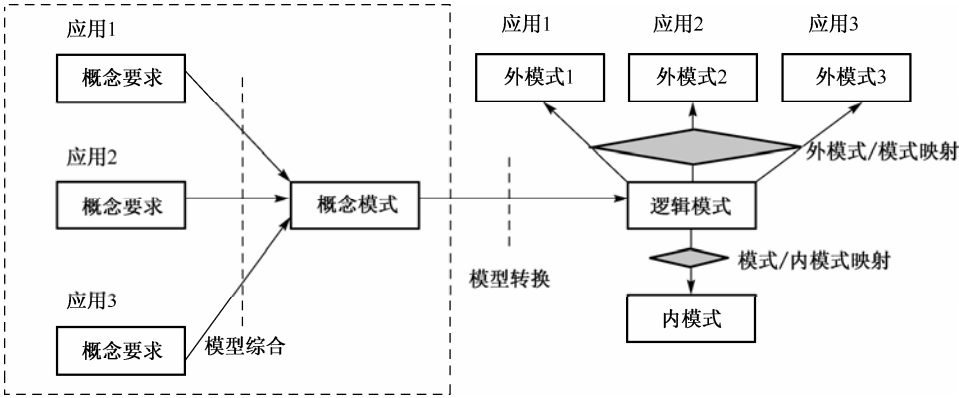


图 2.7 概念设计的主要任务

1. E-R图的基本组成

E-R 图是描述概念模型的基本工具，E-R 图一般由三部分组成，如图2.8所示。

- ① 矩形框，用于表示实体类型（考虑问题的对象）。
- ② 菱形框，用于表示联系类型（实体间联系）。
- ③ 椭圆形框，用于表示实体类型和联系类型的属性。

2. E-R图的绘制步骤

E-R 图可经由以下步骤获得：

- ① 首先确定实体类型；
- ② 确定联系类型；
- ③ 把实体类型和联系类型组合成 E-R 图；
- ④ 确定实体类型和联系类型的属性；
- ⑤ 确定实体类型的键，在 E-R 图中属于键的属性名下画一条横线。

例如，某商业集团中，仓库、商品、商店的实体表示如图2.9、图2.10、图2.11所示。最后获得的实体联系图如图2.12所示。

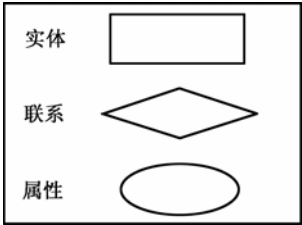


图 2.8 E-R 图的构成元素



图 2.9 仓库实体

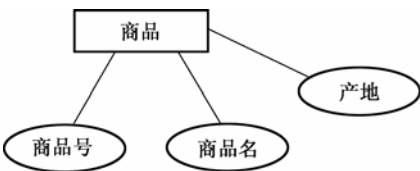


图 2.10 商品实体

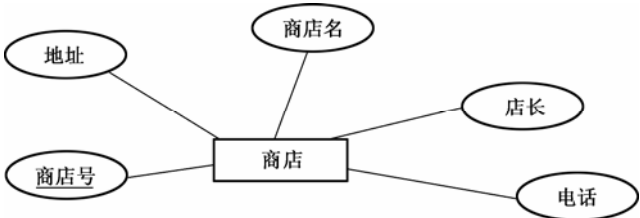


图 2.11 商店实体

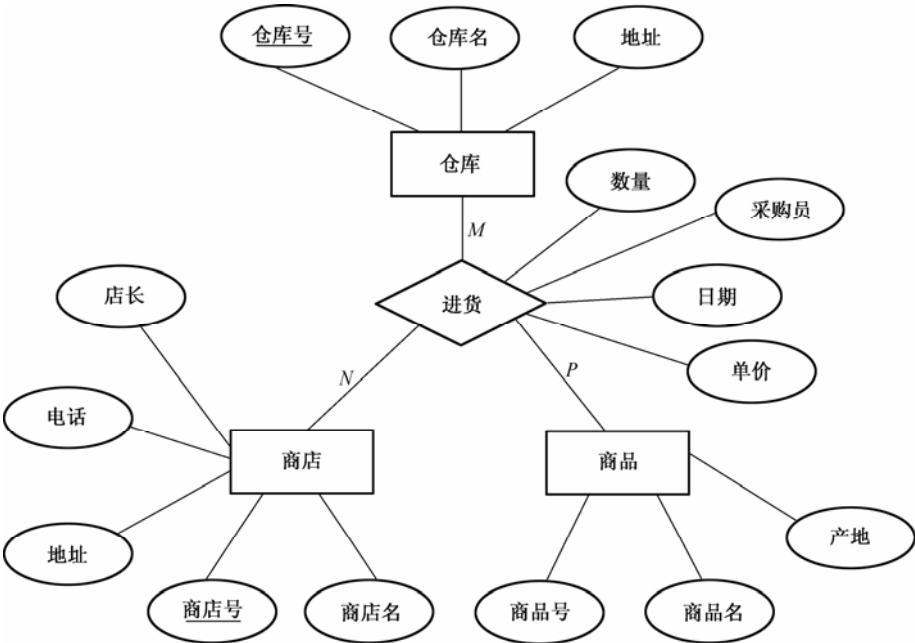


图 2.12 实体联系图

3. 概念设计的主要步骤

- ① 进行数据抽象，设计局部概念模式，其基本过程如图2.13所示。
- ② 将局部概念模式综合成全局概念模式，其基本过程如图2.14所示。
- ③ 评审，若评审通过，则进入下一个设计阶段。

2.3.4 逻辑结构设计

逻辑结构设计的主要任务是将现实世界获取的概念 数据模型设计成数据库能够支持的某种逻辑模型。同时，还需为各种数据处理应用领域产生相应的逻辑子模式，并使其在功能、性能、完整性约束、一致性和可扩充性等方面均满足用户的需求。这一步设计的结果就是所谓逻辑数据库。

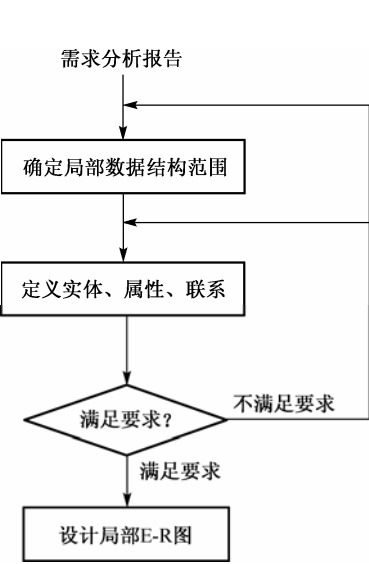


图 2.13 局部 E-R 模型设计步骤

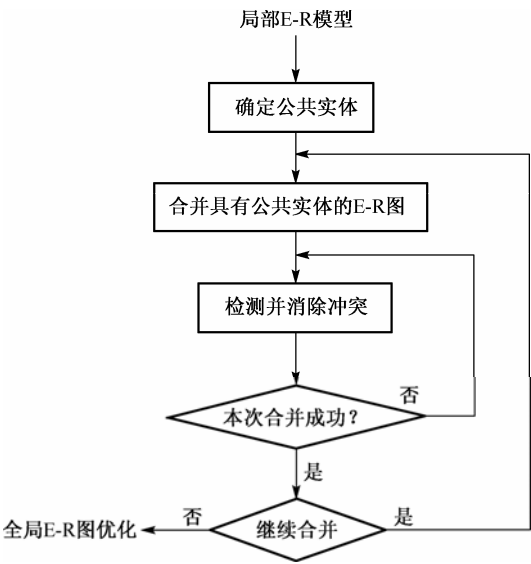


图 2.14 全部 E-R 模型设计步骤

1. 基本步骤

逻辑结构设计的基本步骤如图2.15所示。

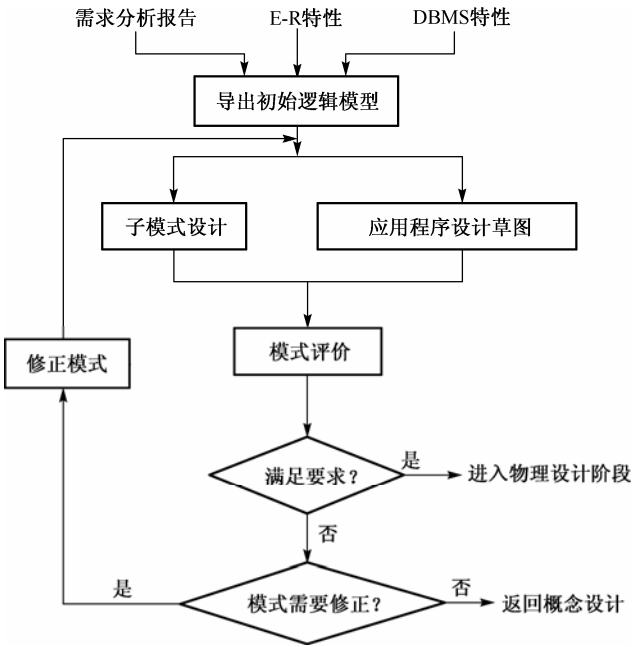


图 2.15 逻辑结构设计步骤

2. 模型转换

E-R 模型向关系模型的转换，实际上就是把 E-R 图转换成关系模式的集合。包括两部分转换：实体型的转换和联系的转换。

(1) 实体类型的转换

将每个实体类型转换成一个关系模式，实体的属性即为关系模式的属性，实体标识符为关系模式的键。

(2) 联系的转换

联系的转换分三种情况：二元联系的转换、一元联系的转换、三元联系的转换。

① 二元联系的转换

二元联系的转换分三种情况：一对一联系的转换、一对多联系的转换、多对多联系的转换。

◆ 一对一联系的转换

若实体间联系是 $1:1$ ，可以在两个实体类型转换成的两个关系模式中任意一个关系模式的属性中加入另一个关系模式的键和联系类型的属性。

◆ 一对多联系的转换

若实体间联系是 $1:N$ ，则在 N 端实体类型转换成的关系模式中加入 1 端实体类型的键和联系类型的属性。

◆ 多对多联系的转换

若实体间联系是 $M:N$ ，则将联系转换成关系模式，其属性为两端实体类型的键加上联系类型的属性，而键为两端实体键的组合。

② 一元联系的转换

同二元联系。

③ 三元联系的转换

将三元联系转换成关系模式，其属性为三端实体类型的键加上联系类型的属性，而键为三端实体键的组合。

对于图2.12而言，转换后得到的关系模式如下：

仓库（仓库号，仓库名，地址）

商店（商店号，商店名，店长，电话，地址）

商品（商品号，商品名，产地）

进货（商店号，商品名，仓库号，日期，数量，采购员，单价）

3. 应该注意的问题

在逻辑模型设计时，需要注意以下问题。

(1) 模式的规范化

模式的规范化有助于消除数据库中的数据冗余。规范化有好几种形式，但第三范式通常被认为在性能、扩展性和数据完整性方面达到了最好平衡。

(2) 考虑变化

在设计数据库的时候要考虑哪些数据字段将来可能会发生变化。例如，姓氏（西方人的姓氏，女性结婚后从夫姓等），在建立系统存储客户信息时，可在单独的一个数据表里存储姓氏字段，而且还附加起始日和终止日等字段，这样就可以跟踪这一数据的变化。

(3) 选择数字类型和文本类型尽量充足

例如，在 SQL 中使用 Smallint 和 Tinyint 类型要特别小心。如果想查看月销售总额，总额字段类型是 Smallint，若总额超过了 \$32,767 就不能进行计算操作了。

而 ID 类型的文本字段，比如客户 ID 或定单号等都应该设置得比一般值更大。假设客户 ID 为 10 位数长，应该把数据库表字段的长度设为 12 或者 13 个字符长。虽然增加了额外空间，但将来数据库的规模增长时，无需重构整个数据库。

(4) 选择键和索引

键设计基本原则为：关联字段创建外键，所有的键都必须唯一，避免使用复合键，外键总是关联唯一的键字段。索引是从数据库中获取数据的高效方式之一，95%的数据库性能问题都可以采用索引技术得到解决。大多数数据库都索引自动创建的主键字段，不要索引大型字段（有很多字符），这样做会让索引占用太多的存储空间。不要索引常用的小型表，因为对插入和删除操作的索引维护可能比扫描表空间消耗的时间更多。

2.3.5 物理设计

数据库的物理设计依赖于具体计算机结构的各项物理设计措施，是指在特定数据库管理系统所提供的多种存储结构和存取方法的基础上，为逻辑数据模型选取一个最适合应用环境的物理结构（包括文件类型、索引结构和数据的存放次序）、存取方法和存取路径等。设计的结果就是所谓“物理数据库”。物理设计步骤如图 2.16 所示，简单地说，包括 5 个阶段。

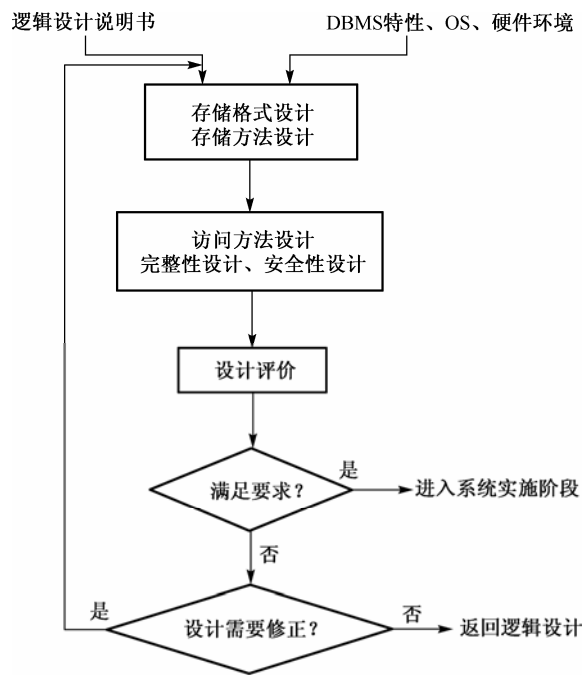


图 2.16 物理设计步骤

(1) 存储记录的格式设计

对所存储的数据项进行类型特征分析，对存储的记录进行格式设计，决定数据的压缩格式和代码格式。

(2) 存储方法设计

存储方法设计主要考虑存储记录在存储空间的物理安排，常见的存储方式有 4 种：顺序存放、杂凑存放、索引存放、聚簇存放。不同的存放方式读/写效率不同。

(3) 访问方法设计

访问方法设计为数据提供存储结构和查询路径。

(4) 完整性和安全性设计

根据数据库的完整性约束条件和系统所采用的具体的 DBMS、OS 特性，以及硬件环境设计数据库所采用的完整性和安全性措施。

(5) 应用设计

应用设计解决应用系统的功能实现设计，主要包括应用系统的界面设计、代码设计、模块设计等。

2.3.6 系统实施

数据库实施阶段的任务是根据逻辑设计和物理设计的结果，在计算机上建立数据库，编制与调试应用程序，组织数据入库，并进行系统测试和试运行，其实施步骤如图2.17所示。

数据库实施阶段应做的工作主要包括以下三个方面。

(1) 用DDL定义数据库结构

根据具体的 DBMS 提供的数据库定义语言和数据库的逻辑结构模型创建数据库的结构，此时所得到仅仅是数据库的框架，里面没有任何的数据。

(2) 组织数据入库

对于创建好的数据库，需要录入数据，对数据量不大的小型系统，可采用人工方式实现数据入库，主要工作包括：筛选数据，转换数据格式，输入数据，校验数据。对大中型系统，则要通过相应的硬件或者软件自动输入，主要工作包括：筛选数据，输入数据，校验数据，转换数据格式，综合数据。

(3) 编制与调试应用程序

系统的许多功能需要通过代码才能够实现，编制程序的目的在于根据选择的程序设计语言实现处理功能。由于程序是一个复杂的逻辑实体，需要通过测试等手段尽可能地发现并改正程序中隐藏的错误。

2.3.7 运行和维护

数据库应用系统正式投入运行后，在数据库系统运行过程中必须不断地对其进行评价、调整与修改。其主要工作包括：

- (1) 数据库的转储和恢复；
- (2) 数据库安全性和完整性控制；
- (3) 数据库性能的监督、分析和改进；
- (4) 数据库的重组和重构造。

开发一个完善的数据库应用系统不可能一蹴而就，它往往是需求分析、概念设计、逻辑设计、物理设计、系统实施、运行和维护这 6 个阶段的不断反复。而这 6 个阶段不仅包含了数据库的设计过程，而且包含了数据库应用系统的设计过程。在设计过程中，应该把数据库

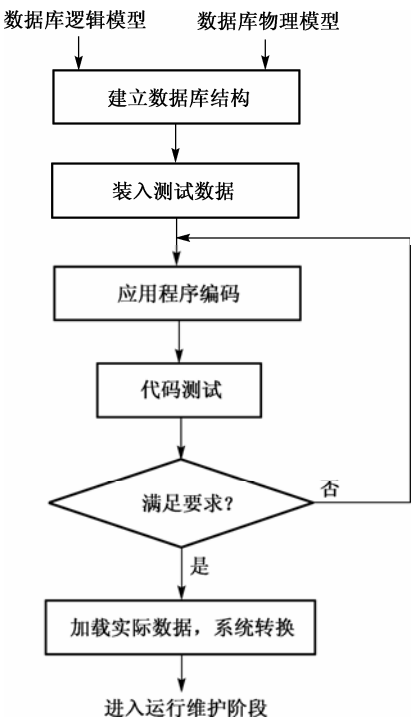


图 2.17 系统实施阶段的主要工作

的结构特性设计和数据库的行为特性设计紧密结合起来，将这两个方面的需求分析、数据抽象、系统设计及实现等各个阶段同时进行，相互参照，相互补充，以完善整体设计。

2.4 关系模式的规范化

关系数据库的规范化设计是指面对一个现实问题，如何选择一个比较好的关系模式集合。规范化设计理论对关系数据库结构的设计起着重要的作用，其主要包括三个方面的内容：数据依赖、范式和模式设计方法。数据依赖研究数据之间的联系，范式是关系模式的标准，模式设计方法是自动化设计的基础，其中数据依赖起着核心的作用。

例如，现有学生成绩表如表 2.1 所示。

表 2.1 学生成绩表

姓名	宿舍号	课程号	课程名	成绩
张玉寒	2-406	C1	大学语文	89
张玉寒	2-406	C2	普通物理	76
张玉寒	2-406	C3	高等数学	90
叶秋	3-201	C4	体育	85
叶秋	3-201	C5	大学英语	83
李梦	5-602	C6	计算机基础	90

对于该关系，可以发现存在着一些问题。

（1）数据冗余

当一个学生选修了多门课程后，该学生的宿舍号就要多次重复存储，因此产生数据冗余。

（2）操作异常

由于数据的冗余，在对数据操作时会引起一些异常，主要包括修改异常、插入异常和删除异常。

① 修改异常

例如学生张玉寒选修了三门课程，在关系中就会有三个元组。如果他的宿舍号变了，这三个元组中的宿舍号都需要改变。若有一个元组中的地址未更改，就会造成这个学生的宿舍号不唯一，产生不一致现象。

② 插入异常

如果一个学生前来登记，但他尚未选择课程，那么将学生的姓名和宿舍号存储到关系中时，在属性课程号和课程名中就没有值（空值）。在数据库技术中，空值的语义是非常复杂的，对带空值元组的检索和操作也十分麻烦。

③ 删除异常

如果想取消学生李梦的计算机基础成绩，就需要把对应的元组删去，同时也会把李梦的宿舍信息从表中删去了，这是不合理的。

由此可见，表 2.1 对应的关系模式学生成绩表（姓名，宿舍号，课程号，课程名，成绩）并不是一个好的关系模式，需要对其进行规范化，以解决数据冗余、修改异常、插入异常和删除异常等问题。

2.4.1 函数依赖和键

在关系模式的规范化过程中，有两个重要的概念需要了解，那就是函数依赖和键。

1. 函数依赖

现实世界中，一个实体往往拥有多个属性，而这些属性之间相互制约、相互依赖，这种关系被称为数据依赖，数据依赖形式多样，而最为重要的数据依赖就是函数依赖。函数依赖可定义如下：

设有关系模式 $R(U)$ ， X 和 Y 是属性集 U 的子集，函数依赖 (Functional Dependency, FD) 是形如 $X \rightarrow Y$ 的一个命题，只要 r 是 R 的当前关系，对 r 中任意两个元组 t 和 s ，若有 $t[X]=s[X]$ 就必有 $t[Y]=s[Y]$ (若它们在 X 上的值相等，则它们在 Y 上的属性值也一定相等)，那么称“ X 函数决定 Y ”或“ Y 函数依赖于 X ”，记作 $X \rightarrow Y$ ，并称 X 为决定因素。

例如，有关系模式：学生 (学号，姓名，性别，出生日期，入学日期，院系)。在这个关系模式对应的关系中，所有元组的学号唯一。因此，只要学号确定，则该学生确定，其对应的其他属性：姓名，性别，出生日期，入学日期，院系等属性也就唯一确定。可以说姓名、性别、出生日期、入学日期、院系函数依赖于学号；也可以说学号函数决定姓名、性别、出生日期、入学日期、院系。其依赖关系可表示如图2.18。

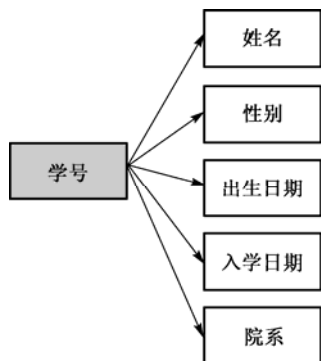


图 2.18 函数依赖关系

再例如，有一个关于学生选课关系模式：

$R(S\#, SNAME, C\#, GRADE, CNAME, TNAME, TAGE)$

属性分别表示学生学号、姓名、选修课程的课程号、成绩、课程名、任课教师姓名和年龄等意义。

如果规定，每个学号只能有一个学生姓名，每个课程号只能决定一门课程，则有以下函数依赖关系：

$S\# \rightarrow SNAME, C\# \rightarrow CNAME$

每个学生每学一门课程，有一个成绩，那么可写出下列函数决定：

$(S\#, C\#) \rightarrow GRADE$

还可以写出其他一些函数决定：

$C\# \rightarrow (CNAME, TNAME, TAGE),$

$TNAME \rightarrow TAGE$

其依赖关系可表示为如图2.19所示。

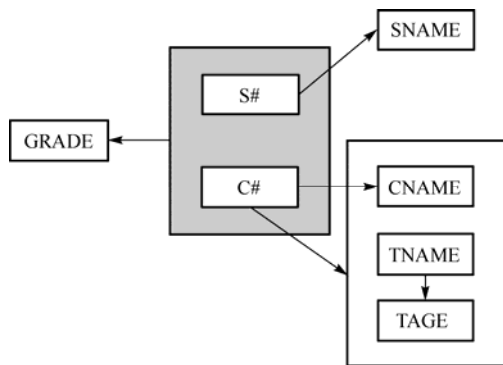


图 2.19 学生选课表函数依赖关系

2. 键

所谓键，简单地说，就是在函数依赖中起决定作用的属性或属性集，也称为候选键。候选键定义如下：

设 U 是关系模式 R 的属性集， X 是 U 的一个子集。如果 $X \rightarrow U$ 在 R 上成立，那么称 X 是 R 的一个超键。如果 $X \rightarrow U$ 在 R 上成立，但对于 X 的任一真子集 X_1 ， $X_1 \rightarrow U$ 都不成立，那么称 X 是 R 上的一个候选键。

例如, 在学生选课关系模式 $R(S\#, SNAME, C\#, GRADE, CNAME, TNAME, TAGE)$ 中, 如果规定每个学生每学一门课只有一个成绩, 每个学生只有一个姓名, 每个课程号只有一个课程名, 每门课程只有一个任课教师。根据这些规则, 可以知道 $(S\#, C\#)$ 能函数决定 R 的全部属性, 是一个候选键。虽然 $(S\#, SNAME, C\#, TNAME)$ 也能函数决定 R 的全部属性, 但相比之下, 只能说是一个超键, 而不能说是候选键, 因为其中含有多余属性。

和候选键相对应的另外一个概念就是外键, 外键的定义为: 有关系 R 和 S , 在 R 中存在属性或属性组 X , X 不是 R 的主键, 在 S 存在同样的属性或属性组 X , X 是 S 的主键, 则称 X 是关系 R 的外键。

2.4.2 关系模式的范式

关系模式的好与坏需要有一个衡量标准, 这个标准就是模式的范式 (Normal Forms, NF)。范式的种类与数据依赖有着直接的联系, 基于函数依赖的范式有 1NF、2NF、3NF、BCNF 等多种。

低一级范式的关系模式可以分解转换为若干个高一级范式的关系模式集合。在数据设计过程中, 一般要求设计的关系模式必须达到第三范式的要求。

1. 第一范式 (1NF)

如果关系模式 R 的每个关系 r 的属性值都是不可分的原子值, 那么称 R 是第一范式的模式。

满足 1NF 的关系称为规范化的关系, 否则称为非规范化的关系。关系数据库研究的关系都是规范化的关系。1NF 是关系模式应具备的最起码的条件。

满足第一范式的关系虽然是规范化的关系, 但其仍存在数据冗余、插入异常、更新异常和修改异常等问题, 需要将其转换为满足更高一级范式要求的关系模式。

2. 第二范式 (2NF)

对于函数决定 $W \rightarrow A$, 如果存在 $X \subset W$, 有 $X \rightarrow A$ 成立, 那么称 $W \rightarrow A$ 是局部依赖 (A 局部依赖于 W); 否则称 $W \rightarrow A$ 是完全依赖。完全依赖也称为“左部不可约依赖”。

如果 X 是关系模式 R 的候选键中属性, 那么称 X 是 R 的主属性; 否则称 X 是 R 的非主属性。

如果关系模式 R 是 1NF, 且每个非主属性完全函数依赖于候选键, 那么称 R 是第二范式 (2NF) 的模式。如果数据库模式中每个关系模式都是 2NF, 则称数据库模式为 2NF 的数据库模式。

1NF 分解成 2NF 模式集的方法如下: 设关系模式 $R(U)$, 主键是 W , R 上还存在函数依赖 $X \rightarrow Z$, 并且 Z 是非主属性, 且 $X \subset W$, 那么 $W \rightarrow Z$ 就是一个局部依赖。此时应把 R 分解成两个模式:

① $R_1(XZ)$, 主键是 X ;

② $R_2(Y)$, 其中 $Y = U - Z$, 主键仍是 W , 外键是 X 。

利用外键和主键的连接可以从 R_1 和 R_2 重新得到 R 。

如果 R_1 和 R_2 还不是 2NF, 则重复上述过程, 直到数据库模式中每一个关系模式都是 2NF 为止。

例如, 有关系模式 $R(S\#, C\#, GRADE, TNAME, TADDR)$ 的属性分别表示学生学号、选修课程的编号、成绩、任课教师姓名和教师地址。 $(S\#, C\#)$ 是 R 的候选键。

R 上有两个函数决定: $(S\#, C\#) \rightarrow (TNAME, TADDR)$ 和 $C\# \rightarrow (TNAME, TADDR)$, 因此前一个函数决定是局部依赖, 所以, R 不是 2NF 模式。此时 R 的关系就会出现冗余和异

常现象。例如某一门课程有 80 个学生选修，那么在关系中就会存在 80 个元组，因而教师的姓名和地址就会重复 80 次。

如果把 R 分解成 $R_1(C\#, TNAME, TADDR)$ 和 $R_2(S\#, C\#, GRADE)$ 后，局部依赖 $(S\#, C\#) \rightarrow (TNAME, TADDR)$ 就消失了。则 R_1 和 R_2 都是 2NF 模式。

有些关系虽然满足 2NF，但仍然可能存在操作异常问题。存在操作异常的主要原因在于这些关系虽然满足 2NF，但存在着传递依赖。

所谓传递依赖是指：若有 $X \rightarrow Y$, $Y \rightarrow Z$ ，且 Y 不函数决定 X ， Z 不是 Y 的子集，则称 X 传递决定 Z ，或者说 Z 传递依赖于 X 。

3. 第三范式 (3NF)

如果关系模式 R 是 2NF，且每个非主属性都不传递依赖于 R 的主属性，那么称 R 是第三范式 (3NF) 的模式。如果数据库模式中每个关系模式都是 3NF，则称其为 3NF 的数据库模式。

分解成 3NF 模式集的方法如下：

设关系模式 $R(U)$ 是 2NF，主键是 W ， R 上还存在函数依赖 $X \rightarrow Z$ 。并且 Z 是非主属性， $Z \notin X$ ， X 不是候选键，这样 $W \rightarrow Z$ 就是一个传递依赖。此时应把 R 分解成两个模式：

- ① $R_1(XZ)$ ，主键是 X ；
- ② $R_2(Y)$ ，其中 $Y=U-Z$ ，主键仍是 W ，外键是 X 。

利用外键和主键相匹配机制， R_1 和 R_2 通过连接可以重新得到 R 。

如果 R_1 和 R_2 还不是 3NF，则重复上述过程，直到数据库模式中每一个关系模式都是 3NF 为止。

例如，有关系模式 $R_1(C\#, TNAME, TADDR)$ 和 $R_2(S\#, C\#, GRADE)$ ，可以发现 R_2 是 2NF 模式，而且也是 3NF 模式。但 $R_1(C\#, TNAME, TADDR)$ 是 2NF 模式，却不一定是 3NF 模式。如果 R_1 中存在函数依赖 $C\# \rightarrow TNAME$ 和 $TNAME \rightarrow TADDR$ ，那么 $C\# \rightarrow TADDR$ 就是一个传递依赖，即 R_1 不是 3NF 模式。此时 R_1 的关系中也会出现冗余和异常操作。譬如一个教师开设五门课程，那么关系中就会出现五个元组，教师的地址就会重复五次。

如果把 R_1 分解成 $R_{11}(TNAME, TADDR)$ 和 $R_{12}(C\#, TNAME)$ 后， R_{11} 和 R_{12} 都是 3NF 模式。

2.4.3 模式分解

除了 1NF、2NF、3NF 还有更高级的范式。包括，BCNF、4NF、5NF，规范化的程度决定于实际需要，因为分离的越深，产生的关系就越多，关系过多，就需要通过频繁的连接操作来获取数据，这样会影响速度。在进行关系数据库设计时，应作权衡，尽可能使数据库模式保持最好的特性。

1. 模式分解的概念

设有关系模式 $R(U)$ ，属性集为 U ， R_1, R_2, \dots, R_k 都是 U 的子集，并且有 $R_1 \cup R_2 \cup \dots \cup R_k = U$ 。关系模式 R_1, R_2, \dots, R_k 的集合用 ρ 表示， $\rho = \{R_1, R_2, \dots, R_k\}$ 。用 ρ 代替 R 的过程称为关系模式的分解。这里 ρ 称为 R 的一个分解，也称为数据库模式。

设 R 是一个关系模式， F 是 R 上的一个 FD 集。 R 分解成数据库模式 $\rho = \{R_1, R_2, \dots, R_k\}$ 。如果对 R 中满足 F 的每一个关系 r ，都有 $r = \Pi R_1(r) \times \Pi R_2(r) \times \dots \times \Pi R_k(r)$ ，那么称分解 ρ 相对于 F 是“无损连接分解”，简称为“无损分解”，否则称为“损失分解”。

2. 分解中注意的问题

进行关系模式规范化的核心是分析函数依赖，在保证属性不可再分的基础上，将关系模式中存在的部分依赖和传递依赖分离，分别建立新的关系模式，这样就可以满足第三范式要求，规范化过程如图2.20所示。

关系模式在分解时应保持“等价”性，等价分解包含数据等价和语义等价两个方面，分别用无损分解和保持函数依赖两个特征来衡量。前者能保持关系在投影连接以后仍能恢复回来，而后者能保证数据在投影或连接中语义不会发生变化，也就是不会违反函数依赖的语义。但无损分解与保持依赖两者之间没有必然的联系。

范式是衡量模式优劣的标准，范式表达了模式中数据依赖之间应满足的联系。关系模式的规范化过程实际上是一个“分解”过程，把逻辑上独立的信息放在独立的关系模式中。分解是解决数据冗余的重要方法，“关系模式有冗余问题就分解它”是规范化的基本原则。在分解过程中，分解成 BCNF 模式集的算法能保持无损分解，但不一定能保持函数依赖。而分解成 3NF 模式集的算法既能保持无损分解，又能保持函数依赖。

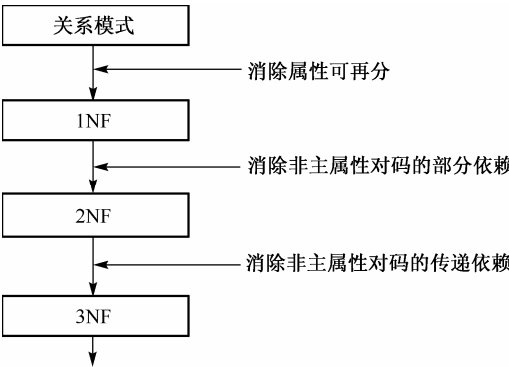


图 2.20 关系模式规范化

3. 模式规范化实例

例如，现有五张数据表，如表 2.2、表 2.3、表 2.4、表 2.5、表 2.6 所示。

表 2.2 学生表

学号	姓名	性别	年龄	系名	系所在楼	父母	
						父亲	母亲
2010118002	张强	男	18	计算机	3	张效民	马敏
2010118009	吴云轩	男	19	计算机	3	吴海	张小花
2010120002	周玉萍	女	18	新闻	5	周俊	王莉荔
2010120006	张志华	男	18	新闻	5	张裕	于小敏
2010119001	雷雨	女	19	英语	4	雷刚	丁菲菲

表 2.3 课程表

课程号	课程名
001	大学语文
002	英语（一）
003	普通物理
004	高等数学
005	计算机应用

表 2.4 成绩表

学号	课程号	成绩
2010118002	001	85
2010118002	005	90
2010118009	003	78
2010120002	003	82
2010119001	004	85

表 2.5 系表

系名	办公楼	系主任
计算机	3	马钢
新闻	5	蒋大强
化学	7	李牧
英语	4	吴敏
考古	1	韩长贵

表 2.6 备课表

课程名	教师	参考书
大学语文	方华	大学语文
普通物理	马湘云	物理概论
高等数学	李理	高等数学应用
计算机应用	周鹏	计算机导论
英语（一）	杨娟	大学英语

五张数据表对应的关系模式为：

学生表（学号，姓名，性别，年龄，系名，系所在楼，父母）

课程表（课程号，课程名）

成绩表（学号，课程号，成绩）

系表（系名，办公楼，系主任）

备课表（课程名，教师，参考书）

可以发现关系模式学生表（学号，姓名，性别，年龄，系所在楼，宿舍，父母）不满足第一范式，因为父母属性可再分。可将父母属性分解，得到新的学生表关系模式：学生表（学号，姓名，性别，年龄，系名，系所在楼，父亲，母亲）。

对关系模式系表（系名，办公楼，系主任）而言，其函数依赖关系如图 2.21 所示。满足第三范式要求。

对于关系模式备课表（课程名，教师，参考书）而言，假如每位老师可以教多门课，每门课可以有多可老师教，一门课可以有多个参考书。则其函数依赖关系如图 2.22 所示。满足第三范式要求。



图 2.21 关系模式“系表”的函数依赖关系

图 2.22 关系模式“备课表”的函数依赖关系

对关系模式课程表（课程号，课程名）而言，其函数依赖关系如图 2.23 所示。满足第三范式要求。

对关系模式成绩表（学号，课程号，成绩）而言，其函数依赖关系如图 2.24 所示。满足第三范式要求。



图 2.23 关系模式“课程表”的函数依赖关系

图 2.24 关系模式“成绩表”的函数依赖关系

对关系模式学生表（学号，姓名，性别，年龄，系名，系所在楼，父亲，母亲）而言，其函数依赖关系如图 2.25 所示。

可以发现，关系模式学生表（学号，姓名，性别，年龄，系名，系所在楼，父亲，母亲）满足第二范式，但不满足第三范式。将其分解为两个新的关系模式：

学生表（学号，姓名，性别，年龄，系名，父亲，母亲）

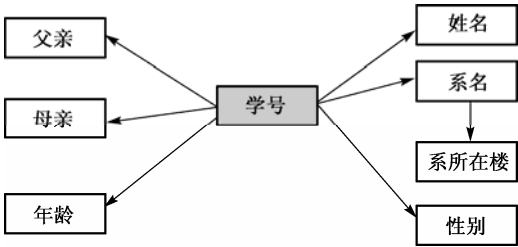


图 2.25 关系模式“学生表”的函数依赖关系

系楼表 (系名, 系所在楼)

两个关系模式均满足第三范式要求, 但关系模式系楼表 (系名, 系所在楼) 包含在关系模式系表 (系名, 办公楼, 系主任) 中, 所以可以删去。

最后得到的关系模型是:

学生表 (学号, 姓名, 性别, 年龄, 系名, 父亲, 母亲)

系表 (系名, 办公楼, 系主任)

成绩表 (学号, 课程号, 成绩)

课程表 (课程号, 课程名)

备课表 (课程名, 教师, 参考书)

习 题 2

一、填空题

1. 一个数据库应用系统的开发过程大致经过____、____、____、____、____和____等 7 个阶段。
2. 在数据库设计的需求分析阶段, 数据流图表达了数据与____的关系。
3. 在数据库设计中, 把数据需求写成文档, 它是各类数据描述的集合, 包括数据项、数据结构、数据流、数据存储、数据加工过程等的描述, 通常称为____。
4. 在数据库设计中, 用 E-R 图来描述信息结构但不涉及信息在计算机中的表示, 它产生于数据库设计的____阶段。
5. 数据库的逻辑模型设计阶段, 任务是将总体 E-R 模型转换成____。
6. 对于较复杂的系统, 概念设计阶段的主要任务是: 首先根据系统的各个局部应用画出各自对应的____, 然后再进行综合和整体设计, 画出____。
7. 由概念设计进入逻辑设计时, 原来的____联系或____联系通常不需要被转换为对应的基本表。
8. 数据库实施阶段包括两项重要的工作, 一项是数据的____, 另一项是应用程序的编码和调试。
9. 关系规范化的目的是____。
10. 对于非规范化的模式, 经过属性单一化形成____, 将 1NF 经过____转变为 2NF, 将 2NF 经过____转变为 3NF。
11. 在一个关系 R 中, 若属性集 X 函数决定属性集 Y , 则记作为____, 称 X 为____。
12. 在一个关系 R 中, 若存在“学号 \rightarrow 系号, 系号 \rightarrow 系主任”, 则隐含存在着____。
13. 在一个关系 R 中, 若 X 能够函数决定关系 R 中的每个属性, 并且 X 的任何真子集都不能决定函数 R 中的每个属性, 则称____为关系 R 的一个____。
14. 已知关系 $R(A, B, C, D)$ 和 R 上的函数依赖集 $F=\{A\rightarrow CD, C\rightarrow B\}$, R 属于第____范式。
15. 对关系进行规范化, 通常只要求规范化到____范式。
16. 关系数据库中的每个关系必须最低达到____范式, 该范式中的每个属性都是____的。
17. 我们通常按属性间的依赖情况来区分关系规范化程度, 其中第二范式属性的特点是____。第三范式的特性是____。

二、选择题

1. 数据流程图是 () 阶段的描述工具。
A. 可行性分析 B. 详细设计 C. 需求分析 D. 程序编码

2. 数据库需求分析时,数据字典的含义是()。
 - A. 数据库中所涉及的属性和文件的名称集合
 - B. 数据库中所涉及的字母、字符及汉字的集合
 - C. 数据库中所有数据的集合
 - D. 数据库中所涉及的数据流、数据项和文件等描述的集合
3. 下列不属于需求分析阶段工作的是()。
 - A. 分析用户活动
 - B. 建立 E-R 图
 - C. 建立数据字典
 - D. 建立数据流程图
4. 需求分析阶段要生成的文档是()和数据字典。
 - A. 数据流程图
 - B. E-R 图
 - C. UML 图
 - D. 功能模块图
5. 有关概念结构设计,下列说法正确的是()。
 - A. 概念结构设计是应用程序模块设计的基础
 - B. 概念结构设计只应用到数据字典
 - C. 概念结构设计与具体 DBMS 无关
 - D. 概念结构设计就是确定关系模式
6. 概念结构设计是整个数据库设计的关键,它通过对用户需求进行综合、归纳与抽象,形成一个独立于具体 DBMS 的()。
 - A. 数据模型
 - B. 概念模型
 - C. 层次模型
 - D. 关系模型
7. ()不属于数据库逻辑结构设计的任务。
 - A. 规范化
 - B. 模式分解
 - C. 模式合并
 - D. 创建视图
8. 数据库设计中,确定数据库存储结构,即确定关系、索引、聚簇、日志、备份等数据的存储安排和存储结构,这是数据库设计的()。
 - A. 需求分析阶段
 - B. 逻辑设计阶段
 - C. 概念设计阶段
 - D. 物理设计阶段
9. 数据库设计每个阶段都有自己的设计内容,“为哪些关系,在哪些属性上建什么样的索引”这一设计内容应该属于()设计阶段。
 - A. 概念设计
 - B. 逻辑设计
 - C. 物理设计
 - D. 全局设计
10. 在关系数据库设计中,对关系进行规范化处理,使关系达到一定的范式,例如达到 3NF,这是()阶段的任务。
 - A. 需求分析阶段
 - B. 概念设计阶段
 - C. 物理设计阶段
 - D. 逻辑设计阶段
11. E-R 图是数据库设计的工具之一,它适用于建立数据库的()。
 - A. 概念模型
 - B. 逻辑模型
 - C. 结构模型
 - D. 物理模型
12. 在数据库的概念设计中,最常用的数据模型是()。
 - A. 形象模型
 - B. 物理模型
 - C. 逻辑模型
 - D. 实体联系模型
13. 数据库物理设计完成后,进入数据库实施阶段,下列各项中不属于实施阶段的工作是()。
 - A. 建立数据库
 - B. 扩充功能
 - C. 加载数据
 - D. 系统调试
14. 若两个实体集间的联系是 1:M,则实现 1:M 联系的方法是()。
 - A. 在“M”端实体转换的关系中加入“1”端实体转换关系的码和联系的属性
 - B. 将“M”端实体转换关系的码和联系的属性加入到“1”端的关系中
 - C. 在两个实体转换的关系中,分别加入另一个关系的码
 - D. 将两个实体转换成一个关系
15. 下列有关 E-R 模型向关系模型转换的叙述中,不正确的是()。
 - A. 一个实体模型转换为一个关系模式
 - B. 一个 1:1 联系可以转换为一个独立的关系模式,也可以与联系的任意一端实体所对应的关系模式合并

- C. 一个 $1:N$ 联系可以转换为一个独立的关系模式, 也可以与联系的任意一端实体所对应的关系模式合并
D. 一个 $M:N$ 联系转换为一个关系模式
16. 设计子模式属于数据库设计的 ()。
A. 需求分析 B. 概念设计 C. 逻辑设计 D. 物理设计
17. 概念结构设计的主要目标是产生数据库的概念结构, 该结构主要反映 ()。
A. 应用程序员的编程需求 B. DBA 的管理信息需求
C. 数据库系统的维护需求 D. 企业的信息需求
18. 数据库设计人员和用户之间沟通信息的桥梁是 ()。
A. 程序流程图 B. 实体联系图 C. 模块结构图 D. 数据结构图
19. 有两个不同的实体集, 它们之间存在着一个 $1:1$ 联系和一个 $M:N$ 联系, 那么根据 E-R 模型转换成关系模型的规则, 转换成的关系模式个数为 ()。
A. 2 个 B. 3 个 C. 4 个 D. 5 个
20. 如果有 10 个不同的实体集, 它们之间存在着 12 个不同的二元联系 (二元联系是指两个实体集之间的联系), 其中 3 个 $1:1$ 联系, 4 个 $1:N$ 联系, 5 个 $M:N$ 联系, 那么根据 ER 模型转换成关系模型的规则, 这个 ER 结构转换成的关系模式个数为 ()。
A. 14 个 B. 15 个 C. 19 个 D. 22 个
21. 在数据库的概念设计中, 最常用的数据模型是 ()。
A. 对象模型 B. 物理模型 C. 逻辑模型 D. 实体联系模型
22. 从 E-R 模型向关系模型转换时, 一个 $M:N$ 联系转换为关系模式时, 该关系模式的关键字是 ()。
A. M 端实体的关键字 B. N 端实体的关键字
C. 两端实体关键字的组合 D. 重新选取其他属性
23. 数据库逻辑设计的主要任务是 ()。
A. 建立 E-R 图 B. 创建数据库说明 C. 建立数据流图 D. 建立数据索引
24. 关系数据规范化是为了解决关系数据中 () 问题而引入的。
A. 插入、删除和数据冗余 B. 提高查询速度
C. 减少数据操作的复杂性 D. 保证数据的安全性和完整性
25. 候选关键字中的属性称为 ()。
A. 非主属性 B. 主属性 C. 复合属性 D. 关键属性
26. 关系数据库的规范化理论主要解决的问题是 ()。
A. 如何构造合适的数据逻辑结构 B. 如何构造合适的数据物理结构
C. 如何构造合适的应用程序界面 D. 如何控制不同用户的数据操作权限
27. 任何一个满足 $2NF$ 但不满足 $3NF$ 的关系模式都不存在 ()。
A. 主属性对候选键的部分依赖 B. 非主属性对候选键的部分依赖
C. 主属性对候选键的传递依赖 D. 非主属性对候选键的传递依赖
28. 设计性能较优的关系模式称为规范化, 规范化的主要理论依据是 ()。
A. 关系规范化理论 B. 关系运算理论 C. 代数运算理论 D. 数理逻辑理论

三、简答题

1. 什么是软件工程? 软件工程的基本目标是什么?
2. 什么是软件生命周期? 软件生命周期一般可分为哪几个阶段?

3. 基于数据库系统生存期的数据库设计分为哪几个阶段？请简述。
4. 什么是数据库设计？数据库设计过程的输入和输出有哪些内容？
5. 数据库设计的规划阶段应做些什么事情？
6. 数据库设计的需求分析阶段是如何实现的？目标是什么？
7. 评审在数据库设计中有什么重要作用？为什么允许设计过程中有多次的回溯与反复？
8. 数据字典的内容和作用是什么？
9. 概念设计的具体步骤是什么？
10. 试述采用 E-R 方法的数据库概念设计的过程。
11. 试述逻辑设计阶段的主要内容。
12. 数据系统投入运行后，有哪些维护工作？
13. 规范化理论对数据库设计有什么指导意义？
14. 什么是函数依赖？
15. 什么是 1NF、2NF、3NF？它们之间有什么关系？

第 3 章 数据库安全

随着计算机应用的普及，越来越多的关键业务系统运行在数据库平台上。数据库中的数据作为企业的资源发挥着越来越重要的作用，同时也成为黑客攻击的主要目标。而如何保证数据库自身的安全，已成为现代数据库系统需要解决的主要问题之一。数据库安全不能仅依赖某一个环节就完成所有的安全防范措施，安全是多个环节共同配合的结果。一个安全的系统需要数据库的安全、操作系统的安全、网络的安全、应用系统自身的安全共同完成。

3.1 数据库安全

Verizon 在 2009 年基于对 2 亿 8500 万次累计数据破坏行为进行分析的调查报告显示，对数据库系统的尝试破坏行为占 75% 左右。主要原因在于：一方面由于数据库系统承载关键业务数据，而这些数据牵涉到企业各个方面的信息，具有重要的价值；另一方面由于数据库系统通常比较复杂，且其对连续性、稳定性有高标准的要求，安全管理人员在缺乏相关知识的情况下会使数据库安全管理工作滞后于业务需求的满足。数据库系统作为信息的聚集体，是计算机信息系统的核心部件，其安全性至关重要。因此，如何有效地保证数据库系统的安全，实现数据的保密性、完整性和有效性至关重要。

3.1.1 系统安全的主要风险

数据库系统在实际应用中存在来自各方面的安全风险，由安全风险最终引起安全问题，常见的数据库系统安全风险来自四个方面。

1. 来自操作系统的风险

来自操作系统的风险主要集中在病毒、后门、数据库系统和操作系统的关联性方面。首先在病毒方面，操作系统中可能存在的木马程序对数据库系统构成极大的威胁。其次在操作系统的后门方面，许多数据库系统的特征参数尽管方便了数据库管理员，但也为数据库服务器主机操作系统留下了后门，这使得黑客可以通过后门访问数据库。

数据库系统和操作系统之间带有很强的关联性。操作系统具有文件管理功能，能够对各类文件包括数据库文件的授权进行读/写和执行等操作，而且操作系统的监控程序能进行用户登录和口令鉴别的控制，如果操作系统允许用户直接存取数据库文件，则在数据库系统中采取最可靠的安全措施也没有用。

2. 来自管理的风险

安全意识薄弱，对信息网络安全重视不够，安全管理措施不落实，导致安全事件的发生，这些都是当前安全管理工作存在的主要问题。从已发生安全事件的原因看，占前两位的分别是“未修补软件安全漏洞”和“登录密码过于简单或未修改”，这表明用户缺乏相关的安全防范意识和基本的安全防范常识。再比如，数据库系统可用但并未正确使用的安全选项、危险的默认设置、给用户不适当的权限、对系统配置未经授权的改动等都会危害数据库的安全。

3. 来自用户的风险

用户的风险主要表现在用户账号、作用和对特定数据库目标的操作许可，例如，对表单和存储步骤的访问。因此必须对数据库系统做范围更广的安全分析，找出所有可能领域内的潜在漏洞，包括销售商提供的相关风险软件的 BUG、缺少操作系统补丁、脆弱的服务和选择不安全的默认配置等。另外密码长度不够，对重要数据的非法访问，以及窃取数据库内容等恶意行动也都是来自用户的风险。

4. 来自数据库系统内部的风险

虽然绝大多数常用的关系数据库系统具有强大的功能，产品非常成熟。但在系统内部安全性方面，特别是那些重要的安全特征方面，很多关系数据库系统并不够成熟。

3.1.2 数据库安全的特征

数据库安全是指保护数据库以防止非法用户的越权使用、窃取、更改或破坏数据。数据库安全涉及很多层面，数据库安全包含两层含义：第一层是指系统运行安全，第二层是指系统信息安全。

数据库系统的安全特性主要是针对数据而言的，包括数据独立性、数据安全性、数据完整性、并发控制、故障恢复等几个方面。

1. 数据独立性

理论上数据库系统的数据独立性包括物理独立性和逻辑独立性两个方面。这两种数据独立性都要靠 DBMS 来实现。到目前为止，物理独立性已经能基本实现，而逻辑独立性实现起来比较困难，数据结构一旦发生变化，一般情况下相应的应用程序都要做一些修改。这也是数据库系统结构复杂的一个重要原因。

2. 数据安全性

一个数据库能否实现防止无关人员得到他不应该知道的数据，是数据库是否实用的一个重要指标。如果一个数据库对所有的人都公开数据，那么这个数据库就不是一个可靠的数据库。操作系统中的对象一般是文件，而数据库支持的应用要求更为精细。通常比较完整的数据库会采取以下安全措施：

- ◆ 将数据库中需要保护的部分与其他部分相隔。
- ◆ 采用授权规则，如账户、口令和权限控制等访问控制方法。
- ◆ 对数据进行加密后存储于数据库。

3. 数据完整性

数据完整性包括数据的正确性、有效性和一致性。正确性是指数据的输入值与数据表对应域的类型一样；有效性是指数据库中的理论数值满足现实应用中对该数值段的约束；一致性是指不同用户使用的同一数据应该是一样的。保证数据的完整性，需要防止合法用户使用数据库时向数据库中加入不合语义的数据。

4. 并发控制

如果数据库应用要实现多用户共享数据，就可能在同一时刻多个用户要存取数据，这种事件叫做并发事件。当一个用户取出数据进行修改，在修改存入数据库之前如有其他用户再

取此数据，那么读出的数据就是不正确的。这时就需要对这种并发操作施行控制，排除和避免这种错误的发生，保证数据的正确性。

5. 故障恢复

如果数据库系统运行时出现物理或逻辑上的错误，系统能尽快地恢复正常，这就是数据库系统的故障恢复功能。数据库管理系统应提供一套方法，及时发现故障和修复故障，从而防止数据被破坏。

3.2 数据库的安全层次

数据库系统的安全除依赖自身内部的安全机制外，还与外部网络环境、应用环境、从业人员素质等因素息息相关，数据库的安全机制如图3.1所示。

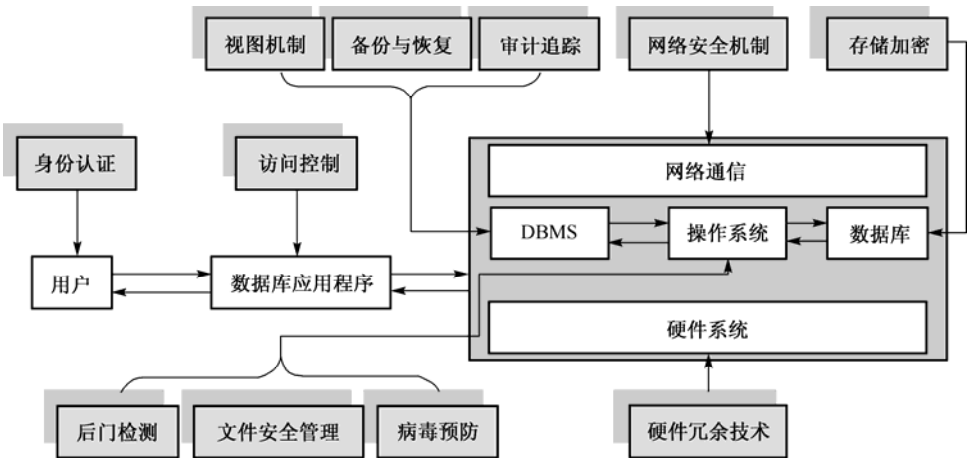


图 3.1 数据库安全机制

从广义上讲，数据库系统的安全框架可以划分为网络系统层、宿主操作系统层和数据库管理系统层三个层次。这三个层次构筑成数据库系统的安全体系，防范的重要性也逐层加强，从外到内、由表及里保证数据的安全。为了实现三个层次的安全性，必须在以下几个层面做好安全措施。

- 1. 物理层面
重要的计算机系统必须在物理上受到保护，以防止入侵者强行进入或暗中潜入。
- 2. 人员层面
数据库系统的建立、应用和维护等工作，一定要由可信的合法用户来操作和管理。
- 3. 操作系统层面
进入数据库系统，首先要经过操作系统，如果操作系统的安全性差，数据库将面临重大的威胁。
- 4. 网络层面
由于几乎所有网络上的数据库系统都允许通过终端或网络进行远程访问，所以网络的安全和操作系统的安

5. 数据库系统层面

数据库系统应该有完善的访问控制机制，以防止非法用户操作。

3.2.1 网络系统层次安全

从广义上讲，数据库的安全首先依赖于网络系统。随着 Internet 的发展和普及，越来越多的公司将其核心业务向互联网转移，各种基于网络的数据库应用系统面向网络用户提供各种信息服务。可以说网络系统是数据库应用的外部环境和基础，数据库系统要发挥其强大作用离不开网络系统的支持，网络系统的安全是数据库安全的第一道屏障，外部入侵首先就是从入侵网络系统开始的。

网络入侵试图破坏信息系统的完整性、机密性和可用性。这些安全威胁无处不在，因此必须采取有效的措施来保障系统的安全。从技术角度讲，网络系统层次的安全防范技术有很多种，包括加密技术、认证技术、数字签名技术、防火墙技术、入侵检测技术等。

3.2.2 操作系统层次安全

操作系统是大型数据库系统的运行平台，为数据库系统提供一定程度的安全保护。目前操作系统平台大多数集中在 Windows NT 和 UNIX，安全级别通常为 C2 级。一个安全的操作系统应该具有访问控制，内存管理，对象重用，审计，加密数据传送，加密文件系统，安全进程间通信机制等功能。主要安全技术有操作系统安全策略、安全管理策略、数据安全等方面。

操作系统安全策略用于配置本地计算机的安全设置，包括密码策略、账户锁定策略、审核策略、IP 安全策略、用户权利指派、加密数据的恢复及其他安全选项。具体可以体现在用户账户、口令、访问权限、审计等方面。

安全管理策略是指网络管理员对系统实施安全管理所采取的方法及策略。针对不同的操作系统和网络环境采取的安全管理策略也不尽相同，其核心是保证服务器的安全和分配好各类用户的权限。

数据安全主要体现在以下几个方面：数据加密技术、数据备份、数据存储的安全性、数据传输的安全性等。可以采用的技术很多，主要有 Kerberos、IPSec、SSL、TLS、VPN 等技术。

3.2.3 数据库管理系统层次安全

数据库系统的安全性很大程度上依赖于数据库管理系统。如果数据库管理系统安全机制非常强大，那么数据库系统的安全性能就较好。目前市场上流行的是关系模型数据库管理系统，其安全性功能很弱，这就导致数据库系统的安全性存在一定的威胁。

由于数据库系统在操作系统下都是以文件形式进行管理的，因此入侵者可以直接利用操作系统的漏洞窃取数据库文件，或者直接利用操作系统工具来非法伪造、篡改数据库文件内容。数据库管理系统层次安全技术主要是用来解决这一问题，保证在网络安全层次和操作系统安全层次被突破的情况下，仍能保障数据库数据的安全。其采用的主要技术就是通过数据库管理系统对数据库文件进行加密处理，即使数据不幸泄露或者丢失，也难以被破译和阅读。

3.3 数据库安全技术

3.3.1 数据库加密技术

对于一些重要的机密数据,例如,一些金融数据、商业秘密、游戏网站玩家的虚拟财产,加密可以保护数据的安全。对数据库安全性的威胁有时候是来自于网络内部,一些内部用户可能非法获取用户名和密码,或利用其他方法越权使用数据库,甚至可以直接打开数据库文件来窃取或篡改信息。因此,有必要对数据库中存储的重要数据进行加密处理,以实现数据存储的安全保护。

数据库加密系统能够有效地保证数据安全,即使黑客窃取了关键数据,仍然难以得到所需的信息。另外,数据库加密以后,不需要了解数据内容的系统管理员不能见到明文,大大提高了关键数据的安全性。

1. 数据库加密的必要性

据有关资料报道,80%的计算机犯罪来自系统内部。在传统的数据库系统中,数据库管理员具有很高的数据管理权力,他既负责各项系统管理工作,例如,资源分配、用户授权、系统审计等,又可查询数据库中的一切信息。为此,不少系统希望通过一定的方法削弱系统管理员的权利来达到数据库安全的目的。

数据库加密以后,各用户(或用户组)的数据由用户用自己的密钥加密,数据库管理员无法进行正常解密,从而保证了用户信息的安全。另外,通过加密,数据库的备份内容成为密文,从而能减少因备份介质失窃或丢失而造成的损失。由此可见,数据库加密对于企业内部安全管理,也是不可或缺的。数据库加密对数据库系统效率的影响很小,如果数据加/解密运算在数据库客户端进行,对数据库服务器的负载及系统运行几乎没有影响。

2. 加密的基本要求

一个好的数据库加密系统应该满足以下基本要求。

(1) 合适的加密层次

① 基于文件的数据库加密技术

把数据库文件作为整体,对整个数据库文件加密,形成密文来保证数据的真实性和完整性。利用这种方法,数据的共享是通过用户用解密密钥对整个数据库文件进行解密来实现的,但多方面的缺点限制了这一方法的实际应用。首先,数据修改的工作将变得十分困难,需要进行解密、修改、复制和加密四个操作步骤,极大地增加了系统的时空开销;其次,即使用户只是需要查看某一条记录,也必须将整个数据库文件解密,这样无法实现对文件中不需要让用户知道的信息的控制。

② 基于记录的数据库加密技术

一般而言,数据库系统中每条记录独立完整地存储了一个实体的数据,因此,基于记录的数据库加密技术是最常用的加密手段。这种方法的基本思路是:在不同密钥的作用下,将数据库的每一个记录加密成密文并存放于数据库文件中;记录的查找是通过将需要查找的值加密成密码文后进行的。然而基于记录的数据库保护有一个缺点,就是在解密一个记录的数据时,无法实现对这个记录中不需要的字段不解密,在选择某个字段的某些记录时,如果不对含有这个字段的所有记录解密就无法进行选择。

③ 基于字段的数据库加密技术

在目前条件下,最好的加密/解密粒度是字段数据。如果以文件或列为单位进行加密,必然会造成密钥的反复使用,从而降低加密系统的可靠性或者因加解密时间过长而无法使用。只有以记录的字段数据为单位进行加解密,才能适应数据库操作,同时进行有效的密钥管理并完成“一次一密”的密码操作。

(2) 密钥动态管理

数据库客体之间隐含着复杂的逻辑关系,一个逻辑结构可能对应着多个数据库物理客体,所以数据库加密不仅密钥量大,而且组织和存储工作比较复杂,需要对密钥实现动态管理。

(3) 合理处理数据

合理处理数据包括几方面的内容,首先要恰当地处理数据类型,否则 DBMS 将会因加密后的数据不符合定义的数据类型而拒绝加载;其次,需要处理数据的存储问题,实现数据库加密后,基本上不增加空间开销。在目前条件下,数据库关系运算中的匹配字段,如表间连接码、索引字段等数据不宜加密。

(4) 不影响合法用户的操作

加密系统影响数据操作响应时间应尽量短,现阶段平均延迟时间不应超过 1/10 秒。此外,对数据库的合法用户来说,数据的录入、修改和检索操作应该是透明的,不需要考虑数据的加密/解密问题。

3. 数据库加密层次

数据库数据的加密可在三个不同层次实现,这三个层次分别是 OS、DBMS 内核层和 DBMS 外层。在 OS 层,由于无法辨认数据库文件中的数据关系,从而无法产生合理的密钥,也无法进行合理的密钥管理和使用。所以,在 OS 层对数据库文件进行加密,对于大型数据库来说,目前还难以实现。在 DBMS 内核层实现加密是指数据在物理存取之前完成加密/解密工作,这种方式要求 DBMS 和加密器(硬件或软件)之间的接口需要 DBMS 开发商的支持。这种加密方式的优点是加密功能强,并且加密功能几乎不会影响 DBMS 的功能。其缺点是加密/解密运算在服务器端进行,会加重数据库服务器的负载。

