



ADO.NET



数据库应用开发

张 骏 崔海波 等编著

- 对ADO.NET的应用开发作了全面而又深入的解析，内容循序渐进，案例导学。
- 希赛顾问团软件开发专家精心编写，引领读者进入项目实战的天堂。
- 随书配套光盘中包含本书所有实例的源代码，供读者学习参考使用。
- 作者在希赛网社区（<http://www.csai.cn>）“书评在线”版块中为读者提供全方位学习指导。



附光盘



机械工业出版社
China Machine Press

本书详细讲解用 ADO.NET 进行数据库开发的方法,分为 3 篇共 12 章,包括 ADO.NET、数据库和集成开发环境的基础知识,应用开发的各项技术与方法,用大量实例讲解技术细节。本书使用的开发环境是 Visual Studio 2005 + SQL Server 2000,全书各知识点的解说由浅入深,并辅以大量的实例说明,最后给出了一个完整的项目案例。

随书光盘中含有全书所有实例的源代码,以及项目案例的源代码,供读者学习参考使用。

本书供有一定的 C#编程基础的程序员作为参考用书,也可供社会 C#编程技术或数据库技术培训班作为教材使用,对于缺乏项目实战经验的程序员来说可用于快速积累项目开发经验。

版权所有,侵权必究。

本书法律顾问 北京市展达律师事务所

图书在版编目(CIP)数据

ADO.NET 数据库应用开发/张骏主编. —北京:机械工业出版社,2008.1
(希赛 IT 技术讲堂)

ISBN 978-7-111-23033-5

I. A… II. 张… III. 软件工具—程序设计 IV. TP311.56

中国版本图书馆 CIP 数据核字(2008)第 001895 号

机械工业出版社(北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑:李南丰

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2008 年 1 月第 1 版第 1 次印刷

184mm×260mm·18.25 印张

标准书号:ISBN 978-7-111-23033-5

ISBN 978-7-89482-504-9(光盘)

定价:42.00 元(附光盘)

凡购本书,如有倒页、脱页、缺页,由本社发行部调换

本社购书热线:(010) 68326294

编写委员会

组编:希赛顾问团

顾问:张友生

主审:邓子云

编委(按姓名拼音排序):

边 伟	陈亿春	崔海波	方海光	冯向科
葛志春	郭 莹	赫 斌	黄 婧	黄少年
李 刚	刘 毅	陆秉炜	刘志成	娄嘉鹏
聂艳明	阮国明	孙鸿飞	施 游	唐 俊
唐天广	王 军	吴吉义	吴伟敏	薛大龙
王红安	王 冀	王 勇	谢 顺	杨 森
张晓燕	张立东	朱小平		

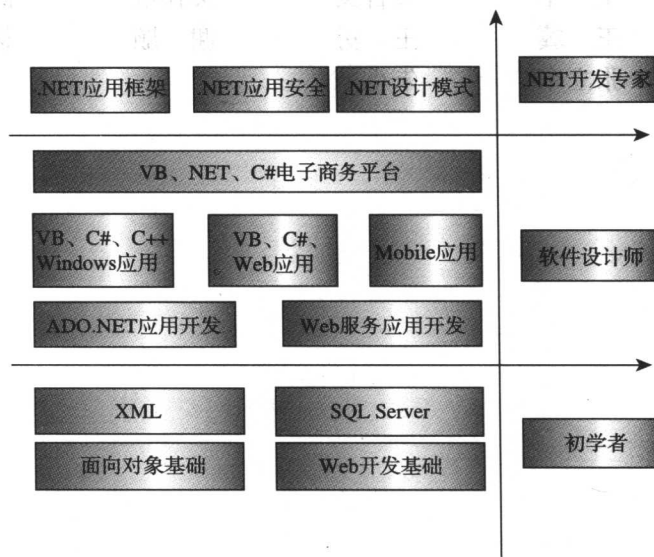
丛书介绍

微软的 .NET 是目前最流行的开发平台之一，它为 Windows 平台下的各种应用系统的开发提供了强大的支持，Visual Studio .NET 是支持这个平台应用开发的一套完善的开发工具，将 VB、C++、C# 等不同语言集成在一个统一的开发环境中，对 Web、桌面、移动等各种应用提供支持，各种语言之间共享开发工具与开发成果。