比较实际的做法是在 DBMS 外层层加密,DBMS 外层层加密是将数据库加密系统做成 DBMS 的一个工具。采用这种加密方式时,加密/解密运算可以放在客户端进行,其优点是不会加重数据库服务器的负载并可实现网上传输加密,缺点是加密功能会受一些限制。

作为一种通过加密方式来保护数据的专门系统,数据库加密系统并不是一个数据库应用系统,而是一个将 DBMS 部分功能、加密器和密钥管理三者紧密结合起来的系统。显然数据库加密系统实际上主要实现的还是加密器的功能,但它既可工作于数据库应用系统与 DBMS 之间,又可作为一个独立工作的系统直接与 DBMS 交互。

3.3.2 存取管理技术

存取管理技术主要包括用户认证技术和访问控制技术两方面。用户认证技术包括用户身份验证和用户身份识别技术。访问控制包括数据的浏览控制和修改控制。浏览控制是为了保护数据的保密性,而修改控制是为了保护数据的正确性和提高数据的可信性。在一个资源共享的环境中,访问控制非常重要。

1. 用户认证技术

用户认证技术是系统提供的最外层安全保护措施。通过用户身份验证,可以阻止未授权用户的访问,而通过用户身份识别,可以防止用户的越权访问。

2. 访问控制技术

访问控制技术是通过某种途径允许或限制用户访问能力及范围的一种方法。访问控制的目的是使用户只能对经过授权的相关数据库进行操作。访问控制从计算机系统的处理功能方面对数据提供保护,是数据库系统内部对已经进入系统的用户的访问控制,它是数据库安全系统中的核心技术,也是最有效的安全手段,限制访问者和执行程序可以进行的操作,这样可防止安全漏洞隐患。DBMS 中对数据库的访问控制是建立在操作系统和网络的安全机制基础之上的。只有被授权的用户才有对数据库中的数据进行输入、删除、修改和查询等权限,通常有以下两种基本的访问控制方法。

(1) 按功能模块对用户授权

每个功能模块对不同用户设置不同权限,如可设置为无权进入本模块、仅可查询、可更新可查询、全部功能可使用等,而且功能模块名、用户名与权限编码可保存在同一数据库。

(2) 基于角色的访问控制

通常为了提高数据库的信息安全访问,用户在进行正常的访问前服务器往往都需要认证用户的身份、确认用户是否被授权。为了加强身份认证和访问控制,适应对大规模用户和海量数据资源的管理,通常 DBMS 主要使用的是基于角色的访问控制 RBAC (Role Based Access Control)。所谓“角色”就是一个或一群用户在组织内可执行操作的集合。角色可以根据组织中不同的工作创建,然后根据用户的职责分配,用户可以轻松地进行角色转换。基于角色的访问控制技术根据用户在组织内所处的角色进行访问授权与控制,只有系统管理员有权定义和分配角色。用户与客体无直接联系,只有通过角色才享有该角色所对应的权限,从而访问相应的客体。

3.3.3 安全审计技术

数据库安全涉及入侵防御、账号管理、访问控制、安全审计、防病毒、评估加固等多个方面。但是,企业内部人员的违规行为与传统的攻击行为不同,对内部的违规行为无法利用攻击机理和漏洞机理进行分析。因此,要防止内部的违规行为,就需要在内部建设审计系统,通过对操作行为的分析,实现对违规行为的及时响应和追溯。

1. 安全审计的概念

数据库安全审计系统主要用于监视并记录对数据库服务器的各类操作行为,通过对网络数据的分析,实时地、智能地解析对数据库服务器的各种操作,并记入审计数据库中以便日后进行查询、分析、过滤,实现对目标数据库系统的用户操作的监控和审计。它可以监控和审计用户对数据库中的数据库表、视图、序列、包、存储过程、函数、库、索引、同义词、快照、触发器等的创建、修改和删除等,分析的内容可以精确到 SQL 操作语句一级。它还可以根据设置的规则,智能地判断出违规操作数据库的行为,并对违规行为进行记录、报警。

由于数据库安全审计系统是以网络旁路的方式工作于数据库主机所在的网络,因此它可以在不改变数据库系统的设置的情况下对数据库的操作实现跟踪记录、定位,实现数据库的在线监控,及时地发现网络上针对数据库的违规操作行为并进行记录、报警和实时阻断,有

效地弥补现有应用业务系统在数据库安全使用上的不足，为数据库系统的安全运行提供了有力保障。一个性能良好的审计系统必须具有以下基本特征：

- ◆ 能够制定确保系统安全审计策略正确实施的规章制度及措施；
- ◆ 能够对重要服务器的访问行为进行审计；
- ◆ 能够包括事件的日期、时间、类型、主体标识、客体标识和结果等；
- ◆ 能够定期对审计记录进行审查分析，对可疑行为及违规操作，采取相应的措施，并及时报告。

2. 常见的数据库安全审计技术

常见的安全审计技术主要有四类，分别是基于日志的审计技术、基于代理的审计技术、基于网络监听的审计技术、基于网关的审计技术。

(1) 基于日志的审计技术

基于日志的审计技术通常是通过数据库自身功能实现，Oracle、DB2 等主流数据库均具备自身审计功能。通过配置数据库的自审计功能，即可实现对数据库的审计，其典型部署如图3.2 所示。

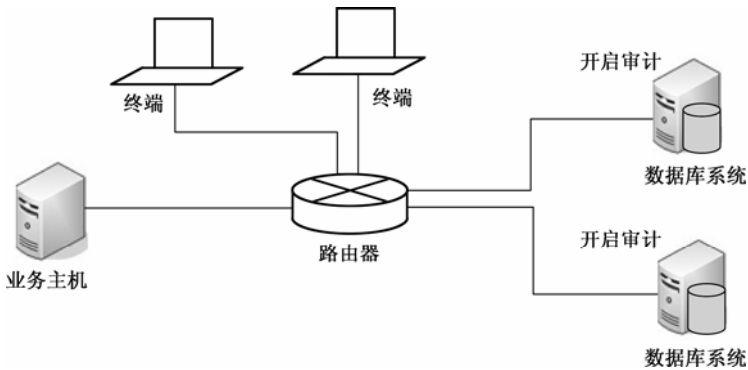


图 3.2 日志审计技术部署

该技术依托于现有数据库管理系统对网络操作及本地操作数据库的行为进行审计，具有很好的兼容性。但这种审计技术的缺点也比较明显，首先，开启自身日志审计对数据库系统的性能有影响，特别是在大流量情况下影响较大；其次，日志审计记录的细粒度差，缺少源 IP、SQL 语句等关键信息，审计溯源效果不好；最后就是日志审计需要到每一台被审计主机上进行配置和查看，较难进行统一的审计策略配置和日志分析。

(2) 基于代理的审计技术

基于代理的审计技术是通过在数据库系统上安装相应的审计代理（Agent），在代理上实现审计策略的配置和日志的采集，常见的产品如 Oracle 公司的 Oracle Audit Vault、IBM 公司的 DB2 Audit Management Expert Tool，以及第三方安全公司提供的产品，其典型部署如图3.3 所示。

与日志审计技术最大的不同是需要在被审计主机上安装代理程序。代理审计技术在审计粒度上优于日志审计技术，但在性能上的损耗大于日志审计技术。由数据库厂商提供的代理审计类产品对自有数据库系统具有良好的兼容性，但是在跨数据库系统的支持上，存在一定的兼容风险。同时在引入代理审计后，原数据库系统的稳定性、可靠性等方面会有一些影响。

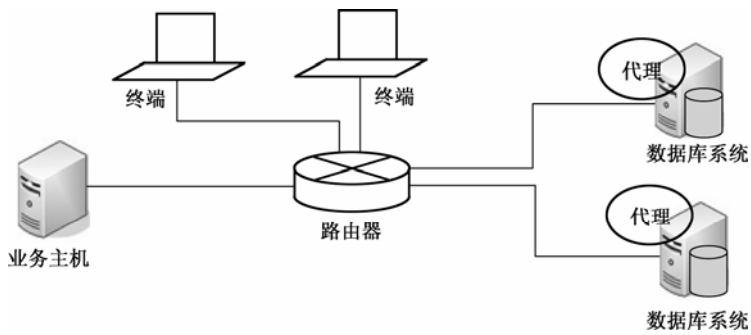


图 3.3 代理审计技术部署

(3) 基于网络监听的审计技术

基于网络监听的审计技术通过将对数据库系统的访问流镜像到交换机某一个端口，然后由专用硬件设备对该端口流量进行分析和还原，从而实现对数据库访问的审计，其典型部署如图3.4所示。

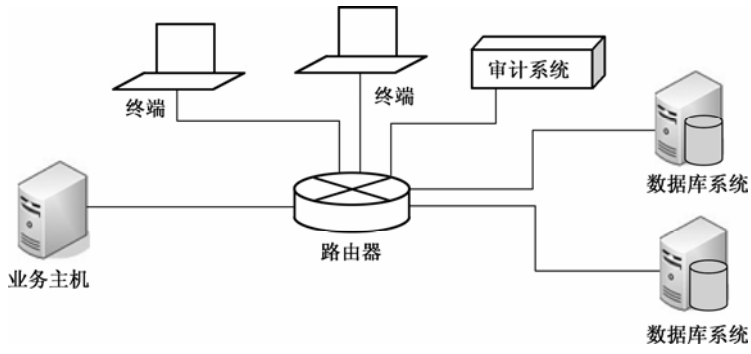


图 3.4 网络监听审计技术部署

虽然在针对加密协议时只能实现到会话级别审计（即可以审计到时间、源 IP、源端口、目的 IP、目的端口等信息），而无法对内容进行审计。但该技术最大的优点就是与现有数据库系统无关，易部署、无风险，部署过程不会给数据库系统带来性能上的负担，故网络监听审计技术在实际的数据库审计项目中应用非常广泛。

(4) 基于网关的审计技术

基于网关的审计技术是通过在数据库系统前部署网关设备，通过在线截获并转发到数据库的流量而实现审计，其典型部署如图3.5所示。

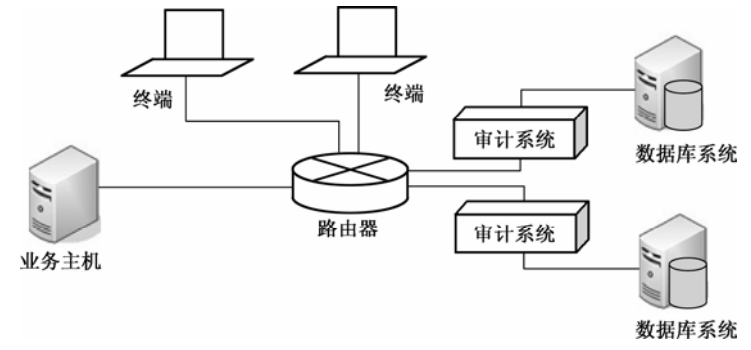


图 3.5 网关审计技术部署

该技术起源于安全审计在互联网审计中的应用，由于数据库环境存在流量大、业务连续性要求高、可靠性要求高的特点，与互联网环境大相径庭，所以网关审计技术主要应用于对数据库运维的审计，而不是对所有对数据库访问行为的审计。

3. 构建审计系统应注意的问题

通过对以上四种技术的分析，在进行数据库审计技术方案的选择时，要注意以下几点。

① 业务保障原则：安全建设的根本目标是能够更好的保障网络上承载的业务，在保证安全的同时，必须保障业务的正常运行和运行效率。

② 结构简化原则：安全建设的直接目的和效果是要将整个网络变得更加安全，简单的网络结构便于整个安全防护体系的管理、执行和维护。

③ 生命周期原则：安全建设不仅仅要考虑静态设计，还要考虑不断的变化，系统应具备适度的灵活性和扩展性。

3.3.4 备份与恢复

计算机在运行过程中会出现多种异常现象，诸如磁盘故障、电源故障、软件故障、灾害故障及人为破坏等。一旦发生这些故障，就有可能造成数据的丢失。而数据库管理系统的备份和恢复机制能保证数据库系统出现故障时，能够将数据库系统还原到正确状态。

1. 故障的种类

数据库系统中发生的故障大致可以归结为以下几类。

(1) 事务故障

所谓事务是用户定义的一个操作序列，这些操作要么全做要么全不做，是一个不可分割的工作单位。事务具有四个特性：原子性、一致性、隔离性和持续性。

原子性事务中包括的诸操作要么都做，要么都不做；一致性保证如果数据库系统运行中发生故障，有些事务尚未完成就被迫中断，系统将事务中对数据库的所有已完成的操作全部撤销，滚回到事务开始时的一致状态；隔离性保证一个事务的执行不能被其他事务干扰，即一个事务内部的操作及使用的数据对其他并发事务是隔离的；持续性也称永久性，指一个事务一旦提交，它对数据库中数据的改变就应该是永久性的，接下来的其他操作或故障不应该对其执行结果有任何影响。

但事务的四个特性可能由于多个事务并行运行时，不同事务的操作交叉执行或者事务在运行过程中被强行停止等因素而受到破坏，这就是事务故障。事务故障意味着事务没有达到预期的终点，因此数据库可能处于不正确状态。

(2) 系统故障

系统故障是指造成系统停止运转，必须重新启动系统的任何事件。例如，特定类型的硬件故障、操作系统故障、DBMS 代码错误、数据库服务器出错以及其他自然原因等。发生系统故障时，一些尚未完成的事务的结果可能已送入物理数据库，有些已完成的事务可能有一部分甚至全部留在缓冲区，尚未写回到磁盘上的物理数据库中，从而造成数据库可能处于不正确的状态。

(3) 介质故障

系统故障常称为软故障，介质故障称为硬故障。硬故障指外存故障，如磁盘损坏、磁头碰撞，瞬时强磁场干扰等。这类故障将破坏数据库全部或部分内容，并影响正在存取这部分

数据的所有事务。这类故障比前两类故障发生的可能性小得多，但破坏性最大，有时会造成数据的无法恢复。

(4) 计算机病毒

计算机病毒是一种人为的故障或破坏，它是由一些人恶意的编制的计算机程序。这种程序与其他程序不同，它可以像病毒一样进行繁殖和传播，并造成对计算机系统包括数据库系统的破坏。

(5) 用户操作错误

在某些情况下，由于用户有意或无意的操作也可能删除数据库中的有用数据或加入错误数据，这同样会造成一些潜在的故障。

2. 数据恢复的基本原理

数据备份与恢复是实现数据库系统安全运行的重要技术。数据库系统总免不了发生系统故障，一旦系统发生故障，重要数据就可能遭到损坏。为防止重要数据的丢失或损坏，数据库管理员应及时做好数据库备份，这样当系统发生故障时，管理员就能利用已有的数据备份，把数据库恢复到原来的状态，以便保持数据的完整性和一致性。

一般来说，数据库备份常用的备份方法有静态备份（关闭数据库时将其备份）、动态备份（数据库运行时将其备份）和逻辑备份（利用软件技术实现原始数据库内容的镜像）等；而数据库恢复则可以通过数据库备份文件、磁盘镜像和数据库在线日志三种方式来完成。

3. 数据备份

数据备份就是数据库管理员（DBA）定期地将整个数据库复制到其他存储介质（如磁带或非数据库所在的另外磁盘）上形成备用文件的过程。这些备用的数据文件称为后备副本或后援副本。当数据库遭到破坏后可以将后备副本重新装入，但重装后备副本只能将数据库恢复到转储时的状态，要想恢复到故障发生时的状态，必须重新运行自转储以后的所有更新事务。

例如，在图3.6中，系统在 Ta 时刻停止运行事务进行数据库转储，在 Tb 时刻转储完毕，得到 Tb 时刻的数据库一致性副本。系统运行到 Tc 时刻发生故障。为恢复数据库，首先由 DBA 重装数据库后备副本，将数据库恢复至 Tb 时刻的状态，然后重新运行自 Tb 时刻至 Tc 时刻的所有更新事务，这样就把数据库恢复到故障发生前的一致状态。

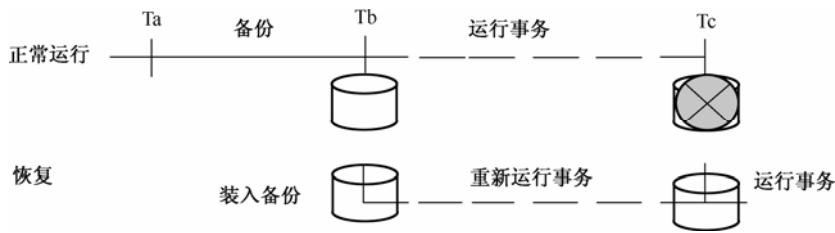


图 3.6 备份和恢复

数据备份十分耗费时间和资源，不能频繁进行。数据库管理员（DBA）应该根据数据库使用情况确定一个适当的转储周期和转储策略。

根据备份过程数据库是否关闭，备份可分为静态备份和动态备份。静态备份是指在备份过程中，系统不运行其他事务，专门进行数据转储工作。而动态备份是指在备份过程中，允许其他事务对数据库进行存取或修改的转储方式。由于动态备份可以动态地进行，这样后备

副本中存储的就可能是过时的数据。因此，有必要把转储期间各事务对数据库的修改活动登记下来，建立日志文件（Log File），使得后援副本加上日志文件能够把数据库恢复到某一时刻的正确状态。

根据备份过的数据量的多少，备份又可分海量备份和增量备份。海量备份每次备份全部数据，海量备份能够得到后备副本，利用后备副本能够比较方便地进行数据恢复工作。但对于数据量大和更新频率高的数据库，不适合频繁地进行海量转储。而增量备份每次只备份上一次转储后更新过的数据，增量备份适用于数据库较大、但是事务处理又十分频繁的数据库系统。

由于数据备份可在动态和静态，海量和增量下进行，因此数据备份方法可以分为4类：动态海量备份、动态增量备份、静态海量备份和静态增量备份。

4. 日志文件

日志文件是用来记录对数据库更新操作的文件。不同的数据库系统采用的日志文件格式基本相同。日志文件能够用来进行事务故障恢复、系统故障恢复，并能够协助后备副本进行介质故障恢复。当数据库文件毁坏后，可重新装入后援副本把数据库恢复到转储结束时刻的正确状态，再利用建立的日志文件，可以把已完成的事务进行重做处理，而对于故障发生时尚未完成的事务则进行撤销处理，这样就可把数据库恢复到故障前某一时刻的正确状态。

日志文件主要有以记录为单位的日志文件和以数据块为单位的日志文件。

以记录为单位的日志文件需要登记的内容包括：每个事务的开始标记、结束标记和所有更新操作，这些内容均作为日志文件中的一个日志记录。对于更新操作的日志记录，其内容主要包括：事务标识、操作的类型、操作对象、更新前数据的旧值及更新后数据的新值。

以数据块为单位的日志文件包括事务标识和更新的数据块。由于更新前后的各数据块都放入了日志文件，所以操作的类型和操作对象等信息就不放入日志记录。

为保证数据库的可恢复性，登记日志文件时必须遵循两条原则：一是登记的次序严格按事务执行的时间次序；二是必须先写日志文件，后写数据库。

5. 数据库恢复策略

当系统运行过程中发生故障时，利用数据库后备副本和日志文件就可以将数据库恢复到故障前的某个一致性状态。不同故障其恢复策略和方法也不一样。

（1）事务故障的恢复

当发生事务故障时，恢复子系统应利用日志文件撤销（UNDO）此事务已对数据库进行的修改。事务故障的恢复通常是由系统自动完成的，用户感知不到系统是如何进行事务恢复的。

系统的恢复步骤是：

① 反向扫描文件日志（即从最后向前扫描日志文件），查找该事务的更新操作。

② 对该事务的更新操作执行逆操作。即将日志记录中“更新前的值”写入数据库。若记录中是插入操作，则相当于做删除操作；若记录中是删除操作，则做插入操作；若是修改操作，则相当于用修改前的值代替修改后的值。

③ 重复执行①和②，恢复该事务的其他更新操作，直至读到该事务的开始标记，事务故障恢复就完成了。

（2）系统故障的恢复

系统故障恢复操作要撤销故障发生时未完成的事务，重做已完成的事务。系统故障的恢复是由系统在重新启动时自动完成的，不需要用户干预。

系统的恢复步骤是:

① 正向扫描日志文件 (即从头扫描日志文件), 找出在故障发生前已经提交的事务 (这些事务既有 **BEGIN** 或 **TRANSACTION** 记录, 也有 **COMMIT** 或 **ROLLBACK** 记录), 将其事务标记记入重做 (**REDO**) 队列。同时找出故障发生时尚未完成的事务 (这些事务只有 **BEGIN** 或 **TRANSACTION** 记录, 无相应的 **COMMIT** 或 **ROLLBACK** 记录), 将其事务标记记入撤销 (**UNDO**) 队列。

② 对撤销队列中的各个事务进行撤销 (**UNDO**) 处理。

进行撤销处理的方法是: 反向扫描日志文件, 对每个事务的更新操作执行逆操作, 即将日志记录中“更新前的值”写入数据库。

③ 对重做队列中的各个事务进行重做 (**REDO**) 处理。

进行重做处理的方法是: 正向扫描日志文件, 对每个重做事务重新执行日志文件登记的操作。即将日志记录中“更新后的值”写入数据库。

(3) 介质故障的恢复

介质故障会破坏磁盘上的物理数据库和日志文件, 这是最严重的一种故障。恢复方法是重装数据库后备副本, 然后重做已完成的事务。

具体恢复步骤是:

① 装入最新的数据库后备副本, 使数据库恢复到最近一次转储时的一致性状态。对于动态转储的数据库副本, 还需要同时装入转储开始时刻的日志文件副本。利用恢复系统故障的方法 (即重做+撤销的方法) 将数据库恢复到一致性状态。

② 装入相应的日志文件副本 (转储结束时刻的日志文件副本), 重做已完成的事务。

利用日志技术进行数据库恢复时, 恢复子系统必须搜索所有的日志, 确定哪些事务需要重做。

6. 数据库镜像技术

随着磁盘容量越来越大, 价格越来越便宜, 为避免磁盘介质出现故障影响数据库的可用性, 许多数据库管理系统提供了数据库镜像功能用于数据库恢复。数据库镜像技术是用来提高数据库可用性的主要软件解决方案。镜像基于每个数据库实现, 并且只适用于使用完整恢复模式的数据库。简单恢复模式和大容量日志恢复模式不支持数据库镜像。因此, 所有大容量操作始终被完整地记入日志。

数据库镜像需要两个数据库, 一个是主体数据库, 另一个是镜像数据库, 两个数据库驻留在不同的服务器上。在任何应用时间, 客户端只能使用一个数据库, 此数据库称为“主体数据库”。客户端对主体数据库进行的更新被同步到“镜像数据库”。

一旦出现介质故障, 可由镜像磁盘继续提供服务, 同时 **DBMS** 自动利用镜像磁盘数据进行数据库的恢复, 不需要关闭系统和重装数据库副本。在没有出现故障时, 数据库镜像还可以用于并发操作, 即当一个用户对数据加排他锁修改数据时, 其他用户也可以读镜像数据库上的数据, 而不必等待该用户释放锁。

由于数据库镜像是通过复制数据实现的, 频繁地复制数据自然会降低系统运行效率, 因此在实际应用中用户往往只选择对关键数据和日志文件镜像, 而不是对整个数据库进行镜像。

习 题 3

一、填空题

1. 数据库系统的安全特性主要是针对数据而言的, 包括____、数据安全性、数据完整性、并发控制、____等几方面。
2. 存取管理技术主要包括____和访问控制技术两方面。
3. 常见的安全审计技术主要有 4 类, 分别是: ____、基于代理的审计技术、基于网络监听的审计技术、____。
4. 事务具有 4 个特性: ____、____、隔离性和____。
5. ____就是数据库管理员 (DBA) 定期地将整个数据库复制到其他存储介质 (如磁带或非数据库所在的另外磁盘) 上保存形成备用文件的过程。
6. ____能够用来进行事务故障恢复、系统故障恢复, 并能够协助后备副本进行介质故障恢复。
7. 数据库镜像需要两个数据库, 一个是____, 另一个是镜像数据库。

二、选择题

1. 数据的完整性是指 ()。
 - A. 数据的存储和使用数据的程序无关
 - B. 防止数据被非法使用
 - C. 数据的正确性、一致性
 - D. 减少重复数据
2. 数据库系统运行过程中, 由于应用程序错误所产生的故障通常称为 ()。
 - A. 设备故障
 - B. 事务故障
 - C. 系统故障
 - D. 介质故障
3. 一个事务的执行, 要么全部完成, 要么全部不做, 一个事务中对数据库的所有操作都是一个不可分割的操作序列的属性是 ()。
 - A. 原子性
 - B. 一致性
 - C. 独立性
 - D. 持久性
4. 表示两个或多个事务可以同时运行而不互相影响的是 ()。
 - A. 原子性
 - B. 一致性
 - C. 独立性
 - D. 持久性
5. 事务的持续性是指 ()。
 - A. 事务中包括的所有操作要么都做, 要么都不做
 - B. 事务一旦提交, 对数据库的改变是永久的
 - C. 一个事务内部的操作对并发的其他事务是隔离的
 - D. 事务必须是使数据库从一个一致性状态变到另一个一致性状态
6. 日志文件是数据库系统出现故障以后, 保证数据正确、一致的重要机制之一。下列关于日志文件的说法错误的是 ()。
 - A. 日志的登记顺序必须严格按照事务执行的时间次序进行
 - B. 为了保证发生故障时能正确地恢复数据, 必须保证先写数据库后写日志
 - C. 检查点记录是日志文件的一种记录, 用于改善恢复效率
 - D. 事务故障恢复和系统故障恢复都必须使用日志文件
7. 对数据库中的数据进行及时转储是保证数据安全可靠的重要手段。下列关于静态转储和动态转储的说法正确的是 ()。
 - A. 静态转储过程中数据库系统不能运行其他事务, 不允许在转储期间执行数据插入、修改和删除操作

- B. 静态转储必须依赖数据库日志才能保证数据的一致性和有效性
- C. 动态转储需要等待正在运行的事务结束后才能开始
- D. 对一个 24 小时都有业务发生的业务系统来说, 比较适合采用静态转储技术

三、简答题

1. 什么是数据库的安全性? 什么是数据库的完整性? 两者之间的联系与区别是什么?
2. 数据库安全性保护通常采用什么方法?
3. 试述事务的概念及事务的 4 个特性。
4. 数据库运行中可能产生的故障有哪几类? 哪些故障影响事务的正常执行? 哪些故障破坏数据库数据?
5. 数据库恢复的基本技术有哪些?
6. 数据库备份的意义是什么? 试比较常见数据备份方法。
7. 什么是日志文件? 为什么要设立日志文件?
8. 针对不同的故障, 试给出恢复的策略和方法。
9. 什么是数据库镜像? 它有什么用途?

Access 应用实践篇

第 4 章 Access简介

Access 是一个多功能的数据库管理系统，是 Office 办公套件中一个极为重要的组成部分。不管是处理公司的客户订单数据，管理自己的个人通讯录，还是大量科研数据的记录和处理，人们都可以通过 Access 高质量地完成。Access 因其使用方便、功能强大，在实际中有着广泛的应用。

4.1 Access简介

Access 是微软公司推出的基于 Windows 的桌面关系数据库管理系统（Relational Database Management System, RDBMS），是 Office 系列应用软件之一。它提供了表、查询、窗体、报表、页、宏、模块等 7 种用来建立数据库系统的对象；通过提供多种向导、生成器、模板，能够把数据存储、数据查询、界面设计、报表生成等操作规范化；普通用户不必编写代码，就可以完成大部分数据管理的任务。

与其他数据库开发系统比较，若通过 Access 开发数据库系统，用户不用编写一行代码，就可以在很短的时间里开发出一个功能强大且相当专业的数据库应用程序，并且这一过程是完全可视的，如果再给它加上一些必要的 VBA 代码，那么就可以开发出更加专业的程序。

Microsoft Access 在很多地方得到广泛使用，例如，小型企业、大公司的某个部门和喜爱编程的开发人员专门利用它来制作处理数据的桌面系统。它也常被用来开发简单的 Web 应用程序，这些应用程序都利用 ASP 技术在 Internet Information Services 运行，比较复杂的 Web 应用程序则使用 PHP、MySQL 或 ASP、Microsoft SQL Server。

4.1.1 Access产生与发展

Access 1.0 发布于 20 世纪 90 年代初期，目前 Access 2003、Access 2007 已经得到广泛应用。历经多次升级改版，其功能越来越强，操作更加简单。尤其是 Access 与 Office 的高度集成，风格统一的操作界面使得许多初学者更容易掌握。

1995 年年末，Access 95 发布，这是世界上第一个 32 位 关系型数据库管理系统。1997 年，Access 97 发布，它的最大特点是在 Access 数据库中开始支持 Web 技术，使得 Access 数据库从桌面向网络发展。

21 世纪初，Microsoft 发布 Access 2000，这是微软桌面数据库管理系统的第六代产品，也是 32 位 Access 的第三个版本。2007 年微软正式发布了 Access 2007，Access 2007 对老版本的数据库程序进行了一次大规模更新和完善，增加了一些重要的新功能，并简化了界面。

4.1.2 Access安全方式

数据库的安全性是一个至关重要的问题，一个数据库管理系统，应具有一套系统而又完整的安全保护机制用来保障数据的安全及完整性。在 Access 中保护 Access 数据库的常见方法有以下 5 种。

1. 加密数据库

最简单的保护方法是对数据库进行加密。加密数据库就是将数据库文件压缩，使某些实用程序不能解读这些文件。加密可以避免在以电子方式传输数据库或者将其存储在软盘、磁带或光盘上时，其他用户访问数据库中的信息。然而 Jet（Access 使用的数据库引擎）使用的加密方法非常弱，因此绝不能用于保护敏感数据。“加密/解密数据库”命令位于“工具”菜单的“安全”子菜单中。解密数据库是对加密过程的逆运算。

2. 使用自定义界面

另一种相对简单的保护方案是使用自定义界面代替 Access 标准界面。和加密一样，它不能保护数据库中的对象和敏感数据的安全。通过选择“工具”菜单中的“启动”选项，可以指定自定义的启动窗体、菜单，甚至自定义的标题和图标。

3. 设置数据库密码

可以通过在数据库上设置密码，要求用户在访问数据和数据库对象时输入密码。使用密码保证数据库或其中对象的安全性也称为共享级安全性。使用该方式不能为用户或组分配权限，因此任何掌握密码的人都可以无限制地访问所有 Access 数据和数据库对象。“设置数据库密码”命令位于“工具”菜单的“安全”子菜单中。

4. 设置模块密码

使用密码可以保护所有标准模块和类模块以免用户不小心修改或查看。设置密码后，只需在每次会话时输入一次密码，便可在 Visual Basic 编辑器中查看或修改代码。除查看和编辑外，在剪切、复制、粘贴、导出或删除任何模块时也都需要密码。但使用这种方法保护代码不能防止运行代码，也不能防止其他用户使用第三方实用程序来查看代码。要完全保护代码，必须将.mdb 文件转换为 MDE 文件。

为项目中的模块设置密码的方法是：从 Visual Basic 编辑器的“工具”菜单中选择“属性”命令。在“项目属性”对话框中单击“保护”选项卡。选中“查看时锁定工程”复选框并键入密码，并在“确认密码”框中，重新键入密码，然后单击“确定”。

5. 使用MDE文件

通过将数据库文件转换为 MDE 文件，可以安全保护 Access 中的代码免受非法访问。将.mdb 文件转换为 MDE 文件时，Access 将编译所有模块，删除所有可编辑的源代码，然后压缩目标数据库。原始的.mdb 文件不会受到影响，新数据库中的 VBA 代码仍然能运行，但不能查看或编辑。数据库将继续正常工作，仍然可以升级数据和运行报表。

将.mdb 文件转换为 MDE 文件的方法是：关闭数据库，单击“工具”菜单中的“数据库实用工具”，单击“生成 MDE 文件”，在“保存数据库为 MDE”对话框中，找到.mdb 文件，然后单击“生成 MDE”。

4.2 Access基本概念

对于数据库来说，最重要的功能就是存取数据库中的数据，所以数据在数据库中各个对象间的流动就成为我们最关心的事情。为了以后建立数据库时能清楚的安排各种结构，应该先了解一下 Access 数据库中对象的作用和联系。

4.2.1 数据库和数据库对象

1. 数据库

Access 数据库是一个默认扩展名为.mdb 的文件，该文件由若干个对象构成，包括用来存储数据的“表”，用于查找数据的“查询”，提供友好用户界面的“窗体”、“报表”、“页”及用于开发系统的“宏”、“模块”等，数据库窗口如图4.1 所示。

一个 Access 数据库可以包含多个对象（可以是表、窗体、报表、查询等的组合），并且只要有充足的资源可用，可以同时打开多个表。可以从其他数据库（如 FoxPro）、客户/服务器数据库（如 Microsoft SQL Server）及电子数据表应用（如 Microsoft Excel 和 Lotus 1-2-3）中导入表。也可以将其他类型的数据库表、格式化文件（Excel 工作表和 ASCII 文本）及其他 Access 数据库链接到 Access 数据库。

2. 表

表是关于特定数据的集合，是数据库的核心。数据库中的全部信息都放在一个或多个表中。表是由行和列组成的二维表格，表中的每一行称为一条记录，反映了某一事物的全部信息，每一列称为一个字段，反映了某一事物的某种属性。能够唯一标识各个记录的字段或字段集称为主关键字。

当信息存储到表中以后，就可以将它们显示在界面更加美观的窗体上。这个过程就是将表中的数据和窗体上的控件建立连接，在 Access 中把这个过程称为绑定。绑定之后就可以通过屏幕上的各种各样的窗体界面来获得存储在表中的数据，表对象窗口如图4.2 所示。

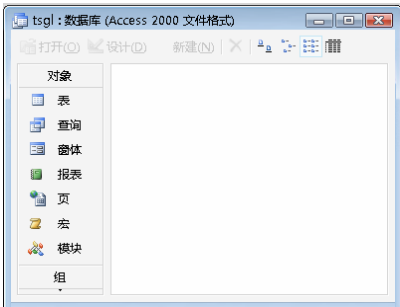


图 4.1 数据库窗口

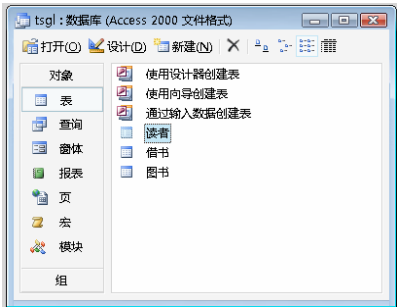


图 4.2 表对象窗口

3. 查询

在数据库的实际应用中，并不是简单地使用这个表或那个表中的数据，而是常常将有“关系”的很多表中的数据调出使用，有时还要对这些数据进行一定的计算以后才使用，对此问题最好的解决办法是使用“查询”。

查询的字段可以来自很多相互之间有“关系”的表，这些字段组合成一个新的数据表称

为视图，但它并不存储任何的数据。当表中数据改变时，查询中的数据也会随之改变，而且还可以通过查询完成复杂的计算工作。查询对象窗口如图4.3所示。

4. 窗体

窗体是数据库和用户联系的界面，用于显示包含在表中或者查询中的数据。它通过计算机屏幕将表或查询中的数据告诉操作者，建立一个友好的使用界面会给操作带来很大的方便，这是建立一个窗体的基本目标。一个好的窗体非常有用，不管数据库中表或查询设计得有多好，如果窗体设计得十分杂乱，而且没有任何提示，操作将变得很不方便。

通过窗体不仅可以显示包含在表或者查询中的数据，还可以向表中添加新的数据及更新或者删除现有的数据，可以在窗体中加入图片和图形，可以在窗体中包含音乐和活动视频，也可以包含子窗体（子窗体是包含在主窗体中的窗体）。Access 窗体还可以在类模块中包含 VBA 代码，为窗体和窗体上的控件提供事件处理子过程。窗体对象窗口如图4.4所示。

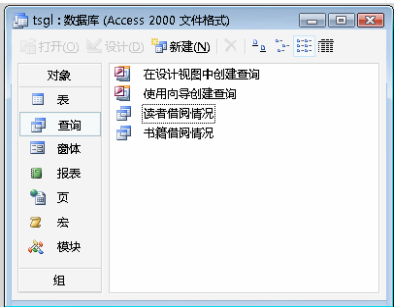


图 4.3 查询对象窗口

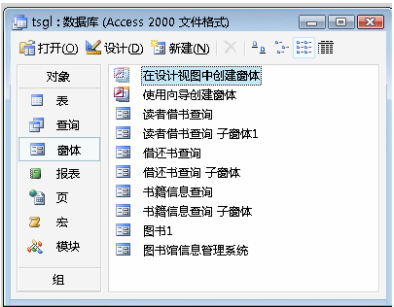


图 4.4 窗体对象窗口

5. 报表

用窗体显示数据虽然很好，却无法满打印要求。Access 中的报表对象可以很好地解决这个问题，该对象的作用就是实现数据的打印。

报表为查看和打印概括性的信息提供了最灵活的方法，可以在报表中控制每个对象的大小和显示方式，并可以按照所需的方式来显示相应的内容，还可以在报表中添加多级汇总、统计比较，甚至加上图片和图表。所以，在 Access 中使用报表来打印格式数据是一种非常有效的方法。报表和窗体的建立过程基本是一样，只是一个显示在屏幕上，一个显示在纸上。窗体可以有交互，而报表没有交互。报表对象窗口如图4.5所示。

6. 页

页就是通常所说的网页。在 Internet 上，很多信息都是以网页的形式来发布和传播。Access 一个突出的功能就是将 Access 数据库中的数据动态地提供给页，页的使用和窗体一样简单，而且这种联接是动态的，不需要创建很多的页面，一切繁琐的工作都由 Access 自己内部处理。掌握这个对象的用法，会使往 Internet 发布数据变得更加方便。页对象窗口如图4.6所示。

7. 宏

宏是一种操作命令，它和菜单操作命令一样，只是它们对数据库施加作用的时间有所不同，作用时的条件有所不同。菜单命令一般用在数据库的设计过程中，而宏命令则用在数据库的执行过程中。菜单命令必须由使用者来施加这个操作，而宏命令则可以在数据库中自动执行。Access 一共有几十种基本宏操作。在使用中，很少单独使用这些基本宏命令，常常是 将这些命令排成一组，形成很多“宏组”操作，按顺序“宏组”中的宏执行，完成一种特定

任务。这些命令既可以通过窗体中控件的某个事件操作来实现，也可以在数据库的运行过程中自动实现。宏对象窗口如图4.7所示。

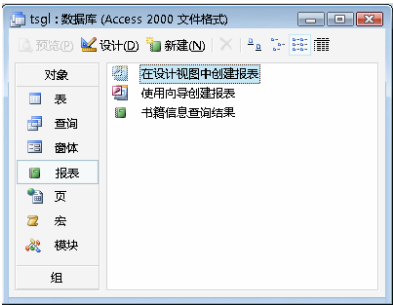


图 4.5 报表对象窗口

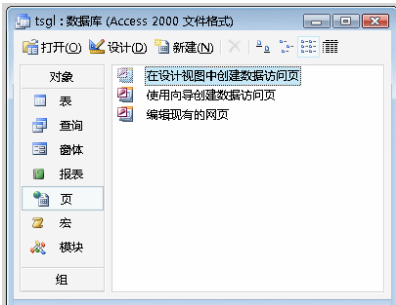


图 4.6 页对象窗口

8. 模块

虽然宏很好用，但它运行的速度比较慢，也不能直接运行 Windows 程序，尤其是不能自定义函数。这样，当我们要对某些数据进行一些特殊分析时，它就无能为力了。由于宏的局限性，所以在设计一些特殊功能时，需要用到模块对象，而这些模块都是由 VBA 来实现的。使用 VBA 编写程序，然后将这些程序编译成拥有特定功能的模块，以便在 Access 中调用。

VB 是微软公司推出的可视化 Basic 语言，用它来编程非常简单。因为简单且功能强大，所以微软公司将它的一部分代码结合到 Office 中，形成我们今天所说的 VBA (Visual Basic for Application, VBA)。可以像编写 VB 程序那样来编写 VBA 程序，来实现某个功能。当这段程序编译通过以后，将这段程序保存在 Access 的一个模块里，并通过类似在窗体中激发宏的方法来启动“模块”，从而实现相应的功能。模块对象窗口如图4.8所示。

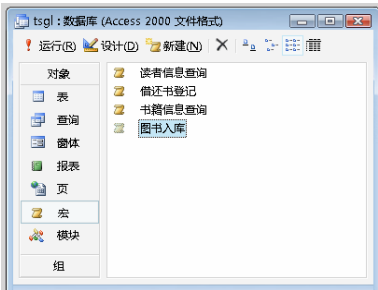


图 4.7 宏对象窗口

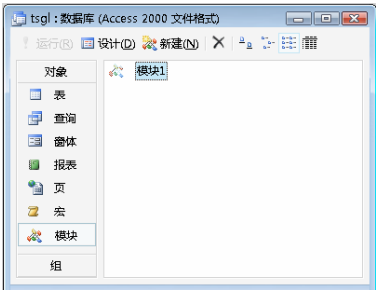


图 4.8 模块对象窗口

4.2.2 Access基本功能

Access 是一个关系数据库管理系统，即使是初学者，也可以通过可视化的操作来完成绝大多数的数据库管理和开发工作。对高级数据库系统开发人员来说，可以通过 VBA 开发高质量的数据库应用软件。另外，由于它拥有与 Office 其他组件类似的用户界面，包括基本相同的菜单系统、工具栏按钮、显示窗口和操作方法，用户可以在短时间内熟悉 Access 的操作环境。Access 具有三大基本功能。

1. 建立数据库

根据实际问题的需要建立数据库，在数据库中建立若干个表结构，并给这些表输入具体的数据，建立表间的联系。

2. 数据库操作

建立数据库的目的是对数据库中数据进行操作,以获得有用的数据或信息。对于数据库中的数据进行增加、删除、修改、索引、排序、检索(查询)、统计分析、打印显示报表、制作发布网页等操作。其中增加、删除、修改、索引、排序等操作属于数据库的维护,检索(查询)、统计分析、打印显示报表、制作发布网页等操作属于数据库的使用。

3. 数据通信

Access 可与其他应用软件如 Excel、Word 等之间进行数据的传输和交换,便于 Access 利用其他软件的处理结果,或其他软件利用 Access 的处理结果。

4.2.3 Access操作模式

Access 具有三个基本操作模式如下。

1. 启动模式

能够压缩、转换、加密、解密、修复数据库,只需在打开一个数据库之前从“工具”菜单的“数据库实用工具”和“安全”子菜单中选择相应的菜单命令即可。

2. 设计模式

能够创建和查询表和查询的结构,开发窗体显示和编辑数据,格式化报表打印。Access 称设计模式为“设计”视图。

3. 运行模式

在单个文档窗口(默认的模式)中显示表、窗体和报表设计。执行宏时可以选择其中的一个,然后选择运行模式。运行模式并不适用于 VBA 模块,因为函数的执行基本上作为查询、窗体和报表的元素来执行的。模块中的过程可以由宏命令或者直接从窗体和报表的事件中执行。表和查询的运行模式称为“数据表”视图,窗体的运行模式称为“窗体”视图,数据访问页(DAP)的运行模式称为“页”视图,而报表的运行模式称为“打印预览”。

可以通过选择数据表中的命令按钮、工具栏上的按钮或者从“视图”菜单中选择命令来选择设计或者运行模式。

4.3 Access的启动和退出

4.3.1 启动Access

启动 Access 有多种方法:在桌面上双击快捷图标;通过系统菜单选择运行等。

通过系统菜单选择运行的步骤如下:

- ① 在 Windows 桌面上单击“开始”,出现“开始”菜单。
- ② 单击“程序”,出现“程序”菜单。
- ③ 单击“Microsoft Access”菜单项即可。

启动界面如图4.9所示,通常可以分成五个大的部分:标题栏、菜单栏、工具栏、状态栏和数据库窗口。



图 4.9 Access 启动后界面

1. 标题栏

标题栏在屏幕的最上方，用于指明现在是谁的窗口；标题栏靠右的位置上有窗口控制按钮，利用这些按钮可以方便地对整个窗口进行放大（恢复）、缩小和关闭操作。

2. 菜单栏

菜单栏在标题栏的下面，用鼠标单击任意一个菜单项，就可打开相应的菜单，选择相应的数据库操作命令，就能够实现 Access 提供的某个功能。

3. 工具栏

菜单栏的下面是工具栏，工具栏中有很多工具按钮，这些工具按钮是常用菜单选项的快捷运行方式，通过使用工具栏上的按钮，可以提高工作效率。

4. 状态栏

状态栏在屏幕的最下方，用于显示正在进行的操作信息。

5. 数据库窗口

数据库窗口是 Access 中非常重要的部分，通过它可以方便、快捷地对数据库进行各种操作。数据库窗口本身又包括窗口菜单、数据库组件选项卡、创建方法和已有对象列表三个部分。

窗口左侧包含两个方面的内容，上面是对象，下面是组。

对象下分类列出了 Access 数据库中的所有对象，比如用鼠标单击“表”对象按钮，窗口右侧就会列出创建“表”对象的基本方法和本数据库中已经创建的所有表。

组提供了另一种管理对象的方法：可以把那些关系比较紧密的对象分为同一组，不同类别的对象也可以归到同一组中。比如，要做一个通讯簿数据库，就可以将通讯簿表和通讯簿窗体归为一组。当数据库中的对象很多时，用分组的方法可以更方便地管理各种对象。但删除组中的对象时，不是将这个对象从数据库中删去，因为组中只是真实对象的快捷方式。将组中对象删除，只是将对象在组中建立的这个快捷方式删除了，它仍然存在于数据库中。

4.3.2 退出Access

退出 Access 的方法如下：

- ① 在 Microsoft Access 的应用文件窗口的菜单栏中单击“文件”菜单；
- ② 选择“退出”。

如果已经改变了数据库的内容而没有保存过, Access 将询问是否保存文件, 可以根据需要进行选择。

习 题 4

一、填空题

1. Access 数据库是一个默认扩展名为_____的文件, 该文件由若干个对象构成。
2. 表是关于_____的集合, 是数据库的核心。
3. 表是由行和列组成的二维表格, 表中的每一行称为一条_____, 反映了某一事物的全部信息。
4. 表中的每一列称为一个_____, 反映了某一事物的某种属性。
5. _____为查看和打印概括性的信息提供了最灵活的方法, 可以在其中控制每个对象的大小和显示方式。
6. _____的字段可以来自很多相互之间有“关系”的表, 这些字段组合成一个新的数据表视图, 但它并不存储任何的数据。
7. _____是数据库和用户联系的界面, 用于显示包含在表中或者查询中的数据。
8. 在 Internet 上, 很多信息都是以网页的形式来发布和传播。Access 一个突出的功能就是将 Access 数据库中的数据动态地提供给_____。
9. Access 一共有几十种基本宏操作, 这些基本操作还可以组合成很多_____操作。
10. 有多个操作构成的宏, 执行时是按_____执行的。
11. 由于宏的局限性, 所以在设计一些特殊功能时, 需要用到模块对象, 而这些模块都是由_____来实现的。
12. Access 可与其他应用软件如 Excel、Word 等之间进行数据的_____, 便于 Access 利用其他软件的处理结果, 或其他软件利用 Access 的处理结果。

二、选择题

1. Access 中表和数据库的关系是 ()。
A. 一个数据库可以包含多个表
B. 一个表只能包含两个数据库
C. 一个表可以包含多个数据库
D. 一个数据库只能包含一个表
2. 以下关于报表的叙述正确的是 ()。
A. 报表只能输入数据
B. 报表只能输出数据
C. 报表可以输入和输出数据
D. 报表不能输入和输出数据
3. 在 Access 中需要发布数据库中的数据的时候, 可以采用的对象是 ()。
A. 数据访问页
B. 表
C. 窗体
D. 查询
4. Access 数据库属于 () 数据库。
A. 层次模型
B. 网状模型
C. 关系模型
D. 面向对象模型
5. 打开 Access 数据库时, 应打开扩展名为 () 的文件。
A. mda
B. mdb
C. Mde
D. DBF
6. 下列 () 不是 Access 数据库的对象类型。
A. 表
B. 向导
C. 窗体
D. 报表
7. 不能退出 Access 的方法是 ()。
A. 单击“文件”菜单“退出”
B. 单击窗口右上角“关闭”按钮

- C. ESC
D. Alt+F4
8. 宏是由一个或多个 () 组成的集合。
A. 命令 B. 操作 C. 对象 D. 表达式
9. 以下软件 () 不是数据库管理系统。
A. VB B. Access C. Sybase D. Oracle
10. 以下 () 不是 Access 的数据库对象。
A. 表 B. 查询 C. 窗体 D. 文件夹
11. Access 是 () 公司的产品。
A. 微软 B. IBM C. Intel D. Sony
12. 在 Access 中, 可用于设计输入界面的对象是 ()。
A. 窗体 B. 报表 C. 查询 D. 表
13. Access 中表和数据库的关系是 ()。
A. 一个数据库可以包含多个表 B. 一个表只能包含两个数据库
C. 一个表可以包含多个数据库 D. 一个数据库只能包含一个表
14. 在 Access 数据库中, 专用于打印的是 ()。
A. 表 B. 查询 C. 报表 D. 页
15. Access 数据库依赖于 () 操作系统。
A. DOS B. Windows C. UNIX D. UCOS

三、简答题

1. Access 具有哪些能基本优点?
2. Access 数据库对象之间有何关系?
3. 启动 Access 有哪些基本方法?
4. 简要说明 Access 的界面组成。
5. 简要说明 Access 有哪几种操作模式?

第 5 章 创建数据库

Access 数据库管理系统是基于对象的，允许用户定义并操作“表”、“查询”、“窗体”、“报表”等对象。建立 Access 数据库有两种基本方法，一种是创建一个空数据库，然后再添加“表”、“查询”、“窗体”、“报表”等对象；另一种是使用数据库向导创建数据库，在创建过程中，用户只需进行简单的选择，就可以在建立框架的同时建立相应的“表”、“查询”、“窗体”、“报表”等对象，从而建立一个完整的数据库。无论采用哪种方法创建的数据库，都可以在任意时刻根据需要进行数据库的修改和扩充。

5.1 创建数据库

Access 提供了三种基本的创建数据库的方法：创建一个空的数据库，根据模板创建数据库，根据文件创建数据库。

5.1.1 创建一个空数据库

- 在 Access 中，新建一个空数据库的具体步骤如下：
- ① 选择“文件”菜单中的“新建”命令。
 - ② 在“新建文件”面板中选择“空数据库”选项。系统弹出“文件新建数据库”对话框，如图5.1所示。
 - ③ 在“文件新建数据库”对话框中，选择保存位置，指定数据库文件名，单击“创建”按钮。例中给数据库取名为“学生成绩管理”。
- 一个创建好的空数据库如图5.2所示，对于该数据库，可以在需要时添加“表”、“查询”、“窗体”、“报表”等对象。

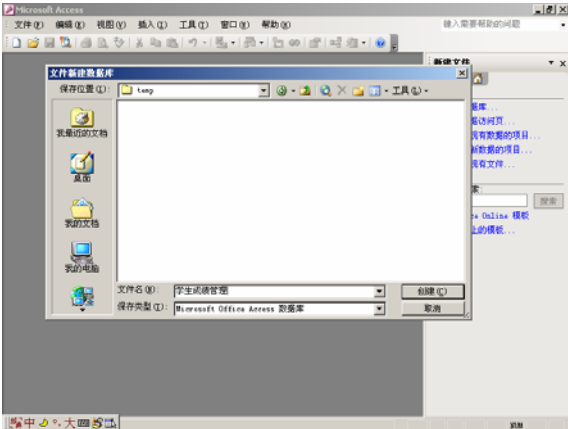


图 5.1 “文件新建数据库”对话框



图 5.2 “学生成绩管理：数据库”窗口

5.1.2 通过模板建立数据库

为了方便用户的使用，Access 提供了一些标准的数据框架，称为“模板”。这些模板不一定符合用户的实际要求，但在向导的帮助下，可以对这些模板稍加修改，即可建立一个符合用户要求的数据库。另外，通过这些模板还可以学习如何组织构造一个数据库。

通过模板，只要回答几个问题就可以轻松地创建一个数据库，可以提高工作效率。通过模板建立数据库的步骤如下：

① 选择“文件”菜单中的“新建”命令。

② 在“新建文件”任务窗的模板中选择“本机上的模板”选项。系统弹出“模板”对话框，在“常用”和“数据库”两个选项卡中选择“数据库”选项，如图5.3所示。

“数据库”选项卡里有很多向导图标，这些图标代表不同的数据库向导。如果需要建立一个关于公司客户、订单等情况的数据库，可以通过双击“订单”图标来完成。

③ 系统弹出“文件新建数据库”对话框。在“文件新建数据库”对话框中，选择保存位置，输入数据库文件名（例中输入“产品订单”），单击“创建”按钮。

④ 选择数据库中的表和表中的字段。在屏幕上出现“数据库向导”对话框，单击“下一步”按钮，进入下一步操作，如图5.4所示。左边是数据库中的表，右边是表中的字段。



图 5.3 “模板”对话框

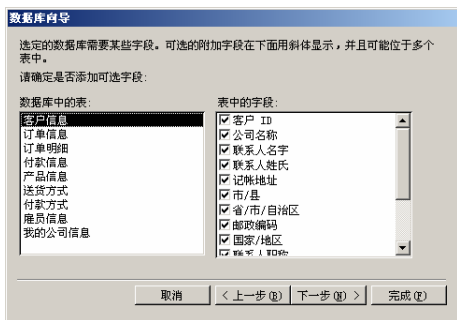


图 5.4 “数据库向导”对话框

字段的前面有一个复选框，有的复选框中有一个“√”，表示此字段被选中，被选中的字段将会出现在数据库表中，没有选中的字段就不会出现在数据库表中。可以通过单击字段的复选框来决定数据库表中是否要包含某些字段。

⑤ 设置屏幕的显示方式。如果没有别的要求，单击“下一步”按钮。可以根据需要选择屏幕的显示方式。也就是选择将要建立的数据库中窗口的背景、窗口上的默认字体大小和颜色。例中选择“标准”方式。

⑥ 单击“下一步”按钮，选择在打印时所用的格式。例中选择“紧凑”样式。

⑦ 为数据库指定标题。在选定报表的打印样式后单击“下一步”按钮，给新建的数据库指定一个标题，例中定名为“产品订单”。

⑧ 启动数据库。单击“下一步”按钮，在出现的对话框中单击“完成”按钮，数据库建立成功。屏幕上出现“主切换版面”对话框，如图5.5所示，显示的就是新建的数据库“产品订单”的主窗体。

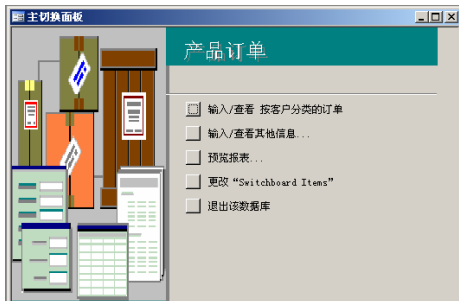


图 5.5 “主切换版面”对话框

新建的数据库中没有存储任何数据，向导的作用主要是为数据库管理建好数据库框架，而数据需要用户自己输入。

通过模板建立数据库虽然简单，但是有时候它根本满足不了实际的需要。一般来说，对数据库有了进一步了解后，我们就不再去用向导创建数据库了，熟练的用户很少使用向导。

5.2 创建表

表是数据库最主要的对象，在创建了数据库后，就可以创建表。表的创建有多种方法，使用向导、设计器、通过输入数据都可以建立表。但主要的方式有两种常见方法：一种使用表向导创建表，另一种通过设计视图创建新表。

5.2.1 用表向导建立表

表向导提供两类表：商务表和个人表。商务表包括客户、雇员和产品等常见表模板；个人表包括家庭物品清单、食谱、植物和运动日志等表模板。使用表向导创建基本表的过程如下：

① 在“文件”菜单中选择“打开”，打开“产品订单”数据库后出现“产品订单”数据库窗口。

② 在“数据库”窗口中，单击对象栏中的“表”按钮，然后双击“使用表向导创建表”按钮，弹出“表向导”对话框，结果如图5.6所示。

③ 在“表向导”对话框中设置新表中的字段。在“示例表”中选择表名，在“示例字段”中选择相应的字段，最后可以将选中的字段组成一个新的表（形成该表的结构）。

例中，定义“客户表”（用来记录一个公司的客户信息）。首先选择表的类型（若该表用于商务，则单击“商务”选项），然后在“示例表”中选择“客户”，在“示例字段”中选择所需字段。接下来单击“>”按钮将所需的“示例字段”加入新表，可重复增加“示例字段”，选择结果如图5.7所示。

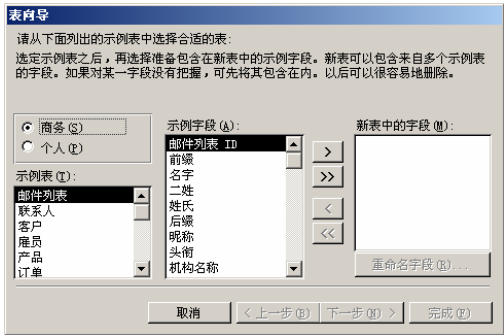


图 5.6 “表向导”对话框

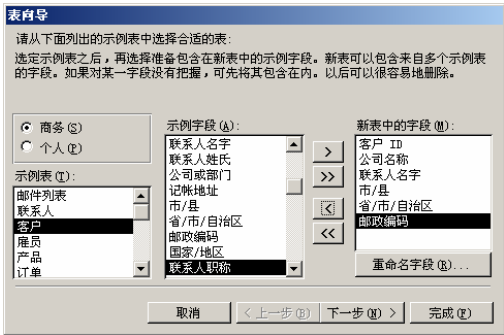


图 5.7 创建客户表

④ 在“表向导”对话框中修改字段名。如果认为示例字段的名称不合要求，通过“重命名字段...”按钮修改字段名。

字段名修改完成后，单击“下一步”，出现“表命名”对话框，给表命名为“客户信息”，并选择“是，设置一个主键”（主键是记录之间的唯一性区分标志）。

⑤ 数据输入。单击“下一步”按钮，弹出“相关性选择”对话框，单击“下一步”按钮。出现“请选择表创建完之后的动作”对话框。如果要立即输入数据，选择“直接向表中输入数据”，然后单击“完成”按钮，出现数据录入窗口，如图 5.8 所示，在这个表窗口中，可以输入所要处理的数据。

在进行输入时，可以通过窗口左下角的按钮及其中间的文本框来控制表中记录的操作。按钮功能如表 5.1 所示。文本框用于指明当前记录在表中的行数，可以通过修改框里数字而改变当前记录。

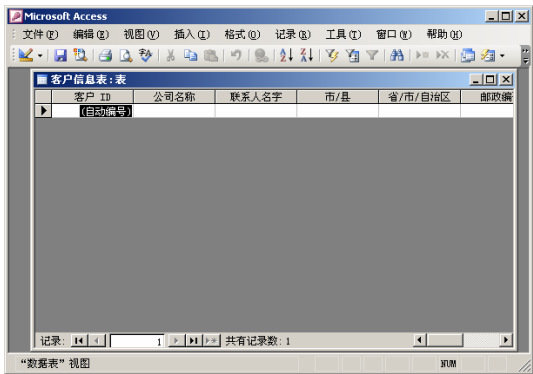


图 5.8 数据录入窗口

表 5.1 按钮功能表

按钮	功能	按钮	功能
>	移动到下一条记录。	>	跳到最后一条记录。
<	移动到上一条记录。	<	跳到第一条记录。
>*	在表中插入新记录。		

5.2.2 通过设计器创建表

虽然向导提供了一种简单快捷的方法来建立表，但如果向导不能提供用户所需要的字段，则用户还需重新创建。因此，大多数用户都通过表设计器设计表。

1. 表结构

用表向导建立表时，系统会自动定义各种字段数据的属性。而通过表设计器进行表设计时，则要对表中的每一字段数据的属性进行设置，比如将表中的某个字段定义为数字类型，那么这个字段就只能输入数字。在设计器中可以方便而直观地进行表结构的设计，在使用表设计器之前，需要先了解与表关系密切的几个基本概念：字段名、数据类型和字段属性。

(1) 字段名

字段名是表中某一列的名称，用来标识字段，由英文、中文、数字构成。命名规则如下：

- ◆ 字段名长度 1~64 个字符。
- ◆ 不能以空格开头。
- ◆ 字段名不能含有“.”、“!”、“[”、“]”等字符。

(2) 数据类型

字段取值的类型称为数据类型。Access 中常见的数据类型有：文本、备注、数字、日期/时间、货币、自动编号、是/否、OLE 对象、超级链接、查询向导等。

- ◆ 文本：最大允许 255 个字符或数字，默认大小是 50 个字符，系统只保存输入到字段中的字符，而不保存文本字段中未用位置上的空字符。设置“字段大小”属性可以控制输入的最大字符长度。
- ◆ 备注：保存长度较长的文本及数字，允许字段能够存储长达 65535 个字符的内容。Access 能在备注字段中搜索文本，但不能对备注字段进行排序或索引。

- ◆ 数字：用来存储进行数值计算的数据，设置“字段大小”属性可以定义一个特定的数字类型（“整数”、“长整数”、“单精度数”、“双精度数”、“小数”五种类型），在 Access 中通常默认为“双精度数”。
- ◆ 日期/时间：用来存储日期、时间或日期时间，可以存放从 100 到 9999 年的日期与时间值。每个日期/时间字段需要 8 字节的存储空间。
- ◆ 货币：一种等价于双精度数的特殊数字类型，占 8 字节。向货币字段输入数据时，不必输入人民币符号和千位处的逗号，Access 会自动显示人民币符号和逗号，并添加两位小数到货币字段。当小数部分多于两位时，Access 会对数据进行四舍五入，精确度为小数点左方 15 位数和右方 4 位数。
- ◆ 自动编号：每次向表格添加新记录时，Access 会自动插入唯一顺序或者随机编号。自动编号一旦被指定，就会永久地与记录连接，如果删除了表格中含有自动编号字段的一个记录后，Access 并不会为表格自动编号字段重新编号。当添加某一记录时，Access 不再使用已被删除的自动编号字段的数值，而是重新按递增的规律重新赋值。
- ◆ 是/否：是针对某一字段中只包含两个不同的可选值而设立的字段，通过“是/否”数据类型的格式特性，用户可以对“是/否”字段进行选择。
- ◆ OLE 对象：允许单独链接或嵌入 OLE 对象。添加数据到 OLE 对象字段时，可以链接或嵌入 Word 文档、Excel 电子表格、图像、声音或其他二进制数据，存储空间最大为 1GB。
- ◆ 超级链接：主要是用来保存超级链接地址。当单击一个超级链接时，Web 浏览器将根据超级链接地址打开网页，在这个字段中插入超级链接地址最简单的方法就是在“插入”菜单中单击“超级链接”命令。
- ◆ 查询向导：为用户提供了一个字段允许取值的列表，在输入字段时，可以直接从这个列表中选择字段值。

对于某一具体数据而言，可以使用的数据类型可能有多种，例如，电话号码可以使用数字型，也可使用文本型，但只有一种是最合适的。在选择数据类型时，主要考虑以下几个方面的因素：

- ◆ 字段中可以使用什么类型的值。
- ◆ 需要用多少存储空间来保存字段的值。
- ◆ 是否需要对数据进行计算（主要区分是否用数字，文本、备注等）。
- ◆ 是否需要建立排序或索引（备注、超链接及 OLE 对象型字段不能使用排序和索引）。
- ◆ 是否需要排序（数字和文本的排序有区别）。
- ◆ 是否需要在查询或报表中对记录进行分组（备注、超链接及 OLE 对象型字段不能用于分组记录）。

（3）字段属性

字段有一些基本属性（如字段名、字段类型、字段宽度及小数点位数），另外对于不同的字段，还会有一些不同的其他属性。

- ◆ 字段大小：文本型默认值为 50 字节，不超过 255 字节。不同种类的数字型所占存储空间不一样。
- ◆ 格式：利用格式属性可在不改变数据存储情况的条件下，改变数据显示与打印的格式。文本和备注型数据的格式最多可由三个区段组成，每个区段包含字段内不同数据格式

的规格。第一区段描述文本字段的格式，第二区段描述零长度字符串的格式，第三区段描述 Null 值字段的格式。可以用 4 种格式符号来控制输入数据的格式：@符号表示输入字符为文本或空格；&符号表示不需要使用文本字符；<符号表示输入的所有字母全部小写(放在格式开始)；>符号表示输入的所有字母全部大写(放在格式开始)。

- ◆ 小数位数：小数位数只有数字和货币型数据可以使用。小数位数为 0 ~ 15 位，视数字或货币型数据的字段大小而定。
- ◆ 标题：在报表和窗体中替代字段名称。要求简短、明确，便于管理和使用。
- ◆ 默认值：新记录在数据表中自动显示的值。默认值只是开始值，可在输入时改变，其作用是为了减少输入时的重复操作。
- ◆ 有效性规则：检查字段中的值是否有效。在该字段的“有效性规则”框中输入一个表达式，Access 会判断输入的值是否满足这个表达式，如果满足才能输入。也可以单击这个属性输入文本框右面的“生成”按钮激活“表达式生成器”来生成这些表达式。
- ◆ 输入掩码：用于控制输入到字段中的值。设置字段的输入掩码，只要单击“输入掩码”文本框右面的“生成”按钮，就会出现“输入掩码向导”对话框，在对话框上的列表框选择相应选项即可。比如要让这个文本字段的输入值以密码的方式输入，则单击列表框中的“密码”选项，然后单击“完成”按钮。
- ◆ 输入法模式：为选择性属性，共有三个选项，分别是“随意”、“输入法开启”、“输入法关闭”。选中“输入法开启”，当光标移动到这个字段内的时候，屏幕上就会自动弹出首选的中文输入法；选择“输入法关闭”时，只能在这个字段内输入英文和数字；选择“随意”就可以启动和关闭中文输入法。
- ◆ 标题：一般情况下不设。默认取字段的字段名作为标题，这样当在窗体上用到这个字段的时候就会把字段名作为它的标题来显示。
- ◆ 必填字段：在填写一个表的时候，常常会遇到一些必须填写的重要字段，像本表中的“姓名”字段就必须填写，所以将这个字段的“必填字段”属性设为“是”。而对于那些要求得不那么严格的数据就可以设定对应字段的属性设置为“否”。
- ◆ 允许空字符串：是否让这个字段里存在“零长度字符串”，通常将它设置为“否”。
- ◆ 索引：是否将这个字段定义为表中的索引字段。“无”表示不把这个字段作为索引；“有(有重复)”表示建立索引并允许在表的这个字段中存在同样的值；“有(无重复)”表示建立索引而且在该字段中绝对禁止相同的值。
- ◆ UNICODE：微软公司为了使一个产品在不同的国家各种语言情况下都能正常运行而编写的一种文字代码。对字段的这个属性一般都选择“有”。

2. 创建表

通过设计视图创建新表的步骤如下。

① 打开“学生成绩管理”数据库，出现“学生成绩管理”数据库窗口，选择“使用设计器创建表”，出现“设计器”窗口，如图5.9所示。

“设计器”窗口分为上、下两个部分。上半部分是表设计器，包含“字段名称”、“数据类型”和“说明”三列，用来定义字段名称和类型。下半部分用来定义表中字段的属性。建立一个表时，只要在设计器“字段名称”列中输入字段名称，并在“数据类型”列定义字段的“数据类型”就可以了。“说明”列中主要包括字段的说明信息，主要目的在于以后修改表

结构时能知道当时设计该字段的原因。

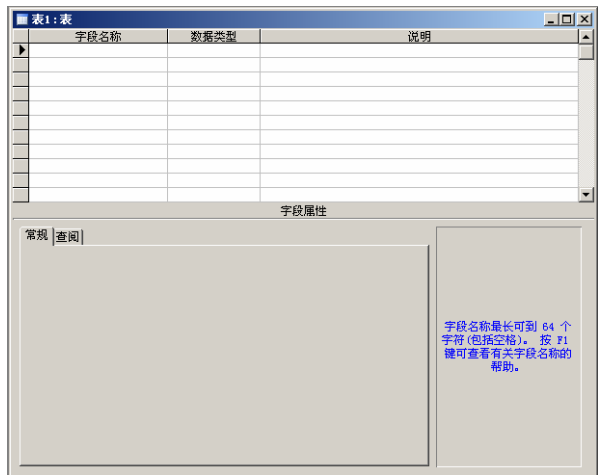


图 5.9 “设计器”窗口

② 定义字段名和类型。

定义学生表的表结构：学号（类型：文本）、姓名（类型：文本）、性别（类型：文本），出生年月（类型：日期/时间）、系别（类型：文本）、简历（类型：备注）。在表设计器的“字段名称”列中按顺序输入这些字段的名称，在“数据类型”列选择相应的类型。表就初步建好了，结果如图5.10所示。



图 5.10 学生表设计结果

③ 设置主键。

主键唯一标识表中每条记录。主键不允许为 Null，并且必须始终具有唯一索引。在表中设置主键的过程非常简单。比如要将“学号”字段作为表的“主键”，只要单击“学号”这一行中的任何位置，将该行设置为当前行，然后单击工具栏上的“主键”按钮，在“学号”一行最左面的方格中出现“钥匙”符号，主键设置完成。如果想取消主键，先选中字段，然后单击工具栏上的“主键”按钮即可。

④ 设置字段属性。

表设计器的下半部分是用来设置表中字段的“字段属性”。字段属性一般包括“字段大小”、

“格式”、“输入法模式”等，对它们进行的设置不同，会对表中的数值产生不同的影响。

设置“学号”字段的“字段大小”属性。默认的“字段大小”为“50”，表示这个字段中最多可以输入 50 个字符。而学号一般不超过 10 个字符，所以可将字段大小定为“10”。只要选中字段大小文本框，然后修改里面的数值就可以了。

⑤ 在文件菜单中选择“另存为”，保存新建的表。

为了完整地进行学生的成绩管理，现在建立“成绩”表，“课程”表。

其中“成绩”表的结构为：学号、课程代码、分数。“课程”表的结构为：课程代码、课程名称、任课教师、学分。具体结果如图 5.11、图 5.12 所示。

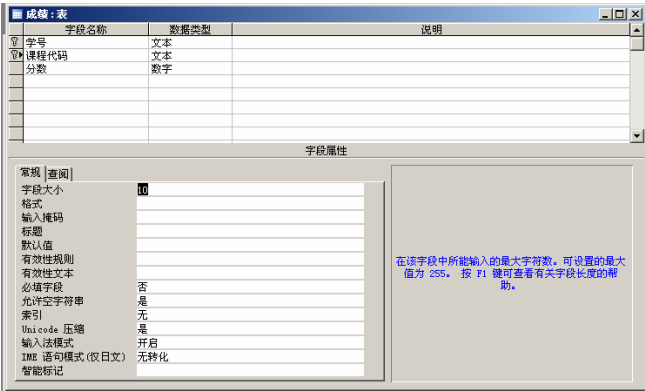


图 5.11 成绩表

“学生”表、“成绩”表、“课程”表建立后，“学生成绩管理”数据库窗口如图 5.13 所示。

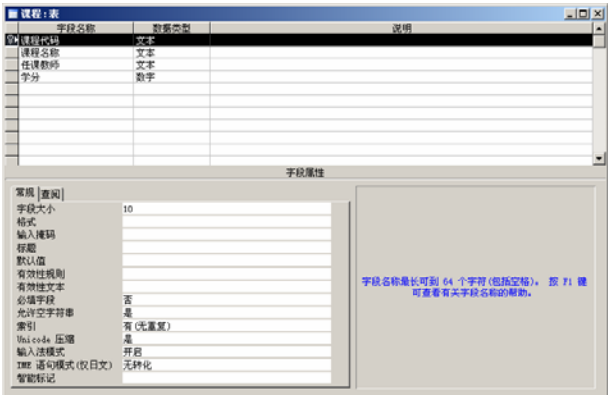


图 5.12 课程表

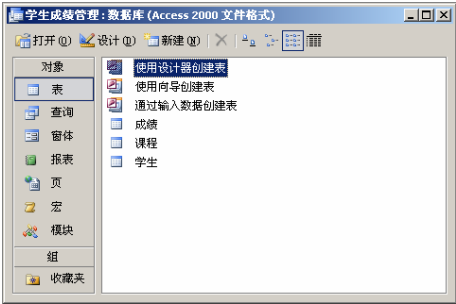


图 5.13 “学生成绩管理”数据库窗口

5.3 表间关系的创建

5.3.1 表的索引

当表中的数据很多时，需要利用索引帮助用户更有效地查询数据。

1. 索引的概念

索引的概念涉及记录的物理顺序与逻辑顺序。文件中的记录一般按其磁盘存储顺序输出，

这种顺序称为物理顺序。索引不改变文件中记录的物理顺序，而是按某个索引关键字（或表达式）来建立记录的逻辑顺序。在索引文件中，所有关键字值按升序或降序排列，每个值对应原文件中相应记录的记录号，这样便确定了记录的逻辑顺序。对文件记录的操作可以依据这个索引建立的逻辑顺序来进行。

索引文件会增加系统开销，因此，一般只对需要频繁查询或排序的字段创建索引。而且，如果字段中许多值是相同的，索引不会显著提高查询效率。在 Access 中，表的主键将自动被设置为索引，而备注、超链接及 OLE 对象等类型的字段则不能设置索引。Access 为每个字段提供了三个索引选项：“无”、“有（有重复）”、“有（无重复）”。

2. 单字段索引

索引可分为单一字段索引和多字段索引两种。一般情况下，表中的索引为单一字段索引。建立单一字段索引的方法如下：

- ① 打开表设计视图，单击要创建索引的字段，该字段属性将出现在“字段属性”区域中。
- ② 打开“常规”选项卡的“索引”下拉列表，在其中选择“有（有重复）”选项或“有（无重复）”选项即可。
- ③ 保存修改。

3. 多字段索引

如果经常需要同时搜索或排序更多的字段，那么就需要为组合字段设置索引。建立多字段索引的操作步骤如下：

- ① 在表的设计视图中单击工具栏中的“索引”按钮，弹出“索引”对话框。
- ② 在“索引名称”列的第一个空行内输入索引名称，索引名称一般与索引字段名相同。
- ③ 选字段名称，设置排序次序。

5.3.2 表间关系

一个数据库中可能有很多表，而且一般情况下这些表之间都有联系。表和表之间的关系由字段来联系，字段分别在两个表中，它们的类型和大小必须相同，字段名可以相同，也可以不同。关系将数据库里表之间的记录相互联系起来，使得对一个表中数据的操作有可能影响其他表的数据。表之间的关系存在以下四种情况。

1. 一对一关系

一个表中仅有一个记录的关键字字段的值与另外一个表中相关字段的某个单独的值相匹配。例如，若一门课程只有一个教师讲授，而每位教师只讲授一门课程，则课程与教师之间是一对一联系。

2. 多对一关系

允许一个表的关键字字段中可以有多个值与另外一个表中相关字段中某个单独的值相匹配。例如，若有若干个教师可以讲授一门课程，若干本参考书可供一门课程使用，则教师与课程之间、参考书与课程之间的联系是多对一联系。

3. 一对多关系

一个表的主键字段唯一，但是表主键字段中的值可以和另外一个表中相关字段中的多个

项相匹配。在这种类型中，这个表中的主键字段具有和另外一个表中有关的字段一对多关系。若一门课程可以有若干个教师讲授，可使用若干本参考书，而每位教师只讲授一门课程，每一本参考书只供一门课程使用，则课程与教师之间、课程与参考书之间的联系是一对多联系。

在实际中，多对一关系和一对多关系没有本质区别，只是从不同的角度观察而已。

4. 多对多关系

一种完全没有限制的关系，不要求现有表或者新表中的关键字段之间存在唯一的关系，并且两个表的外键字段可以包含重复的值。例如，有供应商、项目、零件，一个供应商可以供应多个项目多种零件，而每个项目可以使用多个供应商供应的零件，每种零件也可由不同供应商供给，由此看出供应商、项目、零件三者之间是多对多的联系。

注意：如果两个表存在一对一关系，则建立关系的字段均为主键，如果两个表存在一对多关系，在“一”方表中，该字段为主键。在“多”方表中，该字段不是主键。

5.3.3 创建关系

在表与表之间建立关系，不仅在于确立了数据表之间的关联，它还确定了数据库的参照完整性。即在设定了关系后，用户不能随意更改建立关联的字段。参照完整性要求关系中一张表内的记录在关系的另一张表中有一条或多条相对应的记录。不同的表之间的关联是通过表的主键来确定的。因此当数据表的主键更改时，Access 会进行检查。

创建关系的过程如下：

- ① 打开“学生成绩管理”数据库，单击工具栏上的“关系”按钮，打开“关系”窗口。
- ② 单击工具栏中的“显示表”按钮，打开“显示表”对话框，选择需要建立关系的表，单击“添加”，将其加入到“关系”窗口中，直至将相关的表均加入到“关系”窗口中。关闭“显示表”对话框，结果如图5.14所示。
- ③ 在如图5.14所示的“关系”窗口中，从一个表中将要建立关系的字段拖曳到其他表中的相关字段上，结果如图5.15所示。



图 5.14 “关系”窗口

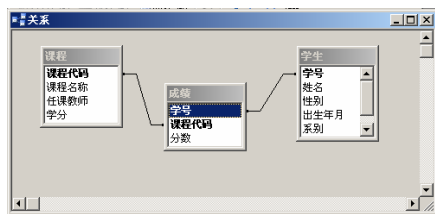


图 5.15 关系建立后窗口

- ④ 单击“关闭窗口”按钮，关闭“关系”窗口，返回“学生成绩管理”数据库窗口。当需要确定是否保存对关系图表布局的更改时，单击“是”按钮。

注意：Access 需要两个具有完全相同数据类型的字段来参与关系。在字段数据类型为“数字”的情况下，两个字段的“字段大小”属性必须相同。例如，不能在自动编号类型的字段（使用长整数数据类型）和包含字节、整型、单精度型、双精度型或者货币型数据的字段之间创建关系。在另一方面，Access 允许通过具有不同长度的文本字段将两个表联系起来，原则上，文本字段之间建立关系时应该使用相同长度的字段。

创建一个新关系时，拖放操作的顺序是相当重要的。必须从一对多关系中的一方将字段拖到多方。这个次序可以保证作为关系中一方的主表（或者说基表）出现在“表/查询”列表中，而多方出现在“相关表/查询”列表中。

5.3.4 编辑和删除关系

用户可以编辑已有的关系，或删除不需要的关系。右键单击连线，选择“删除”，可删除关系；双击关系连线，可编辑关系。编辑关系如图5.16所示。

在数据库图表中创建关系时，默认实施表之间的参照完整性。已实施的关系确保每一个在外键列中输入的值与相关主键列中的现有值相匹配。可以通过编辑关系的属性，来更改参照完整性实施所基于的条件。

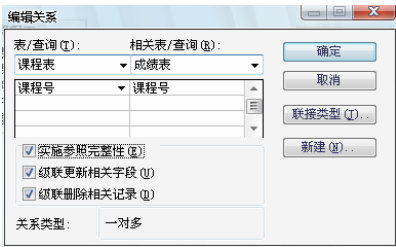


图 5.16 “编辑关系”对话框

1. 实施参照完整性

参照完整性是一个规则的系统，Microsoft Access 使用这些规则保证在相关表中记录间的关系是有效关系，并保证用户不会意外地删除或更改相关的数据。选中该复选框可以为关系实施参照完整性，但前提是应用以下条件：主表的匹配字段必须为主键或具有唯一索引，而且匹配字段要具有相同的数据类型，同时两个表都必须保存在同一个 Access 数据库中。如果清除该复选框，则允许更改可能会破坏参照完整性规则的相关表。

2. 级联更新相关字段

选择“实施参照完整性”，然后选中“级联更新相关字段”，可以在主表的主键值更改时，自动更新相关表中的对应数值。选择“实施参照完整性”，然后清除“级联更新相关字段”，则只要相关表中有相关记录，主表中的主键值就都不能更改。

3. 级联删除相关字段

选择“实施参照完整性”，然后选中“级联删除相关记录”，即可在删除主表中的记录时，自动地删除相关表中的有关记录。选中“实施参照完整性”，然后清除“级联删除相关记录”，则只要相关表中有相关记录，就不能删除主表中的记录。

5.4 修改数据库结构

在创建数据库及表，设定表间关系、表的索引、表的主键之后，随着用户对自己所建数据库的用途更加深入了解，有时候会发现，当初所建数据库及表有很多需要改动的地方，这就涉及修改数据库、表的工作。

5.4.1 对表的操作

在使用中，用户可能会需要对已有的表进行修改，在修改之前，用户应该考虑全面。因为表是数据库的核心，它的修改将会影响到整个数据库。不能修改打开的或正在使用的表，必须先将其关闭。如果在网络环境下使用，必须保证所有用户均已退出使用。关系表中的关联字段也是无法修改的，如果确实要修改，必须先将关联去掉。

1. 表的备份和复原

如果用户需要修改多个表，那么最好将整个数据库备份。数据库的备份，与 Windows 下

普通文件的备份一样，复制一份即可。复制方法很多而且简单，其中一种好方法就是“文件”菜单下的“另存为”选项。

2. 删除表

如果数据库中含有用户不再需要的表，可以将其删除，删除数据库表须慎重考虑。

3. 更改表名

有时需要将表名更改，使其具有新的意义，以方便数据库的管理。通过选中需要改名的表，再右键单击“重命名”可以很快地更改表名。

4. 设置表属性

有两种类型的表属性可以用于 Access 数据库，一种是表对象属性，另一种是表定义属性。

表对象属性包括名称、拥有者、创建日期、最后修正日期，以及诸如“隐藏”、“或复制”等特性。在设计视图中打开表，同时打开“属性”对话框，可以查看和定义当前设计表的属性。如果要更改默认的表设计属性，可选择“工具”菜单下的“选项”命令，打开“表/查询”选项卡进行设置。如图 5.17 所示。

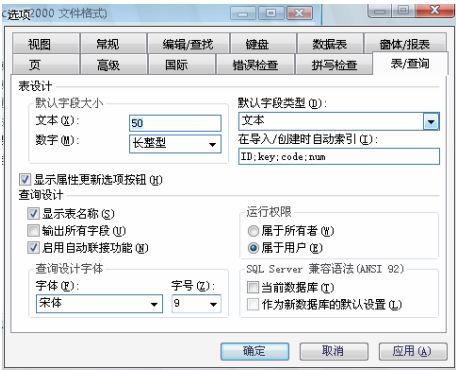


图 5.17 表设计属性窗口

5.4.2 对字段的操作

当用户对字段名称进行修改时，可能影响到字段中存放的一些相关数据。如果查询、报表、窗体等对象中使用了这个更名的字段，那么这些对象中也要相应地更改字段名的引用。更名的方法有两种，一是设计视图，二是数据表视图。

打开“学生成绩管理”数据库，单击选择“表”选项，单击需要修改结构的表，再单击设计按钮，如图 5.18 所示为打开的学生表。在已有的表中不仅能添加字段，而且还可以删除字段。



图 5.18 学生表

1. 加入新字段

将鼠标移动到该字段的标题上，单击鼠标右键，在快捷菜单中单击“插入行”，这样就在该字段前面插入了一个新的字段。

2. 删除字段

将鼠标移动到该字段的标题处，单击鼠标右键，在快捷菜单中单击“删除行”，在对话框中单击“是”按钮，则可将该字段删掉。在删除字段时要注意，删除一个字段的同时也会将表中该字段的值全部删除。

3. 调整字段位置

首先将鼠标移动到该字段的标题处，单击鼠标右键选中这个字段，等整行都变成黑色后，按住鼠标左键拖动字段到指定位置便可。

4. 修改字段属性

用户可以在设计表结构之后，重新更改字段的属性。其中最主要的是更改字段的数据类型和字段长度。

5.4.3 对表的行与列操作

1. 行操作

行操作可以调整行高，通过对话框方式或直接用鼠标完成此操作。

2. 列操作

由于屏幕大小限制，有时需要隐藏某些字段。隐藏列的操作十分简单：使某一列宽为 0 即可将该列隐藏。恢复隐藏列的操作须在数据视图下选择“格式”菜单下的“取消隐藏列”命令，弹出“取消隐藏列”对话框，如图 5.19 所示。要取消隐藏哪些列，只需选择这些列便可。

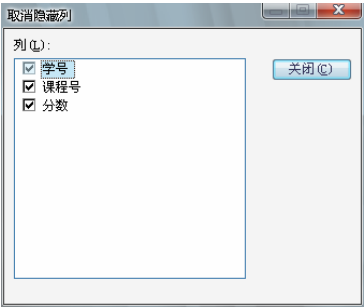


图 5.19 “取消隐藏列”窗口

5.5 表中数据的操作

5.5.1 普通数据的添加、修改和删除

在一个空表中输入数据时，只有第一行中可以输入。当要给某个字段输入内容时，只需将鼠标指向该字段，单击左键，然后便可输入。也可使用键盘上的左、右方向键移动光标。如果输入时出现错误，首先选中所要删除的数据，然后按键盘上的“Delete”键即可将原来的值删掉。

1. 文本、数字、货币型数据输入

该类数据直接在单元格中输入。

2. 是/否型数据输入

选择复选框表示“是”，不选择表示“否”。

3. 日期/时间型数据输入

年、月、日顺序，中间用“-”或者“/”分隔。例如“1992/12/23”或者“1992-12-23”。

5.5.2 图片、声音和影像的输入

要在数据表中插入图片、声音和影像。首先要在设计视图中把需要输入这些数据的字段的数据类型定义为“OLE 对象”。然后在数据表视图中用鼠标右键单击该字段，在弹出的菜单中选择“插入对象”，这时出现“插入对象”窗口，在窗口中选择要插入的对象的类型或要插入的对象的文件名。

若要插入的对象是在插入时才建立的，就需要选中“新建”单选框，并在对象类型这个列表栏中选择插入对象，要插入图片就在这个列表栏中选择“图片”，要插入影像就在这个列表框中选择“影像剪辑”，然后单击“确定”按钮。

例如，若需要输入一段录音，首先选择“新建”单选框，然后在列表栏中选择“音效”，单击“确定”按钮。出现“录音”对话框。

单击“录音”按钮进行录音，录音完成后，单击“文件”，选择“退出并返回表”。

如果要在这个值中加入一个已经建立好的一个图片、声音或影像文件，就需要在“插入对象”窗口中选择“由文件创建”，如图5.20所示，并通过单击“浏览”按钮，在浏览窗口中找到需要插入的文件，双击将它选定，单击“确定”按钮，退出“插入对象”窗口。



图 5.20 由文件创建对象

这时表的相应位置就出现了要插入文件的文件名，双击文件名，就可以看到或听到相应文件的内容。

例如，要插入学生的照片，首先选择“由文件创建”单选框，然后单击“浏览”按钮，选择照片文件，最后单击“确定”按钮。

5.5.3 建立超链接

如果要在表中插入超链接，需要将相应字段的字段类型定义为超链接，然后在表的数据表视图中将鼠标指向要建立超链接单元格，单击鼠标右键，在弹出的菜单中选择“超链接”，并在弹出的更下一级菜单中选择“编辑超链接”，这时弹出“插入超链接”的窗口，如图5.21所示。

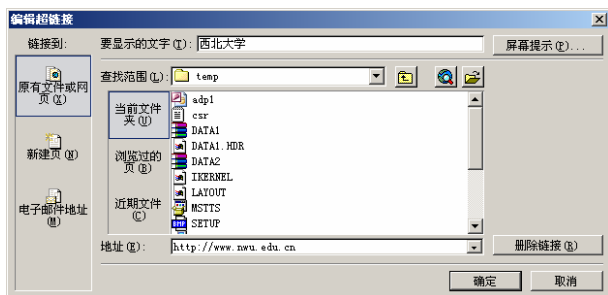


图 5.21 “插入超链接”对话框

在“要显示的文字”栏输入名称，在“地址”栏输入想要链接的网址，单击“确定”按钮就可以在表中建立一个超链接了。

这时用鼠标单击超链接的名称，系统就会自动运行 Internet 浏览器访问 Web 页面了。

5.5.4 设置数据表格形式

Access 的数据表可以有有很多种格式，鼠标单击“格式”菜单中的“数据表...”命令，弹出“设置数据表格式”对话框，如图5.22所示。

在这个对话框中就可以设置数据表的格式，有“单元格效果”、“网格线显示方法”、“背景色”、“网格线颜色”、“边框和线条样式”5个主要选项，通过这些选项所做的修改可以在“示例”栏预览结果。单击“确定”按钮就可以看到 Access 数据表的格式已经改变了。



图 5.22 “设置数据表格式”对话框

5.5.5 查找和替换数据

在使用数据库时，经常需要查看或修改表中的一些数据。如果表很大，人工逐行查找会非常麻烦，这时就需要有一个查找工具能够快速地进行查找，在 Access 中，“查找”命令可以实现这个功能。Access 还包含有替换工具，可以使用这些工具定位到与说明值匹配的每一个记录，接下来就可以随意改变其值。

在一个表中，当需要查找某个值时，用鼠标单击工具栏上的“查找”按钮，弹出“查找和替换”窗口，如图5.23所示。

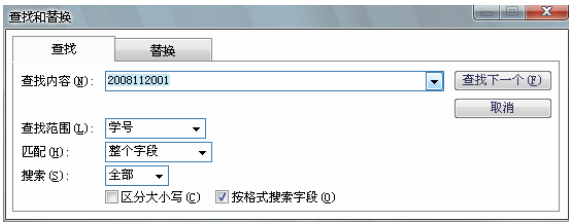


图 5.23 “查找和替换”窗口

在“查找范围”栏中选择需要查找的数据所在的范围，在“匹配”栏选择匹配的方式，在“查找内容”栏中输入所要查找的数据，最后单击“查找下一个”按钮，这样就可以在指定范围中找出第一个相应的数据值，如果这个数据值不是所需要的，再单击“查找下一个”按钮，反复执行就可以找到所需要的数据值的位置。单击“取消”按钮，可以关闭窗口。

在“查找和替换”窗口中还有一个“替换”选项卡，选择该选项卡，可以在数据表中查找某个数据并替换它。在“查找内容”中输入所需替换的内容，在“替换为”中输入替换后的内容。如果只替换一个数据值时，单击对话框上的“替换”按钮，如果要具有这个数据值的所有记录都替换，单击“全部替换”按钮，这样所有的数据值都被新数据所替换。

5.5.6 排序与筛选

当使用表中数据创建邮件列表或者打印一组特殊记录时，对表中的记录进行排序和筛选是十分有用的。

1. 排序

在窗口工具栏上有很多按钮，“升序”和“降序”用于将表中的各个记录按照一定的顺序

进行排列。单击“升序”按钮后所有记录按照从小到大的方式排列，单击“降序”按钮后所有记录按照从大到小的方式进行排列。光标位于哪个字段，就以那个字段的值作为排序依据。

2. 筛选

筛选数据是将符合筛选条件的数据记录显示出来，以便用户查看。筛选方法有4种，分别为按窗体筛选、按选定内容筛选、内容排除筛选、高级筛选/排序。

(1) 按窗体筛选

在数据表视图下，工具栏上有三个按钮：“按窗体筛选”按钮、“按选定内容筛选”按钮、“应用筛选”按钮。单击“按窗体筛选”按钮，表中只剩下一个记录，同时在方格的右侧出现一个“下拉”按钮。单击“下拉”按钮，就会发现下拉框中包括了这个字段中所有的值。

(2) 按选定内容筛选

按选定内容筛选是指先选定数据表中的值，然后在数据表中找出包含此值的记录。

例如，将光标移动到字段“性别”的一个值为“男”的方格内，单击“按选定内容筛选”按钮进行筛选，这时只有在字段“性别”中的值是“男”的记录才显示出来，结果如图5.24所示。单击“取消筛选”按钮，其余的记录又会显示出来。

学号	姓名	性别	出生日期	系别	简历	照片	个人主页
2008112001	张小强	男	1992-12-24	新闻		位图图像	http://www.news
2008112002	王华	男	1991-4-6	新闻			
2008118110	周明明	男	1992-1-8	化学			
2008118111	李想	男	1991-2-18	化学			
2008118112	马虎	男	1990-12-28	化学			
2008119001	王刚	男	1992-1-8	物理		位图图像	http://www.news

图 5.24 按选定内容筛选的结果

(3) 内容排除筛选

用户有时不需要查看某些记录，或已经查看过记录而不想再将其显示出来，这时就要用排除筛选。具体过程如下：

先在数据表中选中字段中某记录的值，然后选择“记录”菜单下的“筛选”子菜单下的“内容排除筛选”命令。例如，选择系别为“新闻”，使用内容排除筛选后的结果如图5.25所示。

(4) 高级筛选/排序

高级筛选/排序可以应用于一个或多个字段的排序或筛选。如图5.26所示，高级筛选/排序窗口分为上下两部分，上半部分是含有表的字段列表，下半部分是设计网格。

学号	姓名	性别	出生日期	系别	简历	照片	个人主页
2008118110	周明明	男	1992-1-8	化学			
2008118111	李想	男	1991-2-18	化学			
2008118112	马虎	男	1990-12-28	化学			
2008119001	王刚	男	1992-1-8	物理		位图图像	http://www.s

图 5.25 内容排除筛选的结果

字段	条件
系别	= "新闻"
性别	= "男"

图 5.26 高级筛选

① 创建筛选

要创建一个高级筛选，首先要把字段添加到用于排序和规定筛选准则的设计网格中。

② 设置筛选条件

在“条件”行中，可添加要显示记录的条件，它的设置方法与按窗体筛选的设置方法一样。

③ 筛选的使用

如图 5.27 所示是一个应用高级筛选后的结果。用户如果保存了筛选，则该筛选与表一起保存，而不作为独立的对象保存。当用户再次打开该表时，筛选不再起作用。如果用户想在一个表中使用多个筛选或永久保存一个筛选，必须将其作为一个查询保存起来。



图 5.27 应用高级筛选

④ 筛选的取消和删除

如果删除一个筛选，将会永久去掉该筛选。而取消某筛选却并非删除该筛选。可以在应用某筛选前先取消该筛选，以显示全部记录。在取消筛选后，可以重新应用它。

用户还可以取消和删除筛选：单击工具栏上的“取消筛选/排序”按钮。

若要完全删除一个筛选，则在要删除其筛选数据表中单击鼠标，切换到“高级筛选/排序”窗口，在“编辑”菜单上单击“清除网格”，最后单击工具栏上的“应用筛选”按钮。

5.6 数据的导入和导出

在 Access 中可以很方便地从外部数据源获取数据，这些外部数据源可以是文本文件，也可以是 Excel、dBASE、Sybase、Oracle、FoxPro 等文件。同时，Access 也可以导出数据库，即将 Access 数据库保存为其他数据库形式，如 FoxPro、dBASE 数据库等。

5.6.1 导入和链接

在“文件”菜单的“获取外部数据”的子菜单中，有“导入”和“链接表”两个选项。两者中的任一选项均可以实现导入或联接一个外部的数据库。虽然两者功能相近，但用法有别。“导入”将创建一个新表来保存外部数据，导入的目的在于获取数据，而不需以前的数据格式。“链接表”将在数据源和目标之间建立一个同步映像，外部数据源的修改将自动地反映到目标数据中，同时在 Access 中对链接的修改也会同步的反映到数据源中。

5.6.2 链接外部数据库

下面介绍 Access 数据库和 Visual Foxpro 数据库建立链接。

① 打开“学生成绩管理”数据库，单击“文件”菜单上的“获取外部数据”项，单击其子菜单上的“链接表”命令，弹出“链接”窗口，如图 5.28 所示。

② 将鼠标移动到“文件类型”下拉框上，单击鼠标左键，选择 ODBC 数据库，弹出“选择数据源”窗口，如图 5.29 所示。

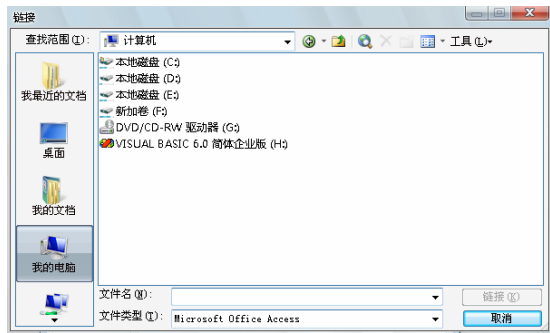


图 5.28 “链接”窗口

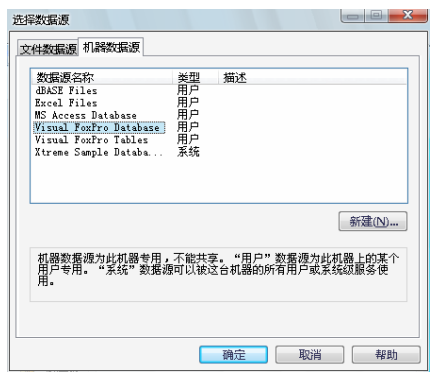


图 5.29 “选择数据源”窗口

单击“机器数据源”选项卡，选择“Visual Foxpro”选项，单击“确定”按钮。出现“配置链接”窗口，在表中选择所需连接的数据库的路径，结果如图5.30所示。

③ 单击“OK”按钮，出现“链接表”对话框，如图5.31所示。

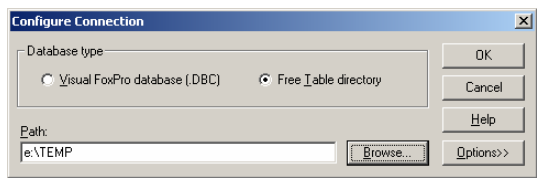


图 5.30 “配置链接”窗口

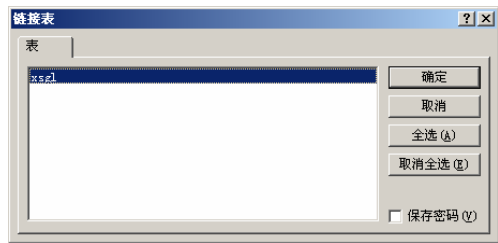


图 5.31 “链接表”对话框

④ 选择文件后单击“确定”按钮，出现“选择唯一的记录标识符”对话框，如图5.32所示。

⑤ 选择“学号”字段，单击“确定”按钮。链接建立完成，结果如图5.33所示。

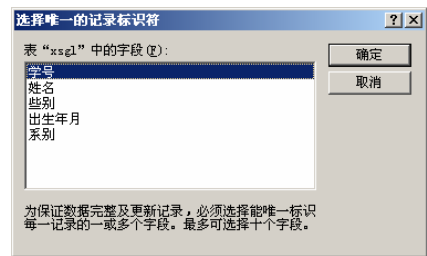


图 5.32 “选择唯一的记录标识符”对话框

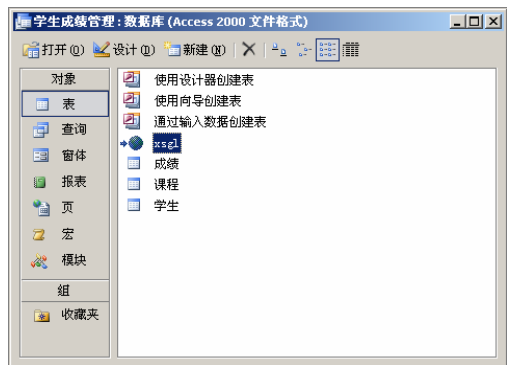


图 5.33 “学生成绩管理”数据库窗口

现在会发现在数据库窗口的表对象中已经有了一个名字为“xsxl”的表。对表中的内容进行修改，Visual Foxpro 数据库中相应表的内容也被同步修改。

5.6.3 导入外部数据库

我们已经知道如何使用“链接”的方法来获得 ODBC 数据库中的数据，现在通过“导入”方式来使用外部数据。

- ① 打开“学生成绩管理”数据库，单击“文件”菜单上的“获取外部数据”项，单击其子菜单上的“导入”命令。弹出“导入”对话框。如图5.34所示。
- ② 在“文件类型”下拉框上，选择“Microsoft Excel”，然后选择所要导入的 Excel 表格，最后单击“确定”按钮。
- ③ 弹出“导入数据表向导”对话框，如图5.35所示。

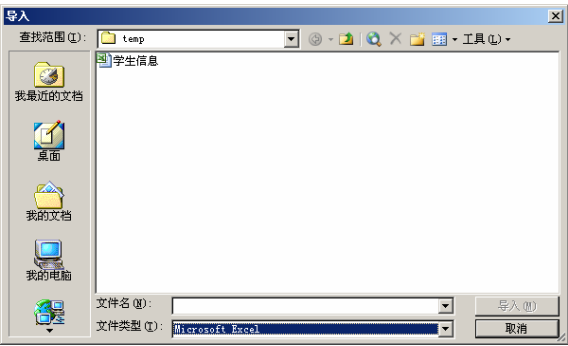


图 5.34 “导入”对话框

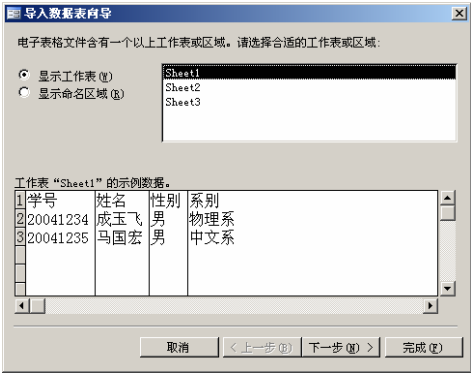


图 5.35 “导入数据表向导”对话框

- ④ 单击“下一步”按钮，在弹出的对话框中选择“第一行包含列标题”，单击“下一步”按钮。弹出“选择数据保存位置”对话框。
- 现在有两种方式可以选择。一是将数据导入到“新表”中，二是将数据导入到“现有的表”中。若要导入到“新表”中，就需要“选择字段”、“设置主键”、“定义表名”。若要导入到“现有的表”中，则需要选择“现有的数据表”。
- ⑤ 最后单击“完成”按钮，数据导入成功。

5.6.4 导出数据

- 数据不仅需要导入，有时也需要将 Access 生成的数据库中的数据导出，成为其他类型数据库文件中的数据，所以 Access 也提供了导出功能。它和导入功能正好相反。凡是能导入的数据库文件格式，也可导出成该格式。
- ① 打开“学生成绩管理”数据库，选择所要导出数据表，然后单击“文件”菜单上的“导出”项，弹出“导出”对话框。
 - ② 在导出的“保存类型”下拉框中选中“Microsoft Excel 97-2003”，然后输入保存后的文件名，最后单击“导出”按钮。
- 现在已经生成一个独立的 Excel 文件，可以随时使用 Excel 对该文件进行操作。

习 题 5

- 一、填空题
- 1. 为了在属性窗口列出 Access 类对象，应首先打开这些类对象的_____视图。
 - 2. Access 中，对数据库表中的记录进行排序时，数据类型为_____、超链接或 OLE 对象的字段不能排序。
 - 3. Access 每个记录由若干个以_____加以分类的数据项组成。

4. 如果在创建表中建立字段“姓名”，其数据类型应当是_____。
5. 一般情况下，一个表可以建立多个索引，每一个索引可以确定表中记录的一种_____。
6. 文本类型的字段最多可容纳_____个中文字。
7. 如果在创建表中建立字段“基本工资”，其数据类型应当是_____。
8. 在人事数据库中，记录人员简历，建立字段“简历”，其数据类型应当是_____。
9. 将表中的字段定义为_____，其作用是保证字段中的每一个值都必须是唯一的（即不能重复）便于索引，并且该字段也会成为默认的排序依据。
10. 在 Access 中，表间的关系有_____、“一对多”、“多对多”。
11. 有一个学生选课的关系，其中学生的关系模式为：学生（学号，姓名，班级，年龄），课程的关系模式为：课程（课号，课程名，学时），其中两个关系模式的键分别是学号和课号，则关系模式选课可定义为：选课（学号，_____, 成绩）。
12. 在数据库技术中，实体集之间的联系可以是“一对一”或“一对多”的，那么“学生”和“可选课程”的联系为_____。

二、选择题

1. Access 中，在“查找和替换”时可以使用通配符，其中可以用来通配任何单个字符的通配符是（ ）。
 - A. ?
 - B. !
 - C. &
 - D. *
2. 在 Access 数据库表中，筛选操作有多种类型，下列筛选不能通过工具栏来实现的是（ ）。
 - A. 高级筛选/排序
 - B. 按选定内容筛选
 - C. 按筛选目标筛选
 - D. 按窗体筛选
3. 在 Access 中，表中有“年龄”字段，数据类型定义为数字数据类型，并且在字段属性的有效性规则内输入“>18 and <40”，那么该字段（ ）。
 - A. 要求输入大于 18 且小于 40 的数字
 - B. 要求输入大于 18 或小于 40 的数字
 - C. 要求输入大于 18 的数字
 - D. 要求输入小于 40 的数字
4. 定义字段的默认值是指（ ）。
 - A. 不得使字段为空
 - B. 不允许字段的值超出某个范围
 - C. 在未输入数值之前，系统自动提供数值
 - D. 系统自动把小写字母转换为大写字母
5. 在 Access 中，下列有关调整数据库表中字段显示宽度和高度的叙述不正确的是（ ）。
 - A. 数据库中“数据表”视图下，用鼠标和菜单命令都可以调整字段的显示宽度
 - B. 数据库中“数据表”视图下，用鼠标和菜单命令都可以调整字段的显示高度
 - C. 数据库中表的“设计”视图下，不能调整字段的显示宽度
 - D. 数据库中表的“设计”视图下，能够调整字段的显示高度
6. 在 Access 中，向数据库中添加数据时，当输入的数据没有通过设定的有效性规则，Access 会弹出一消息框，消息框中的文字可以通过下列哪个属性来设置（ ）。
 - A. 标题
 - B. 有效性文本
 - C. 输入掩码
 - D. 默认值
7. 在 Access 数据库中，可以对表结构进行修改，下列有关删除表中字段的操作叙述不正确的是（ ）。
 - A. 在表的“设计”视图删除某一字段后，则该字段在“数据表”视图下也已经被删除
 - B. 在数据表“视图”下删除某一字段后，该字段在“设计”视图下还存在
 - C. 若该字段是某一个关系的一部分则不能删除
 - D. 在“数据表”视图下，可通过右击选中的某一列，单击快捷菜单中的“删除列”命令删除字段

8. 下列关于 OLE 对象的叙述中, 正确的是 ()。
- A. 用于输入文本数据 B. 用于处理超链接数据
C. 用于生成自动编号数据 D. 用于链接或内嵌 Windows 支持的对象
9. 在关系窗口中, 双击两个表之间的连接线, 会出现 ()。
- A. 数据表分析向导 B. 数据关系图窗口 C. 连接线粗细变化 D. 编辑关系对话框
10. 在设计表时, 若输入掩码属性设置为“LLLL”, 则能够接收的输入是 ()。
- A. Abcd B. 1234 C. AB+C D. ABa9
11. 在数据表中筛选记录, 操作的结果是 ()。
- A. 将满足筛选条件的记录存入一个新表中
B. 将满足筛选条件的记录追加到一个表中
C. 将满足筛选条件的记录显示在屏幕上
D. 用满足筛选条件的记录修改另一个表中已存在的记录
12. 在创建数据库之前, 应该 ()。
- A. 使用设计视图设计表 B. 使用表向导设计表
C. 思考如何组织数据库 D. 给数据库添加字段
13. 表是由 () 组成的。
- A. 字段和记录 B. 查询和字段 C. 记录和窗体 D. 报表和字段
14. 创建子数据表通常需要两个表之间具有 () 的关系。
- A. 没有关系 B. 随意 C. 一对多或者一对一 D. 多对多
15. 用来存储图片的字段对象是 () 类型字段。
- A. OLE B. 备注 C. 超链接 D. 查阅向导
16. Access 中, 可以在 () 命令执行后设置筛选条件。
- A. 按窗体筛选 B. 按选定内容筛选 C. 内容排除筛选 D. 高级筛选/排序
17. Access 中, 设置为主键的字段 ()。
- A. 不能设置索引 B. 可设置为“有(重复)”索引
C. 系统自动设置索引 D. 可不设置索引
18. Access 中, 要改变字段的数据类型, 应在 () 下设置。
- A. 数据表视图 B. 表设计视图 C. 查询设计视图 D. 报表图
19. 在已经建立的“工资库”中, 要在表中直接显示出我们想要看的记录, 凡是姓“李”的记录, 可用 () 的方法。
- A. 排序 B. 筛选 C. 隐藏 D. 冻结
20. 数据表中的“列标题的名称”叫做 ()。
- A. 字段 B. 数据 C. 记录 D. 数据视图
21. 在 Access 的下列数据类型中, 不能建立索引的数据类型是 ()。
- A. 文本型 B. 备注型 C. 数字型 D. 日期时间型
22. 在数据表视图中, 不可以 ()。
- A. 修改字段的类型 C. 修改字段的名称
B. 删除一个字段 D. 删除一条记录
23. 用于记录基本数据的是 ()。
- A. 表 B. 查询 C. 窗体 D. 宏

三、操作题

1. 新建名为“班级管理”的空数据库，使用表设计器，在数据库中创建“学生信息”表、“课程信息”表、“成绩”表，以存储学生的各种信息。表结构如下所示：

“成绩”表的表结构：

字段名称	数据类型	字段大小	小数位数	有效性规则
课程代号	文本	5		
学号	文本	6		
成绩	数字	单精度	1	0~100 之间

“学生信息”表的表结构：

字段名称	数据类型	字段大小
学号	文本	6
姓名	文本	10
性别	文本	2
出生日期	短日期型	
照片	OLE 对象	
家长姓名	文本	10
家庭住址	文本	50
邮政编码	文本	6

“课程信息”表的表结构：

字段名称	数据类型	字段大小
课程代号	文本	5
课程名	文本	50
任课教师	文本	8

对数据库进行如下操作：

- (1) 设置表对象“学生信息”表的出生日期字段默认值为系统日期。
- (2) 设置表对象“学生信息”表的性别字段有效性规则为男或女，同时设置相应有效性文本为“请输入男或女”。
- (3) 删除“学生信息”表中姓名字段含有“江”字的所有纪录。
- (4) 将表对象“学生信息”表导出到一个空数据库文件中，要求只导出表结构定义，导出的表命名为“学生信息表 BK”。
- (5) 建立当前数据库的表间关系，并实施参照完整性。

2. 建立一个空数据库，在数据库中新建表 emp 和 salary，表的结构和记录如下所示。

“Emp”表的表结构：

字段名	数据类型	字段大小
职工号	文本	6
姓名	文本	6
性别	文本	2
出生日期	日期/时间	
政治面貌	文本	6

“Salary”表的表结构：

字段名	数据类型	字段大小
职工号	文本	6
基本工资	数字	双精度
津贴	数字	双精度
奖金	数字	双精度
扣除	数字	双精度

对该表进行如下操作：

- (1) 录入数据。
- (2) 在设计视图中修改“Salary”表的结构，将“基本工资”字段的默认值设为 800、有效性规则设为大于等于 800 且小于等于 5000，有效性文本输入：基本工资在 800~5000 元之间。
- (3) 对“emp”表按“职工号”建立主索引，对“Salary”表按“职工号”建立主索引。定义“emp”和“salary”之间的关系，“emp”为主表，实施参照完整性。

3. 已知 A 单位要建立工资数据库，该单位的工资单结构如下：

编号	日期	姓名	性别	基本工资	加班费	妇女劳保	月奖金	工资总额	所得税	实发金额

建立空数据库，并命名为“工资库”。根据上述工资单结构，确定各字段的数据类型，建立数据库表，表的名称为“工资基本表”。

第6章 查询数据

查询是数据库管理系统中一个最为基本的功能。使用查询将不同表中的信息结合起来，提供一个相关数据项的统一视图。使用查询还可以选择记录、更新表、向表中添加新记录，最为常见的情形是使用查询功能选择一组满足指定条件的特定记录。

6.1 查询与表

在使用数据库中的数据时，并不是简单地使用某个表中的数据，而常常是将有“关系”的很多表中的数据关联起来使用，有时还可能要对这些数据进行一定的计算才使用。对于这样的要求，建立“查询”对象可以很轻松地解决，查询就是依据一定的查询条件，对数据库中的数据信息进行查找。查询的字段来自互相之间有“关系”的表，这些字段组合成一个新的数据表视图，但它并不存储任何的数据。当改变表中的数据时，查询得到的数据也会发生改变。

在运行查询时，查询所生成的数据值动态地来源于表对象，是表中数据的一个镜像。所以在查询数据表中无法加入或删除字段，也不能修改查询字段的字段名。查询结果将以工作表的形式显示出来。显示查询结果的工作表又称为结果集，它虽然与基本表有十分相似的外观，但它并不是一个基本表，而是符合查询条件的记录集合，其内容是动态的。

6.2 常见的查询

通过查询可以按照不同的方式查看、更改和分析数据。同时，用查询也可以作为窗体、报表和数据访问页的数据源。Access 中常见的查询有：选择查询、参数查询、交叉表查询、操作查询、SQL查询。

6.2.1 选择查询

选择查询是最常见的一种查询，它从一个或多个有关系的表中将满足要求的数据提取出来，并把这些数据显示在新的查询数据表中，并能对记录进行分组、总计、计数、求平均值，以及其他类型的计算。

6.2.2 参数查询

如果用户查询时需要通过在对话框中输入要查询的数据，就要创建参数查询。参数查询可以在运行查询的过程中修改查询的规则，并执行参数查询时会显示一个输入对话框以提示用户输入信息。

Access 的参数查询是建立在选择查询或交叉查询的基础之上的，在运行选择查询或交叉查询之前，为用户提供了一个设置条件的参数对话框，可以很方便地更改查询限制或对象。当然不仅可以建立单个参数的查询，还可以建立多字段参数查询。

例如，可以设计用它来提示输入两个日期，然后 Access 检索在这两个日期之间的所有记录。

6.2.3 交叉表查询

Access 支持一种特殊类型的总计查询，称为交叉表查询，交叉表查询允许用户精确确定汇总数据如何在屏幕上显示。利用该查询，可以在类似电子表格的格式中查看计算值，也能够计算数据的总计、平均值、计数或其他类型的操作。交叉表查询以传统的行列电子数据表形式显示汇总数据并且与 Excel 数据透视表密切相关。

6.2.4 操作查询

使用操作查询只需进行一次操作就可对许多记录进行更改和移动。操作查询有四种：删除查询、更新查询、追加查询、生成表查询。

1. 删除查询

从一个或多个表中删除一组记录。例如，可以使用删除查询来删除已经离校的学生信息。使用删除查询，会删除整个记录。

2. 更新查询

对一个或多个表中的一组记录做全局的更改。例如，给所有职工的工资增加 200 元。使用更新查询，可以更改已有表中的数据。

3. 追加查询

将一个或多个表中的一组记录添加到一个或多个表的末尾。例如，新入校学生的信息存放在新生表中，可以通过追加查询将其追加总表中。

4. 生成表查询

生成表查询主要用于创建表以导出到其他数据库中。生成表查询可以根据一定的准则来新建表格，然后再将所生成的表导出到其他数据库中，或者在窗体和报表中加以利用。

6.2.5 SQL查询

SQL 查询是用户使用 SQL 语句直接创建的一种查询。实际上，Access 所有的查询都可以认为是一个 SQL 查询，因为 Access 就是以 SQL 语句为基础来实现查询的功能。

1. SQL查询的分类

SQL 查询可以分为以下四类：联合查询、传递查询、数据定义查询和子查询。

联合查询：该查询使用 UNION 运算符来合并两个或更多选择查询的结果。

传递查询：SQL 特定查询，可以用于直接向 ODBC 数据库服务器发送命令。通过传递查询，可以直接使用服务器上的表，而不是由 Access 数据库引擎处理后的数据。

数据定义查询：包含数据定义语言（DDL）语句的 SQL 特有查询，这些语句可用来创建或更改数据库中的对象。

子查询：在另一个选择查询或动作查询内的 SQL 语句。

2. SQL语句的使用场合

SQL 语句可以在 Access 中的很多场合使用，只要这些场合能够输入表、查询或字段的名称即可。某些情况下，Access 会自动填入 SQL 语句。例如，当使用向导创建窗体或报表以便从多个表中获得数据时，Access 会自动创建一个 SQL 语句，并将该语句用作窗体或报表的“记

录源”属性。在通过向导创建列表框或组合框时，Access 会创建一个 SQL 语句，并将该语句用作列表框或组合框的“行来源”属性设置。

如果不使用向导，也可以为“记录源”或“行来源”属性生成一个 SQL 语句，方法是单击这些属性旁的任意一个“生成”按钮，然后在查询“设计”视图中创建查询。

6.3 创建选择查询

6.3.1 利用查询设计视图建立

直接使用查询设计视图建立查询有利于更好地理解数据库中表之间的关系，这对建立一个优秀的数据库非常有用。

1. 创建过程

建立一个“学生成绩表”查询，通过这个查询可以显示学生的学习成绩，包括“学号”、“姓名”、“课程名称”、“任课教师”、“成绩”等字段。

① 首先打开“学生成绩管理”数据库，然后单击“对象”列表中的“查询”项，如图 6.1 所示。

② 在创建方法列表中双击“在设计视图中创建查询”项。出现“选择查询”窗口和“显示表”对话框。

在“显示表”对话框中，“表”选项卡中列出了所有的表，“查询”选项卡中列出了所有的查询，而“两者都有”可以把数据库中所有“表”和“查询”对象都显示出来，这样有助于从选择的表或查询中选取新建查询的字段。

单击“显示表”对话框上的“两者都有”选项，在列表框中选择需要的表或查询。然后单击对话框上的“添加”按钮，这样就可以将表添加到查询窗口中。

③ 关闭“显示表”窗口，回到“选择查询”窗口，如图 6.2 所示。



图 6.1 “学生成绩管理”窗口



图 6.2 “选择查询”窗口

选择查询窗口分为两大部分，上部是“表/查询显示”窗口，下面是“示例查询设计”窗口。

“表/查询显示”窗口用于显示查询的数据来源（包括表、查询等），方便选择查询字段。

“示例查询设计”窗口则是用来显示查询中所用到的查询字段和查询准则。“示例查询设计”窗口中有如下行标题。

- ◆ 字段：查询工作表中所使用的字段名。
- ◆ 表：该字段所来自的数据表。

- ◆ 排序：是否按该字段排序。
- ◆ 显示：该字段是否在结果集工作表中显示。
- ◆ 条件：查询条件。
- ◆ 或：用来提供多个查询条件。

④ 添加或删除目标字段。在查询设计表格中添加的字段称为“目标字段”，添加目标字段有两种方法。

- ◆ 第一种方法：在表格中选择一个空白的列，单击第一行对应的一格，格子的右边出现一个带下箭头的按钮，单击这个按钮出现下拉框，在下拉框中就可以选择相应的目标字段。
- ◆ 第二种方法：选中目标字段所在的表，然后在它的列表框中找到需要添加的字段，将鼠标移动到列表框中标有这个字段的选项上，按住鼠标左键，这时鼠标光标变成一个长方块，拖动鼠标将长方块拖到下方查询表格中的一个空白列，放开鼠标左键，这样就可以将目标字段添加到查询表格中。

如果要删除一个目标字段，将鼠标移动到要删除的目标字段所在列的选择条上，光标会变成一个向下的箭头，单击鼠标左键将这一列都选中，按下键盘上的“Delete”键，选中的目标字段就被删除。

现在加入“学号”、“姓名”、“课程名”、“任课教师”、“成绩”字段，如图6.3所示。

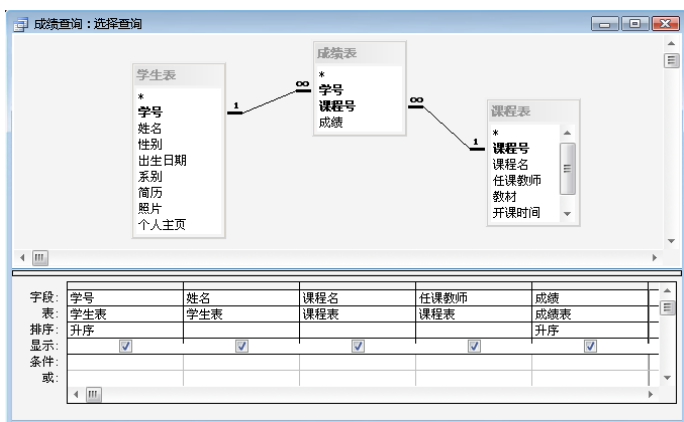


图 6.3 设置结果

⑤ 查看查询的数据表视图。

现在，已经把需要的字段都添加到了查询中，可以来看看建立的“查询”的结果。“查询”可以在设计视图和数据表视图中切换。Access 中，视图之间切换非常简单。只要将鼠标移动到工具栏左上角的第一个工具按钮处，就会弹出“视图”提示标签，单击就可以从查询设计视图切换到数据表视图。

⑥ 保存查询。查询已经基本建立成功，现在需要进行查询的保存。单击“文件”，选择“保存”，然后输入查询的名称，单击“确定”按钮。

2. 设置查询准则

查询设计视图中的准则就是查询记录应符合的条件。它与在设计表时设置字段有效性规则的方法相似。

- (1) 使用准则表达式，准则表达式中相关运算符如表 6.1 所示。
- (2) 在表达式中使用日期与时间，相关内部函数如表 6.2 所示。

表 6.1 准则表达式

运算符	功 能	举 例
And	与操作	“ A ” And “ B ”
Or	或操作	“ A ” Or “ B ”
Between...And	指定范围操作	Between “ A ” And “ B ”
In	指定枚举范围	In(“ A, B, C ”)
Like	指定模式字符串	Like “ A?[A~f]#[!0~9]* ”

表 6.2 日期函数

函 数	功 能	函 数	功 能
Date()	返回系统当前日期	Weekday()	返回日期中的星期数
Year()	返回日期中的年份	Hour()	返回时间中的小时数
Month()	返回日期中的月份	Now()	返回系统当前的日期与时间
Day()	返回日期中的日数		

在准则表达式中使用日期/时间时，必须要在日期值两边加上“#”。例如下面的写法：

#Feb12,98#、#2/12/98#、#1221998#。

- (3) 在表达式中进行计算，相关运算符如表 6.3 所示。

表 6.3 基本运算符

运算符	功 能	举 例
+	两个数字型字段值相加，两个文本字符串连接	A+B
-	两个数字型字段值相减	A-B
*	两个数字型字段值相乘	A*B
/	两个数字型字段值相除	A/B
\	两个数字型字段值相除四舍五入取整	A\B
^	A 的 B 次幂	A^B
Mod	取余，A 除以 B 得余数	Mod(A,B)
&	文本型字段 A 和 B 连接	A&B

给查询添加选择准则，有两个问题需要考虑：首先是为哪个字段添加准则，其次就是要为这个字段添加什么样的准则。如果只想看物理系学生的考试成绩，很明显就是为“系别”字段添加准则，而添加的准则就是“系别”字段的值只能等于“物理系”。限定了这两个条件，就很容易实现任何一种选择准则。

在查询中添加准则的具体过程如下：

① 打开“学生成绩”查询，然后将“学生”表中的“系别”字段加入查询，由于不需要将该字段的值显示在数据表中，将它的“显示”属性定为“否”（即去掉勾选），如图6.4 所示。