一、本丛书的知识体系

希赛 IT 技术讲堂(之 .NET 篇)丛书从 .NET 的入门基础课开始，带您遨游 .NET 所涉及到的方方面面，让您快速全面地了解 .NET 环境、基本概念和各种应用开发技术。丛书中每本书基本独立，由浅入深地介绍 .NET 中的某项技术。但丛书之间又相互关联，统一为一个整体，让读者对整个 .NET 平台有一个清楚的理解和把握，逐步成长为一名 .NET 领域的开发专家。

对于初学者，希赛 IT 技术讲堂(之 .NET 篇)丛书给出如下图所示的技术体系：



根据这个技术体系，我们编写了希赛 IT 技术讲堂(之 .NET 篇)丛书，希望能给正在学习 .NET 或有志于在 .NET 方向有所作为的读者提供帮助。初学者通过本丛书的逐步深入学习，可以成长为一名 .NET 方向的开发专家。有一定的 .NET 开发经验的人员也可以从中找到所需学习的知识。

本丛书以 VB.NET 和 C# 两种开发语言为主线，对 .NET 的整个知识技术体系做了详细的讲解，在书中还附有精选的项目案例，以总结所学的知识点，并引领读者在接近真实的工程实战环境中进行演练，以便快速积累开发经验。不同语言的爱好者可以选择丛书中的 VB.NET 或 C# 的相应书籍进行学习。

二、本丛书的特点

知识新，覆盖范围广。本丛书对 .NET 的整个技术体系做了详细的讲述，不管你是 .NET 的初

学者还是在 .NET 方向做了多年的软件开发人员，都能从本丛书中找到您想要的学习内容。

作者经验丰富，实践水平高。希赛顾问团拥有近 500 名顾问，有的是活跃在各行各业信息化工作一线的高级工程师、项目经理等，有的是高校计算机专业的教学骨干，有的是 IT 公司的工程师、技术总监等，本丛书由希赛顾问团中具有丰富的工程项目实践经验和教学经验的专家编写，是他们多年实践经验与技术积累的体现。

案例驱动，层次递进。每本书的知识点都是理论与实例的完美结合，让读者学习起来更加容易、轻松。丛书循序渐进，由浅入深，每本书都有精心设计的案例，让读者对所学知识的应用有明确的理解。在项目案例的阐述中，以软件工程思想为主线，潜移默化地让读者掌握软件开发的

平台优势，与读者互动。为更好地为读者服务，借助希赛网 (<http://www.csai.cn>) 的平台优势，作者们还将在希赛网上开办技术论坛，辅导答疑活动。有关本系列丛书的意见反馈和咨询，读者可在希赛网社区 (<http://bbs.csai.cn>) “书评在线”版块中的“机械工业出版社”栏目中与作者进行交流。本系列丛书各书配套光盘中的内容，读者也可以在希赛网下载中心 (<http://data.csai.cn>) 下载。

丛书编委会
2007 年 11 月

前言

ADO.NET 是专门为开发人员设计的,它主要面向在 Intranet 或 Internet 上使用的高效多层数据库应用程序。作为微软最新的数据访问技术,ADO.NET 已经在企业信息系统的开发中得到了广泛的应用。它的断开连接方式,极大地提高了系统的使用效率,并且使系统容易维护与扩展。