② 在“系别”字段的“条件”属性中写上“=“物理系””，如图6.5 所示。

有时候需要对查询记录中的多个信息同时进行限制，就需要将所有这些限制规则全部添加到需要的字段上，只有完全满足限制条件的那些记录才能显示出来。

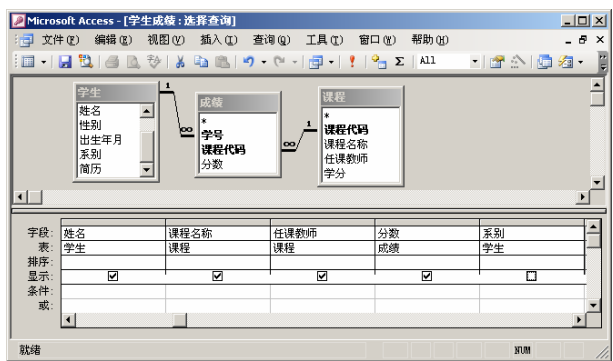


图 6.4 添加系列字段

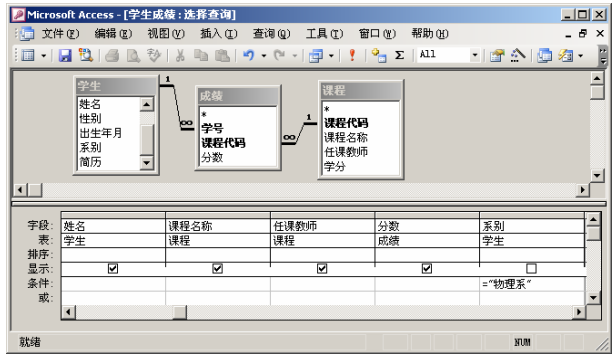


图 6.5 设置系列准则

6.3.2 利用查询向导建立

- 利用查询向导创建查询的基本步骤如下：
- ① 首先打开“学生成绩管理”数据库，单击“对象”列表中的“查询”项。
 - ② 在创建方法列表中双击“使用向导创建查询”项，弹出“简单查询向导”对话框，如图6.6所示。
 - ③ 在“简单查询向导”对话框上选择新建查询中所需的字段名称。

由于字段可能在不同的表或查询中，先要在“表/查询”下拉列表中选择需要的表或查询，然后在“可用字段”列表框中选择需要的“字段”，本例中选择后，在“选定的字段”列表框中显示“课程名称”和“任课教师”，如图6.6所示。

- ④ 将所有需要的字段全部选定以后，单击“下一步”按钮，在下一个窗口中为新建的查询取名，并单击“完成”按钮，就可以创建一个新的查询。

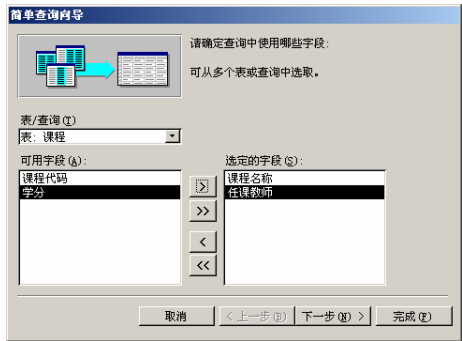


图 6.6 “简单查询向导”对话框

6.4 创建参数查询

参数查询可以在运行查询的过程中自动修改查询的规则，在执行参数查询时会显示一个输入对话框提示用户输入信息。

当需要对某个字段进行参数查询时，设置过程如下：

- ① 打开“学生成绩”查询，切换到设计视图。
- ② 在作为参数使用的字段下的“条件”单元格中，在方括号内输入相应的提示文本，如图6.7所示。

注意：不能省略方括号。

输入完毕后，将查询切换到数据表视图，这时在屏幕中就会出现一个对话框，如图6.8所示。

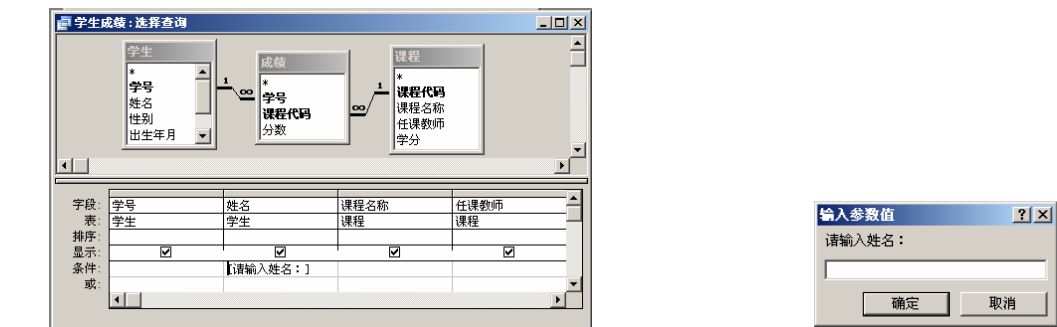


图 6.7 设置输入参数

图 6.8 “输入参数”对话框

此时，输入条件就可以看到满足条件的记录。不仅可以建立单个参数的查询，还可以根据需要同时为多个字段建立参数查询。

6.5 创建交叉表查询

交叉表查询是一种 选择查询，在运行交叉表查询时，结果显示在一个数据表中，该数据表的结构不同于其他类型的数据表。与显示相同数据的简单选择查询相比，交叉表查询的结构让数据更易于阅读。

使用交叉表查询向导是创建交叉表查询最快、最简单的方法。该向导会为用户完成大部分工作，但有些选项它没有提供。使用向导创建一个交叉表查询的步骤如下：

- ① 打开“学生成绩管理”数据库，然后单击“对象”列表中的“查询”项。
- ② 在数据库窗口中单击“新建”按钮，并在弹出的窗口中选择“交叉表查询向导”项，然后单击“确定”按钮，出现“交叉表查询向导”窗口。选择“查询：成绩查询”，在“视图”列表中选择“两者”，如图6.9所示。
- ③ 单击“下一步”按钮，出现“选择行标题字段”窗口，如图6.10所示。

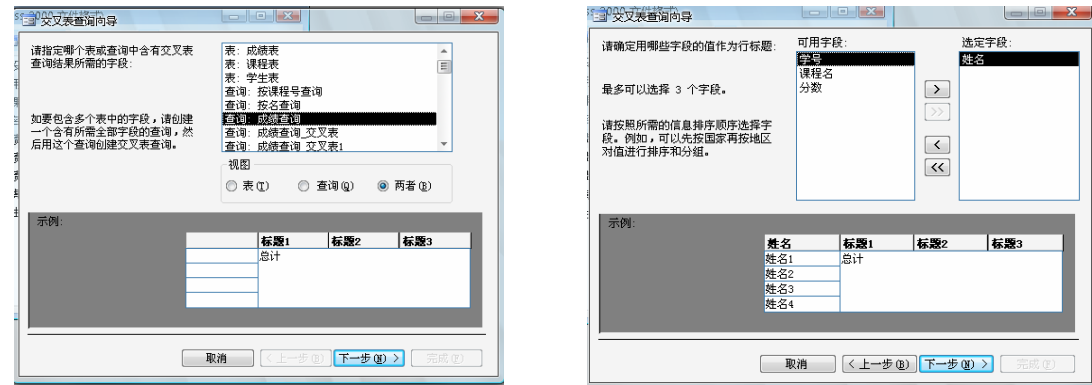


图 6.9 选择数据源

图 6.10 “选择行标题字段”窗口

在创建交叉表查询时，需要指定哪些字段包含行标题，哪些字段包含列标题，以及哪些字段包含要汇总的值。在指定行标题时，最多可使用三个字段。使用的行标题越少，交叉表查询数据表就越容易阅读。本例选择“姓名”作为行标题字段。

④ 单击“下一步”按钮。出现“选择列标题字段”窗口，如图6.11所示。

慎重选择列标题字段，当列标题的数量保持相对较少时，交叉表数据往往更容易阅读。在确定要用作标题的字段后，应使用具有最少明确值的字段来生成列标题。本例选择“课程名”作为列标题字段。

⑤ 单击“下一步”按钮，选择在表中的交叉点计算出什么数值，所选字段的数据类型将决定哪些函数可用。如图6.12所示。

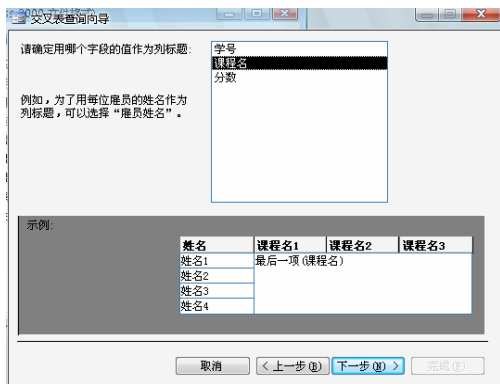


图 6.11 “选择列标题字段”窗口

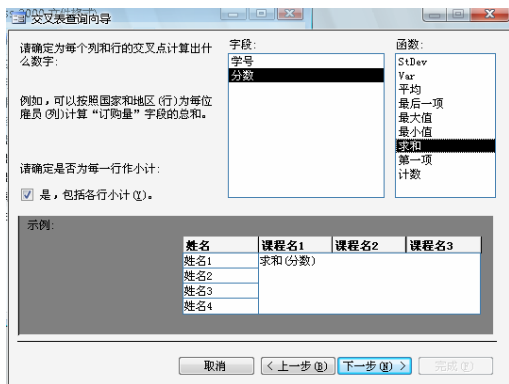


图 6.12 选择计算内容

在“字段”选项中选择“分数”，“函数”选项中选择“求和”，然后选定“是，包含各项小计”。如果包含行小计，则交叉表查询中有一个附加行标题，该标题与字段值使用相同的字段和函数。包含行小计还会插入一个对其余列进行汇总的附加列。

⑥ 为新建的查询取名，并单击“完成”按钮。这样一个交叉表查询就完成了，如图6.13所示。

姓名	总计 分数	计算机	数学	物理	英语
李想	300	80	67	83	70
李想	218	68	80		70
马虎	297	70	80	80	67
王华	150	83			67
张小强	321	86	70	80	85
周明明	225		80	78	67

图 6.13 运行结果

6.6 操作查询

操作查询用于同时对一个或多个表执行全局数据管理操作。操作查询可以对数据表中已有的数据内容进行编辑，对符合条件的数据进行批量修改。

6.6.1 删除查询

删除查询是所有操作查询中最危险的一个。删除查询是将整个记录全部删除而不只是删除查询所使用的字段，查询所使用的字段只是用来作为查询的条件。可以从单个表删除记录，也可以通过级联删除相关记录而从相关表中记录被删除。

创建删除查询的过程如下：

- ① 通过设计视图创建“选择查询”视图，如图 6.14 所示，例中选择成绩表作为数据源。
- ② 鼠标右键单击“选择查询”视图的空白处，在弹出的快捷菜单中，选择“查询类型”下的“删除查询”。输入删除条件，例中删除学号为“2008112001”的学生的成绩，结果如图 6.15 所示。

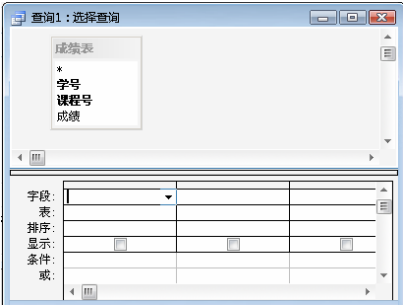


图 6.14 创建“选择查询”视图

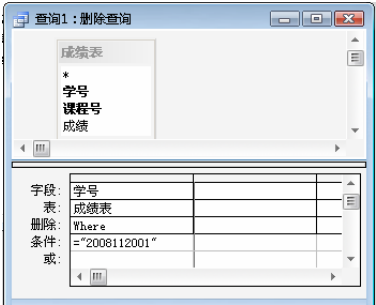


图 6.15 删除查询

- ③ 保存删除查询。
- ④ 运行删除查询。运行时，系统会出现如图 6.16 的提示信息，单击“是”按钮。

打开成绩表，会发现学号为“2008112001”的学生的所有成绩已经被删除了。

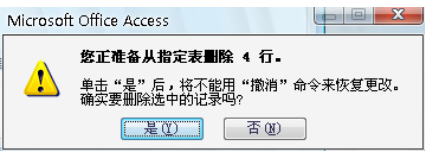


图 6.16 删除确认窗口

6.6.2 更新查询

更新查询用于同时更改多个记录中的一个或多个字段值，用户可以通过添加条件来选择所要更新的记录。大部分更新查询可以用表达式来规定更新规则。规定更新规则的常用表达式如表 6.4 所示。

给所有 80 分以下的学生的成绩加 5 分的更新查询设置如图 6.17 所示。

打开成绩表，会发现 80 分以下的学生成绩均增加了 5 分。

表 6.4 规定更新规则的常用表达式

字段类型	表达式	结 果
货币	[单价]*1.15	把单价增加 15%
日期	#9/19/09#	把日期更改为 2009 年 9 月 19 日
文本	“完成”	把数据更改为“完成”
文本	“平均” & [单价]	把字符“平均”连接到“单价”字段数据
是/否	Yes	把特定的“否”数据更改为“是”

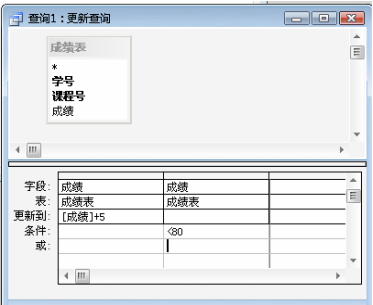


图 6.17 更新查询

6.6.3 生成表查询

生成表查询可以从一个或多个表/查询的记录中制作一个新表。

建立一个生成表查询，将成绩在 80 分之上的学生存放到“80 分以上学生情况”表中。操作步骤如下：

① 打开查询设计视图，并将“学生表”、“课程表”和“成绩表”添加到查询设计视图上半部分的窗口中，并选择新表的构成字段，设置选择条件，如图6.18所示。

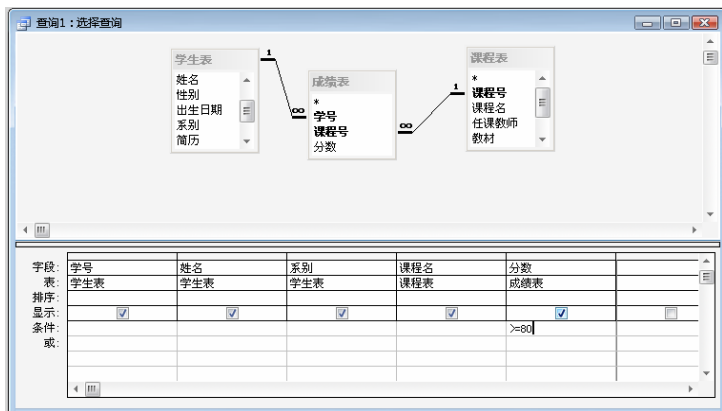


图 6.18 设置选择条件

② 单击工具栏上的“查询类型”按钮右侧的向下箭头按钮，然后从下拉列表中选择“生成表”选项，弹出“生成表”对话框。在“表名称”文本框中输入“80分以上学生情况表”，如图6.19所示。

③ 单击“确定”按钮，保存查询，如图6.20所示。



图 6.19 “生成表”对话框

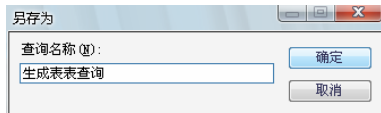


图 6.20 保存查询

④ 运行查询，可以发现产生了一个新表，即 80 分以上学生情况表。

6.6.4 追加查询

当用户要把一个或多个表的记录添加到其他表时，就要用到追加查询。追加查询可以从另一个数据库表中读取数据记录并向当前表内添加记录，由于两个表之间的字段定义可能不同，追加查询只能添加相互匹配的字段内容，而那些不匹配的字段将被忽略。

建立一个追加查询，将选课成绩在 75~80 分之间的学生添加到已建立的“80 分以上学生情况表”中，操作步骤如下：

① 打开查询设计视图，并将“学生表”、“课程表”和“成绩表”添加到查询设计视图上半部分的窗口中，并选择构成字段，设置选择条件，如图 6.21 所示。

② 单击工具栏上的“查询类型”按钮右侧的向下箭头按钮，然后从下拉列表中选择“追加查询”选项，这时屏幕上显示“追加”对话框。在“表名称”文本框中输入“80 分以上学生情况表”或从下拉列表中选择“80 分以上学生情况表”，表示将查询的记录追加到“80 分以上学生情况表”中；选中“当前数据库”单选按钮，如图6.22所示。

③ 保存并运行查询，发现 75~80 分之间的成绩已经追加。

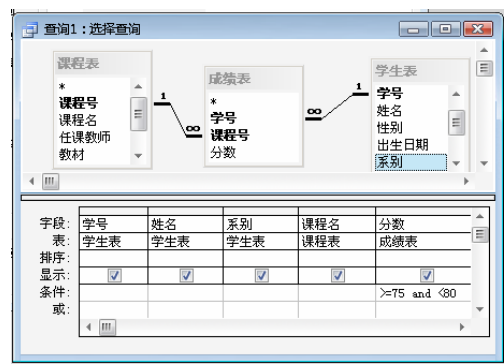


图 6.21 设计视图

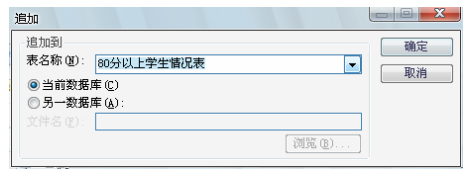


图 6.22 追加对话框

执行操作查询时，可能更改许多记录，并且在执行操作查询后，不能撤销刚做过的更改。因此，在使用操作查询时应注意在执行之前，最好单击工具栏上的“视图”按钮，预览即将更改的记录，如果预览到的记录就是要操作的对象，再执行操作查询，这样可防止误操作。另外，在使用操作查询之前，应该备份数据。

6.7 Access SQL查询

SQL（Structured Query Language）意思为结构化查询语言。SQL 语言的主要功能就是各种数据库之间建立联系，进行沟通。按照 ANSI（美国国家标准协会）的规定，SQL 是为关系型数据库管理系统的标准语言。SQL 语句可以用来执行各种各样的操作，例如，更新表中的数据，从表中提取数据等。目前，绝大多数流行的关系型数据库管理系统，如 Oracle、Sybase、Microsoft SQL Server、Access 等都采用了 SQL 语言标准。虽然很多数据库都对 SQL 语句进行了再开发和扩展，但是包括 SELECT、INSERT、UPDATE、DELETE、CREATE 及 DROP 在内的标准的 SQL 命令仍然可以用来完成几乎所有的数据库操作。

6.7.1 SQL的特点

SQL 语言之所以能够为用户所接受，成为国际标准，主要原因在于它是一种综合的、通用的、功能强大且简单易学的语言。SQL 语言集数据查询、数据操纵、数据定义和数据控制功能于一体，充分体现了关系数据语言的优点。其主要包括以下特点：

- （1）一体化特性
包括数据定义、数据查询、数据操纵和数据控制等方面的功能，可以完成数据库活动中的全部工作。
- （2）高度非过程化
它无须告诉计算机如何去做，而只需要用户描述清楚要做什么，SQL 语言就可以将要求交给系统，自动完成全部工作。
- （3）语言简洁
虽然 SQL 语言功能很强，但它只有为数不多的几条命令，另外 SQL 的语法也非常简单，它很接近自然语言，因此容易学习掌握。
- （4）支持多种使用方式
SQL 语言可以直接以命令方式交互使用，也可以嵌入到程序设计语言中以程序方式使用。

现在很多数据库应用开发工具都将 SQL 语言直接融入到自身的语言之中，使用起来更方便。尽管 SQL 的使用方式不同，但 SQL 语言的语法基本是一致的。

总之，SQL 语言功能强大、语言简洁。完成数据定义、数据操纵、数据控制的核心功能只用了 9 条语句：CREATE、DROP、ALTER、SELECT、INSERT、UPDATE、DELETE、GRANT、REVOKE，如表 6.5 所示。

表 6.5 SQL 操作的语句

SQL 功能	语 句
数据查询	SELECT
数据定义	CREAT, DROP, ALTER
数据操纵	INSERT, UPDATE, DELETE
数据控制	GRANT, REVOKE

6.7.2 SQL数据库的体系结构

SQL 语言支持数据库三级体系结构，如图6.23所示。

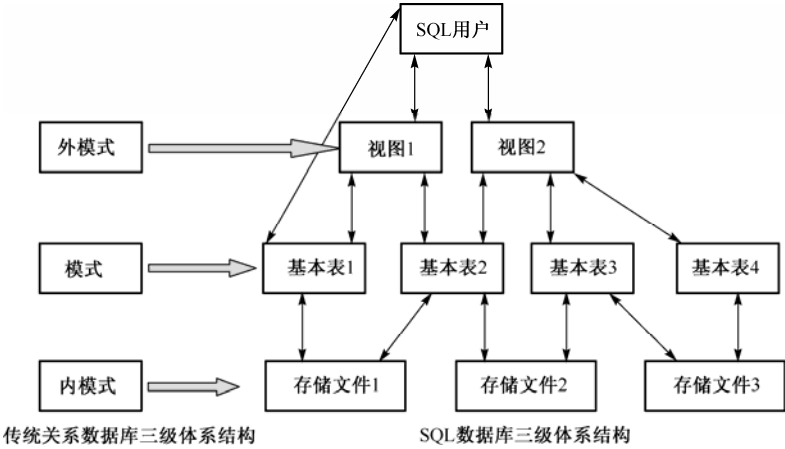


图 6.23 SQL 语言的体系结构

SQL 语言的三级体系结构和传统的数据库体系结构没有本质的不同，只不过有些术语有所不同。在 SQL 中，模式对应于基本表，内模式对应于存储文件，外模式对应于视图和部分基本表。元组对应于表中的行，属性对应于表中的列，SQL 数据库具有如下基本特点：

- (1) 一个 SQL 数据库是表的集合。
- (2) 一个 SQL 表由若干行构成，一行是列的序列，每列对应一个数据项。
- (3) 一个表可以带若干的索引，索引也存放在存储文件中。
- (4) 存储文件的逻辑结构组成了关系数据库的内模式，存储文件的物理结构是任意的，对用户是透明的。
- (5) 一个表可以是一个基本表，也可以是一个视图。基本表是实际存储在数据库中的表；视图是从一个或几个基本表或其他视图导出的表，数据库中只存放视图的定义，而不存放视图对应的数据，视图是一张虚表。
- (6) 一个基本表可以存放在一个或多个存储文件中；一个存储文件可以存放一个或多个基本表。每个存储文件对应外部存储器上的一个物理文件。
- (7) SQL 用户可以是应用程序，也可以是终端用户。

6.7.3 Access SQL的特点

1. 基本数据类型

在 Access 中使用 SQL 语句时，可以使用的数据类型及其特点如表 6.6 所示。

表 6.6 基本数据类型及其特点

数据类型	存储大小	说 明
BINARY	每字符 1 字节	任何类型的数据都可存储在这种类型的字段中
BIT	1 字节	Yes/No(True/False, ON/OFF, -1/0) 只包含两值之一的字段
TINYINT	1 字节	0~255 之间的整数
MONEY	8 字节	-922,337,203,685,477.5808~-922,337,203,685,477.5807 之间
DATETIME	8 字节	100~9999 年的日期或时间数值
UNIQUEIDENTIFIER	128 位	用于远程过程调用的唯一识别数字
REAL	4 字节	单精度浮点数
FLOAT	8 字节	双精度浮点数
SMALLINT	2 字节	-32,768~32,767 的短整型数
INTEGER	4 字节	-2,147,483,648~2,147,483,647 的长整型数
DECIMAL	17 字节	可以定义精度 (1-28) 和符号 (0-定义精度)。默认精度和符号分别是 18 和 0
TEXT	每字符 2 字节	从 0 到最大 2 GB
IMAGE	视实际需要而定	从 0 到最大 2 GB 用于 OLE 对象
CHAR	每字符 2 字节	长度 0~255 字符

- (1) BINARY：二进制型，可以指定长度，否则默认长度为 510。
- (2) BIT：位型，可用格式 (Yes/No, True/False, On/OFF)。
- (3) TINYINT, BYTE：数字-字节，不要指定长度和精度，否则会报错。
- (4) MONEY, CURRENCY：货币型，不要指定长度和精度，否则会报错。
- (5) DATETIME：日期时间型，不要指定长度，否则会报错。
- (6) UNIQUE/DENTIFIER：用于远程过程调用的唯一识别数字。
- (7) REAL, SINGLE：数字-单精度型，不要指定长度和精度，否则会报错。
- (8) FLOAT, DOUBLE, NUMBER：数字-双精度型，不要指定长度和精度，否则会报错。
- (9) SMALLINT, SHORT：数字-整型，不要指定长度和精度，否则会报错。
- (10) INTEGER, INT, LONG：数字-长整型，不要指定长度和精度，否则会报错。
- (11) DECIMAL, NUMERIC：数字-小数，可以指定长度和精度，如只指定长度，那精度默认为 0，如都不指定，那么默认长度 18，默认精度 0。
- (12) TEXT：文本型（指定长度时），备注型（不指定长度时）。
- (13) IMAGE, OLEOBJECT：OLE OBJECT 型，不要指定长度，否则会报错。
- (14) CHAR, NCHAR, VARCHAR, NVARCHAR：文本型，可以指定长度，否则默认值为 255。
- (15) MEMO：备注型，不要指定长度，否则会报错。

2. Access SQL语句与SQL的区别

Access 中提供查询对象，在设计时可以采用设计视图和 SQL 视图，非常方便，SQL 视图中的 SQL 语句可以在 SQL Server 中使用，但两者之间还有一些微小的差别。表 6.7 总结了 Microsoft Access 与 Microsoft SQL Servers 中 SQL 语法的区别。

表 6.7 Access SQL 与 Microsoft SQL Servers 中 SQL 语法的区别

SQL 语法元素	Microsoft Access	Microsoft SQL Server
标识符	限制不超过 64 个字符； 允许使用关键字和特殊字符； 可以用任何字符开头	SQL Server 6.5：限制不超过 30 个字符；不允许使用关键字和特殊字符；必须用字母字符开头。 SQL Server 7.0 的标识符与 Access 完全兼容

(续表)

SQL 语法元素	Microsoft Access	Microsoft SQL Server
输出字段	允许多个输出字段具有相同名称	在视图中不支持多个相同输出字段名
日期分隔符号	英镑符 (#)	撇号 (')
Boolean 常量	True、False; On、Off; Yes、No。	整数: 1 (真)、0 (假)
字符串连接	和号 (&)	加号 (+)
通配符	星号 (*) 与零个或更多字符匹配; 问号 (?) 与单个字符匹配; 叹号 (!) 意味着不在列表中; 英镑符 (#) 意味着单个数字	百分号 (%) 与零个或更多字符匹配; 下划线 () 与单个字符匹配; 上插入符 (^) 意味着不在列表中; 没有与英镑符 (#) 对应的字符
TOP	如果有一个 ORDER BY 子句, 自动包含层次	SQL Server 6.5 不支持; SQL Server 7.0 需要一个明确的 WITH TIES 子句
CREATE INDEX	允许创建升序和降序索引; 允许声明主键, 没有 Null 值, 并且忽略 Null 值	不支持
DROP INDEX	语法是: DROP INDEX <index name> ON <table name>	语法是: DROP INDEX <table name>, <index name>
DISTINCTROW	支持 (允许选择单个记录)	不支持
OWNERACCESS	支持 (在执行时控制许可权)	不支持
TABLE in UNION	支持 (允许使用下列语法指定表: TABLE <tablename>)	不支持
ORDER BY in UNIONS	支持。允许通过联合查询中的子句实现多种排序	支持。允许通过语句末尾的子句实现一种排序
TRANSFORM	支持。用于交叉表查询	不支持
PARAMETERS	支持 (在 SQL 中记录)	不支持

6.7.4 Access SQL的数据定义

标准 SQL 的数据定义功能非常广泛, 一般包括数据库的定义、表的定义、视图的定义、存储过程的定义、规则的定义和索引的定义等若干部分。

SQL 语言中数据定义的基本语句有 4 个:

- ◆ 建立新表: CREATE TABLE...
- ◆ 添加字段: ALTER TABLE...ADD...
- ◆ 删除字段: ALTER TABLE...DROP...
- ◆ 基本表删除: DROP TABLE...

1. 表的建立

SQL 语言中的 CREATE TABLE 语句用来建立新表, CREATE TABLE 语句的使用格式如下:

CREATE TABLE [数据库名.]表名

(列名 1 类型 [(宽度, [小数位数])] [NULL/NOT NULL],
列名 2 类型 [(宽度, [小数位数])] [NULL/NOT NULL],
.....
.....)

说明:

[数据库名.]: 用于指明所建立的表隶属于哪个数据库。

基本表名: 所创建表的名称。

列名: 所建立表的字段名。

类型：指明对应字段的数据类型。

[(宽度, [小数位数])]：指明对应字段的宽度。如有小数部分，还需指出小数位数。

[NULL/NOT NULL]：指明字段是否可以取空值。

基本类型及其符号表示如表 6.6 所示。

【例 6-1】 创建教学数据库中的基本表 S,C,SC。

```
CREATE TABLE S
(SNO CHARACTER(4) NOT NULL, SNAME CHARACTER(8) NOT NULL, AGE
TINYINT, SEX CHARACTER(1))
```

其中：S 为基本表名，SNO, SNAME, AGE, SEX 为列名，NOT NULL 用于说明列值不能为空。

```
CREATE TABLE C
(CNO CHARACTER(4) NOT NULL, CNAME CHARACTER(10),TEACHER CHARACTER(8))
CREATE TABLE SC
(SNO CHARACTER(4),CNO CHARACTER(4),GRADE TINYINT)
```

【例 6-2】 创建表 XCUST。

```
CREATE TABLE XCUST
(CUSTNO SHORT NOT NULL, CUSTNAME TEXT(40) NOT NULL, ADDRESS TEXT(60),
PRICE DECIMAL(15, 4) NOT NULL, SITE DATETIME, TELNO TEXT (30), FAXNO TEXT(30))
```

在使用 CREATE TABLE 语句创建基本表时，最初得到的只是一个空的框架（表结构），用户可以使用 INSERT 命令插入内容。

注意：

使用 SQL 语句创建的表和表中列的名称必须以字母开头，后面可以使用字母，数字或下划线。用户在选择表名时不要使用 SQL 语言中的保留关键词，如 SELECT, CREATE, INSERT 等作为表或列的名称。

2. 修改表结构

基本表创建以后，经过一段时间的使用，表的结构可能会无法满足实际的要求，这时就需要对表的结构进行修改，增加新列或者删除无用列。

(1) 增加新列：“ALTER TABLE...ADD...”

语句格式：ALTER TABLE 基本表名 ADD 列名 类型

【例 6-3】 在基本表 S 中增加 ADDRESS 列。

```
ALTER TABLE S ADD ADDRESS CHARACTER(40)
```

(2) 删除无用列：“ALTER TABLE...DROP...”

语句格式：ALTER TABLE 基本表名 DROP 列名

【例 6-4】 删除基本表 S 中的 ADDRESS 列。

```
ALTER TABLE S DROP ADDRESS
```

【例 6-5】 对 XCUST 表完成要求的表结构操作：增加列、删除列、修改列。

增加列：ALTER TABLE XCUST ADD COLUMN CITY TEXT(30)

删除列：ALTER TABLE XCUST DROP COLUMN CITY

修改列：ALTER TABLE XCUST ALTER COLUMN CITY TEXT (40)

3. 删除基本表

在 SQL 语言中使用 DROP TABLE 命令删除某个表格及该表格中的所有记录。DROP TABLE 命令的使用格式为：

```
DROP TABLE TABLENAME
```

【例 6-6】 删除基本表 SC。

```
DROP TABLE SC
```

6.7.5 Access SQL的数据查询

数据查询是我们对数据库进行的最基本的操作，查询效率的高低对软件有着重要的影响。在 SQL 中提供了 SELECT 查询语句，其功能强大且内容丰富。

1. SELECT的语法格式

SELECT 的语法格式如下：

```
SELECT 目标表的列名序列
FROM 基本表视图序列
[INTO 目标位置]
[WHERE 行条件表达式]
[GROUP BY 列名序列]
[HAVING 组条件表达式]
[ORDER BY 列名[ASC|DESC]...]
```

[]：表示可选项，用户根据实际需要进行选择。

各子句的含义及功能如下：

① 目标表的列名序列：指明查询结果的字段构成，可以是字段名、表达式、常量等。

② 基本表视图序列：用于指明查询信息的数据来源。

③ [INTO 目标位置]：指明查询结果的输出位置，输出位置包括：

◆ ARRAY 数组名：存放到指定的数组中。

◆ CURSOR 临时表名：存放到一个临时表中。

◆ TO FILE 文件名：存放到一个文件中。

◆ TO PRINTER：打印查询结果。

◆ TO SCREEN：将结果在屏幕上显示（默认方式）。

④ [WHERE 行条件表达式]：用于在连接结果中选择满足条件的元组。

⑤ [GROUP BY 列名序列]：用于对结果进行分组。分组记录的字段可以有多个，这些字段的顺序决定最高到最低的分组层次。

⑥ [HAVING 组条件表达式]：用于选择满足条件的组。

⑦ [ORDER BY 列名[ASC|DESC]...]：用于设置查询结果的排序方式。ASC 表示升序，DESC 表示降序。使用的目的是将查询的结果依照指定字段加以排序。若没有 ORDER BY，查询出的数据集将不会排序。

2. 语句的执行过程

SELECT 的语句的执行过程如下：

① 首先读取 FROM 子句中基本表和视图，然后对其进行笛卡儿积运算。

- ② 根据 WHERE 子句，选出满足条件表达式的元组。
- ③ 按照 GROUP 子句中指定列的值进行分组，并从这些分组中选择满足 HAVING 子句中条件的分组。
- ④ 按照 SELECT 子句给出的列求值得到目标表。
- ⑤ 用 ORDER 子句对目标表进行排序。

【例 6-7】 查询职员表中所有工资大于 3900 元的职员的姓名。

```
SELECT 姓名
FROM 职员表格
WHERE 工资>3900
```

3. WHERE子句中的运算符

SQL 语言是完备的，也就是说，只要数据是按关系方式存入数据库的，就能构造合适的 SQL 命令把它检索出来。事实上，SQL 不仅具有一般的检索能力，还可以通过在 WHERE 子句中加入运算符进行计算方式的检索。在 WHERE 子句中使用的一些运算符如表 6.8 所示。

表 6.8 WHERE 子句中的运算符

运算符号	运算符
算术比较运算符	=（等于）、>（大于）、<（小于）、>=（大于等于）、<=（小于等于）、<>（不等于）
逻辑运算符	AND、OR、NOT
集合运算符	UNION（并）、INTERSECT（交）、EXCEPT（差）
集成员资格运算	IN、NOT IN
谓词	EXISTS（存在）、ALL、SOME、UNIQUE
数字函数	AVG、MIN、MAX、SUM、COUNT、FIRST、LAST
其他	LIKE、BETWEEN…AND…

通过使用 LIKE 运算符可以只选择与用户规定格式相同的记录。可以将一字符串与另一特定字符串样式比较，并将符合该字符串样式的记录过滤出来。

若要查询出所有姓“李”的人，可以利用下面的式子：

```
LIKE "李*"
```

【例 6-8】 计算职员身高超过 165 厘米的职员平均身高。

```
SELECT AVG（身高）AS 平均身高
FROM 职员表
WHERE 身高>165
```

【例 6-9】 统计出业务部门的职员人数。

```
SELECT COUNT（姓名）
FROM 职员表
WHERE 部门名称="业务部"
```

【例 6-10】 找出货品数量字段的第一条数据与货品价格字段的最后一条数据。

```
SELECT FIRST（货品数量），LAST（货品价格）
FROM 订单表
```

【例 6-11】 计算出货品总价。

```
SELECT SUM（单位价格*货品数量）AS 货品总价
FROM 订单表
```

4. 常见查询方式

(1) 简单查询

所谓简单查询是指查询仅涉及数据库中的一个表。

【例 6-12】 查询全体学生的详细信息。

```
SELECT * FROM S
```

【例 6-13】 查询名字的第二个字母为 K 的学生的姓名与学号。

```
SELECT SNAME, SNO FROM S WHERE SNAME LIKE "? K%"
```

【例 6-14】 查询姓张的学生的姓名与学号。

```
SELECT SNAME, SNO FROM S WHERE SNAME LIKE "张%"
```

说明：“？”代表任意一个字符，“*”表任意长的字符串。

【例 6-15】 查询全体学生情况，查询结果按所在系的系号升序排列，同一系的学生按年龄的降序排列。

```
SELECT *
FROM S
ORDER BY DEPT, AGE DESC
```

(2) 嵌套查询

嵌套查询也称为子查询，是指一个 **SELECT...FROM...WHERE** 查询可以嵌套在另外一个查询中。SQL 允许多层嵌套。每个子查询在上一级查询处理之前求解，即嵌套查询是由里向外处理的，这样外查询可以利用内查询的结果。

当查询涉及多个关系时用嵌套查询逐次求解，层次分明，易于理解，易于书写，具有结构化程序设计的优点。

【例 6-16】 查询和张明在同一系的学生的学号、姓名。

```
SELECT SNO, SNAME FROM S
WHERE DEPT=(SELECT DEPT FROM S WHERE SNAME="张明")
```

【例 6-17】 查询选修了课程 1 或者选修了课程 2 的学生学号。

```
SELECT SNO FROM SC WHERE CNO="C1"
UNION
SELECT SNO FROM SC WHERE CNO="C2"
```

(3) 联接查询

联接查询涉及两个以上的表，联接查询是关系数据库最主要的查询，包括等值联接、自然联接、非等值联接、自身联接、外联接和复合联接查询等。

【例 6-18】 查询每个学生及其选修情况。

```
SELECT S.*, SC.* FROM S, SC
WHERE S.SNO=SC.SNO
```

【例 6-19】 对 S, SC, C 中的数据进行查询操作。

① 统计每一年龄选修课程的学生人数。

```
SELECT AGE, COUNT(*)
FROM S, SC
WHERE S.SNO=SC.SNO
GROUP BY AGE
```

② 求基本表 S 中男同学的每一年龄组（超过 20 人）的人数，要求查询结果按人数的升序排列，人数相同按年龄的降序排列。

```
SELECT AGE COUNT(SNO)
FROM S
WHERE SEX="男"
GROUP BY AGE
HAVING COUNT(*)>20
ORDER BY 2,AGE DESC
```

③ 检索男同学选修的课程的课程号。

```
SELECT DISTINCT CNO FROM S,SC
WHERE S.SNO=SC.SNO AND SEX="男"
```

6.7.6 Access SQL 的数据更新

Access SQL 的数据更新包括插入数据、删除数据、修改数据三种基本操作。

1. 插入数据

Access SQL 语言使用 INSERT 语句向表中插入或添加新的数据行。INSERT 语句的基本格式为：

```
INSERT INTO 基本表名（列名表）
VALUES（元组值）
```

【例 6-20】 给 SC 表添加一条记录。

```
INSERT INTO SC
VALUES ("S4","C4",80)
```

简单来说，当向数据库表格中添加新记录时，在关键词 INSERT INTO 后面输入所要添加的表名称，然后在括号中列出将要添加新值的列的名称。若省略了列名表，则按表结构中列的顺序输入相应的内容。

【例 6-21】 对 XCUST 表完成插入数据操作。

```
INSERT INTO XCUST (CUST, CITY) VALUES ("0659", "中国北京")
INSERT INTO XCUST VALUES ("0619", 18, "2007-09-22", "中国西安")
INSERT INTO XCUST SELECT * FROM XCUST1 WHERE CITY LIKE "中国"
```

2. 删除数据

SQL 语言使用 DELETE 语句删除数据库表格中的行或记录。DELETE 语句的格式为：

```
DELETE FROM 基本表名[WHERE 条件表达式]
```

从基本表中删除满足条件的表达式的元组，该语句每次只能从一个基本表中删除元组。当无条件时，表示删除所有的元组。

【例 6-22】 删除学号为 0004 的学生，将 C2 课程中小于该科平均成绩的元组从基本表 SC 中删除。

```
DELETE FROM S
WHERE SNO="0004"
DELETE FROM SC
WHERE CNO="C2" AND
```

```

GRADE<(SELECT AVG(GRADE)
FROM SC
WHERE CNO="C2")

```

简单来说,当需要删除某一行或某个记录时,在 **DELETE FROM** 关键词之后输入表格名称,然后在 **WHERE** 从句中设定删除记录的判断条件。注意,如果用户在使用 **DELETE** 语句时不设定 **WHERE** 从句,则表格中的所有记录将全部被删除。

3. 修改数据

SQL 语言使用 **UPDATE** 语句更新或修改满足规定条件的记录。**UPDATE** 语句的格式为:

```

UPDATE 基本表名
SET 列名=值表达式[,SET 列名=值表达式...]
[WHERE 条件表达式]

```

命令功能:修改基本表中满足条件表达式的那些元组指定列的值,所需修改的列由 **SET** 子句指出。

【例 6-23】 将 C2 任课教师的姓名改为 ZHANG。

```

UPDATE C
SET TEACHER="ZHANG"
WHERE CNO="C2"

```

【例 6-24】 将所有男同学的成绩提高 15%。

```

UPDATE SC
SET GRADE=GRADE*1.15
WHERE SNO IN
(SELECT SNO FROM S WHERE SEX="M")

```

使用 **UPDATE** 语句时,关键一点就是要设定好用于进行判断的 **WHERE** 条件从句。

【例 6-25】 对 XCUST 表完成修改操作。

修改日期: **UPDATE XCUST SET NDATE="2007/07/17" WHERE CUST="0659"**

修改数值: **UPDATE XCUST SET CAGE=33 WHERE CUST="0659"**

修改文本: **UPDATE XCUST SET CUST="0699" WHERE CUST="0659"**

6.7.7 建立SQL查询

SQL 查询是用户使用 SQL 语句直接创建的一种查询。实际上,Access 查询就是以 SQL 语句为基础来实现查询的功能。在使用过程中,经常会使用到一些特殊的查询,这些查询用各种查询向导和设计器都无法实现,对于此类查询可以通过 SQL 语句来实现。当建立一个涉及大量字段的查询时,就需要输入大量文字。所以,建立特殊查询的时候也都是先在查询设计视图将基本的查询功能都实现后,再切换到 SQL 视图通过编写 SQL 语句完成一些特殊的查询。

在数据库窗口中,如图 6.24 所示,选择“查询”对象,在菜单栏单击“新建”命令,系统弹出“新建查询”对话框,选择“设计视图”选项,系统弹出“选择查询”窗口。

切换到 SQL 视图,单击工具栏最左面的“SQL 视图”按钮切换到 SQL 状态,如图 6.25 所示,就可以输入相应的 SQL 命令了。

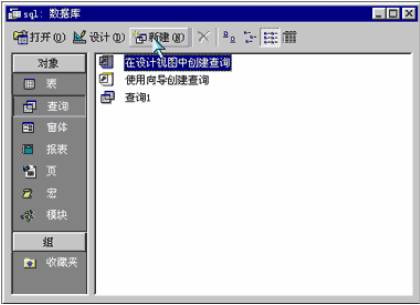


图 6.24 数据库窗口



图 6.25 SQL 视图设计窗口

如图6.26所示，在设计窗口中输入 SQL 语句。
保存并运行查询，其结果如图6.27所示。

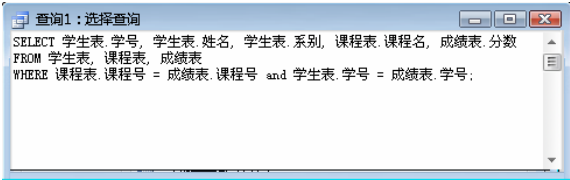


图 6.26 输入 SQL 语句

The image shows the '查询1: 选择查询' (Query1: Select Query) window displaying the results of the query. It contains a table with 5 columns: 学号 (Student ID), 姓名 (Name), 系别 (Department), 课程名 (Course Name), and 分数 (Score).

学号	姓名	系别	课程名	分数
2008112001	张小强	新闻	数学	75
2008112001	张小强	新闻	英语	85
2008112001	张小强	新闻	计算机	86
2008112001	张小强	新闻	物理	80
2008112002	张小强	新闻	英语	72
2008112002	张小强	新闻	计算机	83
2008112003	李娜	新闻	数学	72
2008112003	李娜	新闻	英语	75
2008112003	李娜	新闻	计算机	80
2008112003	李娜	新闻	物理	83
2008118110	周明明	化学	数学	80
2008118110	周明明	化学	英语	72

记录: 1 共有记录数: 20

图 6.27 运行结果

6.8 查询的打开与修改

6.8.1 打开查询

创建了一个查询以后，如果需要查看满足查询条件的查询结果，就需要打开查询。打开查询的步骤如下：

- ① 打开数据库，单击对象栏中的查询。
- ② 单击查询名称，选择某一个查询。
- ③ 单击数据库窗口的打开按钮，出现查询窗口。

6.8.2 修改查询

创建了查询以后，还可以根据需要进行修改。如图 6.28 所示，在设计视图中，可以给查询增加字段、删除字段、增加条件、在查询中排序字段。

1. 增加字段

在出现查询设计视图后，单击下拉箭头，在下拉列表中单击要添加的字段名，该字段名就被增加到“字段”行中。还有两种方法可以给查询增加字段：

- ① 在查询设计窗口的上方有查询中要使用的表，在表中找到需要增加的字段名并双击，该字段名就出现在“字段”行的第一个可用列内。
- ② 在表中找到需要增加的字段名后，拖动该字段名到“字段”行的第一个可用列内。

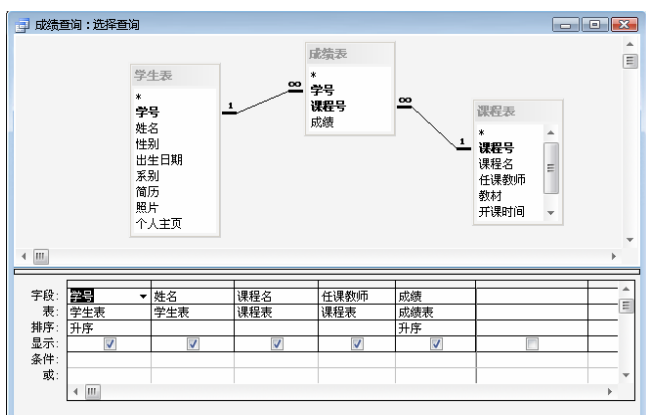


图 6.28 查询设计视图

2. 删除字段

删除字段的方法是在该列的任何地方单击，然后在“编辑”菜单中选择“删除列”，或者把鼠标箭头放在该列上方，当指针变成一个指向下方的箭头时，单击选中该列，再按 Delete 键删除。

3. 增加条件

在查询设计视图中，单击某个字段的“准则”行，输入需要使用的准则。

4. 在查询中排序字段

在查询设计视图中，单击某个字段的“排序”行，再单击出现下拉箭头，在下拉列表中选择“升序”或“降序”。

习 题 6

一、填空题

1. 查询是以_____为数据来源的再生表。
2. 查询主要有选择查询、参数查询及操作查询，其中操作查询包括更新查询、追加查询、_____和生成表查询等。
3. Access 数据库中，SQL 查询的 GROUP BY 语句用于_____。
4. 已知一个 Access 数据库，其中含有系别、男、女等字段，若要统计每个系男女教师的人数，则应使用_____查询。
5. 条件语句“WHERE 工资额>1000”的意思是_____。
6. 利用对话框提示用户输入查询条件，这样的查询属于_____。
7. SQL 语言是在数据库系统中应用广泛的数据库查询语言，它包括了数据定义、数据查询、_____和_____4 种功能。
8. 使用查询向导创建交叉表查询的数据源必须来自_____个表或查询。

二、选择题

1. Access 数据库中，在创建交叉表查询时，用户需要指定三种字段，下面（ ）不是交叉表查询所需求指定的字段。

A. 格式字段 B. 列标题字段 C. 行标题字段 D. 总计类型字段

2. 在 Access 数据库中,带条件的查询需要通过准则来实现,准则是运算符、常量、字段值等的任意组合,而()不是准则中的元素。
- A. SQL 语句 B. 函数 C. 属性 D. 字段名
3. 将 Access 数据库中 C 语言课程不及格的学生从“学生”表中删除,要用()查询。
- A. 追加查询 B. 生成表查询 C. 更新查询 D. 删除查询
4. 在 Access 的查询中,可以只选择表中的部分字段,也可以通过选择一个表中的不同字段生成所需的多个表,这体现了查询的()功能。
- A. 建立新表 B. 选择字段 C. 选择记录 D. 实现计算
5. 从表中抽取选中信息的对象类型是()。
- A. 模块 B. 报表 C. 查询 D. 窗体
6. 完整的交叉表查询必须选择()。
- A. 行标题、列标题和值 B. 只选行标题即可
C. 只选列标题即可 D. 只选值
7. 在 Access 中,可以把()作为创建查询的数据源。
- A. 查询 B. 报表 C. 窗体 D. 外部数据表
8. 在创建查询时,当查询的字段中包含数值型字段时,系统将会提示选择()。
- A. 明细查询、按选定内容查询 B. 明细查询、汇总查询
C. 汇总查询、按选定内容查询 D. 明细查询、按选定内容查询
9. 在 Access 数据库中,对数据表进行删除的是()。
- A. 汇总查询 B. 操作查询 C. 选择查询 D. SQL 查询
10. 在 Access 数据库中,从数据表找到符合特定准则的数据信息的是()。
- A. 汇总查询 B. 动作查询 C. 选择查询 D. SQL 查询
11. 条件语句“性别=“女”AND 工资额>2000”的意思是()。
- A. 性别为“女”并且工资额大于 2000 的记录
B. 性别为“女”或者且工资额大于 2000 的记录
C. 性别为“女”并非工资额大于 2000 的记录
D. 性别为“女”或者工资额大于 2000,且二者择一的记录
12. 条件语句“NOT 工资额>2000”的意思是()。
- A. 除了工资额大于 2000 之外的工资额的记录 B. 工资额大于 2000 的记录
C. 并非工资额大于 2000 的记录 D. 字段工资额大于 2000,且二者择一的记录
13. 已知“借阅”表中有“借阅编号”、“学号”和“借阅图书编号”等字段,每个学生每借阅一本书生成一条记录,要求按学生学号统计出每个学生的借阅次数,下列 SQL 语句中,正确的是()。
- A. SELECT 学号 COUNT(学号) FROM 借阅
B. SELECT 学号 COUNT(学号) FROM 借阅 GROUP BY 学号
C. SELECT 学号 SUM(学号) FROM 借阅
D. SELECT 学号 SUM(学号) FROM 借阅 ORDER BY 学号
14. 将信息系 1999 年以前参加工作的教师的职称改为副教授,合适的查询为()。
- A. 生成表查询 B. 更新查询 C. 删除查询 D. 追加查询
15. 以下不属于操作查询的是()。
- A. 交叉表查询 B. 生成表查询 C. 更新查询 D. 追加查询

16. 下面对查询功能的叙述中正确的是（ ）。
- A. 在查询中，选择查询可以只选择表中的部分字段，通过选择一个表中的不同字段生成同一个表
- B. 在查询中，编辑记录主要包括添加记录、修改记录、删除记录和导入、导出记录
- C. 在查询中，查询不仅可以找到满足条件的记录，而且还可以在建立查询的过程中进行各种统计计算
- D. 以上说法均不对
17. 用 SQL 语言描述“在教师表中查找男教师的全部信息”，以下描述正确的是（ ）。
- A. SELECT FROM 教师表 IF (性别="男")
- B. SELECT 性别 FROM 教师表 IF (性别="男")
- C. SELECT * FROM 教师表 WHERE (性别="男")
- D. SELECT * FROM 性别 WHERE (性别="男")
18. （ ）会在执行时弹出对话框，提示用户输入必要的信息，再按照这些信息进行查询。
- A. 选择查询 B. 参数查询 C. 交叉表查询 D. 操作查询
19. 查询能实现的功能有（ ）。
- A. 选择字段，选择记录，编辑记录，实现计算，建立新表，建立数据库
- B. 选择字段，选择记录，编辑记录，实现计算，建立新表，更新关系
- C. 选择字段，选择记录，编辑记录，实现计算，建立新表，设置格式
- D. 选择字段，选择记录，编辑记录，实现计算，建立新表，建立基于查询的报表和窗体
20. 在 Access 数据库中使用向导向导创建查询，其数据可以来自（ ）。
- A. 多个表 B. 一个表 C. 一个表的一部分 D. 表或查询
21. 在 Access 数据库中，下列查询的计算表达式中，求两门课的平均分数，正确的是（ ）。
- A. [语文]+[数学]/2 B. "([语文]+[数学])/2"
- C. ([语文]+[数学])/2 D. "[语文]"+"[数学]"/2
22. 在 Access 数据库中，要查询的条件是语文成绩处在 60 分数段的记录，则在语文字段的准则中应当输入（ ）。
- A. >60 AND <70 B. >=60 AND <70 C. >60 OR <70 D. >=60 OR <70
23. 在 Access 数据库中，查询姓名字段中所有姓张的同学记录时，在姓名准则中应输入（ ）。
- A. 张 B. 张* C. *张 D. *张*
24. 下列 SELECT 语句正确的是（ ）。
- A. SELECT * FROM"学生表"WHERE 姓名="张三"
- B. SELECT * FROM"学生表"WHERE 姓名=张三
- C. SELECT * FROM 学生表 WHERE 姓名="张三"
- D. SELECT * FROM 学生表 WHERE 姓名=张三

三、简答题

- 举例说明在什么情况下，需要设计生成表查询？
- 举例说明在什么情况下，需要设计追加查询？
- 现在已知某单位“职工工资”表中的性别用汉字“男”或“女”表示，已有字段为“姓名，性别，标准工资，加班费，标准工资”，在查询中除已知字段外，现在要求自动求出“妇女保健补贴”字段中的数据，即每位妇女月补贴 25 元，如何自动求出所有职工的工资总额？
- 设有下列关系模型的样本数据：
书店表：由书店号、书店名、地址组成，书店号为主码；

图书馆表：由图书馆号、图书馆名、城市、电话组成，图书馆号为主码；

图书表：由 ISBN 号、书名、价钱组成，ISBN 号为主码；

图书发行表：由图书馆号、图书 ISBN 号、书店号、册数组成，图书馆号、图书 ISBN 号、书店号为主码。

试用 SQL 语言写出以下查询：

① 查找馆名为“A 馆”的图书馆从书店“B 店”购买的图书书名及其册数；

② 取出馆址在“西安”的馆名及电话号码；

③ 取出书店“太白路店”发行的图书书名和数量。

5. 设有一个销售管理数据库系统，其关系模式如下：

销售员（工号，姓名，柜台，部门）；

商品（商品号，商品名，价格）；

业绩（工号，商品号，销售额）；

用 SQL 语言表示下列数据查询操作：

① 查询“服装”部门女装柜台销售员的工号和姓名；

② 查询销售以“霓裳”开头的服装的销售员的工号、姓名和该类产品的销售额；

③ 查询每种商品的商品号、各销售员销售该商品的最高销售额、最低销售额和所有员工的平均销售额。

四、操作题

1. 创建一个数据表结构如下：序号（自动编号），学号（文本，8，必填，非空），姓名（文本，8），性别（文本，2），出生日期（日期），党员否（是/否），入学成绩（整型，一位小数），籍贯（文本，10），简历（备注），照片（OLE）。

字段属性定义：定义“学号”为主键，“入学成绩”定义有效性规则为大于 500 且小于 750。

查询要求：创建一个籍贯包含“西安”和“北京”学生的查询，显示“姓名”“籍贯”；创建一个入学成绩大于 600 的男性的查询，查询结果按成绩降序排列，显示“学号”“入学成绩”。

2. 创建“学生成绩.mdb”数据库，数据库包括学生表（学号，姓名，系名，性别，出生日期，爱好，照片，简历）、课程表（课程编号，课程名称，学分，开课时间）和成绩表（学号，课程编号，成绩），按下列要求进行操作。

（1）创建表间关系。

（2）录入数据。

（3）写出满足如下条件的 SQL 语句：

① 从选出语文、数学、计算机的各科成绩在 90 分以上的学生姓名；

② 选出“计算机系”的男学生的学号和姓名；

③ 选出姓“刘”的学生的姓名、性别。

（4）创建一个查询，查询每位学生的总分，要求输出学号、姓名、总分，查询保存为“总分查询”。

（5）创建一个选择查询，查找并显示简历信息为空的学生的“学号”、“姓名”、“性别”和“出生年月”四个字段内容，所建查询命名为“基本信息查询”；

（6）创建一个选择查询，按系别统计各自男女学生的平均年龄，显示字段标题为“所属院系”、“性别”，所建查询命名为“按系统计查询”；

（7）创建一个操作查询，将没有书法爱好的学生的“学号”、“姓名”、和“出生年月”三个字段内容追加到目标表“临时”表的对应字段内，所建查询命名为“追加查询”。

(8) 创建一个查询, 当运行该查询时, 应显示参数提示信息“请输入爱好”, 输入爱好后, 在简历字段中查找具有指定爱好的学生, 显示“学号”、“姓名”、“性别”、“年龄”、“照片”和“简历”字段的内容, 所建查询命名为“按爱好查询”;

(9) 创建一个查询, 查找学生的课程成绩大于等于 80 且小于等于 100 的学生情况, 显示“姓名”、“课程名称”和“成绩”三个字段的内容, 所建查询名为“成绩查询”;

(10) 创建一个查询, 按“课程编号”分类统计最高分成绩与最低分成绩的差, 并显示“课程名称”、“最高分与最低分的差”等内容。其中, 最高分与最低分的差由计算得到, 所建查询名为“高低分差别查询”。

3. 创建“库存管理系统”数据库, 在库中创建“产品定额储备”表和“库存情况”表。表结构要求根据自己对实际的情况理解构造。按要求创建如下查询:

(1) 以“库存管理系统”数据库中的“产品定额储备”和“库存情况”两张表为数据源创建一个查询, 查找并显示库存量超过 1000 只的产品名称和库存数量, 查询名为“数量查询”。

(2) 以“库存管理系统”数据库中的“产品定额储备”和“库存情况”两张表为数据源, 创建一个查询, 按出厂价计算每种库存产品的总金额, 并显示其产品名称和总金额。总金额的计算方法为: 总金额=出厂价*库存数量。查询名为“资金查询”。

第 7 章 窗体的使用

友好的操作界面（窗体）给用户使用数据库带来很大的便利，使得更多的用户不用进行专门学习就能根据窗口中的提示完成自己的工作。不管数据库中表或查询设计的有多好，那仅是数据的内部存储方式。如果窗体设计的杂乱，而且又没有足够的提示，就会给使用者带来很大的不便。所以，一个好的窗体非常有用。

7.1 窗体的功能与构成

窗体是 Access 中的一种对象，它通过计算机屏幕将数据库中的数据显示给用户。

7.1.1 窗体的功能

窗体和报表都用于数据库中数据的维护，但两者的作用是不同的。窗体主要用来输入数据，报表则用来输出数据。具体来说，窗体具有以下几种功能。

1. 数据的显示与编辑

窗体最基本的功能是显示与编辑数据。窗体可以显示来自多个数据表中的数据，此外，用户可以利用窗体对数据库中的相关数据进行添加、删除和修改，并可以设置数据的属性。用窗体来显示并浏览数据比用表和查询显示数据更加灵活，不过窗体每次只能浏览一条记录。

2. 数据输入

用户可以根据需要来设计窗体，并作为数据库中数据输入的接口，通过窗体可以节省数据录入的时间并提高数据输入的准确度。

3. 应用程序流控制

与 VB 窗体类似，Access 2003 中的窗体也可以与函数、子程序相结合。在每个窗体中，用户可以使用 VBA 编写代码，并利用代码执行相应的功能。

7.1.2 窗体的构成

窗体一般由窗体页眉、页面页眉、主体、页面页脚、窗体页脚等几部分组成。如图 7.1 所示。

1. 窗体页眉

可用于显示窗体的标题和使用说明，也可以用于打开相关窗体或执行其他任务的命令按钮。显示在窗体视图中顶部或打印页的开头。

2. 页面页眉

用于在窗体中每页的顶部显示标题、列标题、日期或页码。

3. 主体

用于显示窗体或报表的主要部分，该节通常包含绑定到记录源中字段的控件。但也可能包含未绑定控件，如字段或标签等。

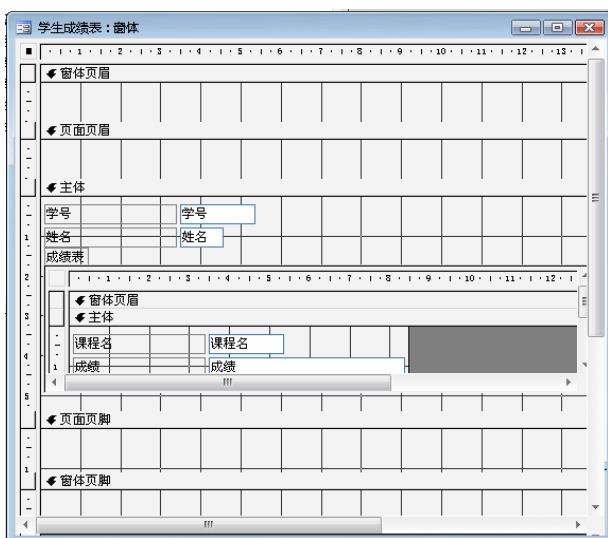


图 7.1 窗体设计视图

4. 页面页脚

用于在窗体和报表中每页的底部显示汇总、日期或页码。

5. 窗体页脚

用于显示窗体的使用说明、命令按钮或接受输入的未绑定控件，显示在窗体视图中的底部和打印页的尾部。

7.2 创建窗体

创建窗体有三种方法：一种方法是使用“自动窗体”创建基于单个表或查询的窗体。另一种方法是使用“向导”创建基于一个或多个表或查询的窗体。还有一种方法是通过“设计视图”创建窗体。

7.2.1 自动窗体创建

用 Access 自动创建一个纵栏式表格的窗体。

① 打开“学生成绩管理”数据库，单击“对象”列表中的“窗体”项。

② 单击数据库菜单上“新建”项，在弹出的“新建窗体”对话框中选择“自动创建窗体：纵栏式”选项。

在“该对象的数据来源表或查询”下拉框中选择需要的表或查询，选取“课程”表，单击“确定”按钮。

③ 单击“文件”，选择“保存”，取名为“课程表（纵栏式）窗体”。

可以发现，这个窗体中数据的显示格式和表中看到的数据显示格式有所不同。纵栏式表格每次只能显示一个记录的内容，文本框及所附标签并排显示在两栏中，标签显示在每个文本框的左面并标识文本框中的数据，如图7.2所示。



图 7.2 纵栏式窗体

7.2.2 向导创建

使用向导创建窗体的步骤如下：

① 打开“学生成绩管理”数据库，单击“对象”列表中的“窗体”项，鼠标双击创建方法框中的“使用向导创建窗体”项，出现“窗体向导”对话框，如图7.3所示。选择窗体上需要的各种字段，这些字段可以来自不同的表、查询。

在“表/查询”下拉列表中选取字段所在的表或查询，再将所需的字段添加到“选定的字段”列表框中。在选取字段时，可以通过选取次序来调整字段在窗体中排列的次序，先选取的字段位于窗体的前面。选择“学生”表的“学号”、“姓名”字段，“课程”表的“课程名”字段，“成绩”表的“分数”字段。

② 单击“下一步”按钮。在“确定数据查看方式”中选择“通过学生”，并选择“带有子窗体的窗体”，单击“下一步”按钮。

③ 选择窗体布局方式。本例选择“表格”布局方式。

④ 选择窗体风格。根据要求选择一种风格，本例选择“标准”风格，单击“下一步”按钮。

⑤ 为创建的新窗体指定标题为“学生”，并选择该窗体首次被打开的方式。

⑥ 如果不需要对前面的设置进行修改，单击“完成”按钮，系统会根据用户在向导中的设置生成窗体，运行结果如图7.4所示。

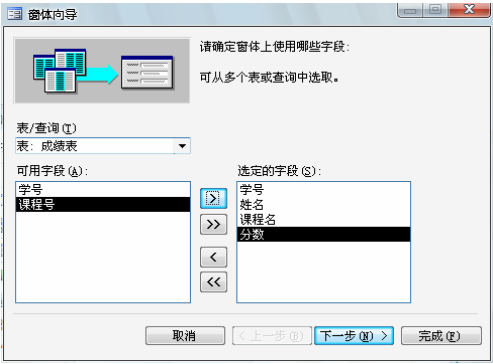


图 7.3 “窗体向导”窗口



图 7.4 窗体运行结果

7.3 窗体设计视图与控件

在建立窗体的时候，一般先通过向导或自动窗体的方式建立基本窗体，然后再使用设计视图对窗体进行修改和美化。其一般步骤是打开窗体设计视图添加控件，然后可以对控件进行移动、改变大小、删除、设置边框、阴影和粗体、斜体等特殊字体效果等操作，来更改控件的外观。另外，通过“属性”窗口，可以对控件或工作区部分的格式、数据事件等属性进行设置。

7.3.1 窗体的设计视图

打开“学生表”窗体，将鼠标移动到工具栏上最左边的“视图”按钮上，单击鼠标左键，切换到设计模式。如图7.5所示。在设计模式下，可以根据实际需要来修改窗体。

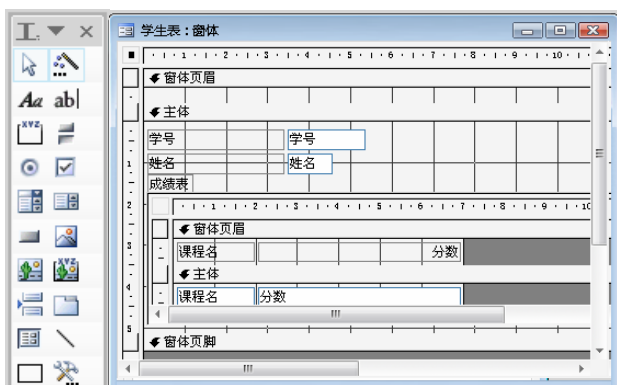


图 7.5 “学生表：窗体”设计视图

Access 中，窗体上所有控件都可以根据自己的需要进行摆放，同时还可以调整窗口的大小、文字的颜色。在屏幕上同时出现的还有一个工具箱，工具箱中包含很多按钮，每个按钮都是构成窗体的一个功能控件。

7.3.2 工具箱的使用

Access 提供了一个可视化的窗体设计工具——窗体设计工具箱。

1. 工具箱的打开和关闭

在窗体设计视图中，如果屏幕上没有显示工具箱，单击“窗体设计”工具栏上“工具箱”按钮，或者单击“视图”菜单中的“工具箱”命令，将工具箱显示在屏幕上。

2. 工具箱的移动和锁定

如果“工具箱”覆盖在设计视图的某区域上，可以通过拖动方式将其移动到其他位置。如果要重复单击工具箱中的按钮。例如，要添加多个标签，就可以将该按钮锁定。锁定后，就不必每次执行重复操作时都要单击某个按钮。锁定的方法是：双击工具箱中要锁定的按钮，如果要解锁，则按“ESC”键。

7.3.3 窗体中的常见控件

控件是窗体上用于显示数据、执行操作、装饰窗体的对象。Microsoft Access 包含的控件有：文本框、标签、选项组、复选框、切换按钮、组合框、列表框、命令按钮、图像控件、绑定对象框、未绑定对象框、子窗体/子报表、分页符、线条、矩形，以及 ActiveX 自定义控件等，它们可以通过工具箱访问。根据作用的不同，可以将控件分为三类：结合型、非结合型与计算型。结合型控件主要用于显示、输入、更新数据库中的字段；非结合型控件没有数据源，用来显示信息、线条、矩形或图像；计算型控件主要用表达式作为数据源，显示运算结果。

1. 标签

使用标签在窗体、报表或数据访问页上显示说明性文本，例如，标题、提示等。标签是为未绑定的，它的值不会随记录的改变而改变。

2. 文本框

使用文本框在窗体、报表或数据访问页上显示记录源上的数据。如果文本框与某个字段中的数据绑定，就称这种文本框类型为绑定文本框。当然，文本框也可以是未绑定的，例如，可以创建一个未绑定文本框来显示计算结果或接收用户输入的数据。

3. 复选框、切换按钮、选项按钮控件

复选框、切换按钮、选项按钮控件作为单独控件来显示基础表、查询或 SQL 语句中的“是/否”值。若在复选框内包含了检查符号，则其值为“是”；若不包含，则其值为“否”。若选择了选项按钮，则其值为“是”；若未选择，则其值为“否”。

4. 选项按钮组

选项组含有一个组框和一组复选框、选项按钮或切换按钮。

如果选项组绑定到某个字段，则只有组框架本身绑定到此字段，而不是组框架内的复选框、选项按钮或切换按钮。可以为每个复选框、选项按钮或切换按钮的“选项值”（窗体或报表）或“Value”（数据访问页）属性设置相应的数字。

5. 列表框和组合框控件

在许多情况下，从列表中选择一個值，要比记住一个值后输入它更快更容易。选择列表也可以帮助用户确保在字段之中输入的值是正确的。窗体上的列表框可以包含一列或几列数据，用户只能从列表中选择值，而不能输入新值。列表框中的列表是由数据行组成的，在窗体或列表框中可以有一个或多个字段。组合框的列表是由多行数据组成，但平时只显示一行，需要显示时可以单击右侧的向下按钮。组合框既可以进行选择，也可以输入文本，组合框就如同文本框和列表框合并在一起。

6. 命令按钮

在窗体上可以使用命令按钮来执行某个操作或某些操作。例如，可以创建一个命令按钮来打开另一个窗体。如果要使命令按钮执行窗体中的某个事件，可编写相应的宏或事件过程并将它附加在按钮的“单击”属性中。

7.3.4 给对象添加控件

给对象添加控件的基本过程如下：在数据库窗口中选中需要设计的对象，单击“设计”按钮，在“设计视图”中打开对象，这时供用户选择和添加控件的工具箱同时打开，通过这些工具它可以添加复选框和列表签等各种控件。

1. 增加标签与画线控件

好的窗体应该具有标题，在窗体中增加标题是通过添加“标签”控件来实现的。例如，给“学生表”窗体增加标题“学生成绩表”，过程如下：

① 切换“学生表”窗体到设计模式下。选择需要移动位置的控件（“Shift”键+鼠标左键单击）。稍微挪动鼠标，鼠标的光标变成手形，通过鼠标拖动，将控件移动到合适的位置。

② 单击“工具箱”的“标签”按钮。在窗体页眉上单击鼠标左键，拖动鼠标，就会出现一个标签。在标签中输入“学生成绩表”，一个标签就插入到窗体中了。如图7.6所示。

③ 设置标签属性。单击标签边缘，出现一个黑色的边框，表示这个控件标签已经被选中，同时在 Access 窗口上出现了标签工具栏。如图7.7所示。

标签工具栏是用来定义标签控件中文字的属性以及 Word 中文字对齐方式、字体大小、颜色等属性的。如果需要对标签进行精确的设置，只需选中标签，然后单击工具栏上的“属性”按钮，屏幕上出现“标签属性”窗口。在窗口的“宽度”和“高度”项中输入相应的数值即可（单位：厘米）。

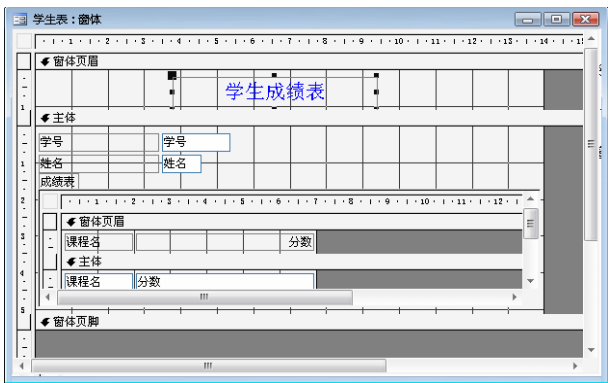


图 7.6 增加标签



图 7.7 标签工具栏

如果想在窗体上添加一条直线，可在工具箱单击“直线”按钮，将鼠标移动到窗体上，拖动鼠标画一条直线。如果要使线变粗一些，先选中“线”这个对象，单击工具栏上“属性”按钮，屏幕上出现“标签属性”窗口，设置颜色为“蓝色”，宽度为“4磅”，设置后显示结果如图7.8所示。



图 7.8 运行结果

2. 为窗体添加背景

为了使窗体更为美观，可以为窗体增加背景图案。

将窗体切换到设计视图，在视图中单击非窗体的部分，再单击工具栏上的“属性”按钮，屏幕上出现“窗体”属性对话框，如图7.9所示。在对话框中选择“格式”选项卡，并在这个选项卡中“图片”提示项的右边选择要添加的图片文件名。

关闭“窗体”属性对话框，会发现在窗体有了一个新背景，如图7.10所示。

3. 建立控件与字段联系

在窗体中增加学生的性别信息，建立控件与字段的联系，其过程如下。

① 在窗体的适当位置增加一个标签，在标签中输入“性别”。同时在新建的标签后面增加一个文本框。在文本框向导中定义文本框中文字的字体、字号、输入法模式及文本框的名称，结果如图7.11所示。

现在窗体中的控件与字段列表中的字段之间还没有联系（未绑定）。为了能够正确的显示内容，需要建立控件和字段之间的联系。

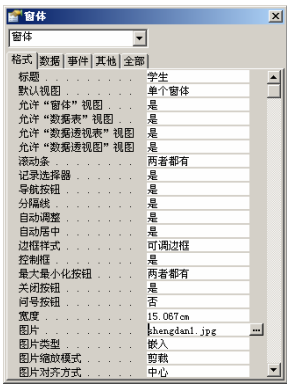


图 7.9 设置背景



图 7.10 背景设置效果



图 7.11 添加文本框控件

② 选择新建的文本框，单击工具栏上“属性”按钮，弹出“文本框”窗口，如图7.12所示。单击“数据”选项卡，单击“控件来源”后面的下拉按钮，在弹出的下拉菜单中选择“性别”字段。

③ 保存窗体，切换视图，结果如图7.13所示。

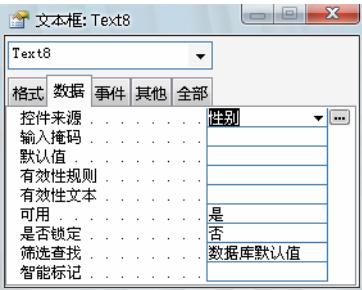


图 7.12 设置“文本框”窗口



图 7.13 设置结果

4. 建立计算控件

计算控件可以处理数据并产生临时结果，例如，通过控件来计算每个月的销售收入之和，从而得到全年的销售收入之和。计算控件的核心是表达式，它可以使用表、查询或另一个控件的数据。表达式由算术运算符、字段、预建的公式和数值等构成。

新建一学生信息窗体，如图7.14所示。

现在要将“出生日期”改为“年龄”来显示。切换到设计视图下，增加文本框，修改标题为“年龄”，如图7.15所示。



图 7.14 新建学生信息窗体

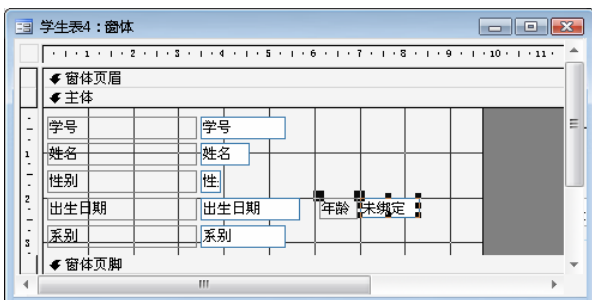


图 7.15 增加文本框

接下来修改文本框的属性。将鼠标指向文本框，右键单击控件，选择快捷菜单中的“属性”命令，在弹出的窗口中选择“数据”选项卡，单击“控件来源”属性框旁边的按钮，就可以打开“表达式生成器”窗口，如图7.16所示。

表达式生成器自上而下可以分为三个部分：

① 表达式框。生成器的上方是一个表达式框，用户生成的表达式就显示在里面。

② 运算符按钮。生成器的中部排列着常用运算符按钮，单击某个按钮即可在表达式框的插入点插入运算符。

③ 下方三个框。左侧的框中包含文件夹，它列出了表、查询、窗体、报表等数据库对象，以及内置和用户定义的函数、常量、操作符和通用表达式。中间的框列出左侧选中文件夹内的特定元素或元素类别。例如，单击左边框中的“内置函数”，中间的框便会列出 Microsoft Access 函数的类别。右侧的框列出了左侧和中间框选中元素的值。假如单击了左侧框中的“内置函数”，然后选中了中间框中的“日期/时间”，则右侧的框将列出“日期/时间”类别中所有内置函数。选中需要的内置函数单击“粘贴”按钮，就可以把这个函数放入表达式框。

在表达式生成器窗口输入计算表达式，如图7.17所示。

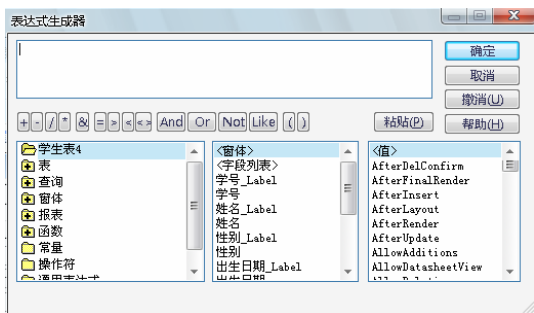


图 7.16 “表达式生成器”窗口

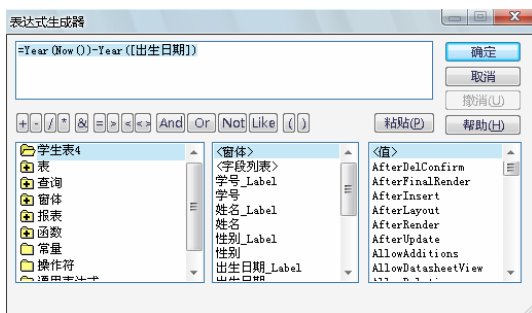


图 7.17 输入计算表达式

保存窗体，运行结果如图7.18所示。

5. 修改控件属性

在设计视图中右键单击对象上放置的控件，选择快捷菜单中的“属性”命令，就可以在“全部”选项卡中看到控件的属性了。其中，下列属性是比较常用的。



图 7.18 运行结果

(1) 标题：所有的窗体和标识控件都有一个标题属性。当作为一个窗体的属性时，标题属性定义了窗口标题栏中的内容。如果“标题”属性为空，窗口标题栏则显示窗体中字段所在表格的名称。当作为一个控件的属性时，标题属性定义了标识控件的文字内容。

(2) 控件提示文本：该属性供用户输入控件的提示文本，用户将鼠标放在控件上会显示提示文字。

(3) 控件来源：在一个独立的控件中，“控件来源”告诉系统如何检索或保存在窗体显示的数据。如果一个控件的作用是更新数据，那么“控件来源”属性可以设置为字段名。如果“控件来源”属性中含有一个计算表达式则该控件又称为计算控件。

(4) 是否锁定：这个属性决定了一个控件中的数据是否能够被改变。如果设置为“是”，则该控件中的数据被锁定且不能被改变，该控件处于锁定状态，在窗体中呈灰色显示。

6. 整体移动控件

设计窗体或报表等对象时需要移动多个控件，一次移动多个控件可以按如下方法操作。在设计视图下将窗体或报表等打开，鼠标在操作对象上画出一个矩形，即可将围在里面的对象全部选中。接着单击“格式”菜单中的“组合”命令，刚才被选中的控件就被组合为一个整体，就可以将它们当成一个控件那样移动。组合后的控件如果需要取消组合，只需按相同方法将它们选中，单击“格式”菜单中的“取消组合”命令即可。

7.4 创建子窗体

子窗体是指在一个窗体中插入的窗体。将多个窗体合并时，其中一个窗体作为主窗体，其余作为子窗体。

下面创建一个课程表的主窗体，然后增加一个子窗体来显示每门课的成绩单明细情况。

- ① 通过设计视图创建课程表主窗体，如图 7.19 所示。
- ② 在设计视图下添加要显示的内容，包括：课程号、任课教师、课程名、开课时间，同时添加一个子窗体，结果如图 7.20 所示。

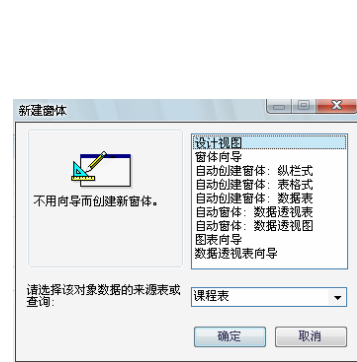


图 7.19 创建窗体

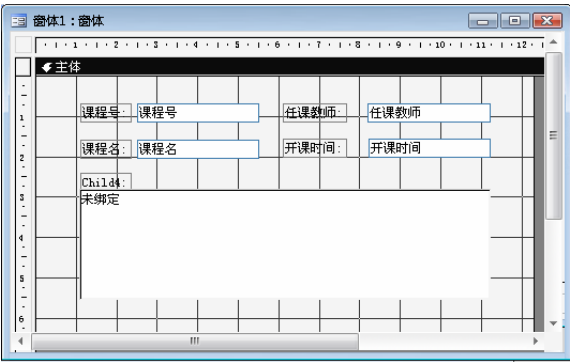


图 7.20 添加控件创建子窗体

- ③ 将子窗体和成绩表绑定，结果如图 7.21 所示。
- ④ 运行窗体结果如图 7.22 所示。

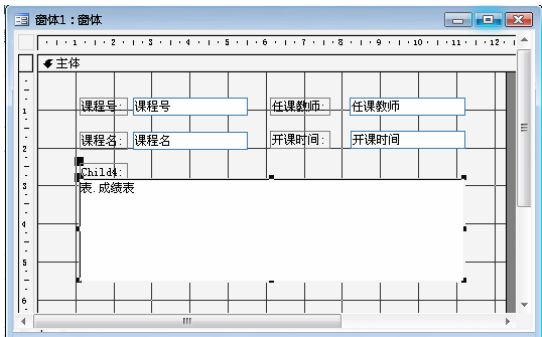


图 7.21 绑定子窗体

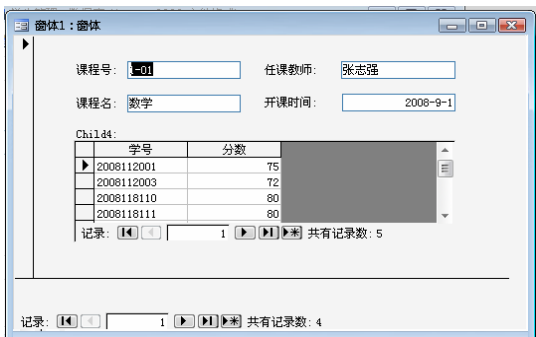


图 7.22 运行结果

7.5 通过窗体处理数据

创建完窗体之后，可以对窗体中的数据进行进一步操作，如数据的查看、添加修改、以及删除等。除此之外，还可以对数据进行查找、排序和筛选等。

7.5.1 窗体视图工具栏

窗体视图工具栏，如图7.23所示。主要的操作按钮有：视图、升/降序、按选定内容筛选、按窗体筛选、应用筛选、查找、新记录、删除记录、属性、数据库窗口、新对象等。



图 7.23 窗体视图工具栏

有关窗体中的查找、排序、筛选等按钮的使用方法与表和查询中的方法类似，详细方法参见第 5 章 5.5 节内容。

7.5.2 记录导航按钮集

当通过“窗体向导”创建窗体时，在窗体的底部的会有标准记录导航按钮集，如图7.24所示。记录导航按钮集在窗体中的功能和它们在表和查询中的作用相同。既可以为主窗体选择作为数据源的表或者查询的第一个或最后一个记录，也可以选择下一个或者前一个记录。子窗体还包含它们自己的记录选择按钮集，在操作上和主窗体中的按钮集相互独立。

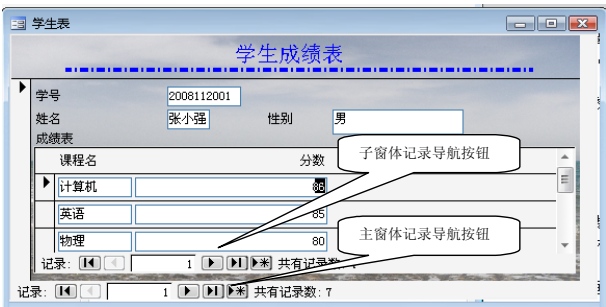


图 7.24 记录导航按钮

在窗体中用于输入或者编辑数据的文本框之间的导航与在表或者查询的数据表视图中的导航类似，只是上下箭头键用于在字段之间而不是在记录之间移动。输入完成时，可以按“Enter”或者“Tab”键。

7.5.3 处理数据

1. 追加记录

在表或者查询的“数据表”视图中，数据表中的最后一个记录是作为假设追加记录提供的（在其记录选择按钮上有一个星号作为指示）。如果在这个记录中输入数据，则该数据将自动地追加到表中，Access 将启动另一个新的假设追加记录。窗体也提供了假设追加记录，除非将窗体的“允许添加”属性设为“否”。

追加一个新的记录并输入必填字段数据的步骤如下：

- ① 打开“学生表”窗体（如果当前在设计视图中，则单击“窗体视图”按钮）。
- ② 单击记录导航按钮集上的“追加记录”按钮（最右边的按钮），窗体出现一条空记录，如图 7.25 所示，输入各字段的内容，可使用 Tab 键在不同字段间移动。



图 7.25 追加记录

2. 编辑数据

可以用与添加新记录同样的方式编辑现有记录。首先单击“下一个”按钮找到想要编辑的记录，然后进行修改。也可以使用工具栏的“查找”按钮来查找特定的记录。

3. 删除数据

浏览记录，使需要删除的记录出现在窗体中。在窗体允许更新的前提下，在“编辑”菜单中选择“删除记录”命令删除记录。

4. 确认和撤销对表的修改

像处理假设追加记录一样，在移动带有记录选择按钮集的记录指针（或者单击“记录”、“保存记录”按钮）之前，Access 不会将记录编辑应用到后台表中。即便记录保存到表之后，马上单击工具栏上的“撤销”按钮也可以撤销刚才的保存。

7.6 创建切换面板

7.6.1 切换面板的作用

切换面板是一种带有按钮的特殊窗体，用户可以通过单击这些按钮在数据库的窗体、报表、查询和其他对象中查看、编辑或添加数据。切换面板上的每一个条目都连接到切换面板

中的其他页，或是链接到某个动作。切换面板不仅提供了一个友好的界面，还可以避免用户进入数据库窗口，特别是窗体或报表的设计视图。通过切换面板管理器，用户可以对向导提供的切换面板进行修改，也可以自己创建切换面板。

7.6.2 切换面板的创建

(1) 打开数据库。在“工具”菜单上，选择“数据库实用工具”下的“切换面板管理器”。系统询问是否要新建切换面板时，请单击“是”。系统弹出“切换面板管理器”窗口，如图7.26所示。

(2) 单击“编辑”按钮。系统弹出“编辑切换面板页”窗口，如图7.27所示。

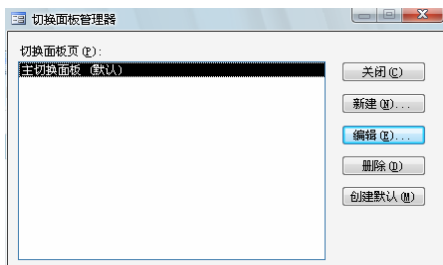


图 7.26 “切换面板管理器”窗口

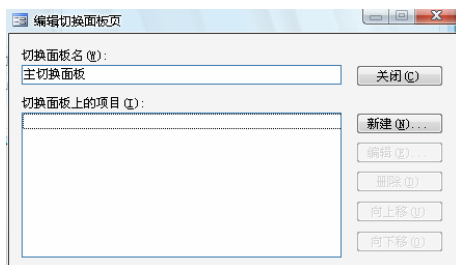


图 7.27 “编辑切换面板页”窗口

(3) 单击“新建”按钮。系统弹出“编辑切换面板项目”窗口。在“文本”框中为第一个切换面板项目键入文本，然后在“命令”框中选择相应的命令。

例如，在“文本”框中输入“成绩表”，在“命令”框中选择“在‘编辑’模式下打开窗体”。在“窗体”框中选择“成绩表子窗体”作为打开的对象，结果如图7.28所示。

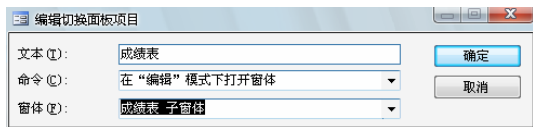


图 7.28 “编辑切换面板项目”窗口

Access 2003 提供了一些命令类型，具体说明如下：

- ① 转至“切换面板”：打开另一个切换面板并关闭自身面板，参数为目标面板名。
- ② 在“添加”模式下打开窗体：打开输入用窗体，出现一个空记录，参数为窗体名。
- ③ 在“编辑”模式下打开窗体：打开查看和编辑数据用窗体，参数为窗体名。
- ④ 打开报表：打开打印预览中的报表，参数为报表名。
- ⑤ 设计应用程序：打开切换面板管理器以对当前面板进行更改，无参数。
- ⑥ 退出应用程序：关闭当前数据库，无参数。
- ⑦ 运行宏：运行宏，参数为宏名。
- ⑧ 运行代码：运行一个 VB 过程，参数为 VB 过程。

(4) 单击“确定”按钮。

(5) 重复(3)、(4)步直到已经在切换面板中添加了所有要添加的项目为止。本例中在编辑切换面板上再增加两个项目：学生表和课程表，具体设置如图7.29和图7.30所示。最后的设置结果如图7.31所示。

若要编辑或删除某项目，可在“切换面板上的项目”框中选中该项目，然后右键单击“编辑”或“删除”。若要重新排列项目顺序，则选中相应的项目，然后单击“向上移”或“向下移”按钮。

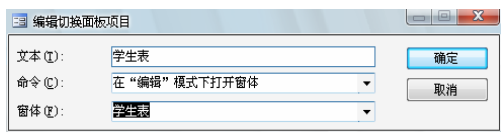


图 7.29 学生表项目设置

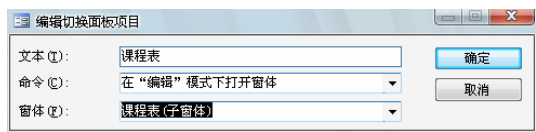


图 7.30 课程表项目设置

(6) 在完成创建切换面板后，单击“关闭”按钮。
运行切换面板，结果如图7.32所示。

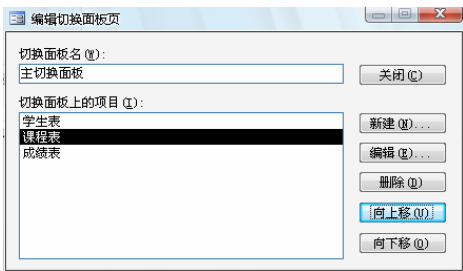


图 7.31 设置结果

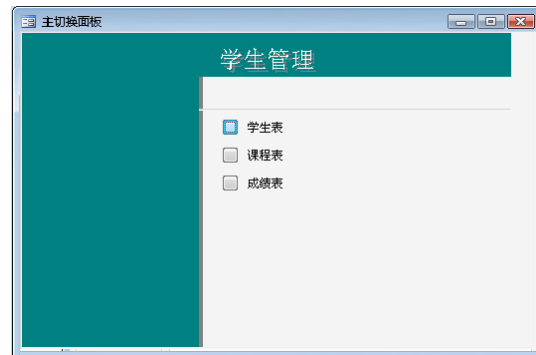


图 7.32 “切换面板”窗口

要使其中的某个切换面板在打开数据库的同时自动被打开，可以在“切换面板管理器”窗口中选中该切换面板的名称，然后单击“创建默认”按钮。
通过“工具”菜单“启动”项进行设置。设置如图7.33所示。

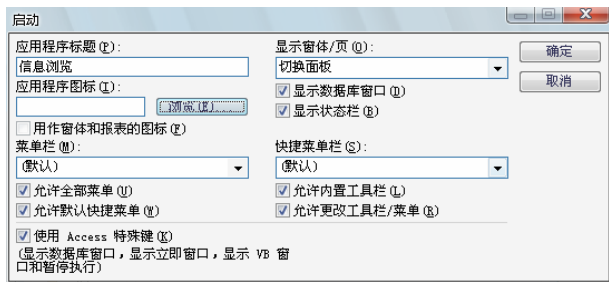


图 7.33 设置启动选项

当用“切换面板管理器”创建切换面板时，Access会创建一个“切换面板项目”表，用来描述窗体上的按钮显示的内容，以及其具有的功能。如果以后在窗体“设计”视图中更改了“切换面板”窗体，该应用程序可能无法再正常运行。

习 题 7

一、填空题

- _____是窗体上用于显示数据、执行操作、装饰窗体的对象。
- 要为新建的窗体添加一个标题，必须使用_____控件。
- Access 中，窗体上显示的字段为表或_____中的字段。
- 窗体通常由窗体页眉、窗体页脚、页面页眉、页面页脚及_____五部分组成。

15. 关于窗体的显示视图说法正确的是()。
- A. 不可以进行数据的修改 B. 不可以进行数据排序
- C. 不可以进行数据筛选 D. 不可以进行窗体控件的修改
16. Access 自动创建窗体的方式有()种。
- A. 2 B. 3 C. 4 D. 6
17. 使用窗体向导创建基于一个表的窗体, 可选择的布局方式有()种。
- A. 4 B. 5 C. 2 D. 3
18. 内部计算函数“Sum”的意思是求所在字段内所有的值的()。
- A. 和 B. 平均值 C. 最小值 D. 第一个值
19. 内部计算函数“Avg”的意思是求所在字段内所有的值的()。
- A. 和 B. 平均值 C. 最小值 D. 第一个值
20. 内部计算函数“Min”的意思是求所在字段内所有的值的()。
- A. 和 B. 平均值 C. 最小值 D. 第一个值

三、简答题

1. 简述窗体的主要功能。
2. 与自动窗体比较, 窗体向导有什么优点?
3. 子窗体与链接窗体有什么区别?
4. 窗体有几种视图? 各有什么作用?
5. 举例说明在属性窗口中设置对象属性值的方法。
6. 如何为窗体设定数据源?
7. 什么是控件? 控件可分为哪几类?
8. 如何给窗体上添加绑定控件?
9. 举例说明如何创建计算型控件。

四、操作题

对学生成绩数据库进行如下操作:

(1) 创建窗体。窗体名为“总分计算”, 要求输入一个学号后, 单击“计算”按钮, 将计算出来的总分显示在表示总分的文本框里。

(2) 创建窗体。创建“学生基本资料信息输入”窗体、“课程信息输入”窗体、“学生成绩输入”窗体, 目的是为输入数据建立良好的用户界面。分别以“学生信息”查询、“课程信息”查询、“学生成绩”查询为数据源, 建立“学生信息查询”窗体、“课程信息查询”窗体、“学生成绩查询”窗体。

(3) 创建主窗体, 在主窗体上创建六个命令按钮, 单击这些按钮可以激活“学生基本资料信息输入”窗体、“课程信息输入”窗体、“学生成绩输入”窗体、“学生信息查询”窗体、“课程信息查询”窗体、“学生成绩查询”窗体, 从而实现学生信息的管理。

第8章 数据报表

输出数据可以通过报表来实现，报表为查看和打印概括性的信息提供了最灵活的方法，在报表中，不仅可以控制每个对象的大小和显示方式，按照所需的方式来显示相应的内容，同时还可以添加多级汇总、统计比较，甚至加上图片和图表。

8.1 报表的功能与类型

8.1.1 报表的功能

尽管数据表和查询都可以打印，但是报表才是打印信息的最佳方式。报表作为 Access 数据库的一个重要组成部分，不仅可用于数据分组，单独提供各项数据和执行计算，还可以制成各种丰富的格式，使用户的报表更易于阅读和理解；可以使用剪贴画、图片或者扫描图像来美化报表的外观；通过页眉和页脚，可以在每页的顶部和底部打印标识信息；可以利用图表和图形来帮助说明数据的含义。与其他的打印数据方法相比，报表具有以下两个优点：

① 报表不仅可以执行简单的数据浏览和打印功能，还可以对大量原始数据进行比较、汇总和小计。

② 报表可生成清单、订单及其他所需的输出内容，可以方便有效地处理信息。

8.1.2 报表类型

在 Access 中，一般可以创建六种类型的报表：单列报表、表格式报表、多列报表、分组/总计报表、邮件标签报表、未绑定报表。

1. 单列报表

单击工具栏上的“自动报表”按钮，Access 的“自动报表”功能便会创建一个单列报表，单列报表在很长的一列文本框列表中列出了表或者查询中每个记录的每个字段值。字段的名字用标签指示，而值则由标签右边的文本框提供。单列报表很少用到，因为这种格式太浪费纸张。

2. 表格式报表

表格式报表为表或者查询的每个字段都提供了一列，并且在列标题下将每个记录的所有字段值都打印在一行上。如果列数太多，一页放不下，将使用额外的页依次进行打印，直到所有的列打印为止，然后再打印下一组记录。

3. 多列报表

多列报表中，第一列中放不下的信息转到第二列，依此类推。多列表的格式较为节约纸张，但是其使用也很受限，因为列对齐有时并非与期望相吻合。

4. 分组/总计报表

分组/总计报表是最常见的一类报表，分组/总计报表对记录组进行合计，然后在报表的末尾添加重要的总计。

5. 邮件标签报表

邮件标签报表是一种特殊的多列报表，它以组的形式打印地址和名字(或者其他多字段数据)。每个字段组构成网格中的一个单元，为打印而设计的常备胶粘标签决定了在一页上打印多少行和列。

6. 未绑定报表

未绑定报表包含了一个或多个基于不相关数据源（例如表或者查询）的子报表。前五种报表类型使用表或者查询作为数据源，这些种类的报表称为绑定到数据源。未绑定报表的主报表没有链接到作为数据源的表或者查询。但是，未绑定报表包含的子报表必须绑定到数据源上，未绑定报表允许集成一个或多个绑定到独立表或者查询的子报表。

8.2 报表设计的组成

在开始报表的设计之前，必须了解报表的几个基本组成部分。报表利用其不同的组成部分将从数据库数据中所提取的信息有机地展现在用户面前。对报表所进行的设计也就是围绕这些组成部分中的控件进行的。

8.2.1 报表的节

报表的设计视图，如图8.1所示。可以在该视图中清楚地看到报表的每个组成部分，报表中的内容是以节作为单位划分。所有的节在设计视图中水平方向无限伸展。

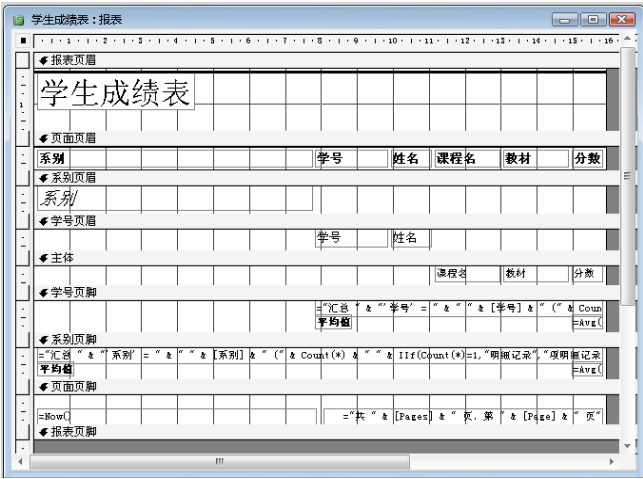


图 8.1 报表的设计视图

报表一般由 5 个节组成，它们分别是：报表页眉、报表页脚、页面页眉、页面页脚、主体，其中报表主体还包括它的标头和注脚。每个节都有特定目的，而且按照一定的顺序打印在页面及报表上，节的位置和其作用如表 8.1 所示。

在设计视图中，节代表着各个不同的带区，而且报表所包含的每节一般只能被指定一次。可以通过放置控件来确定在每节中显示内容的位置，例如，选项卡和文本框。通过对使用相同数值的记录进行分组，可以进行一些计算或简化报表使其易于阅读。在此报表中，相同系别的学生分在同一个组中。报表的最后输出结果如图8.2所示（图 8.1 是它的设计视图）。

表 8.1 报表中节的位置和作用

节	位 置	作 用
报表页眉	只出现在报表开始的位置	在报表中显示标题、徽标、图片，以及其他报表的标识物
页眉	位于每页的最上部	显示字段标题、页号、日期和时间
组标头	位于每个字段组的开始处，需对字段排序或分组的地方	显示标题和总结性的文字
报表主体	每个有下划线的记录都有相对应的详细内容	显示记录的详细内容
主体注脚	位于每个字段的末端，报表主体之后和页脚之前	显示计算和汇总信息
页脚	位于每页的最下部	可以是日期、页号、报表标题、其他信息
报表页脚	只出现在报表结束位置	总结性文字

在默认情况下，通过“新建报表”窗口中“设计视图”选项新建一个报表，新报表中自动包含页眉、页脚和主体这三个节。下面将对报表中各个节的详细内容加以介绍。

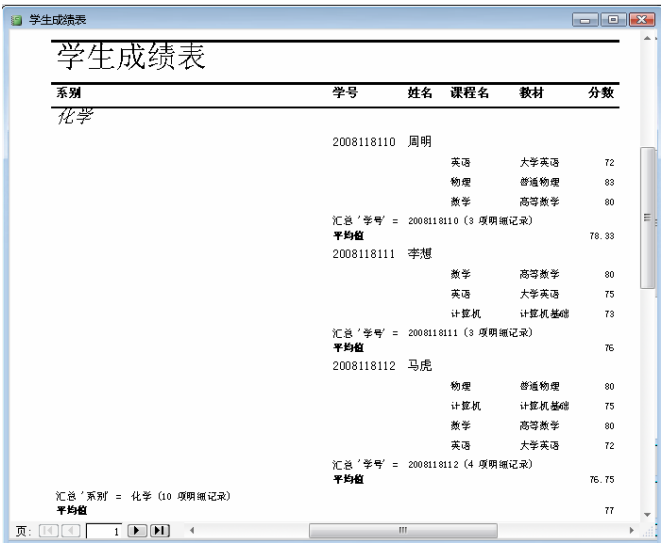


图 8.2 打印预览视图中的报表输出

8.2.2 报表的常见节

1. 报表的页眉

报表的页眉以大字体将该份报表的标题放在报表顶端。只有报表的第一页才显示报表页眉内容。报表页眉的作用是用作封面。报表页眉只能在报表中出现一次，利用分页控件的功能可以将报表中的报表页眉单独作成一份报表的开始或封面，把报表页脚作成一份总结或结束页。如果要在设计视图中显示或隐藏报表页眉，可以在“视图”菜单中选中或取消“报表页眉/页脚”命令即可。

2. 页面页眉

页面页眉中的文字或字段，通常会打印在每页的顶端。如果报表页眉和页面页眉共同存在于第一页，则页面页眉数据会打印在报表页眉的数据下。一个典型的页眉包括页数、报表标题或字段选项卡等。同样，可以利用某些特殊效果对页眉进行处理，突出其包含的内容。如果要在设计视图中显示或隐藏报表页眉，可以在“视图”菜单中选中或取消“页面页眉/页脚”命令即可。

3. 报表的组标头

组标头通常用一个特殊的选项卡来识别,表明报表主体中显示出来的是字段内容。当报表主体使用 Access 报表的排序和分组特性的时候,Access 就会为报表中每个字段添加一个页眉字段选项卡,按照这些字段对记录进行分组。在页面页眉和页面页脚之间是报表主体,即报表的详细信息段。主体中显示每条记录。组标头可以有多级,利用多级可以对显示的主要信息内容进行细分。可以利用组标头属性表对组标头的显示进行设定。

4. 报表的主体

报表的主体用于处理每一条记录,其中的每个值都要被打印。主体区段是报表内容的主体区域,通常是含有计算的字段。可以通过报表设计窗口的“可见性”属性设置是否需要显示报表中的某个字段。通过关闭主体的显示,就能够显示一个不带主体或只显示某些组的总计报表。

5. 报表的组注脚

使用组注脚可以对报表主体中分组之后的所有记录进行统计。对于每个组注脚中文本框,都有一个“运行总和”属性,可以利用该属性改变计算总计的方法。

6. 页面页脚

页面页脚通常包含有页码或控件总计。在一个很大的报表中,可能报表主体中的记录很多,使得在每页中没有一个统计数字,这就需要在报表设计中既包含页码总数又有报表主体中被分组的记录总数。一般来说,在页脚显示页脚时,都是采用将文本控件 Page 与表达式 Page 结合在一起打印出页码,页码文本框中的内容一般为: = “第'&[Page]&'页”。

7. 报表的页脚

报表注脚位于一个报表设计视图中的最底部,是在所有记录数据和报表主体都输出之后打印在报表的结束处。在输出时和报表页眉一样,只出现一次。典型的报表页脚显示所有记录的汇总结果,例如记录的总计数、平均值和百分率等。当在报表页脚中添加了输出的内容,最后一页中的页面页脚会在报表页脚的输出内容之后输出。

8.3 使用报表向导建立报表

创建报表有三种方法:用向导基于一个或多个表或查询创建报表,使用自动报表基于一个表或查询创建报表,在设计视图中手动创建报表。

使用报表向导建立报表的步骤如下。

① 打开“学生成绩管理”数据库,在数据库窗口左侧的选项卡上选择“报表”对象。

② 双击数据库窗口中创建方式栏中的“使用向导创建报表”项,这时在屏幕上会弹出“报表向导”窗口。

这个窗口中要求确定报表的数据来源和构成字段。在“表/查询”下面的下拉框中选择相应的表或查询(以“成绩表”为例),在“可用字段”列表框中便出现所选表的构成字段,选择所需字段,结果如图8.3所示。

③ 单击“下一步”按钮。Access 询问是否要对报表添加分组级别,如图8.4所示。

所谓分组级别就是报表在打印的时候,各个字段是否是按照阶梯的方式排列,分几组,就有几级台阶。当“报表”有多个分组级别时,可以通过两个优先级按钮来调整各个分组级间的优先关系,排在最上面的优先级最高。

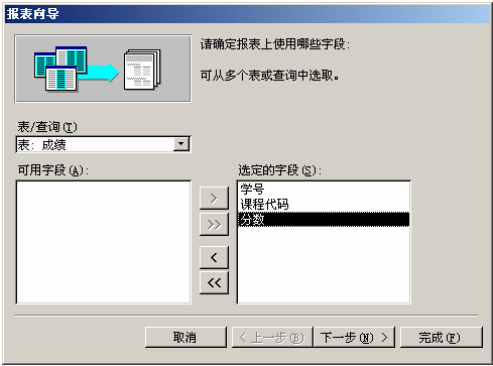


图 8.3 “报表向导”窗口

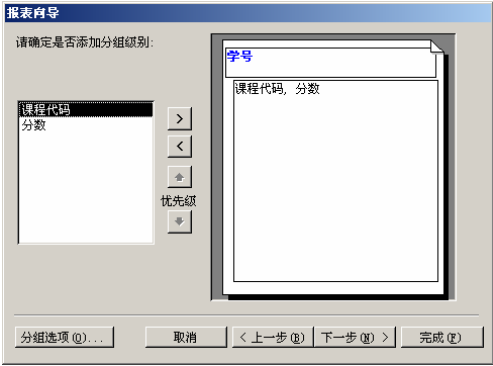


图 8.4 报表分组

④ 单击“下一步”按钮。现在需要确定记录所用的排序次序，即确定报表中各个记录按照什么顺序从报表的上面排到下面，如图8.5所示。例中选择“分数”的升序方式。

⑤ 单击“下一步”按钮。在这一步需要确定报表的布局方式，如图8.6所示。

⑥ 单击“下一步”按钮，选择报表的“样式”，例中选择为“组织”样式。

⑦ 单击“下一步”按钮，为报表确定“标题”，例中输入为“成绩表”，然后单击“完成”按钮，结果如图8.7所示。

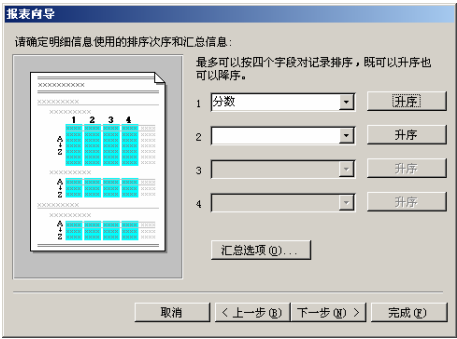


图 8.5 选择记录顺序

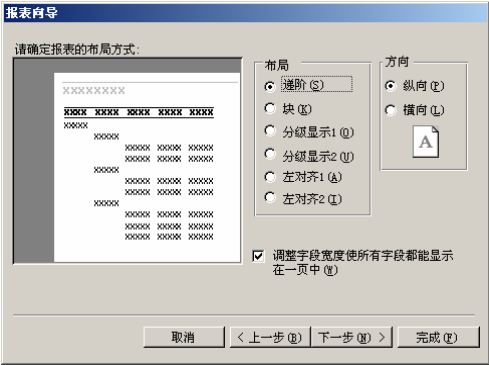


图 8.6 选择布局

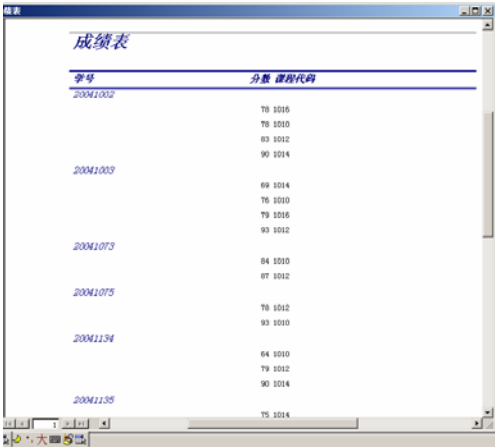


图 8.7 成绩表

8.4 自动报表与图表向导

8.4.1 创建自动报表

自动报表是基于一个表或查询来创建报表，自动创建的报表能够显示基础表或查询（查询中所存储的数据问题）。创建自动报表的过程如下。



图 8.12 选择图表数据所在字段

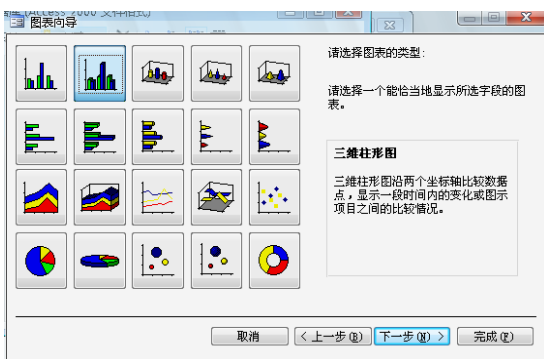


图 8.13 选择图表的类型

⑤ 指定图表的标题，运行图形报表，结果如图8.15所示。

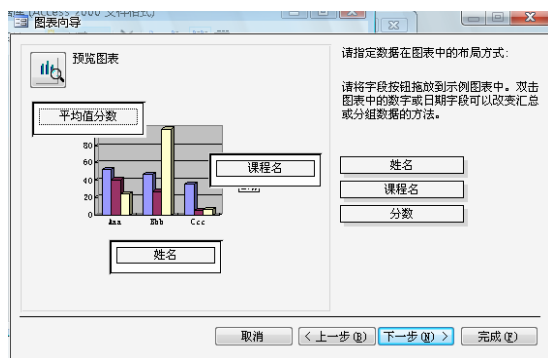


图 8.14 指定图表的布局

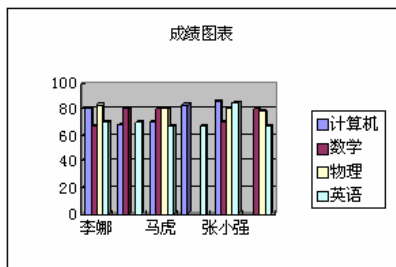


图 8.15 运行结果

8.5 通过设计视图创建报表

如同数据库中创建的大多数对象一样，用户可以采用多种方式来创建所需的报表。

8.5.1 创建过程

通过设计视图创建报表的过程如下。

首先，打开数据库窗口，选择“对象”栏下的“报表”对象，然后单击“报表设计”工具栏上的“新建”按钮，弹出“新建报表”窗口。如图8.16所示。

选择“设计视图”，单击“确定”按钮，出现新建报表工作区，如图8.17所示。

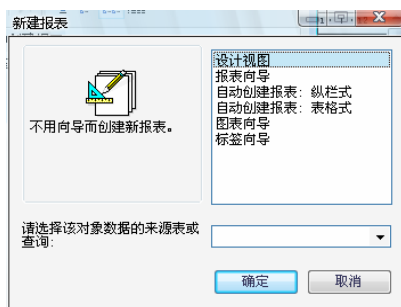


图 8.16 “新建报表”窗口

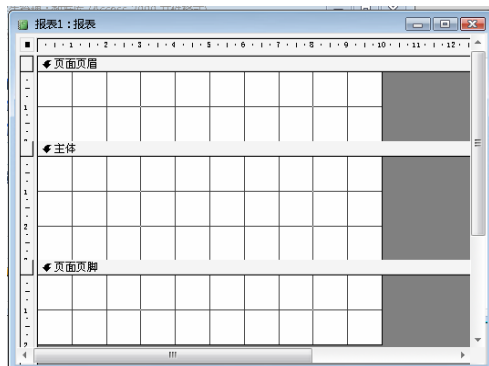


图 8.17 新建报表工作区

8.5.2 报表控件

1. 控件的分类

报表中的每一个对象，都称为控件，控件分为三种。

(1) 绑定控件

绑定控件与表字段绑定在一起。在向绑定控件输入值时，Access 自动更新当前记录中的表字段值。大多数允许输入信息的控件都是绑定控件。绑定控件可以与大多数数据类型捆绑在一起，包括文本、日期、数值、是/否、图片、备注字段等。

(2) 非绑定控件

非绑定控件保留所输入的值，不更新表字段值。这些控件用于显示文本、传递数据。

(3) 计算控件

计算控件是建立在表达式（如函数和计算）基础之上的。计算控件也是非绑定控件，它不能更新字段值。

2. 添加控件

用户可以在设计视图中对控件进行如下操作：

- ◆ 通过鼠标拖动创建新控件、移动控件；
- ◆ 通过按 Delete 键删除控件；
- ◆ 激活控件对象，拖动控件的边界调整控件大小；
- ◆ 利用属性窗口改变控件属性；
- ◆ 通过格式化改变控件外观，可以运用边框、粗体等效果；
- ◆ 对控件增加边框和阴影等效果。

要想在报表中添加非绑定控件，可通过在“工具箱”中选择相应的控件，拖动到报表上即可。向报表中添加绑定控件是一项重要工作，这类控件主要是文本框，它与字段列表中的字段相结合来显示数据。在报表中创建计算控件时，如果控件是文本框，可以直接在控件中输入计算表达式。另外不管控件是不是文本框，都可以使用表达式生成器来创建表达式。

3. 控件的更改和设置

更改控件的方法通常有两种：即在窗体内直接修改，或利用属性窗口进行修改。除可以移动控件的位置和改变控件的尺寸外，还可以通过属性窗口设置控件的属性。方法是右键单击需要进行属性设置的控件，在弹出的快捷菜单中选择属性。

添加控件的过程如下：

- ① 在“新建报表”窗口中选择创建方式和数据源，如图8.18所示。
- ② 在主体节中添加所要显示的内容，结果如图 8.19所示。
- ③ 保存报表，显示结果如图8.20所示。

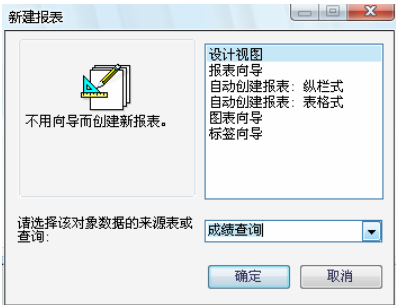


图 8.18 设计视图



图 8.19 添加控件



图 8.20 显示结果

可以看出，没有分组的报表，结构不清晰，数据有大量的重复，内容显示没有规律。因此，需要对内容进行分组和排序，来修正这些不足。

8.5.3 在报表中添加分组

组就是指由具有某种相同信息的记录组成的集合，在报表中计算汇总信息之前，必须进行分组。将报表分组之后，不仅同一类型的记录显示在一起，而且还可以为每个组显示概要和汇总信息，可以提高报表的可读性和易懂性。当按不同字段分组时，除了可以利用整个字段本身作为分组原则，还可以指定分组字段的细节内容作为分组依据。例如，在利用“日期/时间”字段进行分组时，可以指定按照年、月、日等对记录进行分组，将属于相同年份、月份和日子的记录归到同一组中。在利用“文本”字段进行分组时，可以只取字段的前几个字符作为分组依据。

如果要在报表中对记录进行分组，首先要利用报表“设计”视图中的“排序与分组”命令，建立分组所依据的字段或表达式，并通过设置分组的组属性来实现报表的汇总功能。给报表添加分组的过程如下。

① 在报表设计视图中，单击“视图”菜单下的“排序和分组”命令，系统弹出“排序和分组”窗口，在窗口中添加用于排序和分组的字段，结果如图8.21所示。

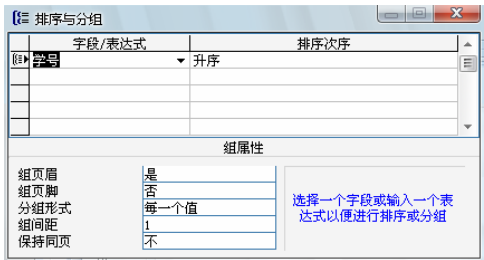


图 8.21 设置排序和分组字段

② 在设计视图中增加一个分组节，在节中添加字段，如图8.22所示。

③ 保存报表并运行，结果如图8.23所示。可以发现，经过分组之后，内容清晰明了。

在“字段/表达式”选项选取的字段不一定是报表的一个分组依据，有可能只是按照该字段对报表中记录所进行的排序。查看一个字段是否是分组字段，需要看组属性中的“组标头”和“组页脚”两项是否被选中。如果其中一个属性为“是”，那么可以认为是报表的一个分组字段。每一个分组字段同时也是一个排序字段，但不是报表中所有的排序字段都是分组字段，有可能都不是，要区分好两者的不同。

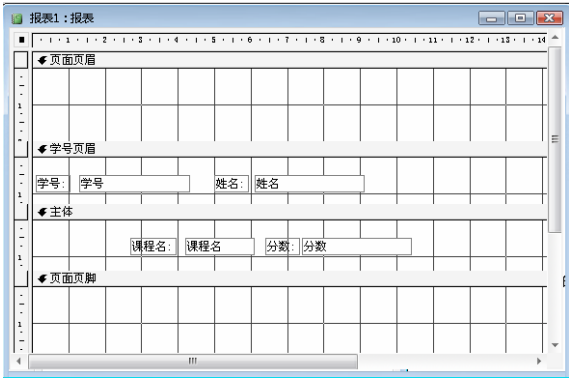


图 8.22 在节中添加字段



图 8.23 添加分组后的显示结果

当把用于分组的字段放到组标头中时，Access 就按指定的字段对记录进行分组，把属于同一组的记录放在一起。如果在“排序与分组”窗口中选取多个分组依据，创建的报表将按照多个字段或表达式对记录进行分组。Access 在分组时，首先按第一字段或表达式分组，当记录属于同一组时再按照下一个字段或表达式分组，依此类推。

8.5.4 添加计算字段

要在报表中显示汇总数据，就必须建立计算字段，利用计算字段计算所需的数据，并把它在计算控件中显示出来。计算控件就是以表达式作为数据来源的控件，表达式可以使用表或查询中的字段数据。一般来说，具有“数据来源”属性的控件都可以作为一个计算控件，建立计算控件的方法与在窗体中建立计算控件一样。操作过程如下。

- ① 在报表设计视图中，添加分组页脚，利用工具箱中的“文本框”按钮在需要输出计算结果的设计页面中添加一个文本框，如图8.24所示。
- ② 在文本框处于选中状态时，单击工具栏上的“属性”按钮，打开该控件的属性表对话框。在属性表中，单击“数据来源”属性框右边的“生成器”按钮，打开“表达式生成器”窗口，利用“表达式生成器”对话框输入计算表达式，例如，计算总分时，输入表达式“=sum(总分)”。单击“确定”按钮，结果如图8.25所示。



图 8.24 添加计算字段



图 8.25 设置计算字段属性

③ 保存报表并运行，结果如图8.26所示。

如果用户对计算表达式比较熟悉，可以直接在属性框中输入计算表达式。计算控件就根据表或查询中的数据动态计算出表达式的数值。如果计算控件是文本框，则可以在设计视图中直接输入表达式，但是每个表达式之前都要加上“=”运算符。在利用“表达式生成器”窗口输入表达式时，可以省略等号的输入。在组注脚的计算控件中输入计算表达式时，除了可以利用数据库字段中的数据，还可以利用其他计算控件的结果。利用前者时，在表达式的计算对象中输入需要的字段名；利用后者时，在表达式的计算对象中输入需要的计算控件的名称即可。但是，对于后者不能利用其他报表区域中的计算控件名作为计算对象。

学号	姓名	课程名	分数	总分
2008112001	张小强	物理	80	326
		计算机	86	
		英语	85	
		数学	75	
2008112002	王华	计算机	83	155
		英语	72	
		总分	155	

图 8.26 显示结果

8.6 修饰报表

8.6.1 添加文字

在设计视图中修改报表的方法和过程与在设计视图中修改窗体的方法基本一致。如果要在报表中添加一行文字，其过程如下：

① 首先将报表切换到设计模式。

② 将鼠标移动到工具箱上单击“标签”图标，将鼠标移到报表需要加文字的地方，按住鼠标左键，拖动鼠标，在屏幕上会出现的矩形虚线框。

③ 在报表上出现一个标签控件，输入文字便可。

当然，可以根据需要移动标签的位置，在报表中移动控件和在窗体上移动是一样的。修改标签控件中文字的字体、大小和颜色也和在窗体中修改这些属性的方法是一样的。

8.6.2 设置内容的显示效果

如果想要报表的内容显示具有特定的格式，那么可以根据自己的实际需求进行显示格式的设置。具体方法是：首先选中需要设置属性的控件，然后单击鼠标右键，在快捷菜单中选择所需要的设置效果（例如，可以设置填充色、字体颜色、以及特殊效果）。

8.6.3 调整显示对齐方式

在报表上进行标签和文本框控件的精确对齐比在窗体上更为重要，因为打印结果是否对齐一目了然。对控件进行格式化可以进一步美化报表的呈现。

对齐控件时，首先要选择对齐行，然后再选择对齐列。Access 提供了控件缩放和对齐选项，使得对齐过程变得容易。为了改变创建的控件大小和对之进行对齐处理，可遵循如下步骤：

① 可以同时调整所有文本框的高度使之适合其内容。选择“编辑”→“全部选定”，便可选择报表中的所有控件。

② 选择“格式”→“大小”→“正好容纳”，调整所选定控件的高度。Access 将调整所有控件到合适的高度。

- ③ 选择“页面页眉”节的所有标签。单击“格式”，选择“对齐”，然后选择“靠上”。这个过程将把所选定的每个标签的顶部和选定标签中最为靠上的那个标签的顶部对齐。
- ④ 选择“主体”节中的所有文本框，在这些文本框上重复步骤③，可以实现对应的设置。

8.6.4 调整行间距

通过调整行间距可使打印的报表显示得更加清晰。调整可以通过改变报表的“页面页眉”和“主体”节间距实现。首先通过选择“格式”→“对齐网格”，可以更为精确地调整控件的大小和行间距。这个命令将在“对齐网格”功能打开和关闭两种状态之间切换。然后通过选择“页眉”中的所有标签，向上移动该组，使它们距离节的顶部越近越好，然后调整节的高度便可。

8.7 打印报表

8.7.1 页面设置

报表最终需要打印，在打印之前，先要进行页面的设置，过程如下：在窗口的主菜单上，单击“文件”按钮，在弹出的菜单中单击“页面设置”项，这时会弹出“页面设置”窗口，如图8.27所示。

在页面设置对话框中可以定制打印纸的一些属性。“边距”选项卡中的页边距是打印纸上正文距离纸边缘的距离。“页”选项卡中的打印方向就是确定内容是横着还是竖着打印出来。

8.7.2 预览与打印报表

选中数据窗口中“对象”栏下的“报表”对象，选择所需预览的报表后，单击工具栏中的“预览”按钮，出现“打印预览”窗口。打印预览与打印结果一致。如果报表记录很多，一页容纳不下，在每页的下面有一个滚动条和页数指示框，可进行翻页操作。

打印报表的最简单方法是直接单击工具栏上的“打印”按钮，出现“打印”窗口，如图8.28所示。在“打印机”选项中选择打印机的型号。然后在“打印范围”选项中指定打印所有页或者确定打印页的范围。在“份数”选项中指定复制的份数和是否需要对其进行分页。最后单击“确定”按钮，就可以进行打印了。

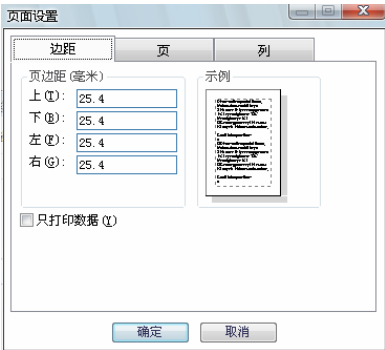


图 8.27 “页面设置”窗口

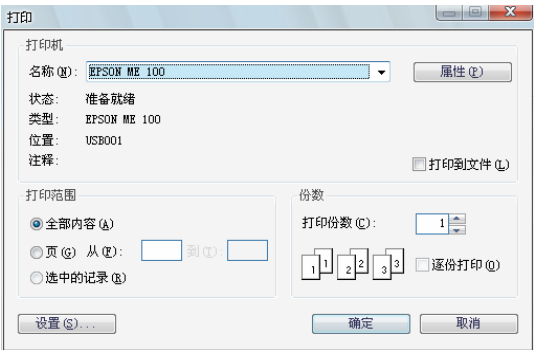


图 8.28 “打印”窗口

习 题 8

一、填空题

1. 利用报表不仅可以创建_____, 而且可以对记录进行分组, 计算各组的汇总数据。
2. 设计报表时, 将各种类型的文本和_____放在报表设计窗体中的各个区域内。
3. 在 Access 中, 创建报表的三种方式: _____、使用向导功能和使用设计视图功能创建。
4. 报表组成区域是: 报表页眉、页面页眉、_____, 报表页脚、页面页脚。
5. 页面页眉中的文字或控件一般输出显示在每页的_____。
6. Access 报表的数据源可以来_____。
7. 在报表的设计视图中, 可以包含标签、文本框、_____。
8. 如果要在某报表结尾处添加整个报表的数量总计, 可以将_____在报表页脚。
9. 要实现报表按某字段分组统计输出, 需要设置该字段组_____。
10. 要在报表每一页的顶部都输出信息, 需要设置_____。

二、选择题

1. 下列说法不正确的是()。
 - A. 可以在单独的报表页眉中输入任何内容
 - B. 为了将标题在每一页都显示出来, 应该将标题放在页面页眉中
 - C. 在实际操作中, 组页眉和组页脚不可以单独设置
 - D. 主体节中可以包含计算的字段数据
2. 下列关于纵栏式报表的描述中, 错误的是()。
 - A. 垂直方式显示
 - B. 可以显示一条或多条记录
 - C. 将记录数据的字段标题信息与字段记录数据一起安排在每页主体节区内显示
 - D. 将记录数据的字段标题信息与字段记录数据一起安排在每页报表页眉节区内显示
3. 用于显示整个报表的计算汇总信息的是()。
 - A. 报表页交节
 - B. 页面页脚节
 - C. 主体节
 - D. 页面页眉节
4. 使用“自动报表”创建的报表只包括()。
 - A. 报表页眉
 - B. 页脚和页面页眉
 - C. 主体区
 - D. 页脚节区
5. 下列叙述中正确的是()。
 - A. 纵栏式报表将记录数据的字段标题信息安排在每页主体节区内显示
 - B. 纵栏式报表将记录数据的字段标题信息安排在页面页眉节区内显示
 - C. 表格式报表其记录数据的字段标题信息被安排在每页主体节区内显示
 - D. 多态性是使该类以统一的方式处理相同数据类型的一种手段
6. 页面页眉的作用是()。
 - A. 用于显示报表的标题、图形或说明性文字
 - B. 用来显示整个报表的汇总说明
 - C. 用来显示报表中的字段名称或对记录的分组名称
 - D. 打印表或查询中的记录数据
7. 只能出现在报表开始处的是()。
 - A. 页面页眉节
 - B. 页面页脚节
 - C. 组页眉节
 - D. 报表页眉节
8. 在 Access 数据库中, 专用于打印的是()。

- A. 表 B. 查询 C. 报表 D. 页
9. 在报表设计过程中, 不适合添加的控制是 ()。
- A. 标签控件 B. 图形控件 C. 文本框控件 D. 选项组控件
10. 下面关于报表对数据的处理中叙述正确的是 ()。
- A. 报表只能输入数据 B. 报表只能输出数据
C. 报表可以输入和输出数据 D. 报表不能输入和输出数据
11. 用于实现报表的分组统计数据操作区间的是 ()。
- A. 报表的主体区域 B. 页面页眉或页面页脚区域
C. 报表页眉或报表页脚区域 D. 组页眉或组页脚区域
12. 为了在报表的每一页底部显示页码号, 应该设置 ()。
- A. 报表页眉 B. 页面页眉 C. 页面页脚 D. 报表页脚
13. 要在报表每一页的顶部都输出信息, 需要设置 ()。
- A. 报表页眉 B. 报表页脚 C. 页面页眉 D. 页面页脚
14. 要在报表上显示格式为“7/总 10 页”的页码, 则计算控件的控件源应设置为 ()。
- A. [Page]/总[Pages] B. =[Page]/总[Pages]
C. [Page]&“/总”&[Pages] D. =[Page]&“/总”&[Pages]
15. 用来查看报表页面数据输出形态的视图是 ()。
- A. “设计”视图 B. “打印预览”视图
C. “报表预览”视图 D. “版面预览”视图
16. 使用 () 创建报表时会提示用户输入相关的数据源、字段和报表版面格式等信息。
- A. “自动报表” B. “报表向导” C. “图标向导” D. “标签向导”
17. 关于报表功能叙述不正确的是 ()。
- A. 可以呈现各种格式的数据 B. 可以包含子报表与图标数据
C. 可以分组组织数据, 进行汇总 D. 可以进行计数、求平均、求和等统计计算
18. 对报表属性中的数据源设置, 下列说法正确的是 ()。
- A. 只能是表对象 B. 只能是查询对象
C. 既可以是表对象也可以是查询对象 D. 以上说法均不正确
19. 报表中的报表页眉是用来 ()。
- A. 显示报表中的字段名称或对记录的分组名称 B. 显示报表的标题、图形或说明性文字
C. 显示本页的汇总说明 D. 显示整份报表的汇总说明

三、简答题

1. 报表的功能是什么?
2. 报表主要有哪三种视图?
3. 常见的报表格式有哪几种?
4. 在报表中最多可按多少个字段或表达式进行分组? 在报表中对记录进行分组的操作步骤是什么?
5. 在报表中最多可按多少个字段或表达式进行排序? 在报表中对记录进行排序的操作步骤是什么?
6. 在报表中如何添加页码?
7. 在报表设计视图中, 报表的结构有几个组成部分?