一、本书的知识体系

由于本书采用 C#语言来编写代码,因此学习 ADO.NET 的应用开发最好有面向对象编程和数据库编程的基础。

本书的知识体系结构如图 1 所示,遵循了循序渐进的原则,逐步引领读者从基础到各个知识点的学习,最后开发出完整的数据库系统。

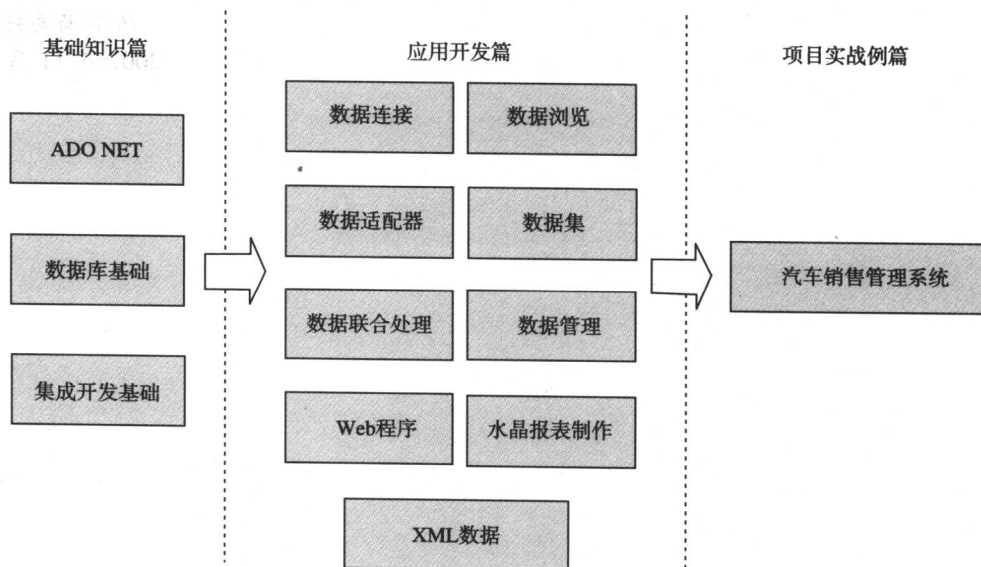


图 1 本书的知识体系结构

二、章节内容介绍

全书共分为三篇,第一篇为基础知识篇,包括 3 章。

第 1 章介绍数据库的基础知识。读者要掌握关系数据库理论,同时对 SQL 语言有一个大概的了解,这有助于了解 ADO.NET,为读者动手实践打下了基础。

第 2 章介绍 ADO.NET 的基础知识。包括 ADO.NET 基础、ADO.NET 的模型结构、ADO.NET 的特点,以及 ADO.NET 与 ADO 的对比。

第 3 章介绍如何通过数据绑定生成 Windows 应用程序。

第二篇为应用开发篇,包括 8 章。

第 4 章介绍数据连接与数据浏览的方法,通过介绍在设计中连接与在代码中创建连接的两种方式,实现与 SQL Server 数据库、Access 以及 Oracle 数据库的连接。然后通过 Command 对象与

DataReader 对象结合实现数据浏览。

第 5 章介绍数据适配器, 介绍如何创建和使用数据适配器。

第 6 章介绍的数据集是 ADO.NET 最核心的知识之一, 包括 DataSet、DataTable、DataColumn 和 DataRow。首先介绍 DataSet 的创建及用法。然后介绍数据表, 它的基本功能是一个包含有 3 种集合的容器: 列集合、行集合及约束。通过使用数据表的行、列, 可以很方便地实现对数据表的操作。

第 7 章说明了通过 DataRelation 对象或者通过 SQL 联合查询获得数据集中相关表的相关记录。

第 8 章介绍数据库的管理, 包括数据筛选与排序, 以及数据操作与数据更新。详细介绍了数据操作中的查找、插入、删除、更改记录, 以及更新数据源中的方法及注意要点。

第 9 章介绍 Web 应用程序, 包括数据连接、显示、数据缓存设置、数据编辑等。

第 10 章介绍水晶报表制作, 说明了水晶报表制作的两种方法, 然后介绍如何根据具体情况来设置报表。

第 11 章介绍 XML 数据的读取与写入方法。

第三篇为项目实战篇, 包括 1 章。

第 12 章中的项目案例是一个汽车销售管理系统。这个案例以开发与实现为主线, 从系统需求分析、系统总体架构的设计、数据库设计、系统的各个模块的详细实现这些方面逐步深入分析, 较为明晰地讲解了这个系统是如何分析、设计与编程实现的, 可综合之前所学的基础知识。

有数据库知识基础的读者可跳过第 1 章的学习, 有 ADO.NET 知识背景的读者可跳过第 2 章的学习。

本书内容由浅入深, 并辅以大量的实例说明。本书可供有一定 C# 编程基础的程序员作为参考书用, 也可供数据库技术培训班作为教材使用, 对于缺乏项目实战经验的程序员来说, 可用于快速积累项目开发经验。

随书的配套光盘含有全书所有实例的源代码, 以及项目案例的源代码, 供读者学习参考使用, 所有程序均经过了作者精心的调试。

三、技术支持

希赛是中国领先的互联网技术和 IT 教育公司, 在互联网服务、图书出版、人才培养方面, 希赛始终保持 IT 业界的领先地位。希赛对国家信息化建设和软件产业化发展具有强烈的使命感, 利用希赛网(www.csai.cn)强大的平台优势, 加强与促进 IT 人士之间的信息交流和共享, 实现 IT 价值。“希赛, 影响 IT”是全体希赛人不懈努力和追求的目标!