四、操作题

对学生成绩数据库完成以下两个操作。

(1) 创建学生成绩统计报表。并对报表进行如下操作:

① 在报表中的报表页眉节区添加一个标签控件, 标题显示为“学生成绩表”;

② 在报表页脚节区添加一个计算控件, 计算并显示学生平均成绩。计算控件放置在距上边 0.3 cm、距左边 6.1 cm, 并命名为“Avg_score”。

(2) 创建学生基本信息报表。并对报表进行如下操作:

① 在报表中的报表页眉节区添加一个标签控件, 标题显示为“学生基本信息表”;

② 在报表的主体节区添加一个文本框控件, 显示“年龄”字段值。该控件放置在距上边 0.1 cm、距左边 6.9 cm, 并命名为“Age”;

③ 在报表页脚节区添加一个计算控件, 计算并显示学生平均年龄。计算控件放置在距上边 0.3 cm、距左边 6.1 cm, 并命名为“Avg”。

第 9 章 数据访问页

Web 页也就是我们通常所说的网页，在 Internet 上，很多的信息都是以网页的形式来发布和传播。Access 的一个最突出的功能就是能够将 Access 数据库中的数据动态地提供给 Web 页。

9.1 数据访问页的基本概念

9.1.1 数据访问页

数据访问页是一种特殊类型的网页，它允许用户在 IE 浏览器上查看和使用 Access 数据库、SQL Server 数据库或 MSDE 数据库中存储的数据。与其他 Access 数据库对象不同，数据访问页不保存在 Access 数据库或 Access 工程文件内，而是保存在一个外部的独立文件(.htm)中。Microsoft Access 无法控制数据访问页文件的安全性。

9.1.2 数据访问页视图

数据访问页是以超文本标记语言（HTML）编码的窗体，包括以下两种视图方式。

1. 页视图

页视图是查看所生成的数据访问页样式的一种视图方式，如图9.1所示。

2. 设计视图

对数据访问页进行修改时，需要以设计视图方式打开数据访问页。选中要打开的数据访问页，然后选择“设计”按钮，即可打开该页并切换到设计视图。设计视图是创建与设计数据访问页的一个可视化的集成界面，在该界面下可以修改数据访问页，如图9.2所示。

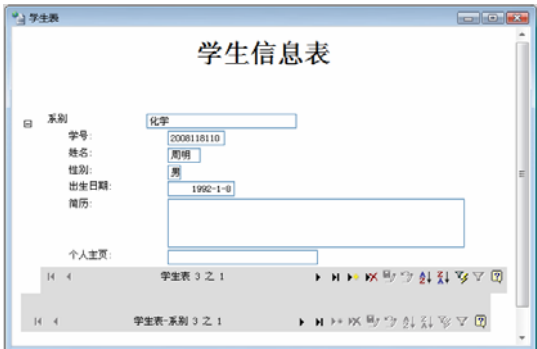


图 9.1 数据访问网页视图



图 9.2 数据访问页设计视图

9.1.3 数据访问页的组成

数据访问页一般由正文、节、组页眉和组页脚、记录导航以及标题 5 个部分组成。

1. 正文

正文是数据访问页的基本设计界面。在支持数据输入的页上，可以用它来显示信息性文本、与数据绑定的控件及节。

2. 节

使用节可以显示文字、数据库中的数据以及工具栏。通常有两种类型的节用在支持数据输入的页上分别是：组页眉和记录导航节。数据访问页中的每个分组级别都有一个记录源。记录源的名称显示在用于分组级别的每一节的节栏上。

3. 组页眉和组页脚

组页眉和组页脚用于显示数据和计算结果值。

4. 记录导航

记录导航控件是数据访问页上用于显示记录导航工具栏的控件。在分组的页中，可以向每个分组级别添加一个导航工具栏，通过更改导航工具栏的属性对其进行自定义。组的记录导航节出现在组页眉节之后，在记录导航节中不能放置。

5. 标题

标题紧挨组页眉并在其前面出现，用于显示文本框和其他控件的标题，在标题节中不能放置绑定控件。

9.2 创建数据访问页

9.2.1 自动创建

自动创建数据访问页是最简单创建方法，操作步骤如下。

① 打开数据库，在“数据库”窗口选择“页”为操作对象后，单击“新建”按钮。

② 系统弹出“新建数据访问页”窗口，选择“自动创建数据页：纵栏式”，然后选择数据源，如图9.3所示。

③ 保存数据访问页，结果如图9.4所示。

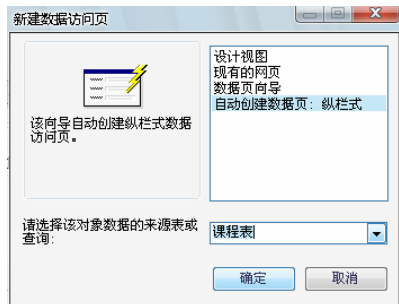


图 9.3 “新建数据访问页”窗口

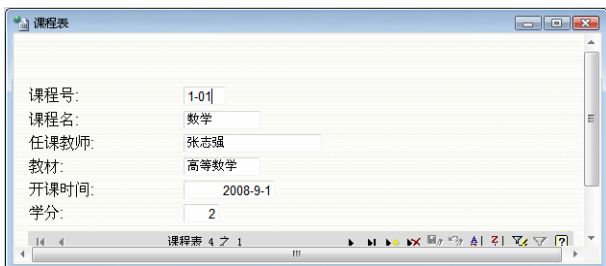


图 9.4 数据访问页页面视图

9.2.2 使用向导创建

- 通过向导也可以很方便的创建数据访问页，步骤如下：
- ① 打开学生成绩管理数据库，在数据库窗口上的选项卡中选择“页”对象，双击“使用向导创建数据访问页”，弹出了“数据页向导”窗口，选择数据源和显示字段，结果如图9.5 所示。
 - ② 确定页中分组级别和排序次序。在创建页的时候必须注意，添加分组级别后的页将是只读页。即在网络浏览器上只能看，不能修改它上面的数据值。而非只读的页在任何时候用户都可以改动数据库中的内容。单击“下一步”按钮。
 - ③ 确定各记录的排序次序。如果不想改变排列顺序，单击“下一步”按钮。
 - ④ 为 Web 页取名。本步为创建访问页的最后一步，主要是输入 Web 页的名字。输入标题“成绩查询”后，单击“完成”按钮。运行结果如图9.6所示。

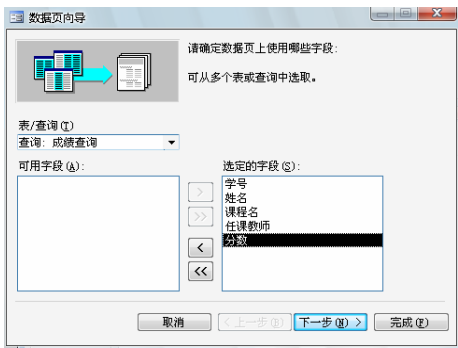


图 9.5 “数据页向导”窗口



图 9.6 运行结果

9.2.3 使用设计器创建

利用前面方法创建的 Web 数据访问页有时需要进行编辑，以修改创建时的不足。对 Web 数据访问页修改时，要在 Access 中的设计视图中进行。

1. 工具箱

当打开 Web 数据访问页的“设计”视图时，系统会自动弹出工具箱，如图9.7所示。



图 9.7 数据访问页设计工具箱

与窗体或报表中的工具箱相比，该工具箱增添了一些新的控件按钮。如滚动文字、Office 数据透视表、Office 图表、Office 电子表格、绑定的动态链接、图像的超级链接、影片等控件。利用这些控件按钮可以在 Web 数据访问页中添加一些在网络中经常见到的装饰内容。例如，我们可以在 Web 数据访问页中利用“Office 数据透视表”控件添加一个数据透视表到 Web 数据访问页中，这样建立的数据透视表具有其相应的 Office 编辑功能。例如可以将表格中的某个字段设置为筛选字段、行字段或列字段，改换数据表的显示等。还可以在 Web 数据访问页中访问数据库的外部文档或数据库内其他已有组件，利用超级链接地址来打开或使用这些文档，使得 Web 数据访问页中所包含的内容更加丰富。

2. 创建过程

操作步骤如下：

① 打开数据库，在“数据库”窗口选择“页”对象，按“新建”按钮，系统弹出“新建数据访问页”窗口，选择“设计视图”，选择数据源，结果如图9.8所示。

② 单击“确定”按钮，系统弹出“数据访问页”设计窗口，选择“数据访问页”中所需字段，如图9.9所示。

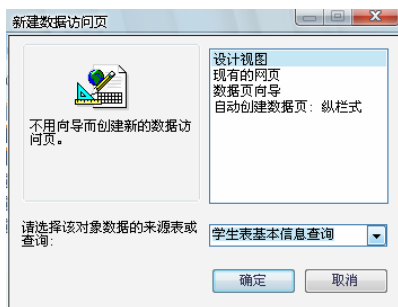


图 9.8 设计视图



图 9.9 “数据访问页”设计窗口

③ 保存数据访问页并运行，结果如图9.10所示。



图 9.10 运行结果

9.3 美化数据访问页

可以在设计视图对数据访问页做进一步的修改美化，页设计视图类似于报表的设计视图，两者操作方法相同。只不过页设计视图中的工具箱里面的工具图标更多一些。

9.3.1 添加标签

如果要创建附加到控件上的标签，只要创建该控件即可，Access 将在创建控件时自动为其附加相应的标签，通过下列方法可以创建单独的标签。

① 在数据访问页“设计”视图中打开数据访问页。

② 单击工具箱中的“标签”按钮。

③ 在数据访问页中，单击要放置标签的位置，然后在标签上输入相应的文本。设置结果如图9.11所示。



图 9.11 添加标签控件

9.3.2 添加命令按钮

可以使用命令按钮向导创建所需的命令按钮。使用向导可以加快创建命令按钮的处理过程，因为向导为用户完成了所有基本的任务。在使用向导时，Access 将提示输入所需的信息，并根据用户的回答来创建命令按钮。

如图9.12所示是“命令按钮向导”对话框，通过这种方式可以创建 30 多种不同类型的命令按钮。例如，可以创建一个命令按钮来查找记录、打印记录或套用窗体筛选。本例中添加四个命令按钮，采用“命令按钮向导”设置命令，如图9.12所示。

删去不需要的导航按钮，最终设置结果如图9.13所示。

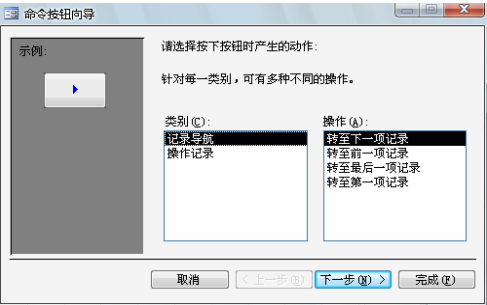


图 9.12 “命令按钮向导”窗口



图 9.13 设置结果

运行结果如图 9.14 所示。

9.3.3 添加滚动文字

1. 创建非绑定滚动文字控件
 - ① 在设计视图中打开数据访问页，单击工具箱中的“滚动文字”工具。
 - ② 拖放控件，直到获取所需的尺寸，然后输入要显示的信息，结果如图9.15所示。
2. 创建绑定滚动文字控件
 - ① 在设计视图中打开数据访问页，单击工具箱中的“滚动文字”控件。
 - ② 在设计视图中添加“滚动文字”控件，如图9.16所示。
 - ③ 鼠标指向控件，单击右键，选择属性，在“属性”窗口中选择绑定字段，如图9.17所示。

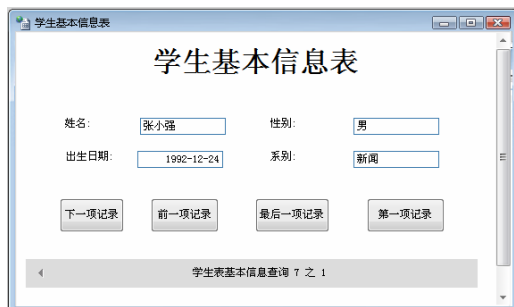


图 9.14 运行结果



图 9.15 设置非绑定动态文字



图 9.16 设置绑定动态文字



图 9.17 设置绑定数据字段

9.3.4 添加主题和背景

主题是项目符号、字体、水平线、背景图像和其他数据访问页元素的设计元素和颜色方案的统一体，使用主题有助于方便地创建专业化的数据访问页。将主题应用于数据访问页时，将会自定义数据访问页中的以下元素：正文和标题样式、背景色彩或图形、表边框颜色、水平线、项目符号、超级链接颜色及控件。

可以使用“格式”菜单中的“主题”命令来实现“首次应用主题”、“应用不同的主题”或“删除主题”。在可用的主题列表中选择主题，然后在“主题示例”框中查看样本数据访问页元素的显示情况，可以在应用主题之前进行预览。在应用主题之前，还可以设置一些选项，例如，给文本和图形应用较亮的颜色，使特定的主题图形变为活动的图形，对数据访问页应用背景等。只有在 Web 浏览器中查看数据访问页时，主题图形才可以是活动的，而在 Access 中主题图形总是静态的。如图9.18就是添加层叠主题后的效果。

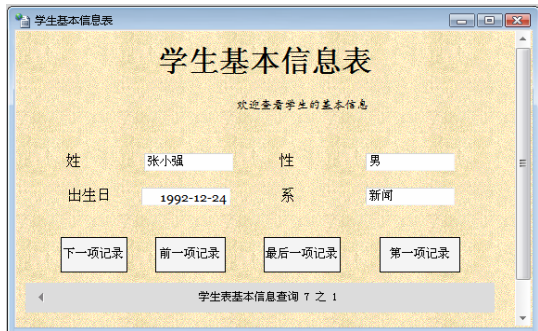


图 9.18 添加层叠主题的效果

9.3.5 添加Office电子表格

Access 数据库中，用户可以在数据访问页中添加 Office 电子表格，用户可以利用数据访问页的页视图或 Internet Explorer 浏览器来查看和分析相关的数据。Office 电子表格类似于 Microsoft Excel 工作表，用户可以在 Office 电子表格中输入原始数据、添加公式及执行电子表格运算。

- ① 在设计视图中打开数据访问页，单击工具箱中“Office 电子表格”工具。在数据访问页中单击控件的左上角将其拖放到适当的位置。
- ② 单击控件内部以激活电子表格。用鼠标右键单击控件内部，然后单击“命令和选项”。在弹出的“命令和选项”对话框中对电子表格的外观、行为、数据源等进行自定义，如图9.19所示。
- ③ 保存数据访问页，结果如图9.20所示。

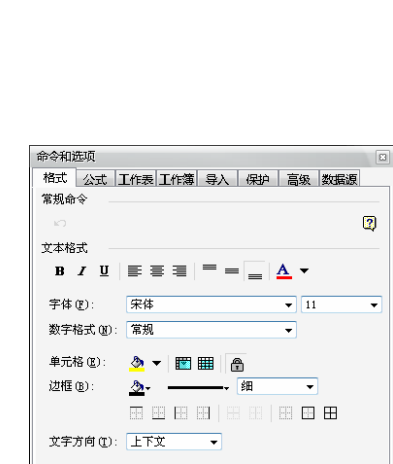


图 9.19 “命令和选项”对话框



图 9.20 添加电子表格

9.3.6 建立超链接

在一个数据页中，除了能够加入标题，进行页面美化外，还可以增加一些对象来增强页面的功能。例如，可以在数据页中加入超链接访问网上的某一个 Web 服务器。在数据页中可以添加 4 种超链接：原有文件或 Web 页、此数据库中的页、新建页和电子邮件地址。

在数据页中加入超链接的步骤如下：

- ① 将数据页切换到设计视图状态下，在工具栏中选择“超链接”按钮。
- ② 在数据页的适当位置通过拖动鼠标建立“超链接”标签。
- ③ 在“插入超链接”对话框中输入相应内容。
- ④ 单击“确定”按钮。

习 题 9

一、填空题

- 1. 数据访问页是用户通过_____进行数据交互的数据库对象。
- 2. 使用向导创建数据访问页时，在确定分组级别步骤中最多可设置_____个分组字段。

3. Access 中若要将数据库中的数据发布到网上, 应采用的对象是_____。
4. 数据访问页是以_____编码的窗体, 有页视图和设计视图两种视图方式。

二、选择题

1. 在 Access 数据访问页中, 命令按钮是一种常用的对象, 下列有关命令按钮对象的叙述不正确的是()。
 - A. 在数据访问页中命令按钮主要用来浏览记录和保存、删除等操作
 - B. 命令按钮控件提供了命令按钮向导, 可以通过向导设置命令按钮的动作、外观等
 - C. 可以在命令按钮上添加图片显示, 从而使其更形象
 - D. 在命令按钮向导中, 可以为命令按钮设置超链接
2. 在 Access 中, 数据访问页中的浏览记录工具栏能够进行多种操作, 下列选项中哪个不是浏览记录工具栏所具有的功能按钮()。
 - A. 保存记录
 - B. 添加记录
 - C. 按窗体筛选
 - D. 以升序排序
3. 在 Access 数据访问页中, 可以用来修改生成的数据访问页的视图方式是()。
 - A. “字段列表”窗口
 - B. “设计”视图
 - C. “属性”窗口
 - D. “页”视图
4. 数据访问页可以简单地认为就是一个()。
 - A. 网页
 - B. 数据库文件
 - C. Word 文件
 - D. 子表
5. Access 通过数据访问页可以发布的数据()。
 - A. 只能是静态数据
 - B. 只能是数据库中保持不变的数据
 - C. 只能是数据库中变化的数据
 - D. 是数据库中保存的数据
6. 数据访问页的视图方式主要有()种。
 - A. 2
 - B. 3
 - C. 4
 - D. 5

三、简答题

1. 数据访问页的功能是什么?
2. 超链接的具体形式有哪些?
3. 如何创建交互式的数据访问页?

第 10 章 宏

宏是能自动执行某种操作的命令，它和菜单操作命令的本质一样，只是它们对数据库施加作用的时间不同、条件不同。菜单命令一般用在数据库的设计过程中，而宏命令则用在数据库的执行过程中。菜单命令必须由使用者来激活，而宏命令则可以在数据库中自动执行。

10.1 理 解 宏

要让控件完成特定的功能，在 Access 中有四种办法：“控件向导”、“宏”、“VBA”和“SQL 语言”。宏虽然只有 53 种基本操作，却可以组合成很多宏组命令，这样就能在 Access 中实现有关窗体、报表、查询的功能，使用起来也非常方便。宏由一些操作和命令组成，其中每个操作可实现特定的功能，命令可辅助完成自动化的操作。例如，可以完成排序、查询、显示窗体、打印报表等各种操作。

10.1.1 宏的分类

宏的优点在于无须编程即可完成对数据库对象的各种操作。在宏中使用的操作与操作系统中的批处理命令非常相似。用户使用宏时，只需给出操作的名称、条件和参数，就可以自动完成指定的操作。宏一般可以分为三类：操作序列、宏组和含有条件的宏。

1. 操作序列

操作序列是结构最简单的一种宏。宏中包含的就是顺序排列的各种操作，每次运行该宏时，Access 都将顺序执行宏中的操作。

2. 宏组

在数据库操作中，如果为了完成一项功能而需要使用多个宏，则可将完成同一项功能的多个宏构成一个宏组，以便对数据库中的宏进行分类管理和分别维护。调用（执行）一个宏组中的宏的格式为：宏组名.宏名。

3. 含有条件的宏

含有条件的宏是指在满足一定条件时才执行宏中的某个或某些操作。条件的设置可以通过逻辑表达式来完成，表达式的真假决定了是否执行宏中的操作。

10.1.2 宏的介绍

1. 和筛选数据有关的宏

筛选数据有关的宏是：ApplyFilter。

使用 ApplyFilter 操作可以对表、窗体或报表应用筛选、查询或 SQL WHERE 子句，以便限制或排序表、窗体、报表和查询中的记录。对于报表，只能在其 OnOpen 事件属性所指定的宏中使用该操作。

2. 和移动记录或页有关的宏

移动记录或页有关的宏包括：FindRecord、FindNext、GoToControl、GoToPage、GotoRecord。

(1) FindRecord 操作：查找符合 FindRecord 参数指定准则的第一个数据。

(2) FindNext 操作：查找下一项记录，该记录满足 FindRecord 操作或“在字段中查找”窗口所指定的准则。

(3) GoToControl：把焦点移到打开的窗体、窗体数据表、表数据表、查询数据表中当前记录的特定字段或控件上。

(4) GoToPage 操作：在活动窗体中将焦点移到某一特定页的第一个控件上。

(5) GoToRecord 操作：使指定的记录成为已打开表、窗体或查询结果集中的当前记录。

3. 和执行有关的宏

执行有关的宏包括：RunCommand、Quit、OpenQuery、RunCode、RunMacro、RunSQL、RunApp、StopAllMacros、StopMacro。

(1) RunCommand 操作：运行 Access 的内置命令。内置命令可以出现在 Access 菜单栏、工具栏或快捷菜单上。

(2) Quit 操作：退出 Access。Quit 操作还可以指定在退出 Access 之前保存数据库对象的几个选项。

(3) OpenQuery 操作：在“数据表”视图、“设计”视图或“打印预览”中打开选择查询或交叉表查询，可以为查询选择数据输入方式。

(4) RunCode 操作：调用 Visual Basic 的 Function 过程。

(5) RunMacro 操作：运行宏，该宏可以在宏组中。

(6) RunSQL 操作：通过使用相应的 SQL 语句，来运行 Access 的操作查询。还可以运行数据定义查询。

(7) RunApp 操作：在 Access 中运行一个 Windows 或 MS-DOS 应用程序，如 Excel、Word 或 PowerPoint。

(8) StopAllMacros 操作：中止当前所有宏的运行。

(9) StopMacro 操作：停止当前正在运行的宏。

4. 和导入导出有关的宏

导入导出有关的宏包括：OutPutTo、SendObject。

(1) OutPutTo 操作：将某特定的 Access 数据库对象（数据表、窗体、报表、模块、数据访问页）中的数据输出到 Excel 2000 文件(*.xls)、MS-DOS 文本文件(*.txt)或多信息文本(*.rtf)格式中。还可将数据输出到 HTML (*.html) 格式文件、Microsoft Internet Information Server (*.htx, *.idc)格式文件或 Microsoft Active Server (*.asp)格式文件中。

(2) SendObject 操作：将指定的 Access 数据表、窗体、报表、模块或数据访问页包含在电子邮件消息中，以便查看和发送。也可将 Excel 2000、MS-DOS 文本文件、多信息文本或 HTML 格式的对象包含在 Microsoft Exchange、Microsoft Mail、Microsoft Windows for Workgroups 邮件或其他使用 Microsoft Mail Applications Programming Interface(MAPI)电子邮件应用程序的邮件消息中。

5. 和数据传递有关的宏

传递数据有关的宏包括：TransferDataBase、TransferSpreadsheet、TransferText。

(1) TransferDatabase 操作: 可以在 Access 数据库或 Access 项目与其他的数据库之间导入与导出数据。对于 Access 数据库还可以从其他的数据库链接表到当前的数据库中, 通过链接表, 其他的数据库中也可以访问其中表的数据。

(2) TransferSpreadsheet 操作: 在当前的 Access 数据库或 Access 项目和电子表格文件之间导入或导出数据。还可将 Excel 电子表格中的数据链接到当前的 Access 数据库中。通过链接的电子表格, 可以用 Access 查看和编辑电子表格中的数据, 同时允许在 Excel 电子表格程序中对数据进行完全访问。还可以链接 Lotus 1-2-3 电子表格文件中的数据, 但这些数据在 Access 中是只读的。

(3) TransferText 操作: 在当前的 Access 数据库或 Access 项目与文本文件之间导入或导出文本。还可以将文本文件中的数据链接到当前的 Access 数据库中。通过链接的文本文件, 在允许字处理程序完全访问该文本文件的同时还可以用 Access 查看该文本数据。也可以导入、导出或链接到 HTML 文件的表或列表中。

6. 和对象处理有关的宏

对象处理有关的宏包括: CopyObject、Rename、Save、DeleteObject、SelectObject。

(1) CopyObject 操作: 将指定的数据库对象复制到不同的 Access 数据库或同一数据库中, 或以新的名称复制到 Access 项目。例如, 可以从另一个数据库中复制或备份一个已有的对象, 或快速地创建一个仅略有不同的相似对象。

(2) Rename 操作: 对一个指定的数据库对象重新命名。

(3) Save 操作: 保存一个特定的 Access 对象。在某些情况下还可以使用新名称保存活动对象。

(4) DeleteObject 操作: 可删除一个特定的数据库对象。

(5) SelectObject 操作: 可选择指定的数据库对象。

7. 和窗体有关的宏

移动或调整窗体大小有关的宏包括: Maximize、Minimize、MoveSize、Restore。

(1) Maximize 操作: 可以放大活动窗口, 使其充满 Access 窗口。该操作可以使用户尽可能多地看到活动窗口中的对象。

(2) Minimize 操作: 可以将活动窗口缩小为 Access 窗口底部的小标题栏。

(3) MoveSize 操作: 可以移动活动窗口或调整其大小。

(4) Restore 操作: 可将处于最大化或最小化的窗口恢复为原来的大小。

8. 和打开或关闭对象有关的宏

打开或关闭对象有关的宏包括: Close、OpenForm、OpenModule、OpenQuery、OpenReport、OpenTable、OpenDataAccess Page、OpenDiagram、OpenStoredProcedure、OpenView。

(1) Close 操作: 关闭指定的 Access 窗口。如果没有指定窗口, 则关闭活动窗口。

(2) OpenForm 操作: 从“窗体”视图、窗体“设计”视图、“打印预览”或“数据表”视图中打开一个窗体。

(3) OpenModule 操作: 在指定的过程中打开特定的 Visual Basic 模块。该过程可以是 Sub 过程、Function 过程或事件过程。

(4) OpenQuery 操作: 在“数据表”视图、“设计”视图或“打印预览”中打开选择查询或交叉表查询。该操作将运行一个操作查询。可以为查询选择数据输入方式。

(5) OpenReport 操作: 在“设计”视图或“打印预览”中打开报表或立即打印报表。

(6) OpenTable 操作: 在“数据表”视图、“设计”视图或“打印预览”中打开表。

(7) **OpenDataAccess Page** 操作：在“页”视图或“设计”视图中打开数据访问页。

(8) **OpenDiagram** 操作：在“设计”视图中打开数据库图表。

(9) **OpenStoredProcedure** 操作：在“数据表”视图、“设计”视图或“打印预览”中打开存储过程。

(10) **OpenView** 操作：在“数据表”视图、“设计”视图或“打印预览”中打开视图。当在“数据表”视图中打开视图时，该操作运行指定的视图。可以选择该视图的数据项，并可限制视图显示的记录数。

9. 和打印对象有关的宏

打印对象有关的宏包括：**PrintOut**。

PrintOut 操作：打印打开数据库中的活动对象，也可以打印数据表、报表、窗体和数据访问页和模块。

10. 和设置字段控件属性有关的宏

设置字段控件属性有关的宏包括：**SetValue**、**RepaintObject**、**Requery**、**ShowAllRecords**。

(1) **SetValue** 操作：对 Microsoft Access 窗体、窗体数据表或报表上的字段、控件或属性的值进行设置。

(2) **RepaintObject** 操作：完成指定数据库对象挂起的屏幕更新。如果没有指定数据库对象，则对活动数据库对象进行更新。

(3) **Requery** 操作：通过重新查询控件的数据源来更新活动对象中特定控件的数据。如果不指定控件，该操作将对对象本身的数据源进行重新查询。

(4) **ShowAllRecords** 操作：删除活动表、查询的结果集合和窗体中所有已应用过的筛选，并且显示表或结果集合及窗体的基本表或查询中的所有记录。

11. 其他的宏

其他宏包括：**AddMenu**、**SetMenuItem**、**Echo**、**Hourglass**、**MsgBox**、**SetWarnings**、**SendKeys**、**ShowToolbar**、**Beep**。

(1) **AddMenu** 操作：创建窗体或报表的自定义菜单栏，窗体、窗体控件或报表的自定义快捷菜单、全局菜单栏、全局快捷菜单。

(2) **SetMenuItem** 操作：设置活动窗口的自定义菜单栏或全局菜单栏上的菜单项状态（可用或不可用，选取或不选取）。

(3) **Echo** 操作：指定是否打开回响。例如，可以使用该操作隐藏或显示宏运行时的结果。

(4) **Hourglass** 操作：使鼠标指针在执行宏时变成沙漏形状（或其他所选择的图标）。当操作或宏本身需要花很长时间执行时，该功能特别有用。

(5) **MsgBox** 操作：显示包含警告信息或其他信息的信息框。例如，可以在有效性验证的宏中使用 **MsgBox** 操作。当控件或记录不符合宏中的验证条件时，消息框将显示错误信息，并提示应该输入的正确数据。

(6) **SetWarnings** 操作：可以打开或关闭系统信息。

(7) **SendKeys** 操作：把按键直接传送到 Access 或其他 Windows 应用程序。

(8) **ShowToolbar** 操作：显示或隐藏内置工具栏或自定义工具栏。它可以在所有的 Access 窗口中显示内置工具栏或只在显示某一工具栏的视图上显示内置工具栏。

(9) **Beep** 操作：发出嘟声。



10.2 宏创建与运行

10.2.1 创建宏

1. 宏设计窗口

将鼠标移动到数据库窗口中的宏对象上，单击左键，即可切换到“宏”对象上。单击数据库窗口的“新建”菜单，屏幕上弹出“宏设计”窗口，如图10.1所示。

通过这个窗口中就可以进行宏设计。在宏的设计窗口中，“操作”列用于指明宏所能执行的各种操作。在 Access 中，一共有 53 种基本操作，这些操作可以通过单击这列中任何一个值后弹出的下拉选单来选择，而“注释”列则用于对操作进行解释、备注。

在宏设计器中，“宏名”和“条件”字段是可选的，现在将鼠标移动到工具栏上，单击 （宏名）按钮，就会发现在宏设计器上出现“宏名”列，单击工具栏上的 （条件）按钮，在设计器上就会出现“条件”列。此时的宏的设计窗口如图10.2所示。

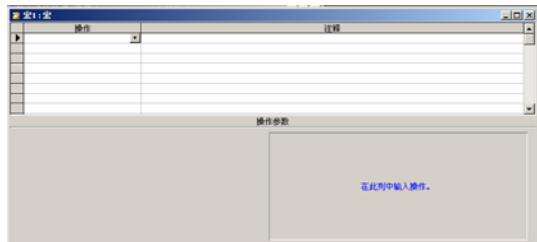


图 10.1 宏设计

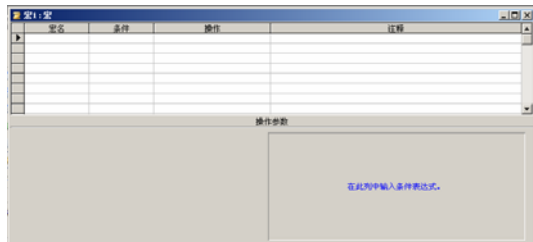


图 10.2 选择宏名和条件的宏设计窗口

“宏名”用来定义一个或一组宏操作的名字，以后要执行这个宏操作时，只要直接运行就可以了。如果想对宏执行进行一些限制，就需要在条件列中输入相应的条件表达式，这样在宏执行之前先判断条件是否满足，如果满足则执行，否则不执行。在操作列中，提供了 50 多种操作，用户可以从这些操作中做选择，创建自己的宏。选定操作后，在“操作参数”区域会出现相应的操作参数，可以在各操作参数对应的文本框中输入数值，以设定操作参数的属性，也可以使用表达式生成器生成的表达式设置操作参数。如果有多个宏，可将相关的宏设置成宏组，使用宏组可以避免单独管理这些宏的麻烦，在数据库窗口中的宏名称列表中将显示宏组名称。如果要指定宏组中的某个宏，格式为：宏组名.宏名。

2. 在宏中添加操作

创建宏所需要的基本操作都由系统提供，用户只需对其中的一些属性进行设置即可。

① 打开“学生成绩管理”数据库，单击数据库窗口选项卡上的“宏”对象，然后单击数据库窗口上的“新建”选项，弹出“宏设计”窗口。

② 选择操作命令。将鼠标移动到宏的操作列，单击下拉框右面的“向下”按钮，弹出下拉菜单，在菜单上选择所需的操作命令。例中选择“MsgBox”宏命令。

③ 输入操作参数。在选择了基本宏操作后，还需要在宏设计窗口下部的操作参数中输入此操作需要的参数。例中输入“欢迎查看学生信息”，如图10.3所示。

基本上每个操作都有操作参数，有的多一些，有的少一些，有的是必须填写的，有的则可以取默认的值。只有设置好参数的宏操作才是合法的。

④ 增加操作。现在需要添加一个操作打开学生表窗体。将鼠标移动到表格的第二行，然后在“操作”字段选取“Openform”，并设置参数，如图10.4所示。

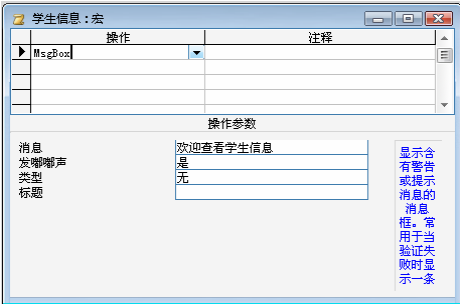


图 10.3 添加操作

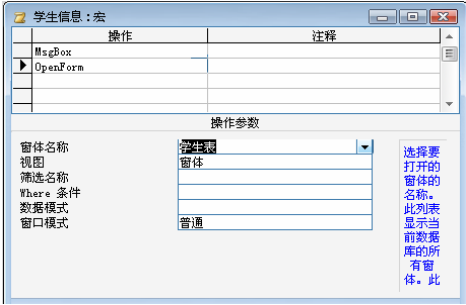


图 10.4 添加新操作

⑤ 单击常用工具栏上的“保存”按钮，在弹出的“另存为”对话框中输入“学生信息”。
双击“学生信息”宏，系统首先运行“MsgBox”宏，如图10.5所示。单击“确定”按钮，系统运行“Openform”宏，结果如图10.6所示。

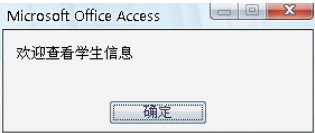


图 10.5 消息框



图 10.6 打开表

3. 建立宏组

建立宏组的操作步骤如下。

① 打开“学生成绩管理”数据库，选择数据库窗口选项卡上的“宏”对象，然后单击数据库窗口上的“新建”选项，弹出“宏设计”窗口。单击“视图”菜单上“宏名”命令，窗口自动增加“宏名”列。在窗口上半部分和下半部分，分别输入相应内容，并选择相应的宏操作命令。结果如图10.7所示。

② 单击常用工具栏上的“保存”按钮，在弹出“另存为”对话框中输入“信息浏览”。
宏组不能直接运行，需要通过窗体命令按钮事件来驱动，即单击窗体上的命令按钮来执行宏。

③ 通过设计视图创建一个窗体，添加三个命令按钮，如图10.8所示。

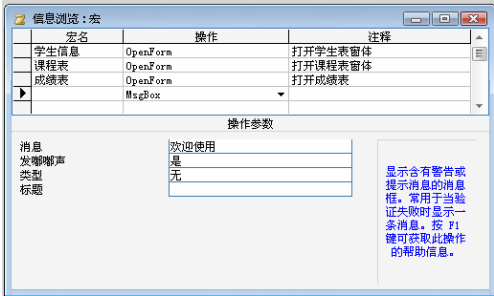


图 10.7 定义宏组



图 10.8 创建窗体

并分别设置属性的单击事件，添加对应宏组：浏览学生信息的“事件”选项的“单击”属性选择“信息浏览.学生信息”；浏览课程信息的“事件”选项的“单击”属性选择“信息浏览.课程表”；浏览成绩信息的“事件”选项的“单击”属性选择“信息浏览.成绩表”。

④ 运行窗体，分别单击命令按钮来执行对应的宏。

4. 创建AutoExec宏

如果在首次打开数据库时执行指定的操作，可以创建一个名为 AutoExec 的特殊宏。该宏可在首次打开数据库时执行一个或一系列的操作。打开数据库时，Access 将查找名为 AutoExec 的宏，如果找到，就自动运行它。创建 AutoExec 宏的方法如下：

- ① 创建一个宏，其中包含在打开数据库时要运行的操作。
- ② 以 AutoExec 为宏名保存该宏。

下次打开数据库时，Access 将自动运行该宏。如果不想在打开数据库时运行 AutoExec 宏，可在打开数据库时按住 Shift 键。

例如，建立一个 AutoExec 宏，打开学生管理系统数据库时，出现一个欢迎消息框，然后运行“学生表”窗体。操作步骤如下：

- ① 在数据库窗口中，单击对象列表中的“宏”对象，然后单击数据库窗口工具栏上的“新建”按钮。
- ② 在操作列表中选择 MsgBox，在消息文本框中输入“欢迎查看学生基本信息”。
- ③ 在操作列表中选择 OpenForm，在窗体名称中选择“学生表”窗体。
- ④ 以 AutoExec 为宏名保存该宏，下次打开数据库时，Access 将首先运行该宏，弹出一个消息框。
- ⑤ 在消息框上按确定后将进入“学生表”窗体。

5. 创建AutoKeys宏组

如果要将一个操作或操作集合赋值给某个特定的按键，可以创建一个 AutoKeys 宏组，在按下特定的按键或组合键时，Access 就会执行相应的操作。命名 AutoKeys 宏时，使用符号“^”表示 Ctrl 键。表 10.1 列出了可用来运行 AutoKeys 宏组的组合键的类型。

表 10.1 组合键的类型

语 法	说 明	示 例
^number	Ctrl+任一数字	^3
{F*}	任一功能键	{F5}
^[F*]	Ctrl+任一功能键	^[F5]
+{F*}	Shift+任一功能键	+{F5}

创建 AutoKeys 宏组的方法与 AutoExec 差不多，只是以 AutoKeys 为名保存宏组。Autokeys 宏通过按下指定给宏的一个键或一个键序触发。例如，名为 F5 的宏将在按下 F5 键时运行。

创建 AutoKeys 宏时，必须定义宏将执行的操作，如打开一个对象，最大化一个窗口或显示一条消息。另外还需要提供操作参数，宏在运行时需要这种参数，例如，要打开的数据库对象、要最大化的窗口或要在对话框中显示的消息的名称等。

例如，建立一个 AutoKeys 宏组，当按下 Ctrl+l 组合键时，执行打开学生表操作；当按下 F8 键时执行“学生成绩”查询。操作步骤如下：

- ① 在数据库窗口中，单击对象列表中的“宏”对象，然后单击数据库窗口工具栏上的“新建”按钮。

② 单击工具栏上的“宏名”按钮，在“宏名”列中输入“^1”组合键。在操作列表中选择 OpenTable，在“表名称”参数中选择“学生表”。

③ 在下一行的“宏名”中输入“{F8}”，在操作列表中 选择 OpenQuery，在“查询名称”参数中选择“学生成绩”查询。

④ 以 AutoKeys 为名称保存宏组。
只要学生管理系统数据库是打开的，在任何情况下按下 Ctrl+I 组合键时，执行打开“学生表”操作，按下 F8 时执行“学生成绩”查询。

6. 创建条件宏

条件宏是满足一定条件后才运行的宏。
创建一个将包含条件的宏组，用来实现对口令的检验，如果口令正确，就先关闭身份核对窗口，再打开另一个窗体；如果口令不正确，将出现信息框要求重新输入口令。

① 建立窗体，添加两个文本框，一个命名为“管理员账户名”，用于输入用户名，另一个命名为“管理员口令”，用于输入口令。再建立一个“确定”按钮和一个“取消”按钮，如图10.9所示。

② 创建宏，命名为“用户登录”，在宏中创建两个宏组“确定”和“取消”，如图10.10所示。



图 10.9 系统登录窗体

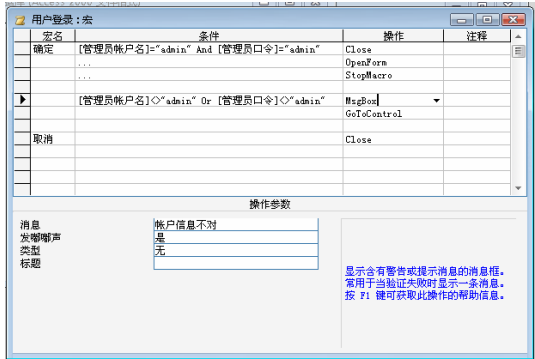


图 10.10 创建宏

“条件”栏内的省略号“...”，表示条件为真时可以连续执行这些操作。其中：
条件 “[管理员账户名]=“admin” And [管理员口令]=“admin”” 成立时执行宏 “close” 操作。

条件 “[管理员账户名] <> “admin” or [管理员口令] <> “admin” ” 成立时执行宏 “MsgBox” 操作。

③ 为窗体上“确定”和“取消”按钮设置事件属性，分别为单击时执行“用户登录.确定”宏、“用户登录.取消”宏。

④ 打开窗体“系统登录”，当输入的口令不正确时，单击确定按钮将出现一个“账户信息不对”的消息提示。如图10.11所示。



图 10.11 消息提示

在这个例子中，管理员账户名和口令被定义为“admin”，这在实际的应用中是不可行的，因为口令会因为需要而随时更改。为了实现这样的功能，可以先建一个用户表，里面保存用户的口令，在宏中通过将用户输入的口令和用户表中的口令比较。如果要更改口令，更改表的数据就行了。

下面，创建一个具有查询功能的宏。通过在窗口中输入待查询的人的姓名，然后通过命令按钮激活相应的宏实现查找。过程如下。

- ① 打开“学生成绩管理”数据库，单击数据库窗口选项卡上的“宏”对象，然后单击数据库窗口上的“新建”选项，弹出“宏设计”窗口。
- ② 确定宏名。例中给宏组取名为“按姓名查询”。
- ③ 选择操作命令。将鼠标移动到宏的操作列，单击下拉框右面的“向下”按钮，弹出下拉菜单，在菜单上选择所需的操作命令。例中选择“GoToControl”宏命令。
- ④ 输入操作参数。在选择了基本宏操作后，还需要在宏设计窗口下部的操作参数“控件名称”中输入此操作需要的参数。例中输入“姓名”，这样当执行这个宏的时候，窗口上的焦点就会移动到“姓名”这个字段上来，如图10.12所示。
- ⑤ 增加操作。增加一个用于对文本框内容进行有效性判断的宏 MsgBox，如图10.13所示。

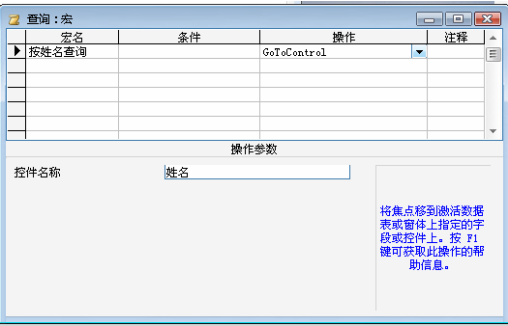


图 10.12 添加 GoTocontrol 宏

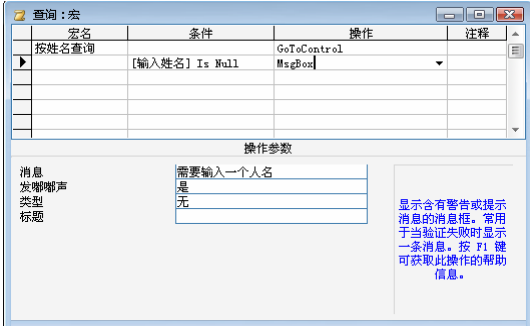


图 10.13 添加 MsgBox 宏

通过添加一个操作来找到“姓名”这个字段中的相应记录，首先将鼠标移动到表格的第三行，然后在“操作”字段中选取“FindRecord”。在参数“查找内容”这一栏中，输入“=[输入姓名].value”，这个“输入姓名”是在窗体上创建的专门用来输入查询值的文本框的名字，而“匹配”字段中选择“整个字段”，其他的几个参数都取默认值，如图10.14所示。

⑥ 保存宏，取名为“查询”。现在创建用于激活宏的“宏”触发器，触发器通常是由窗体、页及其上面的控件的各种事件来担任的。比如在窗体上单击一个按钮，这个单击过程就可以触发一个宏的操作。

下面通过“学生”窗体上的按钮单击事件来触发刚才新建的“查询”宏。

- ① 打开“学生表”窗体，切换到设计视图模式。在窗体上添加一个文本框（用这个文本框来输入要查询的员工姓名），同时将这个文本框的名字改为“输入姓名”（和刚才创建的条件“=[输入姓名].value”对应）。
- ② 在窗体上添加一个按钮，并将该按钮的名字改为“查找”，结果如图10.15所示。
- ③ 设置宏的触发事件。首先选中“查找”按钮，单击右键选择属性。弹出“属性”窗口，



图 10.14 添加 FindRecord 宏

在属性选项卡上选择“事件”选项，然后将鼠标移动到“单击”选项右边的方框内，单击鼠标左键，在这个方框右侧会出现一个“向下”按钮，单击该按钮，在弹出的下拉菜单中单击“查询.按姓名查询”宏。

④ 保存窗体。将视图切换到窗体的数据表视图，如图10.16所示。以后当这个窗体以数据表视图出现的时候，在文本框中输入要查找的人名，单击查询按钮就可以执行刚才建立的宏。

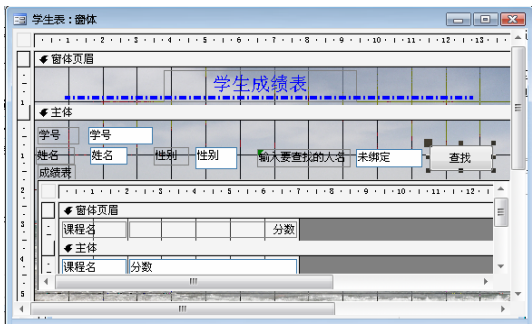


图 10.15 添加控件



图 10.16 运行结果

若没有在文本框中输入文字，而单击“查询”按钮，会出现出错提示，如图 10.17 所示。在文本框内输入要查找人的姓名，单击“查询”按钮，便可启动宏，完成对应的操作，结果如图10.18所示。

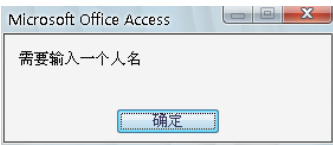


图 10.17 出错提示



图 10.18 运行查找宏

10.2.2 运行宏

创建完一个宏后，就可以运行宏执行各个操作。当运行宏时，Access 会运行宏中的所有操作，直到宏结束。可以直接运行宏，或者从宏组或事件过程中运行宏，也可以作为窗体、报表或控件中出现的事件响应运行宏，还可以创建自定义菜单命令或工具栏按钮来运行宏，也可以将某个宏设定为组合键，或者在打开数据库时自动运行宏。

1. 直接运行宏

如果希望直接运行宏，通过双击宏名、通过菜单或者通过单击工具栏上的“执行”按钮等操作，可以直接运行宏。

2. 从其他宏或VB程序中运行宏

如果要从其他的宏或 VB 过程中运行宏，需要将 RunMacro 操作添加到相应的宏或过程中。将 RunMacro 操作添加到宏中的方法是：在宏的设计视图中，在空白操作行选择 RunMacro 选项，并且将 MacroName 参数设置为相应的宏名即可。RunMacro 操作的参数如表 10.2 所示。

表 10.2 RunMacro 操作的参数

操作参数	描 述
宏名	执行的宏的名称
重复次数	宏执行的最大次数。空白为一次
重复表达式	表达式结果为 True(-1)或 False(0)。如果为假，则宏停止运行

在下列三种情况下使用 RunMacro 操作：

- ◆ 从另一个宏运行宏；
- ◆ 执行基于某个条件的宏；
- ◆ 将宏附加到一个自定义的菜单命令上。

如果用户在“宏名”参数中设置宏组名，则会运行组中第一个宏。

如果要将 RunMacro 操作添加到 VB 过程中，在过程中添加 DoCmd 对象的 RunMacro 方法，然后指定要运行的宏名即可。例如：DoCmd.RunMacro “My Macro”。

3. 从控件中运行宏

如果希望从窗体、报表或控件中运行宏，只需在设计视图中创建相应控件，在控件的“属性”窗口中选择“事件”选项卡，然后在下拉列表框中选择当前数据库中的相应的宏。这样在事件发生时，就会自动执行所设定的宏。

例如，建立一个宏，执行操作“Quit”将某一窗体中的命令按钮的单击事件设置为执行这个宏，则当在窗体中单击按钮时，将退出 Access。

10.3 宏向Visual Basic代码转换

在 Access 中提供了将宏转换为等价的 VBA 事件过程或模块的功能。转换操作分为两种情况：①转换窗体或报表中的宏；②转换不属于任何窗体与报表的全局宏（即转换全局宏）。

10.3.1 转换窗体或报表中的宏

转换窗体或报表中的宏操作步骤如下：

- ① 在设计视图中打开窗体或报表。
- ② 在“工具”菜单中选择“宏”，单击“将窗体的宏转换为 Visual Basic 代码”或“将报表的宏转换为 Visual Basic 代码”。
- ③ 单击对话框的“转换”按钮，再单击“确定”按钮即可。

10.3.2 转换全局宏

转换全局宏操作步骤如下：

- ① 在“数据库”窗口中打开宏对象，选择要转换的宏。
- ② 在“文件”菜单中选择“另存为”命令，在对话框的“保存类型”框中选择“模块”，再单击“确定”按钮。
- ③ 单击“转换”对话框中的“转换”按钮，再单击“确定”按钮即可。

习 题 10

一、填空题

1. 宏是一种特定的编码，是一个或多个_____的集合。

2. 宏的使用一般是通过窗体、报表中的_____控件实现的。
3. 在一个查询集中, 要将指定的记录设置为当前记录, 应该使用的宏操作命令是_____。
4. 宏是 Access 的一个对象, 其主要功能是_____。
5. 在宏中添加了某个操作以后, 可以在宏设计窗体的下部设置这个操作的_____。
6. 有多个操作构成的宏, 执行时是按_____依次执行。
7. 打开查询的宏命令是_____。

二、选择题

1. 下列关于宏的说法中, 错误的是()。
 - A. 宏是多个操作的集合
 - B. 每一个宏操作都有相同的宏操作参数
 - C. 宏操作不能自定义
 - D. 宏通常与窗体、报表中命令按钮相结合来使用
2. 宏中的每个操作都有名称, 用户()。
 - A. 能够更改操作名
 - B. 不能更改操作名
 - C. 能对有些宏名进行更改
 - D. 能够调用外部命令更改操作名
3. 关于宏叙述错误的是()。
 - A. 宏是 Access 的一个对象
 - B. 宏的主要功能是使操作自动进行
 - C. 使用宏可以完成许多繁杂的人工操作
 - D. 只用熟悉掌握各种语法、函数, 才能编写出功能强大的宏命令
4. 怎样才能产生宏操作()。
 - A. 创建宏
 - B. 编辑宏
 - C. 运行宏
 - D. 创建宏组
5. 用于使计算机发出“嘟嘟”声的宏命令是()。
 - A. Echo
 - B. MsgBox
 - C. Beep
 - D. Restore
6. 用于退出 Access 的宏命令是()。
 - A. Creat
 - B. Quit
 - C. Ctrl+Alt+Del
 - D. Close
7. 用于从其他数据库导入和导出数据宏命令是()。
 - A. TransferDatabase
 - B. Transfer Text
 - C. Transfer Test
 - D. Transfer Txt
8. 宏命令 OpenTable 打开数据表, 则可以显示该表的视图是()。
 - A. “数据表”视图
 - B. “设计”视图
 - C. “打印预览”视图
 - D. 以上都是
9. 从宏设计窗体中运行宏, 应单击工具栏上的()。
 - A. Ctrl+空格键
 - B. Ctrl+Break 键
 - C. Alt + Ctrl 键
 - D. Pause 键
10. 在 Access 系统中, 宏是按()调用的。
 - A. 名称
 - B. 标识符
 - C. 编码
 - D. 关键字
11. 如果不指定对象, Close 将会()。
 - A. 关闭正在使用的表
 - B. 关闭正在使用的数据库
 - C. 关闭当前窗体
 - D. 关闭相关的使用对象(窗体、查询、宏)
12. 如果要在宏执行过程中暂停它的执行, 可以按组合键()。
 - A. Ctrl+Break
 - B. Ctrl+Alt
 - C. Alt+Break
 - D. Alt+Shift
13. 下列操作中, 适合使用宏的是()。
 - A. 修改数据表结构
 - B. 创建自定义过程

- C. 打开或关闭报表对象
- D. 处理报表中错误
14. 下列叙述中，错误的是（ ）。
- A. 宏能够一次完成多个操作
- B. 可以将多个宏组成一个宏组
- C. 可以用编程的方法来实现宏
- D. 宏命令一般由动作名和操作参数组成
15. 下列关于宏操作的叙述错误的是（ ）。
- A. 可以使用宏组来管理相关的一系列宏
- B. 使用宏可以启动其他应用程序
- C. 所有宏操作都可以转化为相应的模块代码
- D. 宏的关系表达式中不能应用窗体或报表的控制值
16. 用于最大化激活窗口的宏命令是（ ）。
- A. Minimize
- B. Requery
- C. Maximize
- D. Restore
17. 在宏的表达式中要引用报表 exam 上控件 Name 的值，可以使用引用式（ ）。
- A. Reports! Name
- B. Reports! exam! Name
- C. exam! Name
- D. Reports exam Name
18. 在模块中执行宏 macro1 的格式为是（ ）。
- A. Function.RunMacro MacroName
- B. DoCmd.RunMacro macro1
- C. Sub.RunMacro macro1
- D. RunMacro macro1

三、简答题

1. 宏和宏组的主要功能是什么？
2. Access 中常用的操作数据库对象的宏操作有哪些？
3. Access 中常用的操作数据的宏操作有哪些？
4. 宏的执行方式有哪些？

四、操作题

对学生成绩数据库进行如下操作。

- (1) 创建如图 10.19 所示的窗体，为每个按钮设置宏，通过宏打开对应的表。
- (2) 在打开的学生窗体上，添加文本框控件和命令按钮控件，如图 10.20 所示。为每个按钮设置宏，通过宏可以实现查找。



图 10.19 窗体界面

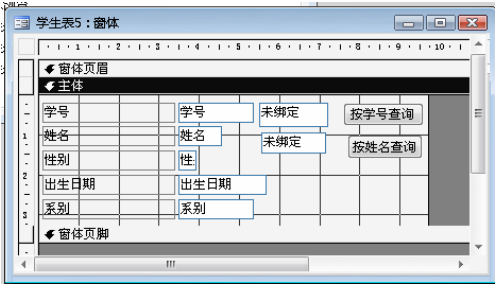


图 10.20 学生表窗体

第 11 章 VBA和模块

宏虽然很好用，但它的运行速度比较慢，不够灵活，而且不能满足用户的一些特殊需求。因此，在设计特殊功能时，需要使用 VBA 语言编写程序，然后将这些程序编译成拥有特定功能的“模块”，在 Access 中调用。

11.1 VBA简介

VB（Visual Basic）是微软公司推出的可视化Basic语言，用它来编程非常简单。微软公司将它的一部分代码结合到Office中，形成今天所说的VBA（Visual Basic for Applications）。VBA的很多语法源自VB，所以可以像编写VB程序那样来编写VBA程序。当程序编译通过以后，将其保存在Access的一个模块里，并通过类似在窗体中激发宏的方式来启动这个模块，从而实现相应的功能。

11.1.1 宏与VBA的差异

宏是多个操作的组合，这些操作是由 Access 定义的，用户无法自定义；VBA 则是 Basic 形式的程序语言，保存在各个模块（包括窗体及报表）内。就某方面而言，宏和模块可达到相同功能。但 VBA 的灵活度远高于宏，即大部分的 VBA 应用都无法用宏来实现。所以要完全发挥 Access 数据库的功能，需使用 VBA。

11.1.2 VBA与VB的区别

VB 是单独包装及执行的程序语言，目前最新版本为.NET，VBA 则是内含于 Office 各软件的语言，二者异同可从以下几个方面比较。

1. 编译执行文件

执行文件的扩展名为.exe。VB 由于内含编译器，故可形成 exe 文件。VBA 由于内含于 Office 系列各软件内，且不提供编译器，故 VBA 程序只可依附于各软件而执行，无法形成可执行文件。

2. 可用的资源

可用的资源也就是程序内可引用的资源，包括对象、函数等。VB 在此方面的资源较 VBA 丰富，VB 是较专业的程序设计语言，而 VBA 的目的则是强化 Office 应用系统，故在可用资源方面 VBA 不及 VB。

3. 基本语法

VBA 和 VB 语法完全相同，只要有 Basic 基础，就可使用 VBA。所以，VBA 可以说是“内置于各软件的 Basic 语法”，提供了很多 VB 中没有的函数和对象，这些函数、对象都是针对应用软件的。

11.2 VBA开发环境

编写程序需要开发环境，开发 VBA 程序也需要开发环境，这个开发环境称为“VBA 开发环境”。

11.2.1 开发环境界面

打开一个数据库，选定数据库窗口上的“模块”选项，鼠标单击数据库窗口上的“新建”按钮，系统弹出一个窗口，这就是 VBA 开发环境，如图11.1所示。

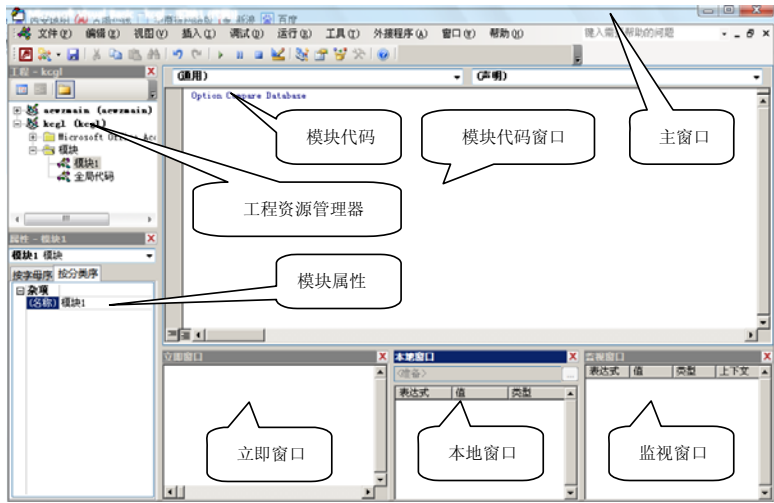


图 11.1 VBA 开发环境

VBA 开发环境分为“主窗口”、“模块代码窗口”、“工程资源管理器”和“模块属性”等几部分。“模块代码窗口”用来输入模块内部的程序代码；“工程资源管理器”用来显示这个数据库中所有的模块。用鼠标单击这个窗口内的一个模块选项时，在“模块代码窗口”中就会显示该模块的程序代码；“模块属性”窗口显示当前选定的模块属性。

在 VBA 开发环境中“本地窗口”、“立即窗口”和“监视窗口”专门用来调试程序，三个窗口的出现和隐藏可以通过“视图”菜单设置。

11.2.2 编写代码

所有 VBA 代码都以模块的方式保存在数据库里。创建模块有两种方法：作为模块对象或作为窗体或报表对象的一部分。

- 1. 模块对象
单击数据库窗口的“模块”按钮可查看模块对象。要创建一个新模块，单击“新建”按钮便可，模块中定义的公共过程可以从应用程序的任何地方调用。
- 2. 窗体和报表模块
在窗体或报表类模块中，可以创建特别命名的事件过程来响应事件。可以通过在设计视图中打开窗体或报表，从“视图”菜单中选择“代码”来编辑该窗体或报表的模块。

11.2.3 模块的调试

程序编写过程中，可能遇到各种问题导致结果不能满足要求，例如，语法错误，算法错误等，因此需要进行程序的调试，也就是说，需要快速找到程序中的问题，消除代码中的错误。模块的调式有多种方法。

1. 设置断点

设置断点就是在代码编辑器窗口中开始调试的位置，点击代码左边的边界标识条（边界标识条就是代码左边的一个宽条。如果没有这个代码标识条，可以点击“工具”→“选项”→“编辑器格式”，将边框标识条勾选上，点击“确定”按钮），默认情况下出现一个咖啡色的圆点，这就是断点，同时这行代码的背景色也变为咖啡色。再运行窗体，点击相应的事件按钮，程序运行到断点时会暂停，可以按“F8”键单步执行。

2. Stop法

另一种方法是在要开始调试的位置前加一句代码 stop，程序运行到这里就会暂停，然后再按“F8”键继续单步执行。

3. 单步执行

还有一种只对模块中的代码有效的方法，就是将光标停在模块内的任何位置，然后按“F8”键就开始单步运行。

以上不管是哪种方法，只要按“F5”键，程序就会继续连续执行。

11.3 VBA基础

11.3.1 VB语言元素

VB 的语言基础是 Basic 语言，VB 程序的语言元素主要包括关键字、函数、表达式、语句等。

1. 关键字

关键字就是程序设计语言里事先定义的，指明特定含义的标识符，也称系统保留字，VB 系统定义的关键字见表 11.1。

表 11.1 VB 系统定义的关键字

关键字	含义	关键字	含义	关键字	含义	关键字	含义
Const	常数	Dim	定义	As	为	Mod	取模
And	并且	Or	或者	Not	非	If	如果
Then	那么	Else	否则	Stop	停止	End	结束
Select	选择	Case	条件	Is	是	For	计数
To	到	Step	步长	Byref	传址	Byval	传值
Sub	子程序	Function	函数	Exit	退出	Do	做
Loop	循环	Until	直到	While	当	Wend	当结束
Let	让	Call	调用	Rem	注释	Integer	整数
Long	长整数	Single	单精度	Double	双精度	Boolean	布尔
String	字符串	Me	我	Private	私有	Public	公共

注意：在 VB 中标识符与关键字不区分字母的大小写；而在 C 语言里，区分大小写，关键字都是小写字母。

2. 函数

函数是一个能完成特定功能的代码段。VB 提供许多系统函数，帮助程序设计者完成特定的任务。这些函数按功能可分为：交互式函数、类型转换函数、数学函数、日期和时间函数、字符串函数、目录和文件函数等，对于系统函数，使用者不必了解函数的内部实现过程，只要知道正确的使用格式即可。

① 交互式函数

用来和用户进行交互的函数，经常使用的有 MsgBox 函数、InputBox 函数。

② 类型转换函数

当要对不同类型的变量进行赋值操作或者不同类型的数据参与运算时，就要进行类型转换，常见的类型转换函数见表 11.2。

表 11.2 常见的类型转换函数

函 数	转换后的 x 类型或返回值	函 数	转换后的 x 类型或返回值
Cbool(x)	Boolean	Cstr(x)	String
Cbyte(x)	Byte	Cvar(x)	Variant
Ccur(x)	Currency	Cerr(x)	Error
Cdate(x)	Date	Cdbl(x)	Double
Cint(x)	Integer	Clng(x)	Long
Fix(x)	去掉小数部分（不四舍五入）	Hex(x)	将十进制数转换为十六进制数
Oct(x)	将十进制数转换为八进制数	Str(x)	把 x 的值转换为一个字符串
Val(x)	返回字符串内的数字	Chr(x)	返回字符码对应的 ASCII 字符

③ 数学函数

数学函数用来完成特定的数学计算，常见的数学函数见表 11.3。

表 11.3 常见数学函数

函 数	功 能	函 数	功 能
Abs(x)	求绝对值	Exp(x)	求指数
Int(x)	求不大于 x 的最大整数	Log(x)	求对数
Rnd(x)	产生一个 0 ~ 1 间的随机数	Tan(x)	正切函数
Sin(x)	正弦函数（返回弧度的正弦）	Cos(x)	余弦函数
Atn(x)	反正切函数	Sqr(x)	求平方根
Sgn(x)	求数字符号。x 为负、为正、为 0，函数分别返回-1、1、0		

④ 日期和时间函数

VB 提供了丰富的日期和时间函数，这些函数不仅可以用来返回和设置当前的时间和日期，还可以从日期和时间中提取年、月、日、时、分、秒，以及对时间和日期进行格式化设置等，常见的日期和时间函数见表 11.4。

表 11.4 常见的日期和时间函数

函 数	功 能	函 数	功 能
Date(x)	取得系统当前日期	Time(x)	取得系统当前时间
Now(x)	取得系统当前日期和时间	Year(x)	返回指定日期中的年份
Month(x)	返回指定日期中的月份	Day(x)	返回指定日期中天
Weekday(x)	返回指定日期为星期几	Hour(x)	返回指定时间量中的小时
Minute(x)	返回指定时间量中的分钟	Second(x)	返回指定时间量中的秒

⑤ 字符串函数

字符串函数用来完成对字符串的操作和处理，如截取字符串、查找和替换字符串、对字

字符串进行大小写处理等。VB6.0 新添了许多字符串函数，加上原有的函数，可见其在对字符串的处理方面功能非常全面，常见的字符串函数见表 11.5。

表 11.5 常见的字符串函数

函 数	功 能	函 数	功 能
StrComp (字符串 1, 字符串 2)	两个字符串比较	Lcase (字符串)	把大写字母转换为小写字母
Ucase (字符串)	把小写字母转换为大写字母	Space(n)	返回 n 个空格
String (n, 字符串)	返回由 n 个字符组成的字符串	Len (字符串)	测试字符串长度
Instr (字符串 1, 字符串 2)	在字符串 1 中查找字符串 2	Mid (字符串, p, n)	从位置 p 开始取字符串的 n 个字符

⑥ 目录和文件函数

目录和文件函数属于文件系统函数。通过它们可以操作目录和文件，如创建目录、删除文件和读写文件等，常见的目录和文件函数见表 11.6。

表 11.6 常见的目录和文件函数

函 数	功 能	函 数	功 能
ChDir	改变当前目录	ChDrive	改变当前驱动器
CurDir	返回当前目录	MkDir	创建目录
RmDir	删除目录	FileCopy	复制文件
Kill	删除文件	FileLen	取得文件的长度
Dir	查找指定的文件和目录		

3. 表达式

将数据（如常量、变量、函数等）用运算符按一定的规则连接起来的有意义的式子称为表达式。例如，算术表达式、逻辑表达式、字符表达式等。

4. 语句

语句是高级编程语言源代码的基本组成元素，VB 源程序也是由 Basic 语句组成。语句一般包括说明性语句、赋值语句、选择语句、循环语句、跳转语句、返回语句、开关语句、设置变量数据类型语句、注释语句和续行标志等。

11.3.2 书写规则

书写 VB 程序时应遵守相应的书写规则，不仅应该满足 VB 的语法要求，同时还应该使程序具有良好的可读性。

1. 程序书写规则

程序是完成特定功能的语句序列，并最终由系统编译并运行，在书写程序时，应遵守相应的规则：

- （1）程序中不区分字母的大小写；
- （2）系统对用户程序代码进行自动转换，对于 VB 中的关键字，首字母被转换成大写，其余转换成小写；若关键字由多个英文单词组成，则将每个单词的首字母转换成大写；对于用户定义的变量、过程名，以第一次定义的为准，之后输入的自动转换成首次定义的形式。

2. 语句书写规则

语句是程序最基本的单位，在书写语句时，应遵守以下的规则：

- （1）一般是一行写一句，若在同一行上书写多行语句，语句间需用冒号即“:”分隔；

(2) 单行语句可以分多行书写, 在本行后加续行符, 续行符为空格和下划线, 即 “□_” (□代表空格);

(3) 一行允许多达 255 个字符。

3. 程序的注释规则

编写的程序应便于阅读和理解, 而最好的解决方法就是给程序添加注释, 在添加注释时, 应遵守相应的规则:

(1) 整行注释一般以 Rem 开头;

(2) 用单引号 “'” 引导的注释, 既可以是整行的, 也可以直接放在语句的后面;

(3) 可以利用 “编辑” 工具栏的 “设置注释块”、“解除注释块” 来将设置多行注释。

11.4 Visual Basic语言基础

11.4.1 数据类型

在编写程序时, 必须做好两件事: 一是描述数据, 二是描述数据的加工方法。前者是通过数据定义语句实现, 而后者是通过若干执行语句来完成。数据的加工建立在数据描述的基础上, 数据描述的好坏直接影响数据加工的质量。对于每种数据类型, 需要注意该类型变量所占空间、取值范围、可进行的运算。

1. 基本数据类型

基本数据类型是系统定义的标准数据类型, 可以直接使用。VB6.0 提供的基本数据类型有: 字符型数据、数值型数据、布尔类型、日期类型、通用类型、对象类型等。下面介绍 VB 基本数据类型的特点。

(1) 字符串: String

字符串用来定义一个字符序列, VB 中字符串长度是以字 (习惯称字符) 为单位, 也就是每个西文字符和每个汉字都作为一个字, 占 2 字节。这与传统的概念有所不同, 原因是编码方式的不同。在 Windows 中采用 DBCS (Double Byte Character Set) 编码方案, 一个汉字在计算机内存中占 2 字节, 一个西文字符 (ASCII 码) 占 1 字节, DBCS 编码实际上是单字节与双字节的混合编码。但在 VB 中采用的是 Unicode (国际标准化组织 ISO 字符标准) 码来存储字符和操作字符串的, Unicode 码全部用两字节表示一个字符的字符集, 所以一个西文或汉字字符都占 2 字节。

在默认字符状态下, String 为变长字符串, 变长字符串的长度变化范围为 $0 \sim (2^{31}-1)$ 个字符; 当然, 也可以声明为定长字符串, 定长字符串的长度变化范围为 $0 \sim (2^{16}-1)$ 个字符。如果字符个数少于固定长度, 自动用空格将不足部分补齐, 否则会自动截去多余部分的字符。

注意:

① "" 表示空字符串, 而 " " 表示空格。

② 如字符串中本身含有双引号, 则用两个连续的双引号表示。例如, 表示字符串 “计算机 ABC”, 正确的表示形式为 “计算机""ABC”。

(2) 数值型数据

数值型数据分为整型和实型, 共两类 4 种。整型数据分为整数和长整数, 实型数据分为单精度和双精度数据。有时把货币型和字节型数据也归为数值型数据。

① 整型数: Integer

整数不带小数点, 范围是-32 768~32 767, 一个整数在内存占用 2 字节 (16 位)。-32 678~32 767 之间的一个数字在尾部加一个 “%” 符号也表示一种整型数据。

例如: 1 345%, -67%。

② 长整型数: Long

长整型数是范围是-2 147 483 648~2 147 483 647 不带小数点的数, 一个长整数在内存中占 4 字节 (32 位)。在 VB 中, 可在数尾加 “&” 表示长整型数据。

例如: 59 872&, 123&。

整数 (Integer) 和长整数 (Long) 均用于保存整数, 其可以是正整数、负整数或者 0。例如: 7 210、-7 869、+3 969 均表示整数, 而 398.0 就不是整数; -9 998 936、12 654 378 均表示长整数, 而 123 654 则是非法数。整型数具有运算速度快、精确、且占用存储空间较小的优点, 但表示数的范围也较小。

③ 单精度: Single

单精度是带小数点的实数, 精确到小数点后 6 位。在内存中用 4 字节 (32 位) 存放一个单精度数。通常以指数形式 (科学记数法) 来表示, 以 “E” 或 “e” 表示指数部分。在 VB 中数尾常加 “!” 表示单精度数据, 也可省略。

例如: 785.98!, 1.378 9E10, 0.572 38e-5。

如果某个数小数点后的有效数字位数超过 6 位, 当把它定义为单精度时, 超出的部分会自动四舍五入。

④ 双精度数据: Double

双精度数据是带小数点的实数, 精确到小数点后 14 位。在内存中用 8 字节 (64 位) 存放一个双精度数。双精度数通常以指数形式 (科学记数法) 来表示, 以 “D” 或 “d” 表示指数部分。在 VB 中数尾常加 “#” 表示双精度数据, 也可省略。

例如: -9 874.547 78#, 0.589 68D5, 0.125 782d-8。

如果某个数小数点后的有效数字位数超过 14 位, 当把它定义为双精度时, 超出部分会自动四舍五入。

⑤ 货币类型: Currency

货币类型: 是为计算货币而设置的定点数据类型, 是一种特殊的小数。在内存中占 8 字节 (64 位)。取值范围在-922 337 203 685 477.580 5~922 337 203 685 477.580 7 之间。货币型数据保留小数点右边 4 位和小数点左边 15 位, 如果数据定义为货币型, 且其小数点后超过 4 位, 那么超过的部分自动四舍五入。在 VB 中数尾常加 “@” 表示货币型数据。

例如: 3 573.443 5@, 565 980@。

⑥ 字节类型: Byte

字节类型是一种数值类型, 以 1 字节的无符号二进制数存储, 取值范围为 0~255, 不能表示负数。

(3) 布尔类型: Boolean

布尔型数据是一个逻辑值, 也称逻辑类型, 用来表示逻辑判断结果, 用 2 字节存储, 它只取两种值: 即 True (真) 或 False (假)。若数据信息是 “true/false”、“yes/no”、“on/off” 信息, 则可将它定义为 Boolean 类型。

从其他数据类型转换为 Boolean 时: 0 转为 False, 其他值转为 True;

从 Boolean 转换为其他整型时: False 转为 0, True 转为-1

(4) 日期类型: Date

日期类型用以表示日期,在内存中一个日期型数据用 8 字节来存放,表示从公元 100 年 1 月 1 日到公元 9999 年 12 月 31 日的日期,时间范围则从 0 点 0 分 0 秒到 23 点 59 分 59 秒,即 0:00:00~23:59:59。日期型数的表示方式有两种:可以用号码符“#”括起来,也可以用数字序列表示(小数点左边的数字代表日期,右边代表时间,0 为午夜,0.5 为中午 12 点,负数表示是 1899 年 12 月 30 日前的日期和时间)。下面给出一个系统识别多种日期格式的例子:

```
Dim birthday As Date
birthday = #12/5/78#
birthday = #1990-12-5 12:00:00 AM#
birthday = #January 1, 1993#
birthday = -2.25
Print birthday
```

打印出来的结果是 1899-12-28 6:00:00

(5) 通用类型: Variant

通用类型也称变体类型,是一种通用的、可变的数据类型,它可以表示上述任何一种数据类型。VB 中没有声明的变量其默认的数据类型为变体型,这种变量可以用来存储各种数据,但所占用的内存比其他类型都多。为提高运行效率(整型效率较高)或达到一定的运算精确度(浮点型精度较高,但运行较慢)应合理的定义数据类型。

定义 a 为通用型变量:

```
Dim a As Variant
```

在变量 a 中可以存放任何类型的数据,例如:

```
a=10           'a 中存放一个整数。
a=20.5         'a 中存放一个实数。
a=#08/15/2003# 'a 中存放一个日期型数据。
```

根据赋给 a 的值的类型不同,变量 a 的类型不断变化,这就是称之为变体类型的由来。当一个变量未定义类型时,VB 自动将变量定义为 Variant 类型。不同类型的数据在 Variant 变量中是按其实际类型存放的(例如将一个整数赋给 a,在内存区中按整型方式存放),用户不必作任何转换的工作,VB 自动完成。

(6) 对象类型: Object

对象型数据主要以变量形式存在,可以引用应用程序中或某些其他应用程序中的对象,用来表示图形、OLE 对象或其他对象,用 4 字节存储。保存对象的引用(地址),只能用 Set 语句赋值。

例如:

```
Dim objDb As Object
Set objDb=OpenDatabase("c:\Vb6\student.mdb")
```

2. 自定义数据类型

在信息处理中,经常会遇到要对某一客观事物及其属性进行描述的情况,如对人的描述,人是具有整体概念的数据,其包含的属性是一些不同数据类型的分量,其属性有名字、性别、身份证号、年龄、住址、职务等数据项。为了方便处理,需要把它们组成为一个有机的整体,并为其命名,这些可以通过自定义数据类型来实现。

(1) 自定义数据类型的定义

自定义数据类型是指由若干标准数据类型组成的一种复合类型，也称为记录类型。自定义数据类型必须先定义其结构。

定义方式：

Type 自定义类型名

元素名 As 类型名

.....

元素名 As 类型名

End Type

其中：

- ① 元素名：表示自定义类型中的一个成员；
- ② 类型名：表示元素的类型。

例如：定义一个学生信息的自定义类型如下：

```
Type StuType
    Num    As Integer
    Name   As String*9
    Sex    As String*2
    Score (1 TO 6) As Single
End Type
```

例中定义了一个自定义类型，类型名为 **StuType**，该类型由 4 个成员组成：第 1 个成员名为 **Num**，整型变量；第 2 个成员名为 **Name**，字符型；第 3 个成员名为 **sex**，字符型；第 4 个成员名为 **Score**，单精度数据，6 个成员。

注意：

- ① 自定义类型一般在标准模块 (.bas) 中定义，默认是 **Public**；
- ② 自定义类型中的元素可以是字符串，但应是定长字符串；
- ③ 不可把自定义类型名与该类型的变量名混淆，上例仅仅是定义了一种新的数据类型，如果需要使用该类型的变量，则需要定义变量；
- ④ 一个自定义类型变量所占内存空间的总长度是各成员所占内存空间长度之和。

(2) 自定义型变量的声明和使用

定义方式：Dim 变量名 As 自定义类型名

例如，Dim stu1 As StuType

自定义类型中元素的表示方法是：变量名.元素名

例如，stu1.name="张小强"

stu1.score(4)=80.5

11.4.2 常量与变量

1. 标识符

程序中有许多对象需要命名，以便在程序中其他地方使用它们。为此，每种程序语言都规定了在程序中为对象命名的规则，这些名字包括：变量名、常量名、数组名、过程名等，通常被统称为标识符。

VB 中标识符的命名规则如下：

- (1) 必须以字母或汉字开头, 由字母、汉字、数字或下划线组成, 最长可达 255 个字符;
- (2) 不能使用 VB 中的关键字, 并尽量不与 VB 中标准函数名同名;
- (3) 不区分大小写, 一般变量首字母用大写, 其余用小写, 常量全部用大写字母表示;
- (4) 为了增加程序的可读性, 可在变量名前加一个缩写的前缀来表明该变量的数据类型。

2. 常量

常量是指在程序运行过程中值不能改变的量。另外, 常量的处理速度比变量快。程序运行时, 常量值不需要查找, 编译器只要把常量名换成常数即可, 这样保证程序执行得更快。

VB 常量包括三类: 系统内部常量、直接常量和符号常量。

(1) 系统内部常量

系统内部常量是应用程序和控件提供的, 可与应用程序的对象、方法和属性一起使用。可以在帮助文件中查找到相应的常量, 系统内部的常量名采用大小写混合的格式, 其前缀表示定义常量的对象库名。

例如, 来自 Visual Basic 和 Visual Basic for Applications 对象库的常数以“VB”开头的常量 VBBlack 表示黑色。来自访问对象库的常数以“db”开头的常量 dbRelationUnique。

(2) 直接常量

直接常量是指在程序中直接给出值的数据。例如, 123、123&、123.45、1.234E2、123D3、“abc”、#2009/10/28#。分别表示整型、长整型、单精度浮点型(小数形式)、单精度浮点型(指数形式)、双精度浮点型、字符型、日期型。

八进制常数在数值前加&O, 如&O123。

十六进制常数在数值前加&H, 如&H123。

(3) 符号常量

符号常量用 Const 来声明, 其声明方式为:

Const 符号常量名 [As 类型] = 表达式

例如:

Const PI=3.14159	'声明了常量 PI, 代表 3.141 59, 单精度型
Const MAX As Integer=&O144	'声明了常量 MAX, 代表八进制数 144, 整型

注意:

- ① 如果程序中有特别频繁使用的值, 且这个值从不变化, 并在整个模块和窗体中使用, 可以用 Public Const 语句来声明一个全局常量。
- ② 在常量声明的同时要对常量赋值, 在程序运行的过程中, 常量不能被重新赋值。
- ③ 可以为声明的常量指定类型, 如: Const Val As Currency=3.37, 省略时为所赋的类型。当然也可以不指定。如果不指定常量的类型, 则系统会根据表达式的数值指定该常量的类型。如果用逗号进行分隔, 则在一行中可放置多个常量声明。

3. 变量

在程序运行过程中其值可以改变的量称为变量。每个变量都应有一个与其他变量不同的名字, 称为变量名。对于任何一个变量, 计算机在内存中为其分配一定的存储空间, 用来存储该变量的值。这就意味着变量具有三个基本要素: 即变量名、类型和值。

变量具有保持当前值的特性, 也就是说, 如果在某个时刻为某变量赋了一个新值, 此后使用这个变量时, 每次得到的将总是这个新值。变量名作为变量的标识, 其命名规则应符合标识符的命名规定。变量有两种类型: 属性变量和用户自己建立的变量。

当我们在窗体中设计用户界面时，VB 会自动为产生的对象（包括窗体本身）创建一组变量，即属性变量，并为每个变量设置其默认值。这类变量可供我们直接使用，比如引用它或给它赋新值。

用户也可以创建自己的变量，以便存放程序执行过程中的临时数据或结果数据。在程序中，这样的变量是非常需要的。

用户自定义变量根据作用域不同可分为全局变量和局部变量。

① 全局变量

用 **Public** 声明的变量就是全局变量，程序中的任何窗体和模块都能访问到它。声明全局变量要小心，任何模块或窗体中对它的修改都会影响其他模块或窗体对该变量的引用。

② 局部变量

在模块和窗体中用 **Dim** 或 **Static** 声明的变量只在本模块或窗体中起作用，这样的变量叫局部变量。如果全局变量和局部变量同名，则在局部变量所在模块或窗体内部，局部变量起作用。离开该窗体或模块，全局变量起作用。

在使用变量之前，大多数语言通常首先需要声明变量。就是说，必须事先告诉编译器在程序中使用了哪些变量，这些变量的数据类型以及变量的长度。VB 中声明变量有两种方式：隐式声明和显式声明。

① 隐式声明

VB 中允许变量不经过声明就直接使用，这种方式称为隐式声明，所有隐式声明的变量都是变体型的。隐式声明容易造成错误，为了调试程序方便，一般对使用的变量应先声明，可以在通用声明段使用 **Option Explicit** 语句来强制显式声明所有变量。

② 显式声明

显式声明中需用声明语句创建变量，用 **Dim** 语句进行显式声明。语句形式如下：

Dim 变量名 [**As** 类型]

例如：**Dim intX As integer, intY As integer, intZ As integer.**

在显式说明时，有以下几点需要注意：

- ◆ 如果没有 **As** 类型，则默认为变体类型；
- ◆ 可在变量名后加类型符来代替 **As** 类型，如：**Dim intX%**；
- ◆ 一条语句可以同时定义多个变量，但每个变量必须有自己的类型声明，同类型声明不能共用；
- ◆ 字符串变量根据其存放的长度是否固定，分为定长字符串定义和不定长字符串定义。定长字符串声明形式为“**Dim strA As String*10**”，表示最多存放 10 个字符，如果赋值不足 10 个，则右补空；若多于 10 个，则多余部分截去。

不定长字符串声明形式为“**Dim strB As String**”。

例如，有以下变量声明：

Dim a,b,c as string * 4

表示在 VB 中声明了 a 和 b 两个变体型变量，以及 c 这个字符串变量。若要声明 a,b,c 三个字符串变量，正确的写法如下：

Dim a as string * 4

Dim b as string * 4

Dim c as string * 4

若想写在同一行，写法如下：

```
Dim a as string * 4, b as string * 4, c as string * 4
```

11.4.3 表达式与运算符

1. 表达式

表达式由常量、变量、运算符、函数和圆括号按一定的规则组成，通过运算后有一个结果，运算结果的类型由数据和运算符共同决定。表达式最终由系统理解和执行，所以在书写时必须遵守相应的书写规则：

- ① 乘号不能省略；
- ② 括号必须成对出现，均使用圆括号，可以嵌套，但必须配对；
- ③ 表达式从左到右在同一基准上书写，无高低、大小之分。

2. 变量赋值

赋值语句是任何程序设计中最基本的语句，变量赋值的格式为：

```
变量名=表达式
```

其功能是计算右边表达式的值，然后赋给左边的变量，当表达式的类型与变量的类型不一致时，系统会自动进行类型转换。

有关自动类型转换的几点说明如下：

- ① 当表达式为数值型而与变量精度不同时，强制转换成左边变量的精度；
- ② 当表达式是数字字符串，左边变量是数值类型，自动转换成数值类型再赋值，但当表达式中有非数字字符或是空串时，会出错；
- ③ 任何非字符类型赋值给字符类型，自动转换为字符类型；
- ④ 当逻辑型赋值给数值型时，True 转换为-1，False 转换为 0；反之，非 0 转换为 True，0 转换为 False；
- ⑤ 赋值号左边只能是变量，不能是常量、常数符号和表达式，否则报错；
- ⑥ 不能在一句赋值语句中，同时给多个变量赋值；
- ⑦ 在条件表达式中出现的“=”是等号，系统会根据“=”号的位置，自动判断是否为赋值号。

3. 算术运算符

算术运算符两边的操作数应该是数值类型，若是数字字符或逻辑型，则自动转换为数值类型后再运算。VB 提供的算术运算符及功能见表 11.7。

表 11.7 算术运算符及功能

基本运算	运算符	示 例	含 义
乘方	^	2^8	表示 2 的 8 次方
负号	-	-3	表示负 3
乘法	*	a*b	表示 a 乘以 b
除法	/	5.2/2	计算结果为 2.6
整除	\	5\2	计算结果为 2
求余数	Mod	17 mod 3	计算结果为 2
加法	+	a+b	表示 a 加 b 的和
减法	-	a-b	表示 a 减 b 的差

优先级由高到低的顺序为：^、-（负号）、*（乘法，和除法同优先级）、\、Mod、+（加法，和减法同优先级）。

4. 字符串运算符

系统提供的字符串连接运算符有两个：&和+，其功能是将两个字符串拼接起来。

&连接符两旁的操作数不管是字符型还是数值型，系统先将操作数转换成字符，然后再连接。

例如："123"&55

结果为："12355"

+连接符比较复杂，如果参加“+”运算的操作数中有数值或逻辑值，则它会将所有操作数转换成数值后再完成加法运算。转换时，逻辑值 True 被转换成-1，False 转换成 0；如果参加运算都是字符串，则该运算演变为字符串连接运算；如果参数运算的操作数中有日期型，则运算按日期型来完成。

例如：

"abc"+"12"， 结果为："abc12"；

"123"+55，结果为：178；

"abc"+12，则出错；

False+ False，结果为：0；

#2007-01-01# +1，结果为：2007-1-2

5. 关系运算符

关系运算符用于比较两个数据的关系是否成立，运算的结果是“成立”或“不成立”。如果成立，则结果为逻辑值“真”，用 True 表示；如果不成立，则结果为逻辑值“假”，用 False 表示，VB 的关系运算符及功能见表 11.8。

表 11.8 VB 的关系运算符及功能

关系运算	运算符	示 例	含 义
大于	>	100>99	结果为 True
小于	<	1.2<-23	结果为 False
大于等于	>=	Sin(1)>=0	结果为 True
小于等于	<=	Sqr(3)<=0	结果为 False
等于	=	12=13	结果为 False
不等于	<>	12<>13	结果为 True
字符串匹配	Like	"ABCDEFGH" Like "*DE*"	结果为 True

注意：

- ① 如果两个操作数都是数值型，则按其大小比较；
- ② 如果两个操作数都是字符型，则按字符的 ASCII 码值从左到右一一比较；
- ③ 汉字字符大于西文字符；
- ④ 关系运算符的优先级相同；
- ⑤ VB6.0 中 Like 运算符与通配符的使用说明如下：
 - ?：表示任何单一字符；
 - *

[字符列表]: 表示字符列表中的任何单一字符;
[!字符列表]: 表示不在字符列表中的任何单一字符;

6. 逻辑运算符

逻辑运算符用于构成逻辑表达式, 从而实现逻辑判断功能。逻辑运算符包括: 逻辑与(And), 逻辑或(Or), 逻辑非(Not), 逻辑异或(Xor), 逻辑相等(Eqv)和逻辑蕴含(Imp)。VB 的逻辑运算符及功能见表 11.9。

表 11.9 逻辑运算符及功能

逻辑运算	运算符	功 能
取反	Not	操作数为假时, 结果为真; 操作数为真时, 结果为假
与	And	两个操作数均为真时, 结果才为真
或	Or	两个操作数中有一个为真时, 结果为真
异或	Xor	两个操作数不相同, 结果才为真, 否则为假
等价	Eqv	两个操作数相同时, 结果才为真
蕴含	Imp	第一个操作数为真, 第二个操作数为假时, 结果才为假, 其余都为真

优先级由高到低的顺序为: Not、And、Or (和 Xor 同优先级)、Eqv、Imp。

注意:

- ① 若有多个条件时, And 必须全部条件为真才为真; Or 只要有一个条件为真就为真。
- ② 如果逻辑运算符对数值进行运算, 则以数字的二进制值逐位进行逻辑运算。And 运算常用于屏蔽某些位; Or 运算常用于把某些位置为 1。

例如: 10 And 7, 表示对 1010 与 0111 进行 And 运算, 得到二进制值 0010, 结果为十进制 2。

- ③ 对一个数连续进行两次 Xor 操作, 可恢复原值。

7. 运算符优先级

在一个表达式中进行多个运算时, 每一部分都会按预先确定的顺序进行计算求解, 这个顺序被称为运算符优先级。括号可改变优先级的顺序, 强制优先处理表达式的某部分。括号内的操作总是比括号外的操作先被执行。但是在括号内, 仍保持正常的运算符优先级。

当表达式有多种运算符时, 先处理算术运算符, 接着处理比较运算符, 然后再处理逻辑运算符。所有比较运算符有相同的优先级, 即按它们出现的顺序从左到右进行处理。不同运算符之间的优先级见表 11.10。

表 11.10 运算符优先级

高 → 低		
算 术	比 较	逻 辑
指数运算 (^)	相等(=)	Not
负数(-)	不等(<>)	And
乘法和除法(*, /)	小于(<)	Or
整除(\)	大于(>)	Xor
求余运算(Mod)	小于或等于(<=)	Eqv
加法和减法(+, -)	大于或等于(>=)	Imp
字符串连接(&)	Is	&

当乘法和除法同时出现在表达式中时, 按照从左到右的顺序处理每个运算符。同样, 当加法和减法同时出现在表达式中时, 也按照从左到右的顺序处理每个运算符。

字符串连接运算符(&)不是算术运算符,但是就其优先级而言,它在所有算术运算符之后,而在所有比较运算符之前。Is 运算符是对象引用的比较运算符,它并不比较对象或对象的值,而只判断两个对象引用是否引用了相同的对象。

11.5 数 组

在程序设计中,为了处理方便,可以把具有相同类型的若干变量按空间连续的形式组织起来,这些具有连续存储空间的同类数据元素的集合称为数组。一个数组可以分解为多个数组元素。数组可以是一维的,也可以是多维的。在实际问题中碰到的一些成批出现的变量,如数学中的向量和矩阵、数据处理中的表格等,都可以用数组对其进行存储和处理。

11.5.1 基本概念

1. 数组

数组是一组变量的集合。这些变量称为数组的元素,每个数组元素都有一个编号,这个编号称为下标,我们可以通过下标来区别这些元素。数组元素的个数有时也称为数组的长度。一般情况下,数组的元素类型必须相同,可以是前面讲过的各种基本数据类型,但当数组类型被指定为变体型时,它的各个元素就可以是不同的类型。数组和变量一样,也有作用域,按作用域的不同可以把数组分为:过程级数组(或称为局部数组)、模块级数组以及全局数组。

2. 数组元素

数组元素即数组中的变量,用下标表示数组中的各个元素。

3. 数组维数

数组维数由数组元素中下标的个数决定,一个下标表示一维数组,二个下标表示二维数组。VB 中有一维数组、二维数组、……最多 60 维数组。

4. 下标

下标表示顺序号,每个数组元素有一个唯一的顺序号,下标不能超过数组声明时的上、下界范围。下标可以是整型的常数、变量、表达式,甚至还可以是一个数组元素。

下标的取值范围是:下界~上界,省略下界时,系统默认取 0。

5. 静态数组和动态数组

静态数组:声明时确定了数组的大小。

动态数组:声明时没有给定数组大小(省略了括号中的下标),使用时需要用 ReDim 语句重新指定其大小。

使用动态数组的优点是根据用户需要,能动态改变数组的大小,它是在程序执行到 ReDim 语句时才分配存储单元,而静态数组是在程序编译时分配存储单元。

11.5.2 一维数组的使用

1. 静态数组的声明

数组应先声明再使用,声明时应包含以下内容:数组名、类型、维数、数组大小。一般情况下,数组中各元素类型必须相同,但若数组为 Variant 时,可包含不同类型的数据。

声明格式:

Dim 数组名 ([下界 TO]上界) [As 类型名]

声明数组时应注意以下问题:

- ① 数组的下界和类型是可选的, 默认下界为 0;
- ② 如果定义数组时不指定其类型, 默认为变体型。

下面举例说明定长数组的定义:

- ① Dim a(1 to 10) As Integer
- ② Dim b(5 to 9) As String
- ③ Dim c(9) As Integer '相当于 Dim c(0 to 9) As Integer
- ④ Dim d(6)

其中:

- ① 中定义了一个具有 10 个元素的整型数组, 其下标从 1~10;
- ② 中定义了一个具有 5 个元素的字符型数组, 其下标从 5~9;
- ③ 中省略了下界, 它定义了一个具有 10 个元素的整型数组, 其下标从 0~9;
- ④ 中省略了下界和类型, 定义的是具有 7 个元素的变体型数组, 其下标从 0~6。

在定义定长数组时, 其上界和下界必须是常数或常量表达式。

2. 动态数组的建立及声明

在很多情况下, 数组的长度事先无法预测, 而且可能需要在程序中改变数组的长度以适应新的情况, 因此需要使用动态数组。动态数组是在定义数组时仅指定数组名及其类型, 等以后知道数组的长度或需要改变数组长度时再用 ReDim 指定它的长度。

建立动态数组的方法是: 利用 Dim、Private、Public 语句声明括号内为空的数组, 然后在过程中用 ReDim 语句指明该数组的大小。

例如:

```
Dim X AS Integer
Dim a() As String
.....
X=10
ReDim a(x)
.....
X=X+3
ReDim a(x)
```

这段代码首先定义了一个整型变量和一个字符型的动态数组, 之后经过一系列运算后使变量 X 得到一个定值, 再使用“ReDim”指定数组的长度, 最后根据需要又一次改变了数组的长度。

使用动态数组时, 应该注意以下问题:

- ① 在动态数组 ReDim 语句中的下标可以是常量, 也可以是有了确定值的变量。
- ② 可以一次或多次改变动态数组的长度, 也可改变数组的维数。但当重新指定数组长度时, 数组内原有的数据会被清除。如果既想改变数组的长度, 又想保留数组原有的数据, 则在使用 ReDim 时需要加上“Preserve”关键字。

例如, ReDim Preserve a(x)。但要注意, 如果缩小数组长度, 那么数组中超出新的下界到上界之间的元素将清除, 即使使用了“Preserve”关键字, 也不会保留这些元素, 如果仍然访问这些元素将会引起下标越界的错误。

③ 当不需要再使用某个动态数组时，可以使用 **Erase** 删除该数组，以释放该数组占用的内存空间。例如，**Erase a()**。

3. 给数组元素赋值

(1) 利用循环结构给数组赋值

例：Dim i As Integer, A(1 To 10) As Integer

```
For i=1 To 10
```

```
    A(i)=0
```

```
Next i
```

(2) 在 VB6.0 中可以直接将一个数组的值赋值给另一个数组

使用数组赋值的格式是：数组 2=数组 1。使用数组赋值有几点需要注意：

① 这里的数组 2，只能声明为 **Variant**，赋值后的数组 2 的大小、维数、类型与数组 1 相同。

② 若将数组 2 声明成变长或静态的数组，例如：

Dim 数组名 2()或 Dim 数组名 2 (下标)

程序在运行到上述赋值语句时，会显示“不能给数组赋值”的出错信息。

阅读下面程序，理解运行结果。

```
Private Sub Command1_Click()
```

```
    Dim i As Integer, A(1 To 10) As Integer
```

```
    Dim s As Variant
```

```
    For i = 1 To 10
```

```
        A(i) = i
```

```
    Next i
```

```
    s = A
```

```
    For i = 1 To 10
```

```
        Print s(i);
```

```
    Next i
```

```
End Sub
```

运行结果为：1 2 3 4 5 6 7 8 9 10

11.5.3 多维数组的使用

一个数组可以是一维的，也可以是多维。当需要表示平面中的一个点坐标，就需要用到二维数组；表示空间中的一个点时，就需要用到三维数组。

多维数组的定义格式为：

Dim 数组名 ([下界 TO]上界[, [下界 TO] 上界 [, ...]]) [AS 类型名]

多维数组的定义格式与一维数组基本上是一致的，只是多加几个上界和下界。

例如：

```
Dim a(1 TO 3,1 TO 4) As Integer
```

```
Dim b(5, 9) As Double
```

```
Dim b(4, 3)
```

这三行语句各自定义了一个二维数组，第一个指定了下界及类型；第二个只指定了类型使用默认的下界；最后一个下界和类型都没有指定，其类型是变体型的。

对数组元素的引用，是数组名加下标的形式。

例如：

```
Dim a(5) as Integer
Dim b(3,4) as Integer
a(1)=3
a(2)=5
a(3)=a(1)+a(2)
b(2,1)=a(2)
```

在此例中，先对数组 a 的前三个元素赋值，再给数组 b 的一个元素赋值。使用数组元素时，其下标不要超出下界至上界的范围。

11.6 基本程序设计

程序是由语句组成的，它描述计算机的执行步骤。程序中语句的执行顺序称为程序结构。如果程序中的语句是按照书写顺序执行，则称其为顺序结构；如果某些语句是按照当时的某个条件来决定是否执行，则称其为选择结构；如果某些语句要反复执行多次，则称其为循环结构。

11.6.1 程序的基本结构

程序设计理论已经证明，任何程序均可由三种基本结构构成。这三种基本结构分别是顺序结构、选择结构和循环结构，如图11.2所示。

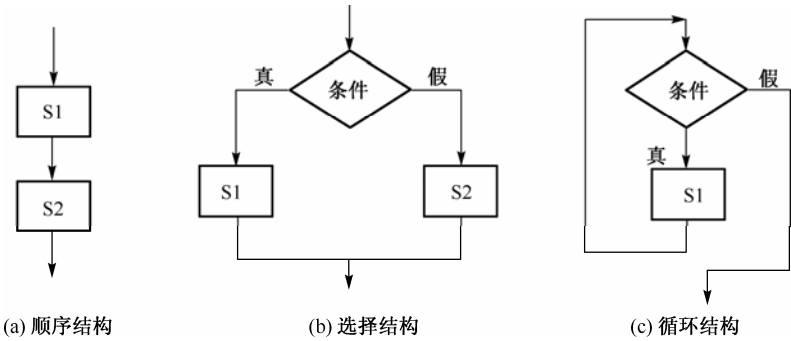


图 11.2 程序的三种基本结构

顺序结构中，从前向后顺序执行 S1 语句和 S2 语句，只有当 S1 语句执行完后才执行 S2 语句。选择结构中，根据条件有选择地执行语句，当条件成立（为真）时执行 S1 语句，当条件不成立（为假）时执行 S2 语句。循环结构中，有条件地反复执行 S1 语句（也称为循环体），直到条件不成立时终止循环，控制转移到循环体外的后继语句。一个程序无论多么复杂和庞大，程序的模块内部或模块之间都是由这三种基本结构组成的。因此，任何一种程序设计语言都会设计与之对应的语句，完成程序中的流程控制。

11.6.2 顺序结构

顺序结构的特点是程序执行时是从上到下依次执行的。

例如，将变量 a、b 中存放的数据交换可以有以下三种方法可以实现。

【方法一】使用中间变量（中转站）

```
c=a:a=b:b=c
```

【方法二】不使用中间变量（先求和）

$a=a+b:b=a-b:a=a-b$

【方法三】不使用中间变量（先作差）

$a=a-b:b=a+b:a=b-a$

以上三种方法都是逐条语言顺序执行的顺序结构。

11.6.3 选择结构

选择结构语句是指程序在运行中，能依据运行时某些变量或表达式的值，确定哪些程序段被执行以及哪些程序段不被执行。

1. if条件语句

(1) If...Then语句（单分支结构）其语句执行过程如图 11.3 所示。

语句形式：

① If <表达式> Then

语句块

End If

② If<表达式>Then<语句>

说明：表达式一般为关系表达式、逻辑表达式，也可以为算术表达式，非 0 为 True，0 为 False。

语句块可以是一句或多句，若用方式②，则最好是一句语句，若多句，语句间需用冒号分隔，而且必须在一行上书写。

【例 11-1】已知两个数 x 和 y，比较它们的大小，使得 x 大于 y。

【方法一】

```
if x<y then
    t=x
    x=y
    y=t
end if
```

【方法二】

```
if x<y then t=x:x=y:y=t
```

(2) If...Then...Else语句（双分支结构）其语句执行过程如图 11.4 所示。

语句形式：

① If <表达式> Then

<语句块 1>

Else

<语句块 2>

End If

② If<表达式>Then<语句 1>Else<语句 2>

(3) If...Then...ElseIf语句（多分支结构）

语句形式：

If <表达式 1> Then

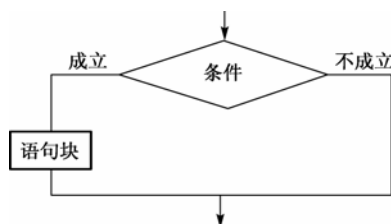


图 11.3 If...Then 语句执行过程

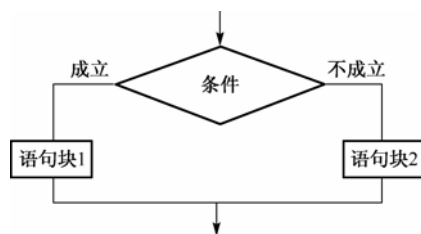


图 11.4 If...Then...Else 语句执行过程

```
<语句块 1>
ElseIf<表达式 2> Then
    <语句块 2>
    .....
[ Else 语句块 n+1 ]
End If
```

注意:

① 不管有几个分支, 程序执行了一个分支后, 其余分支不再执行;

② If...Then...ElseIf 只是 If...Then...Else 的一个特例。可以使用任意数量的 ElseIf 子句, 或者一个也不用。可以有一个 Else 子句, 而不管有没有 ElseIf 子句。ElseIf 不能写成 Else If。总是可以添加更多的 ElseIf 块到 If...Then 结构中去。但是, 当每个 ElseIf 都将相同的表达式比作不同的数值时, 这个结构编写起来很乏味, 此时, 可以使用 Select Case 判定结构;

③ 当多分支中有多个表达式同时满足, 则只执行第一个与之匹配的语句块。

(4) If语句的嵌套

If 语句的嵌套是指 if 或 else 后面的语句块中又包含 If 语句。

语句形式:

```
If <表达式 1> Then
    If <表达式 2> Then
        .....
    End If
    .....
End If
```

注意:

① 对于嵌套结构, 为了增强程序的可读性, 应该采用缩进形式书写;

② If 语句形式若不在一行上书写, 必须与 End If 配对, 多个 If 嵌套, If 与它最近的 End If 配对。

【例 11-2】 编写一个简单的计税程序。

```
Public Sub Command_Click (Money As Integer)
    If Money>10000 Then
        Tax=Money*0.2
    ElseIf Money>1000 Then
        Tax=Money*0.15
    Else
        Tax=Money*0.1
    End If
    Print Tax
End Sub
```

下面的 IF 语句用来判断一个字符是否是字母, 而且判断它的大小写。

```
If Asc(strChar)>=65 And Asc(strChar)<=90 Then
    strCharType="大写字母子"
Elseif Asc(strChar)>=97 And Asc(strChar)<=122 Then
    strCharType="小写字母"
End If
```


3. 条件函数

(1) Iif函数

函数形式：Iif（表达式，当条件为 True 时的值，当条件为 False 时的值）。

例如，求 X、Y 中较大的数，并将其放入变量 Tmax 中，可使用语句：Tmax=Iif(X>Y,X,Y)。

(2) Choose函数

函数形式：Choose（整数表达式，选项列表）

若整数表达式的值是 1，则选择列表中的第 1 项，依此类推；若小于 1 或大于列表项数时，则返回 Null。

例如，判断 nub 为 1~4 中的何值，换算成对应的运算符，可使用语句：OP=Choose (nub, “+”, “-”, “×”, “÷”)。

11.6.4 循环结构

循环语句用来描述具有规律性的重复运算，它可以大大缩短程序的长度，使程序简单明了。VB 中提供了两种类型的循环语句：计数循环语句和条件型循环语句。

1. For循环语句

For 循环语句执行过程如图 11.6 所示。

语句形式：

For 循环变量 = 初值 To 终值[Step 步长]

语句块

[Exit For]

语句块

Next 循环变量

说明：

- ① 循环变量必须为数值型；
- ② 步长可正、可负，为正数时，初值小于终值；若为负，初值大于终值；默认步长为 1；
- ③ 语句块可以是一条或多条语句，称为循环体；
- ④ Exit For 表示当遇到该语句时，退出循环体；
- ⑤ 退出循环后，循环变量的值保持退出时的值；
- ⑥ 在循环体内对循环变量可多次引用，但不要对其赋值，否则影响结果。

【例 11-3】 计算 1+2+3+...+100。

```
Sub Command1_Click()  
    Dim i As Integer, intSum As Integer  
    intSum = 0:i=0  
    For i=1 to 100  
        intSum = intSum +i  
    Next i  
    Print intSum  
End Sub
```

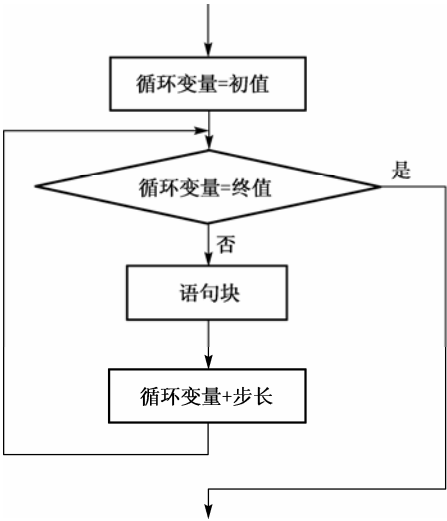


图 11.6 For 语句执行过程

2. Do...Loop循环

是用于控制循环次数未知的循环结构，语法形式有两种。

(1) Do While...Loop

其控制流程如图 11.7 所示。

```
Do [ While | Until 条件 ]
    语句块
[ Exit Do ]
    语句块
Loop
```

【例 11-4】 比较分析下面两个程序。

```
Sub Command1_Click()
    Dim i As Integer, intSum As Integer
    intSum = 0:i=0
    Do while i<100
        intSum = intSum +i
        i=i+1
    loop
    Print intSum
End Sub
```

```
Sub Command1_Click()
    Dim i As Integer, intSum As Integer
    intSum = 0:i=0
    Do until i>100
        intSum = intSum +i
        i=i+1
    loop
    Print intSum
End Sub
```

两个程序功能等价，都可以实现例 11-3 的功能。

(2) Do...Loop While

其控制流程如图 11.8 所示。

```
Do
    语句块
[ Exit Do ]
    语句块
Loop [ While | Until 条件]
```

【例 11-5】 比较分析下面两个程序。

```
Sub Command1_Click()
    Dim i As Integer, intSum As Integer
    intSum = 0:i=0
    Do
        intSum = intSum +i
        i=i+1
```

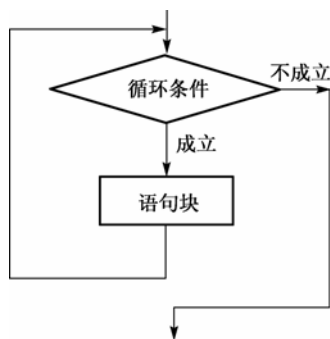


图 11.7 Do While...Loop 语句执行过程

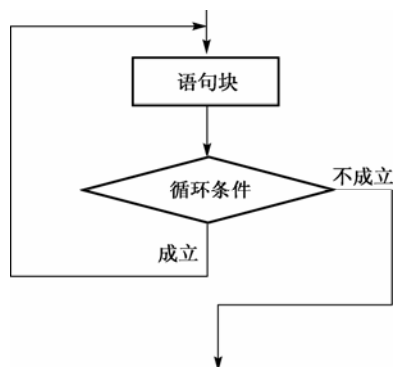


图 11.8 Do ...Loop While 语句执行过程

```
loop while i<100
Print intSum
End Sub

Sub Command1_Click()
Dim i As Integer, intSum As Integer
intSum = 0:i=0
Do
    intSum = intSum +i
    i=i+1
loop until i>100
Print intSum
End Sub
```

两个程序功能等价，也都可以实现例 11-3 的功能。

关于 Do...Loop 循环有以下几点需要注意：

- ① 形式（1）为先判断后执行，有可能一次也不执行；
- ② 形式（2）为先执行后判断，至少执行一次；
- ③ 关键字 While 用于指明条件为真时就执行循环体中的语句，Until 刚好相反；
- ④ 当省略了 While|Until 条件字句，即循环结构仅由 Do...Loop 关键字构成时，表示无条件循环，这时循环体内应该有 Exit Do 语句，否则为死循环；
- ⑤ Exit Do 语句表示当遇到该语句时，退出循环，执行 Loop 的下一语句。

【例 11-6】 求最小公倍数、最大公约数。

```
Sub Command1.Click()
Dim n As Integer, m As Integer ,mn As Integer
n = Val(Text1):m = Val(Text2)
If n <= 0 Or m <= 0 Then
    Print "数据出错"
End
End If
mn = m * n
If m < n Then t = m: m = n: n = t
Do While (n <> 0)
    r = m Mod n
    m = n
    n = r
Loop
End Sub
```

3. 循环的嵌套

循环嵌套指在循环体内又包含了另一个完整的循环结构。循环嵌套对 For 循环和 Do...Loop 循环均适用。

11.6.5 其他辅助控制语句

1. Exit 语句

用于退出某控制结构的执行，VB 的 Exit 语句有多种形式，如：Exit For（退出 For 循环），Exit Do（退出 Do 循环），Exit Sub（退出子过程），Exit Function（退出函数）。

2. End语句

独立的 End 语句用于结束一个程序的执行，可以放在任何事件过程中，形式为“End”。VB 的 End 语句还有多种形式，用于结束一个过程或块，如：End If，End With，End Type，End Select，End Sub，End Function。

3. With语句

它的作用是可以对某个对象执行一系列的语句，而不用重复指出对象的名称。但不能用一个 With 语句设置多个不同的对象。属性前面需要带点号“.”。

语句形式如下：

```
With 对象名
    语句块
End With
```

例：

```
With form1
    .Height=3000
    .Width=4000
    .BackColor=RGB(255,0,0)
End With
```

4. MsgBox函数

通过 MsgBox 函数输出数据，MsgBox 函数的一般格式如下：

```
MsgBox(msg[, type][, title][, helpfile, context])
```

功能：在屏幕上显示一个消息对话框，根据选择确定其后的操作。该函数有 5 个参数，除第一个参数外，其余参数都是可选的。

各参数的含义如下：

- ① msg 项。msg 是一个字符串，其长度不能超过 1 024 个字符，如果超过，那么多余的字符被截掉。该字符串的内容将在 MsgBox 函数产生的对话框内显示。当字符串在一行内显示不完时，将自动换行，当然也可以用“Chr\$(13)+Chr\$(10)”强制换行。
- ② type 项。type 是一个整数值或符号常量，用来控制在对话框内显示的按钮、图标种类及数量。该参数的值由四类数值相加产生，这四类数值或符号常量分别表示按钮的类型、显示图标的种类、活动按钮的位置及强制返回。如果省略，默认值为 0。

Type 的常见取值见表 11.11。

表 11.11 Type 的常见取值

常 数	值	含 义	常 数	值	含 义
vbOKOnly	0	只显示 OK 按钮	VbOKCancel	1	显示 OK 及 Cancel 按钮
VbAbortRetryIgnore	2	显示 Abort、Retry 及 Ignore 按钮。	VbYesNoCancel	3	显示 Yes、No 及 Cancel 按钮
VbYesNo	4	显示 Yes 及 No 按钮	VbRetryCancel	5	显示 Retry 及 Cancel 按钮
VbCritical	16	显示 Critical Message 图标	VbQuestion	32	显示 Warning Query 图标
VbExclamation	48	显示 Warning Message 图标	VbInformation	64	显示 Information Message 图标

③ Title 项。Title 是在对话框标题栏中显示的是字符串表达式。若省略 title，则将应用程序名放在标题栏中。

④ Helpfile 项。Helpfile 字符串表达式，用来识别向对话框提供上下文的帮助文件，如果提供了 helpfile，那么也必须提供 context。

⑤ Context 项。Context 是数值表达式，由帮助文件的作者指定给适当帮助主题的上下文编号。如果提供了 context，那么也必须提供 helpfile。

函数的返回值见表 11.12。

表 11.12 函数的返回值

常 数	值	含 义	常 数	值	含 义
VbOK	1	OK	VbCancel	2	Cancel
VbAbort	3	Abort	VbRetry	4	Retry
VbIgnore	5	Ignore	VbYes	6	Yes
VbNo	7	No			

例如：Dim MyVar

MyVar = MsgBox ("Hello welcom", VbOKCancel+ VbInformation, "测试")

MyVar 的结果需要单击按钮来产生。单击消息框上的“确定”按钮，MyVar 的值为 1，单击“取消”按钮，其值为 2。

11.7 VB的过程

在程序设计中，为各个相对独立的功能模块所编写的一段程序称为过程，通过过程可以将一个复杂的问题变为多个相互之间关联的简单问题来解决。根据是否带有返回值可将 VB 中的自定义过程分为如下两类：

- ① 子程序过程（Sub 过程），包括事件过程和自定义过程；
- ② 函数过程（Function 过程）。

有时也将自定义过程和函数过程称为通用过程，通用过程又可分为公有（Public）过程和私有（Private）过程两种，公有过程可以被应用程序中的任一过程调用，而私有过程只能被同一模块中的过程调用。

事件过程和通用过程之间存在一定的区别：

- ① 命名方法不同：事件过程名由对象名和事件名组成，而通用过程名则由用户自己给出。
- ② 启动机制不同：事件过程一般是因为事件的产生而自动启动，通用过程则必须由另一个过程显式调用才可以启动，通用过程有助于将复杂的应用程序分解成多个易于管理的逻辑单元，使应用程序更简洁、更易于维护。

11.7.1 Sub过程

1. 事件过程

事件过程的主要功能是对产生的事件进行响应处理，即产生某个事件后，系统会自动执行该事件对应的事件过程。事件过程有两种基本形式：窗体事件过程和控件事件过程。

（1）窗体事件

窗体事件过程的格式如下：

```
Private Sub Form_事件名 ([参数列表])  
    [局部变量和常数声明]  
    语句块  
  
End Sub
```

注意:

① 窗体事件过程名由“Form_事件名”组成,多文档窗体用“MDIForm_事件名”;

② 每个窗体事件过程名前都有一个 **Private** 的前缀,表示该事件过程仅能在自己的窗体模块内被调用;

③ 事件过程有无参数,完全由 **VB** 提供的具体事件本身决定,用户不可以随意添加。

(2) 控制事件

控件事件过程的格式如下:

Private Sub 控件名_事件名 ([参数列表])

[局部变量和常数声明]

语句块

End Sub

注意: 其中的控件名必须与窗体中某控件相匹配,否则 **VB** 将认为它是一个通用过程。

(3) 建立事件过程

建立事件过程的基本过程如下:

① 打开代码编辑器窗口(两种方法:双击对象或从工程管理器中单击“查看代码”按钮);

② 在代码编辑器窗口中,选择所需要的“对象”和“事件过程”;

③ 在 **Private Sub**...**End Sub** 之间键入代码;

④ 保存工程和窗体。

(4) 事件过程的调用

事件过程由一个发生在 **VB** 中的事件自动调用或者由同一模块中的其他过程显示调用。

2. 自定义过程

(1) 自定义过程的格式

自定义过程的格式如下:

[**Private** | **Public**] [**Static**] **Sub** 过程名 ([参数列表])

[局部变量和常数声明]

语句块

[**Exit Sub**]

语句块

End Sub

在自定义过程中,定义形参的格式为:

[**ByVal**] 变量名[()] [**As** 数据类型]

式中:

ByVal: 表明其后的形参是按值传递参数(传值参数 **Passed By Value**),若省略或用 **ByRef**,则表明参数是按地址传递的(传址参数)。

变量名[()]: 应为合法的 **VB** 变量名或数组名,无括号表示变量,有括号表示数组。

As: 数据类型。缺省表明该形参是变体型变量,若形参变量的类型声明为 **String**,则只能是不定长的。而在调用该过程时,对应的实际参数可以是定长的字符串或字符串数组,若形参是数组则无限制。

在定义自定义过程时,有以下几点需要注意:

① 省略[**Private** | **Public**]时,系统默认为 **Public**;

② Static 表示过程中的局部变量为“静态”变量;

③ 过程名的命名规则与变量的命名规则相同;

④ 参数列表中的参数称为形式参数,它可以是变量名或数组名,不能是常量、数组元素、表达式;若有多个参数时,各参数之间用逗号分隔。VB 的过程可以没有参数,但圆括号不可以省略,不含参数的过程称为无参过程;

⑤ Sub 过程不能嵌套定义,但可以嵌套调用;

⑥ 过程中可以用 Exit Sub 提前结束过程,并返回到调用该过程语句的下一条语句。

在 Access 中,打开窗体的命令是 DoCmd.OpenForm。要打开一个名为“窗体 1”的窗体,加入错误捕获后代码,则可以编写如下过程:

Sub_打开窗体(stDocName As String)	'参数 stDocName 为需要打开的窗体名称
On Error GoTo Err_打开窗体	'出错时跳转到 Err_打开窗体
DoCmd.OpenForm stDocName	'打开指定窗体
Exit_打开窗体:	
Exit Sub	
Err_打开窗体:	
MsgBox Err.Description	
Resume Exit_打开窗体	'错误处理后跳转到 Exit_打开窗体
End Sub	

注意: resume 和 return 作用一样,但是 resume 是在错误处理中。如果此时需要调用该过程打开名为“窗体 1”的窗体,只需在程序中加入代码:打开窗体(“窗体 1”)。

(2) 建立 Sub 过程的方法

建立 Sub 过程有两种基本方法如下:

【方法一】

- ① 打开代码编辑器窗口;
- ② 选择“工具”菜单中的“添加过程”;
- ③ 从对话框中输入过程名,并选择类型和范围;
- ④ 在新创建的过程中输入内容。

【方法二】

- ① 在代码编辑器窗口的对象中选择“通用”,在文本编辑区输入 Private Sub 过程名;
- ② 按回车键,便可创建一个 Sub 过程模板;
- ③ 在新创建的过程中输入内容。

3. 过程调用

(1) 用 Call 语句调用 Sub 过程

语法: Call 过程名(实际参数列表)

实际参数的个数、类型和顺序,应该与被调用过程的形式参数相匹配,有多个参数时,用逗号分隔。

(2) 把过程名作为一个语句来用

语法: 过程名[实参 1[, 实参 2.....]]

11.7.2 Function 过程

VB 包含许多内部函数,用户可以根据实际的需要调用。另外,还可用 Function 语句编写自己的 Function 过程。在定义了一个 Function 过程之后,可以像调用任何一个 VB 的内部函数一样使用它,可以在任何表达式,语句或函数中引用它。

1. Function过程的定义

(1) Function过程的格式

Function 过程的格式如下:

```
[Public|Private|Friend|Static]Function 过程名[(参数列表)][As 数据类型]
    [程序段]
    [过程名=表达式]
    [Exit Function]
    [程序段]
    [过程名=表达式]
End Function
```

说明:

① Public、Private、Friend、Static、Exit Function 的含义与 Sub 过程相同。“As 数据类型”指明由 Function 过程返回的值的的数据类型,若省略,则为 Variant。

【例 11-7】 已知直角三角形两直角边的值,定义一个 Function 函数计算斜边。

```
Function hypotenuse(A as integer, B as integer) as Single
    hypotenuse=sqr(A*A+B*B)
End function
```

② 调用 Sub 过程相当于执行一个语句,不返回值;而调用 Function 过程要返回一个值,因此可以像内部函数一样在表达式中使用。

在 VB 中调用 Function 过程的方法和调用任何内部函数的方法是一样的,如调用上面的“hypotenuse”函数可以按如下方式:

```
label1.caption=hypotenuse(Cint(text1.text), cint(text2.text))
strX=hypotenuse(width, height)
```

③ 函数过程总是以该函数的名称返回给调用函数一个值。因此,函数中的最后一行语句往往是将函数的最终计算结果放入“过程名”中的赋值语句。

(2) 建立Function过程

建立 Function 过程的方法和建立 Sub 过程的方法类似,只是当用第一种方法建立时,在对话框的“类型”栏内选择“函数”。

【例 11-8】 求圆面积。

```
Public Function A (R As Single) As Single
    On Error GoTo Err_求解出错      '出错时跳转到 Err_求解出错
    A=3.14*R*R
Exit_求解出错:
Exit Function
Err_求解出错:
    MsgBox Err.Description
    Resume Exit_求解出错          '错误处理后跳转到 Exit_求解出错
End Function
```

2. 调用函数过程

调用 Function 过程比较简单,可以像使用 VB 内部函数一样调用。因为 Function 过程返

回一个值，所以完全可以将它看成一个函数，与内部函数（如 Sqr, Str, Chr 等）没什么区别，只不过内部函数由系统提供，而 Function 过程由用户自己定义。

例如，对于例 11-8，如果要计算半径为 5 的圆的面积，则只需调用“x=A(5)”即可。

11.7.3 参数的传递

1. 形参与实参

参数有形参和实参之分，形参是指出现在 Sub 和 Function 过程形参表中的变量名、数组名。过程被调用前，没有为形参分配内存，其作用是说明自变量的类型和形态以及在过程中的角色，形参可以是除定长字符串变量之外的合法变量名，以及后面跟圆括号的数组名。实参是指在调用 Sub 或 Function 过程时，传送给相应过程的变量名、数组名、常数或表达式。在过程调用传递参数时，形参与实参是按位置结合的，形参表和实参表中对应的变量名可以不必相同，但位置必须对应。

2. 按值传递参数

按值传递参数时，是将实参变量的值复制到对应形参中，如果在调用过程中改变了形参的值，不会影响实参变量本身，即实参变量保持调用前的值不变。要实现按值传递，在定义形参时需要加上 ByVal。

3. 按地址传递参数

按地址传递参数时，把实参变量的地址传送给被调用的过程，形参和实参共用内存的同一地址。如果实参是一个常数或表达式，VB 会按“传值”方式来处理。要实现按地址传递，在定义形参时需要加上 ByRef 或者定义时不加修饰词。

4. 数组参数

VB 允许数组作为形参出现在形参表中。

语法：形参数组名 () [As 数据类型]

形参数组只能按地址传递参数，对应的实参也必须是数组，且数据类型相同。调用过程时，把要传递的数组名放在实参表中，数组名后面不跟圆括号。在过程中不可以用 Dim 语句对形参数组进行声明，否则会产生重复声明的错误。但在使用动态数组时，可以用 ReDim 语句改变形参数组的维界，重新定义数组的大小。

11.7.4 过程和变量的作用域

1. 过程的作用域

过程的作用域是指过程在程序中起作用的范围，如表 11.13 所示。

表 11.13 过程的作用域

作用范围	模块级		全局级	
	窗体	标准模块	窗体	标准模块
定义方式	过程名前加 Private 例：Private Sub my1（形参表）		过程名前加 Public 或默认 例：[Public] Sub my2（形参表）	
能否被本模块其他过程调用	能	能	能	能
能否被本应用程序其他模块调用	不能	不能	能，必须在过程名前加窗体名。 例：Call 窗体名.My1（实参表）	能，但过程名必须唯一，否则需要加标准模块名。 例：Call 标准模块名.My2（实参表）

2. 变量和常量的作用域

变量和常量不仅具有值和类型，而且还有作用域。作用域就是变量和常量在程序中起作用的范围。按变量作用域的不同可以把变量分为：过程级变量、模块级变量、全局变量以及静态变量几种；按常量作用域的不同可以把常量分为：过程级常量、模块级常量、全局常量。

(1) 过程级变量和常量

过程级变量和常量只能在过程中定义，它的作用范围只限于定义它的过程之内。也就是说，一个过程级变量或常量只能被定义它的过程使用，别的过程无权访问该变量或常量。因此，可以在不同的过程中使用同名的过程级变量，它们互不影响。当过程结束时，过程级变量所占用的内存空间就会自动释放。

(2) 模块级变量和常量

在一个模块中通常包含了多个过程。模块级的变量和常量为该模块中所有过程所共有，也就是说模块中的所有过程都可以访问属于该模块的模块级变量和常量。因此，可以利用模块级变量在过程之间进行相关数据的传递，模块级的变量和常量是在各模块的声明部分定义的。

(3) 全局变量和常量

在一个工程中通常包含多个模块，全局变量和常量是为该工程中所有模块所共有。对于全局变量和常量，工程内的任一过程都可以对其访问，全局变量和常量一般是在标准模块的声明部分定义的。

(4) 静态变量

静态变量也是在过程中定义的。它与普通的过程级变量有些类似，但又有很大的区别。静态变量在编译时为其分配存储空间，在整个程序运行期间，该变量占用固定的存储空间，程序运行结束后，这部分空间才被释放，这类变量的生存期为整个程序运行期。过程级变量在程序运行过程中，只有当变量所在函数或过程被调用时，编译系统才临时为该变量分配一段内存空间，当函数调用结束时，变量所占用的空间被释放，所以过程级变量生存期仅为函数或过程调用期间。

【例 11-9】 阅读程序，理解静态变量。

```
Private Sub Command1_click()
    Static x As Integer
    x = x + 1
    Print "x="; x
End Sub
Private Sub Command2_click()
    Dim x As Integer
    y = y + 1
    Print "y="; y
End Sub
```

若 Command1 按钮被单击 3 次，则显示结果为：

```
x=1
x=2
x=3
```

若 Command2 按钮被单击 3 次，则显示结果为：

```
y=1
y=1
y=1
```

11.8 面向对象程序设计基础知识

VBA 是面向对象的程序设计语言,掌握面向对象程序设计的基本概念的基本方法,对于使用 VBA 编写程序是非常必要的。

11.8.1 几个基本概念

面向对象的程序设计方法与传统的结构化程序设计方法有较大区别。学习面向对象的程序设计,需要读者透彻地理解对象、类、事件、方法等基本概念。

1. 对象和类

在面向对象的程序设计中,对象是一个具有属性和方法的实体。在 VBA 中,每个可以操作的实体都是对象。例如,窗体、按钮、文本框、列表框等都是常用的对象。VBA 提供了多种标准对象,可以在程序中直接使用这些对象。对象可以包含其他对象,包含其他对象的对象称为容器对象。同类型的对象具有一些共同的属性,将这些共同属性抽象出来就组成一个类,类是同类型对象的模板,基于类可以生成相同类型的任何一个对象,这些对象具有类的所有属性。

在面向对象程序设计中,对象的特性称为对象的属性,其引用方式为“对象.属性”。描述该对象特性的具体数据称为属性值,可以通过修改和设置属性值改变对象的外观和位置。有些属性值可以在程序设计时使用属性窗口来设置,有些属性则需要通过代码来修改。

Access 采用面向对象程序开发环境,Access 数据库在提供表、查询、窗体、报表、页、宏和模块七个对象的基础上,还提供了一个重要的对象: DoCmd 对象。它的主要功能是通过调用方法来实现对 Access 的操作。例如,利用 DoCmd 对象的 OPenReport 方法可以打开报表“学生信息”,其语句格式为:

DoCmd·OPenReport “学生信息”。

2. 事件

事件就是对象可以识别和响应的操作,它是预先定义的特定操作。在 VBA 中,事件可以由用户激发,也可以由程序或系统激发。比如,在 VBA 中,一个窗口上有一个“关闭”按钮,这是两个“对象”。窗口和按钮的颜色、大小是它们的属性。当按下“关闭”按钮时,发生关闭事件,窗口关闭。大部分事件不需要用户干预,Access 按默认的方法处理事件。为了提高系统的功能,可为少量的事件定义自己的处理方法,定义宏或编写 VBA 代码。

3. 方法

方法就是事件发生时对象执行的操作。方法与对象紧密联系。VBA 中的方法是事件发生时执行的一段内部程序,即事件过程代码。这些代码指定事件发生时对象需要完成的操作。VBA 提供了许多常用的方法,可以直接调用这些方法,但有些方法则需要编写代码来实现。

4. 事件驱动模型

在结构化程序中,代码的执行次序由应用程序自身控制。而在事件驱动系统中,事件的触发顺序决定了代码执行的顺序。当某个事件发生后,系统会找到并运行相应的事件处理程序,从而对该事件做出响应。因为事件的出现顺序是不确定的,所以在代码中必须对每个需要处理的事件组织其事件处理程序。在响应事件时,由事件来驱动应用程序执行相应的事件

处理程序。VB 中的每一个对象都有一个预定义的事件集，如果其中一个事件发生，并且在关联的事件过程中存在代码，那么 VB 就会自动调用该代码，当事件处理程序运行后，系统处于等待状态，当另一个事件发生后，程序就去执行该事件的事件过程，当这个事件执行完后，系统又处于等待另一个事件发生的状态，这就是事件驱动程序方式。

11.8.2 VBA程序设计的基本过程

在 VB 程序设计中，基本的设计机制是：改变对象的属性，使用对象的方法，为对象事件编写事件过程。程序设计时要做的工作就是决定应更改哪些属性，调用哪些方法，对哪些事件进行响应，从而得到希望的结果。VBA 充分体现了面向对象程序设计的特点。使用 VBA 开发应用程序的基本过程可概括为：

设计应用程序的界面→设置对象的属性值→编写事件过程代码→运行调试程序→保存程序。

使用 VBA 开发应用程序的基本方法并不是一成不变的。可以添加一个控件后，紧接着就设置控件的属性值和编写事件过程代码，再添加下一个控件，并做相同的操作。不过，按照上述基本方法操作，条理性和规范性要强一些。VBA 还可以将应用程序编译成可执行文件，并且编译生成的可执行文件可以脱离 VBA 开发环境直接在 Windows 系统中运行。

【例 11-10】 编写一个程序，指定在编辑一条记录的数据结束后将记录指针指向下一条记录时，询问用户编辑的结果是否正确，并根据用户的回答进行相应的操作。基本过程如下：

- (1) 新建VBA程序。
 - ① 在数据库窗口中选择“模块”对象，然后单出“新建”按钮，打开“学生管理-模块”代码编辑器窗口，如图 11.9 所示。
 - ② 在数据库窗口菜单中选择“插入”→“过程”命令，打开“添加过程”对话框，如图 11.10 所示，设置过程名称和其他选项。



图 11.9 代码窗口

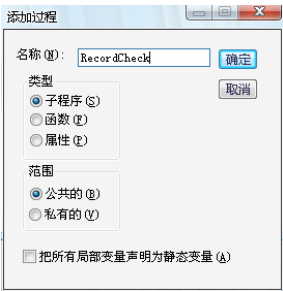


图 11.10 添加过程

- ③ 在图 11.10 中选择“确定”按钮，返回代码编辑器窗口，如图 11.11 所示。
 - ④ 在代码编辑器窗口中输入如下程序代码，如图 11.12 所示。
- 代码如下：

```
Public Sub RecordCheck()  
    Dim Result As Integer  
    Result=MsgBox ("输入数据是否正确", VbOKCancel+ VbInformation, "验证")  
    If Result=1 then  
        DoCmd.GotoRecord , acNext
```

```
Else
    MsgBox ("请修改错误")
End if
End Sub
```

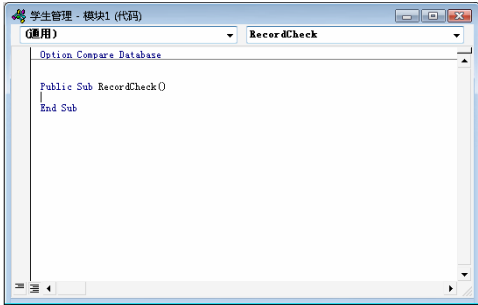


图 11.11 代码窗口

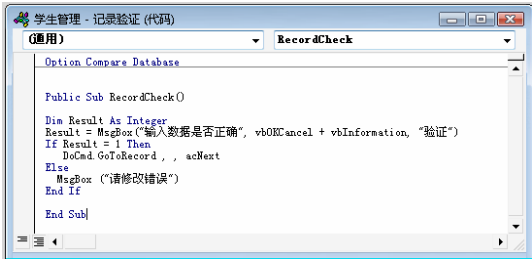


图 11.12 输入代码

⑤ 单击工具栏上的“保存”按钮，弹出“另存为”对话框，输入模块名称，保存程序，如图11.13所示。

⑥ 关闭代码编辑器窗口，返回数据库窗口，新建的程序模块如图11.14所示。

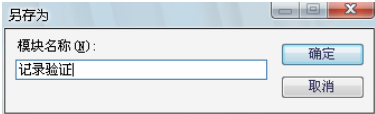


图 11.13 保存模块

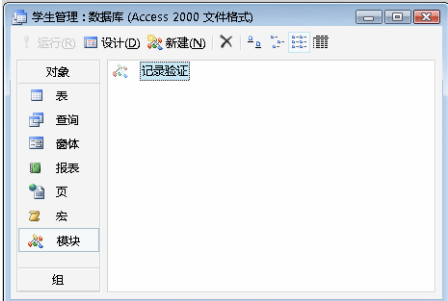


图 11.14 数据库窗口

(2) 指定“下一个”命令按钮运行该程序

① 打开“学生表”窗体。创建一个命令按钮，如图11.15所示。

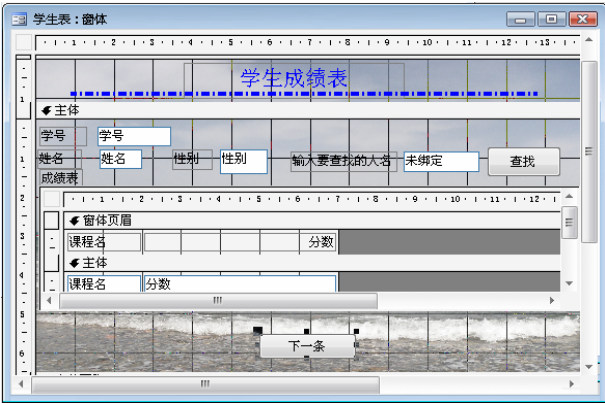


图 11.15 添加命令按钮

② 打开“属性”窗口，按钮名称设置为“下一条”，设置“单击”事件属性为“事件过程”，如图11.16所示。

③ 单击“事件过程”右边的对话按钮，打开代码编辑窗口，输入代码，如图 11.17 所示。



图 11.16 设置事件属性

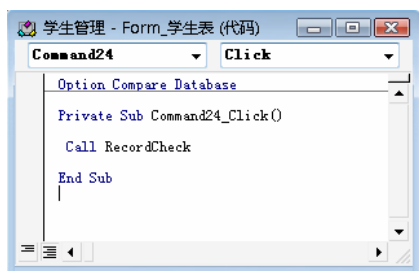


图 11.17 代码窗口

④ 关闭代码编辑器窗口。

⑤ 运行窗体，结果如图 11.18 所示。编辑数据后，单击“下一条”按钮，将显示如图 11.19 所示对话框。



图 11.18 运行结果



图 11.19 验证对话框

⑥ 如果选择“确定”按钮，则继续编辑下一条记录的数据，如图 11.20 所示。如果选择“取消”按钮，将显示如图 11.21 所示对话框。



图 11.20 添加新记录

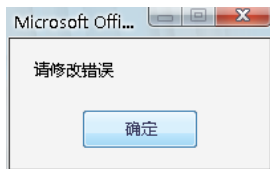


图 11.21 求该确认

⑦ 选择“确定”按钮，可以继续编辑当前记录的数据。

11.9 应用举例

【例 11-11】 打开、关闭窗体。

窗体操作在 VBA 是很重要的，程序中往往不止一个窗体。将大量的窗体在程序中用代码关联起来，程序运行的时候才会有条不紊。窗体操作有两个很重要的命令：DoCmd.OpenForm（打开窗体）和 DoCmd.Close（关闭窗体）。新建一个窗体，放置一个名为“指定窗体”的标签、一个名为“txt 窗体”的文本框，两个命令按钮，名称分别是“com 打开”、“com 关闭”，其标题属性分别为“打开”、“关闭”。然后在窗体模块中写入以下代码：

```
Option Compare Database
Public forName As String
Public Sub 打开窗体(stDocName As String)
    On Error GoTo Err_打开窗体
    Dim stLinkCriteria As String
    DoCmd.openform stDocName, , stLinkCriteria
Exit_打开窗体:
    Exit Sub
Err_打开窗体:
    MsgBox Err.Description
    Resume Exit_打开窗体
End Sub
Public Sub 关闭窗体(stDocName As String)
    On Error GoTo Err_关闭窗体      '关闭指定窗体并保存该窗体
    DoCmd.Close acForm, stDocName, acSaveYes
Exit_关闭窗体:
    Exit Sub
Err_关闭窗体:
    MsgBox Err.Description
    Resume Exit_关闭窗体
End Sub
Sub com 打开_Click()                '单击“打开”按钮所引发的事件
    forName = Me.txt 窗体          '将“txt 窗体”的内容赋给窗体名称变量
    打开窗体 (forName)
End Sub
Sub com 关闭_Click()              '单击“关闭”按钮所产生的事件
    forName = Me.txt 窗体          '将“txt 窗体”的内容赋给窗体名称变量
    关闭窗体(forName)
End Sub
```

运行该窗体，结果如图 11.22 所示。

如果要打开一个名为“成绩显示”的窗体，则在文本框中键入“成绩显示”。单击“打开”按钮，如果数据库项目中存在该窗体，则“成绩显示”窗体被打开。

【例 11-12】 新建表、删除表。

虽然我们可以在设计程序时对数据表进行新建和删除等操作，但在运行程序时或程序编译后却不能通过主设计窗体对其进行上述两个动作，但通过 VBA 很容易实现一些临时表的新建和删除工作。新建一个窗体：放置三个控件：文本框“txt 表名”、命令按钮“com 新建”和命令按钮“com 删除”，需要在其中实现表的新建和删除。



图 11.22 运行结果

窗体模块中代码如下：

```
Option Compare Database
Dim strName As String
Private Sub com 删除_Click()
    strName = Me.txt 表名
    On Error GoTo 删除表_Err
    DoCmd.DeleteObject acTable, strName
    删除表_Exit:
Exit Sub
    删除表_Err:
    MsgBox strName & "表不存在或已被删除"
    Resume 删除表_Exit
End Sub
Private Sub com 新建_Click()
    strName = Me.txt 表名
    On Error GoTo 新建表_Err
    Dim db As Database '声明数据库变量
    Dim tb As New TableDef '声明新表变量
    Dim fldName As New Field '声明新字段（姓名）变量
    Dim fldSex As New Field '声明新字段（性别）变量
    Set db = CurrentDb()
    tb.Name = strName
    fldName.Name = "姓名"
    fldName.Type = dbText
    tb.Fields.Append fldName
    fldSex.Name = "性别"
    fldSex.Type = dbText
    tb.Fields.Append fldSex
    db.TableDefs.Append tb
    新建表_Exit:
Exit Sub
    新建表_Err:
    MsgBox strName & "表已存在" '如果该表已存在，则提示错误
    Resume 新建表_Exit
End Sub
```

新建该表时，如果该表已存在，程序则会提示，并不再新建；删除该表时，如果该表不存在，程序会提示错误。

【例 11-13】 计算个人所得税。

个人所得税的调整可能越来越频繁，为了避免每次调整都去改程序源代码，编写程序实现计算个人所得税。当计税方式发生变化时，只需要在“个人所得税比率”表中修改比率，个人所得税比率表如表 11.14 所示。在“起征点”表中修改起征点就可了，起征总表如表 11.5 所示。

调用格式：所得税=Call 所得税（应纳税所得）。

表 11.14 个人所得税比率

应纳税所得 下限额(元)	纳税比率 (%)	速算扣除数 (元)
0	5	0
500	10	25
2 000	15	125
5 000	20	375
20 000	25	1 375
40 000	30	3 375
60 000	35	6 375
80 000	40	10 375
100 000	45	15 375

表 11.15 起征总表

起征点
2000

应纳税所得是指个人总收入除去个人承担的基本养老保险金、医疗保险金、失业保险金，以及按省级政府规定标准缴纳的住房公积金等。

新建一个窗体：放置五个控件：标签“税前工资”、标签“应纳税所得”、文本框“应纳税额”、文本框“所得税”、命令按钮“com 计算”，设计界面如图 11.23 所示。计算时，在应纳税额文本框中输入应纳税的金额，单击“计算”按钮，完成所得税计算，计算结果在所得税文本框中显示。

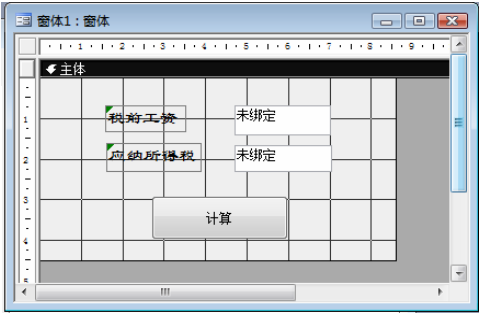


图 11.23 运行结果

模块中代码如下：

```
Public Function Call 所得税(ByVal 应纳税所得额 As Double) As Double
Dim rs As ADODB.Recordset
Set rs = New ADODB.Recordset
rs.ActiveConnection = CurrentProject.Connection
rs.CursorType = adOpenStatic
rs.LockType = adLockOptimistic
rs.Open "Select * FROM 个人所得税比率表 ORDER BY 应纳税所得下限额 DESC"
Dim rsQZD As ADODB.Recordset
Set rsQZD = New ADODB.Recordset
rsQZD.ActiveConnection = CurrentProject.Connection
rsQZD.CursorType = adOpenStatic
rsQZD.LockType = adLockOptimistic
rsQZD.Open "Select 起征点 FROM 起征点表"
Dim temp As Double
temp = 应纳税所得额 - rsQZD!起征点
Do Until rs.EOF
    If temp > rs!应纳税所得下限额 Then
        Call 所得税 = temp * rs!纳税比率 / 100 - rs!速算扣除数
        Exit Function
    Else
        rs.MoveNext
    End If
Loop
End Function

Private Sub com 计算_Click()
Dim a As Double
应纳税额.SetFocus
a = Val (应纳税额.Text)
所得税.SetFocus
所得税.Text = Call 所得税(a)
End Sub
```

【例 11-14】 增加工资。

数据库中有工资表，包括“姓名”、“工资”和“职称”等字段，现要对不同职称的职工增加工资，规定教授职称增加 15%，副教授职称增加 10%，其他人员增加 5%。按照上述规定编程调整每位职工的工资，并显示所涨工资的总和。

程序代码如下：

```
Private Sub Command1_Click()
    Dim ws As DAO.Workspace
    Dim db As DAO.Database
    Dim rs As DAO.Recordset
    Dim gz As DAO.Field
    Dim zc As DAO.Field
    Dim sum As Currency
    Dim rate As Single
    Set db = CurrentDb()
    Set rs = db.OpenRecordset ("工资表")
    Set gz = rs.Fields ("工资")
    Set zc = rs.Fields ("职称")
    sum = 0
    Do While Not rs.eof
        rs.Edit
        Select Case zc
            Case Is = "教授"
                rate = 0.15
            Case Is = "副教授"
                rate = 0.1
            Case Else
                rate = 0.05
        End Select
        sum = sum + gz * rate
        gz = gz + gz * rate
        rs.update
        rs.MoveNext
    Loop
    rs.Close
    db.Close
    Set rs = Nothing
    Set db = Nothing
    MsgBox "涨工资总计:" & sum
End Sub
```

【例 11-15】 新建一个窗体。

新建一个窗体，其中放置两个按钮和一个文本框控件。按钮的功能分别定义为“com 显示”、“com 清除”；按钮的标题分别定义为“显示”、“清除”。文本框的名称定义为“txt 你好”。

窗体模块窗体中代码如下：

Option Compare Database	
Private Sub com 清除_Click()	'单击“清除”按钮事件
Me.txt 你好.SetFocus	'SetFous 方法，使文本框具有焦点
Me.txt 你好.Text = ""	'使文本框的 Text 属性为""（空）
End Sub	'结束过程
Private Sub com 显示_Click()	'单击显示按钮事件
Me.txt 你好.SetFocus	'SetFous 方法，使文本框具有焦点
Me.txt 你好.Text = "你好！ 欢迎学习 ACCESS VBA"	
End Sub	'结束过程
Private Sub Form_Load()	'窗体加载事件
Me.Caption = "欢迎！"	'设使窗体的 Caption（标题）属性为“欢迎！”
End Sub	

运行该窗体，结果如图11.24所示。

【例 11-16】 新建消息框。

友好的人机界面会大大提高用户对程序的好感，使程序变得易用，用 VBA 也可以实现良好的人机界面，其中很重要的就是消息框。消息框可以在程序运行中告诉用户下一步该做什么，不该做什么，以及哪里出了错误，该如何处理等。新建一个窗体，该窗体中放上三个控件：文本框“txt 姓名”、文本框“txt 性别”和命令按钮“com 加入”。

窗体模块中代码如下：

```
Option Compare Database
Private Sub com_加入_Click()           '声明 Name、Sex 变量，分别表示姓名，性别
Dim Name As Variant, Sex As Variant
'声明消息框函数将要用到的三个变量，分别代表消息内容、消息框标题和消息常数
Dim Msg, Title, Response
Name = Me.txt_姓名
Sex = Me.txt_性别
If IsNull(Name) = True Then
    Beep                               '发出“嘟”的警告声
    Msg = ""姓名"不能为空！"
    Style = vbOKOnly                   '定义按钮
    Title = "输入错误"                 '定义标题
    Response = MsgBox(Msg, , Title)
End If
If IsNull(Sex) = True Then
    Beep
    Msg = ""性别"不能为空！"
    Style = vbOKOnly
    Title = "输入错误"
    Response = MsgBox(Msg, , Title)
ElseIf Sex <> "男" Or Sex <> "女" Then
    Beep
    Msg = "“性别”只能“男”或“女”！"
    Style = vbOKOnly
    Title = "输入错误"
    Response = MsgBox(Msg, , Title)
End If
End Sub
```

程序中用到了 MsgBox() 函数，其作用是，运行窗体时，若没输入内容而单击“加入”按钮，会出现出错提示。

【例 11-17】 在 Access 中调用 Windows 系统自带的应用程序。

在程序中，有时需要用到其他已编制好的程序。比如在一个图片管理数据库中也许要用到 Windows 绘图程序来修改图片，也许还要用到 ACDSee 来察看图片。我们不可能也不必要将这些程序逐个开发出来，可以利用 Shell 语句直接运行这些程序即可。下面程序说明如何在 Access 中来调用 Windows 自带的应用程序——计算器。

```
Public Sub jsq()
On Error GoTo Err_计算器
```

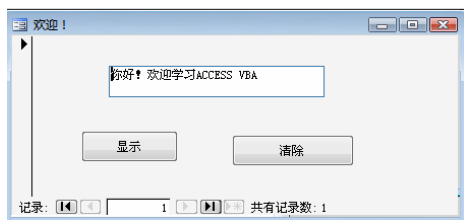


图 11.24 运行结果

```

Dim windowMe
windowMe = Shell("C:\Windows\system32\CALC.EXE", 1)
Exit_计算器:
Exit Sub
Err_计算器:
    MsgBox Err.Description
    Resume Exit_计算器
End Sub

```

这个过程代码很少，其中用到了一个 Shell（）函数，该函数的语法为：

```
Shell(pathname[,windowstyle])
```

Pathname 是必须的参数，它指明要运行的程序。可以在其中指定程序的完整路径，本过程中，计算器程序的路径是“C:\Windows\system32\CALC.EXE”。windowstyle 是可选的参数。表示在程序运行时的窗口样式。其中，1 表示窗口具有焦点；3 表示窗口具有焦点且最大化。

习 题 11

一、填空题

1. VBA 的三种流程控制结构是顺序结构、_____和_____。
2. 在下面窗体文本框 Text1 中输入“456AbC”后，立即窗口上输出的结果是_____。

```

Private Sub Text1_KeyPress(KeyAscii As Integer)
    Select Case DeyAscii
    Case 97 To 122
        Debug.Print Ucase(Chr(KeyAscii));
    Case 65 To 90
        Debug.Print Lcase(Chr(KeyAscii));
    Case 48 To 57
        Debug.Print Chr(KeyAscii);
    Case Else
        KeyAscii = 0
    End Select
End Sub

```

3. 在窗体上有一个命令按钮 Command1，编写事件代码如下：

```

Private Sub Command1_Click()
    Dim a(10), p(3) As Integer
    k = 5
    For i = 1 To 10
        a(i) = i * i
    Next i
    For i = 1 To 3
        p(i) = a(i * i)
    Next i
    For i = 1 To 3
        k = k + p(i) * 2
    Next i
    MsgBox k
End Sub

```

打开窗体运行后，单击命令按钮，消息框中输出的结果是_____。

4. 下列程序的功能是找出被 5、7 除, 余数为 1 的最小的 5 个正整数。请在程序空白处填入适当的语句, 使程序可以完成指定的功能。

```
Private Sub Form_Click()  
    Dim Ncount %, n%  
    n = n + 1  
    If _____ Then  
        Debug.Print n  
        Ncount = Ncount + 1  
    End If  
    Loop Until Ncount = 5  
End Sub
```

5. 以下程序的功能是在立即窗口中输出 100~200 之间所有的素数, 并统计输出素数的个数。请在程序空白处填入适当的语句, 使程序可以完成指定的功能。

```
Private Sub Command2_Click()  
    Dim i%, j%, k%, t %           't 为统计素数的个数  
    Dim b As Boolean  
    For i = 100 To 200  
        b = True  
        k = 2  
        j = Int(Sqr(i))  
        Do While k <= j And b  
            If i Mod k = 0 Then  
                b = _____  
            End If  
            k = _____  
        Loop  
        If b = True Then  
            t = t + 1  
            Debug.Print i  
        End If  
    Next i  
    Debug.Print "t="; t  
End Sub
```

二、选择题

- 下列窗口中可以直接运行临时的编写语句的是 ()。
A. 立即窗口 B. 本地窗口 C. 监视窗口 D. 快速监视窗口
- 在 VBA 中, 下列变量名中不合法的是 ()。
A. 你好 B. ni hao C. Nihao D. ni—hao
- VBA 中定义静态变量可以用关键字 ()。
A. Const B. Dim C. Public D. Static
- 一般情况下, 数组 a(3, 4, 5) 包含的元素个数为 ()。
A. 345 B. 12 C. 120 D. 60
- 下面 () 不是 Access 中的运算符。
A. 关系运算符 B. 逻辑运算符 C. 算术运算符 D. 特殊运算符
- 对于 Access 的系统常量, 编码时 ()。
A. 要先声明才可以使用 B. 可以直接使用
C. 有些可以直接使用, 有些则必须要先声明 D. 可以使用后再声明

7. 对于算术运算符“/”和“\”的区别是（ ）。
- A. 前者是整数除法，后者是普通除法 B. 前者是普通除，后者是整数除法
- C. 前者是除法，后者是分数形式 D. 两者没有本质的区别
8. 对于符号常量的类型，下列说法中正确的是（ ）。
- A. 必须指明数据类型
- B. 不需要指明数据类型，VBA 会自动按存储效率最高的方式来确定其数据类型
- C. 不需要指明数据类型，因为常量本身没有数据类型
- D. 指明不指明均可
9. 在调试 VBA 程序时，能自动被检查出来的错误是（ ）。
- A. 语法错误 B. 逻辑错误 C. 运行错误 D. 语法错误和逻辑错误
10. 在已建窗体中有一命令按钮（名为 Command1），该按钮的单击事件对应的 VBA 代码为：

```
Private Sub Command1_Click()
    subT. Form. RecordSource = "select * from 雇员"
End Sub
```

单击该按钮实现的功能是（ ）。

- A. 使用 select 命令查找“雇员”表中的所有记录
- B. 使用 select 命令查找并显示“雇员”表中的所有记录
- C. 将 subT 窗体的数据来源设置为一个字符串
- D. 将 subT 窗体的数据来源设置为“雇员”表
11. 由“For i=1 To 9 Step 3”决定的循环结构，其循环体将被执行（ ）。
- A. 0 次 B. 1 次 C. 4 次 D. 5 次
12. 在窗体上有一个命令按钮 Command1 和一个文本框 Text1，编写事件代码如下：

```
Private Sub Command1_Click()
    Dim i,j,x
    For i = 1 To 20 step 2
        x = 0
        For j = To 20 step 3
            x = x + 1
        Next j
    Next i
    Text1. Value=Str(x)
End Sub
```

打开窗体运行后，单击命令按钮，文本框中显示的结果是（ ）。

- A. 1 B. 7 C. 17 D. 400
13. 在窗体上有一个命令按钮 Command1，编写事件代码如下：

```
Private Sub Command1_Click()
    Dim y As Integer
    y = 0
    Do
        y = InputBox("y=")
        If (y Mod 10) + Int(y / 10) = 10 Then Debug. Print y;
    Loop Until y = 0
End Sub
```

打开窗体运行后，单击命令按钮，依次输入 10、37、50、55、64、20、28、19、-19、0
立即窗口上输出的结果是（ ）。

- A. 37 55 64 28 19 19 B. 10 50 20
C. 10 50 20 0 D. 37 55 64 28 19

14. 在窗体上有一个命令按钮 Command1，编写事件代码如下：

```
Private Sub Command1_Click()  
    Dim x As Integer, y As Integer  
    x = 12: y = 32  
    Call Proc(x, y)  
    Debug. Print x; y  
End Sub  
Public Sub Proc(n As Integer, ByVal m As Integer)  
    n = n Mod 10  
    m = m Mod 10  
End Sub
```

打开窗体运行后，单击命令按钮，立即窗口上输出的结果是（ ）。

- A. 2 32 B. 12 3 C. 2 2 D. 12 32

15. 在窗体上有一个命令按钮 Command1，编写事件代码如下：

```
Private Sub Command1_Click()  
    Dim d1 As Date  
    Dim d2 As Date  
    d1 = #12/25/2009#  
    d2 = #1/5/2010#  
    MsgBox DateDiff("ww", d1, d2)  
End Sub
```

打开窗体运行后，单击命令按钮，消息框中输出的结果是（ ）。

- A. 1 B. 2 C. 10 D. 11

16. 执行语句“MsgBox “AAAA”, vbOKCancel+vbQuestion, “BBBB””之后，弹出的信息框（ ）。

- A. 标题为“BBBB”、框内提示符为“惊叹号”、提示内容为“AAAA”
B. 标题为“AAAA”、框内提示符为“惊叹号”、提示内容为“BBBB”
C. 标题为“BBBB”、框内提示符为“问号”、提示内容为“AAAA”
D. 标题为“AAAA”、框内提示符为“问号”、提示内容为“BBBB”

17. 以下关于类模块的说法不正确的是（ ）。

- A. 窗体模块和报表模块都属于类模块，它们从属于各自的窗体或报表
B. 窗口模块和报表模块具有局部特性，其作用范围局限在所属窗体或报表内部
C. 窗体模块和报表模块中的过程可以调用标准模块中已经定义好的过程
D. 窗口模块和报表模块生命周期是伴随着应用程序的打开而开始、关闭而结束

18. 阅读下面的 VBA 程序段，执行后 K 的值为（ ）。

```
K=0  
for I=1 to 3  
for J=1 to I  
K=K+J  
Next J  
Next I
```

- A. 8 B. 10 C. 14 D. 21

19. VBA 数据类型符号 “%” 表示的数据类型 ()。

- A. 整型 B. 长整型 C. 单精度型 D. 双精度型

20. 在 Access 中, 如果变量定义在模块的过程内部, 当过程代码执行时才可见, 则这种变量的作用域为 ()。

- A. 程序范围 B. 全局范围 C. 模块范围 D. 局部范围

三、操作题

1. 编写一个对输入字符进行转换的程序。转换规则为, 将其中的大写字母转换成小写字母, 小写字母转换成大写字母, 空格不转换, 其余转换成 “*”。

2. 编写账号和密码输入的检验程序。对输入的账号和密码规定如下: 账号不超过 6 位数字, 密码 4 位字符, 在本题中, 密码假定为 “Dong”。密码输入时在屏幕上不显示输入的字符, 而以 “*” 代替。当输入不正确, 如账号为非数字字符, 密码不正确等, 显示 “输入错误”; 若单击 “重试” 按钮, 则清除原输入的内容, 焦点定位在原输入的文本框, 等待再次输入; 若单击 “取消” 按钮, 则停止程序的运行。

3. 编写加密和解密程序, 即将输入的一行字符串的所有字母加密, 加密后还可再进行解密。

4. 用随机函数生成 20 个两位数在立即窗口中输出, 并求出其中的最小数, 指出它是第几个数。

5. 圆周率 π 可以用级数公式 $(\pi^2)/6=1+1/4+1/9+1/16+1/5^2+\cdots+1/n^2$ 求得, 编程计算当 $n=20$ 时 π 的近似值, 要求使用一个函数过程求整数的平方, 另一个函数过程求解整数的倒数。

第 12 章 综合实例

图书馆管理是实际中最常见的应用之一，本章通过使用工程化思想开发一个图书馆管理系统来说明如何使用 Access 实现数据管理。

12.1 需求分析

在进行一个项目的设计之前，先要进行必要的需求分析。本例中，某图书馆需要管理图书信息，希望实现办公的信息化，通过建立一个图书馆管理系统来管理图书。其基本功能包括图书的入库管理、读者的借阅管理、相关的查询，以及有关读者信息的管理等。

12.2 系统设计

数据库系统的主要功能是进行信息管理，所以，首先需要搞清楚所要存储的信息，以及它们之间的关系。在实际工作中，常见的描述工具是 E-R 图，如图 12.1 所示是图书馆管理系统的实体联系图。

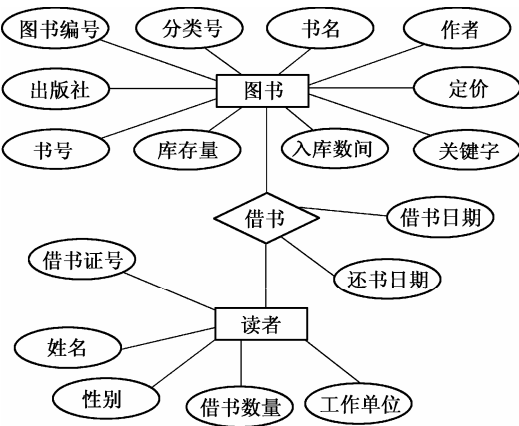


图 12.1 系统的实体联系图

Access 支持关系模型，所以需要将 E-R 图转换为关系模式。根据转换规则，初步得到图书、读者、借书三个关系模式，具体如下：

- ① 图书（图书编号，分类号，书名，作者，出版社，定价，书号，关键字，入库时间，库存量）。
- ② 读者（借书证号，姓名，性别，借书数量，工作单位）。
- ③ 借书（借书证号，图书编号，借书日期，还书日期）。

图书关系模式中“图书编号”作为关键字。
读者关系模式中“借书证号”作为关键字。
借书关系模式中“图书编号”和“借书证号”的组合作为关键字。
对应关系模式的表结构分别如表 12.1、表 12.2、表 12.3 所示。

表 12.1 图书的表结构

字段名	类型	字段长度（字节）	约束
图书编号	文本	30	主键
分类号	文本	30	
书名	文本	200	
作者	文本	150	
出版社	文本	200	
定价	货币		
书号	文本	30	
关键字	文本	100	
入库时间	时间/日期		
库存量	数字（整型）		

表 12.2 读者的表结构

字段名	类型	字段长度（字节）	约束
借书证号	文本	30	主键
姓名	文本	150	
性别	文本	2	
借书数量	数字（整型）		
工作单位	文本	200	

表 12.3 借书的表结构

字段名	类型	字段长度（字节）	约束
借书证号	文本	30	主键
图书编号	文本	30	
借书日期	日期/时间		
还书日期	日期/时间		

系统的功能模块组成如图12.2所示。

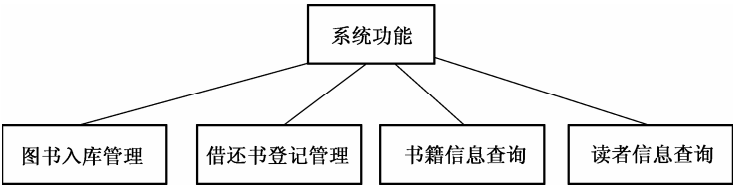


图 12.2 系统功能结构图

软件是否方便使用很大程度上取决于界面设计是否友好合理，根据系统的要求主要界面设计如图12.3、图12.4、图12.5、图12.6所示。

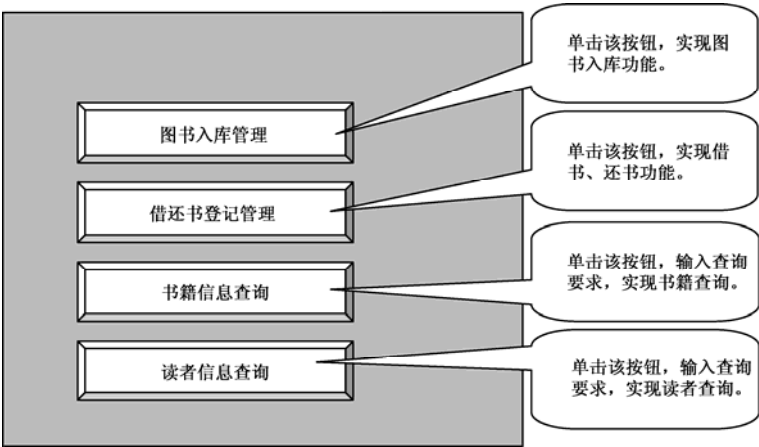


图 12.3 系统主界面

图书编号
作者
书名
库存量

借书证号

借出日期

书籍查询信息

单击该按钮，可以查看书籍的详细信息。
书籍信息

图 12.4 书籍信息查询界面

图书编号
分类号
书名
作者
出版社
定价
书号
库存量
关键字
入库时间

图 12.5 书籍详细信息界面

借书证号	姓名	性别	工作单位

图 12.6 读者信息界面

12.3 系统实现

12.3.1 创建数据库

创建一个空的数据库，在数据库中创建三张表，三张表分别按表 12.1、表 12.2、表 12.3 所示的结构设计，并建立表间关系，结果如图 12.7 所示。

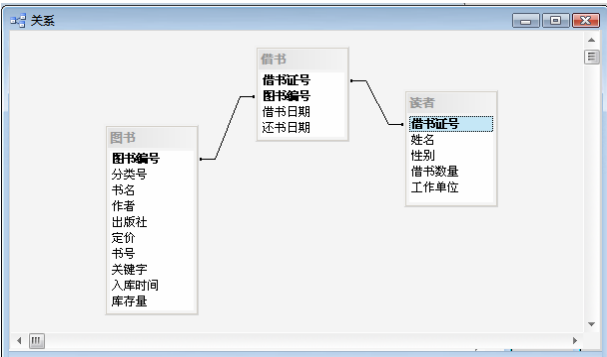


图 12.7 表结构及表间关系

12.3.2 查询设计

1. 读者借阅情况查询

读者借阅情况查询用于查询书籍信息，使用简单查询向导创建查询，并给查询取名“读者借阅情况”，其创建结果如图 12.8 所示。

2. 借书查询

通过设计视图创建查询，并给查询取名“书籍借阅情况”，其创建结果如图 12.9 所示。



图 12.8 “读者借阅情况”窗口



图 12.9 “书籍借阅情况”查询的设计视图

12.3.3 报表设计

通过设计视图创建书籍信息查询结果报表，用于显示书籍查询结果，数据源选择“读者借阅情况”查询，并对其进行美化，在页眉处添加标题，并画直线，在页脚插入页码，结果如图12.10所示。

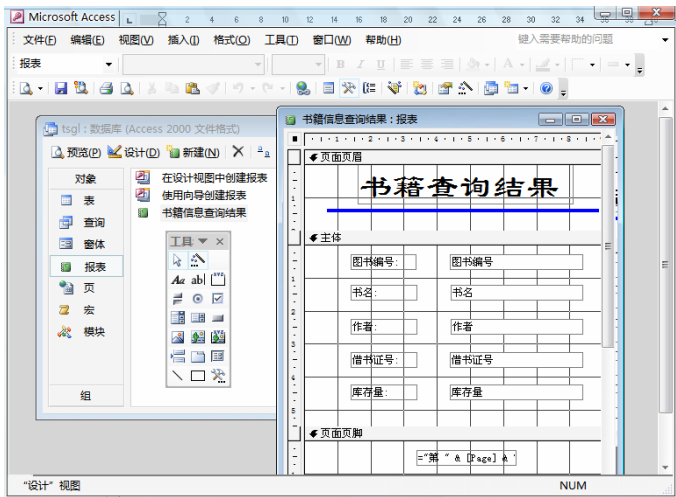


图 12.10 “书籍信息查询结果”报表的设计视图

12.3.4 窗体设计

1. 图书入库管理窗体设计

单击图书馆管理系统主界面上的“图书入库管理”按钮将弹出相应的图书入库管理窗体，用于录入新入库的图书。

使用向导创建该窗体，相关参数设置为：数据来源：“图书”表；布局：“纵栏式”；样式：“水墨画”；窗体命名“图书 1”，结果如图 12.11 所示。

2. 借还书登记管理窗体设计

单击图书馆管理系统主界面上的“借还书登记管理”按钮将弹出相应地借还书登记管理窗体，用于借书和还书的登记。

使用向导创建该窗体，相关参数设置为：数据来源：“借书”表和“图书”表；布局：“表格”；样式：“水墨画”；窗体命名“借还书查询”，结果如图 12.12 所示。

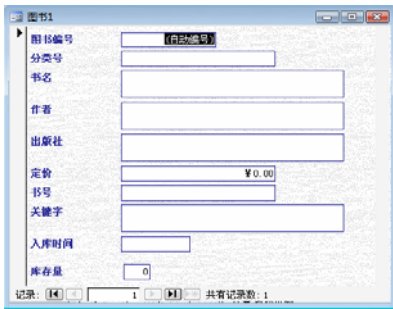


图 12.11 “图书 1”窗体效果



图 12.12 “借还书查询”窗体效果

借书时，在借书子窗体填写借阅者的借书证号和借书日期，同时修改库存量（减 1）；还书时，在借书子窗体填写借阅者的还书日期，同时修改库存量（加 1）。

3. 书籍信息查询窗体设计

单击图书馆管理系统主界面上的“书籍信息查询”按钮将弹出相应地书籍信息查询窗体，用于显示书籍的借阅情况和详细信息。该窗体的创建过程如下。

① 创建书籍信息查询窗体

使用向导创建，相关参数设置为：数据来源：“书籍借阅情况”查询；查看数据方式：“通过图书”，选择“带有子窗体的窗体”；布局：“表格”；样式：“水墨画”；新建窗体命名“书籍信息查询”，子窗体命名“书籍信息查询子窗体”，设置效果如图 12.13 所示。

② 在图 12.12 基础上添加“详细信息”按钮，单击该按钮可以查看书籍的详细信息。切换到设计视图，添加命令按钮 Command，属性设置为：标题：“详细信息”；字体：“宋体”；字号：“10”；字体粗细：“加粗”。

右键单击“详细信息”按钮，在快捷菜单上选择“事件生成器”，在弹出的对话框中选择“代码生成器”，在弹出的 VBA 设计窗口中添加代码，代码如下：

```
Private Sub Command_Click()  
    DoCmd.OpenForm "图书 1"  
End Sub
```

③ 在图 12.13 基础上添加两个按钮：预览按钮和打印按钮，用来操作报表。预览按钮用于预览报表，在命令按钮向导对话框中设置如下属性：“类别”选择：报表操作；“操作”选择：预览报表。

打印按钮用于打印报表，在命令按钮向导对话框中设置如下属性：“类别”选择：报表操作；“操作”选择：打印报表。最终的设置效果如图 12.14 所示。

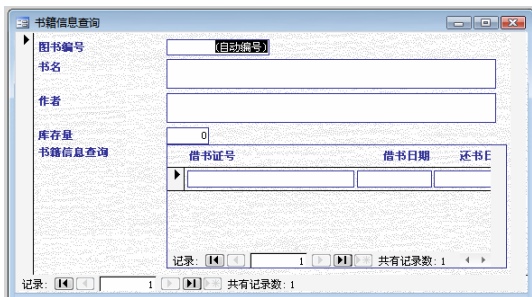


图 12.13 “书籍信息查询”窗体效果



图 12.14 “书籍信息查询”窗体效果

4. 读者信息查询窗体设计

单击图书馆管理系统主界面上的“读者信息查询”按钮将弹出相应地读者信息查询窗体，用于显示某个读者的图书借阅情况。

使用向导创建该窗体，相关参数设置为：数据来源：“读者”表、“借书”表和“图书”表；布局：“表格”；样式：“水墨画”；窗体命名“读者借书查询”，结果如图12.15所示。

5. 图书馆管理系统主窗体设计

应用程序主界面就是图书馆管理系统主窗体，如图12.16所示，其创建方法是通过设计视图创建。窗体属性设置如下：标题：“图书管理信息系统”；滚动条：“两者均无”；记录器选择：“否”；导航按钮：“否”；分隔线：“否”；边框样式：“对话框边框”；图片缩放模式：“拉伸”。



图 12.15 “读者借书查询”窗体效果

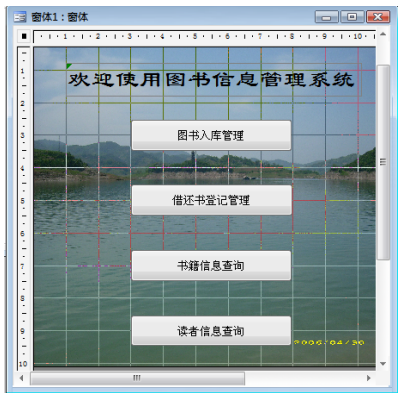


图 12.16 主窗体设置效果

为窗体添加背景图像，在窗体上添加标签，内容为：“欢迎使用图书管理信息系统”。

在窗体上建立 4 个命令按钮：Command0、Command1、Command2、Command3。

Command0 的属性设置如下：标题：“图书入库管理”；字体：“宋体”；字号：“10”；字体粗细：“正常”。

Command1 的属性设置如下：标题：“借还书登记管理”；字体：“宋体”；字号：“10”；字体粗细：“正常”。

Command2 的属性设置如下：标题：“书籍信息查询”；字体：“宋体”；字号：“10”；字体粗细：“正常”。

Command3 的属性设置如下：标题：“读者信息查询”；字体：“宋体”；字号：“10”；字体粗细：“正常”。

主窗体中右键单击“图书入库管理”按钮，在弹出的快捷菜单中选择“事件生成器”，再在弹出的对话框中选择“宏生成器”，宏取名为“图书入库”，如图12.17所示。宏包含两个操作，第一个操作选择：Hourglass，第二个操作选择：OpenForm，其设置如图12.17所示。

另外三个命令按钮的设置方法与“图书入库管理”按钮的设置方法类似。其中“借还书登记管理”按钮的设置如图12.18所示。“书籍信息查询”按钮的设置如图12.19所示。“读者信息查询”按钮的设置如图12.20所示。

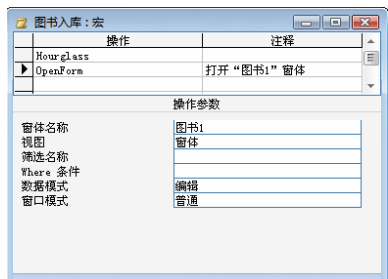


图 12.17 图书入库宏

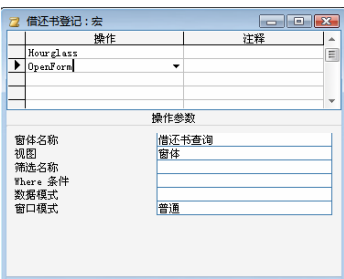


图 12.18 借还书登记宏

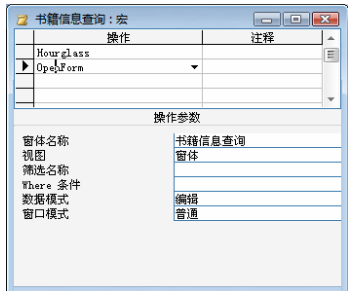


图 12.19 书籍信息查询宏

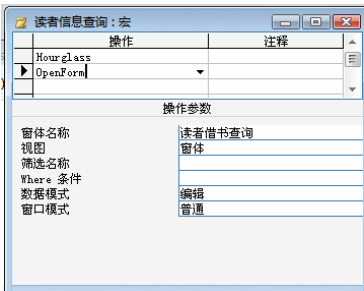


图 12.20 读者信息查询宏

12.4 系统测试及运行

启动系统，输入数据，测试系统是否完成了指定的功能，如果完成则可以使用，如果还没有达到要求，则需要进一步改进。

若设置“图书馆信息管理系统”窗体为主切换面版，则启动数据库后，系统自动启动该窗体。可以通过“工具”菜单的“启动”项打开“启动对话框”进行设置，如图12.21所示。

系统运行关系的结构如图12.22所示。

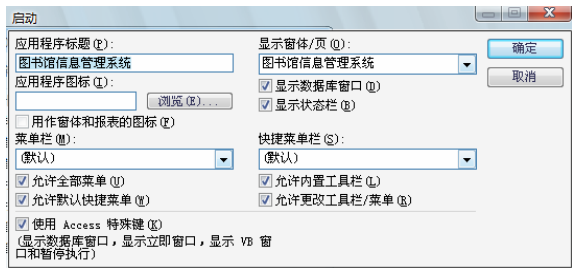


图 12.21 启动项的设置

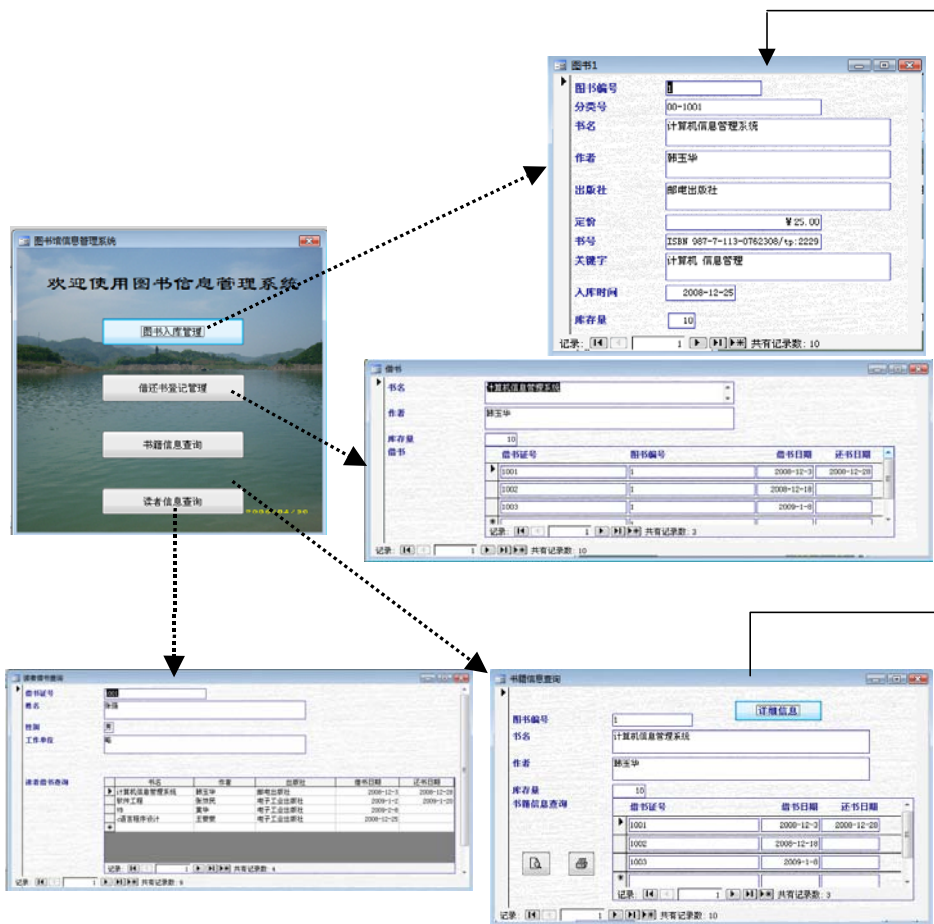


图 12.22 系统运行关系

习 题 12

结合教材内容的学习，完成“学生信息管理系统”的开发，主要包括：数据库表的建立、完整性设置、数据输入、创建查询、建立窗体、建立报表、建立系统菜单等。基本要求如下：

- (1) 建立数据库，数据库名称为：学生信息管理。
- (2) 建立数据表及关联：“学生档案”表、“课程信息”表、“学生成绩”表、“学生操行分”表。
- (3) 建立相关查询。
- (4) 建立如下几个窗体：
 - ① “学生档案登录”窗体：可输入学生的基本信息，窗体中能显示学生档案表中的所有字段，要有添加、保存、退出等相应控制按钮；
 - ② “课程信息登录”窗体：可输入课程信息，窗体中能显示课程信息表中的所有字段，要有添加、保存、退出等相应控制按钮；
 - ③ “学生信息显示”窗体：可显示学生的基本信息，包括学生的选课情况；
 - ④ “学生选课情况”窗体：可显示学生的选课情况，通过按班级查询、按学号查询、按课程名查询等几个按钮对学生的选课情况进行查询；
 - ⑤ “学生选课”窗体：可完成学生的选课录入；
 - ⑥ “学生选课成绩登录”窗体：可完成学生选课成绩的录入；
 - ⑦ “操行分登录”窗体：用于输入学生操行分的情况；
 - ⑧ “操行分查询”窗体：用于学生操行分的查询。
- (5) 建立“学生成绩统”和“学生基本信息”两个报表。
- (6) 建立主切换面板。

参 考 文 献

- [1] David M. Kroenke. 数据库原理. 冯飞译. 北京: 清华大学出版社, 2008
- [2] 李建中. 数据库系统原理. 北京: 电子工业出版社, 2008
- [3] 徐保民. 数据库原理与应用. 北京: 人民邮电出版社, 2010
- [4] 王珊. 数据库系统概论 (第四版). 北京: 高等教育出版社, 2006
- [5] 教育部高等学校文科计算机基础教学指导委员会. 大学计算机教学基本要求 (2010 版). 北京: 高等教育出版社, 2010
- [6] 耿国华. 大学计算机应用基础 (第二版). 北京: 清华大学出版社, 2010
- [7] michael R. Access 2007 教程. 谢俊译. 北京: 人民邮电出版社, 2008