希赛网以希赛顾问团为技术依托, 是中国最大的 IT 资源平台。希赛 IT 教育研发中心是希赛公司属下的一个专门从事 IT 教育、教育产品开发、教育书籍编写的部门, 在 IT 教育方面具有极高的权威性。在国家权威机构发布的“计算机图书出版市场综述”中, 称赞希赛丛书为读者所称道, 希赛的图书已经形成品牌, 在读者心目中具有良好的形象。

本书由希赛顾问团组织策划, 由张骏(系统分析师、希赛顾问团顾问)主编, 崔海波负责第 9、10 章的编写, 余海负责全书的程序编制工作, 扶文奇、周进、肖佳等参与了全书的实例源代码调试工作, 王冀、王勇、史小琴、陈倩、谢顺等参与了书中项目案例的部分程序编制工作, 梁赛负责部分章节的校稿和编辑工作。

由于时间仓促和作者的水平有限, 书中的错误和不妥之处在所难免, 敬请读者批评指正。有关本书的意见反馈和咨询, 读者可在希赛网 IT 社区(bbs.csai.cn)“书评在线”版块中的“机械工业出版社”栏目中与作者进行交流。本书配套光盘中的内容, 读者也可以在希赛网下载中心

(data.csai.cn) 下载。

四、致谢

感谢机械工业出版社的陈冀康编辑，在本书的编写过程中，他提出了许多指导性的意见，并承担了大量的策划与编辑工作。借此还要感谢希赛顾问团的邓子云以及希赛公司的图书创作团队，他们给我提供了很多写作的思路和建议，也时时在鼓励我写这本书。正是因为这么多人的大力支持和辛勤汗水，本书才得以出版。

张 骏

2007 年 7 月于长沙

目 录

编写委员会

丛书介绍

前言

第一篇 基础知识篇

第 1 章 数据库基础知识	2
1.1 数据库基础	2
1.1.1 数据库基本概念	2
1.1.2 数据库管理系统的控制功能	2
1.1.3 常用的数据模型	3
1.1.4 数据库系统的三级模式结构	3
1.2 关系数据库理论	4
1.2.1 关系模型的基本术语	4
1.2.2 关系数据库对关系的限定	4
1.2.3 关系模型的完整性约束	4
1.2.4 关系代数	5
1.2.5 关系数据库的规范化理论	6
1.2.6 范式	6
1.3 SQL 简介	7
1.3.1 SQL 的特点	7
1.3.2 数据定义	8
1.3.3 数据简单查询	9
1.3.4 连接查询	12
1.3.5 嵌套查询	12
1.3.6 数据更新	13
1.3.7 视图操作	14
1.3.8 嵌入式 SQL 语句	14
1.4 小结	15
1.5 思考解答	15
第 2 章 ADO.NET 基础	17
2.1 ADO.NET 概述	17
2.1.1 ADO.NET 发展过程	17
2.1.2 ADO.NET 特点	17
2.1.3 ADO.NET 2.0 中的新增功能	19
2.2 ADO.NET 模型结构	20

2.2.1 DataSet 对象结构	21
2.2.2 .NET 对象结构	24
2.2.3 .NET 的数据提供程序	26
2.3 ADO 和 ADO.NET 比较	30
2.3.1 数据表现形式	30
2.3.2 数据访问	30
2.3.3 断开连接方式的数据访问	30
2.3.4 在应用程序之间共享数据	31
2.4 ADO.NET 的优点	31
2.4.1 支持 XML 格式	31
2.4.2 维护简单易行	31
2.4.3 编程效率高	32
2.4.4 优越的性能	32
2.4.5 可扩展性	32
2.5 小结	32
2.6 思考解答	33
第 3 章 用数据绑定生成 Windows 应用程序	35
3.1 数据绑定基础	35
3.2 Windows 窗体中的数据绑定	35
3.2.1 可以绑定到的数据源结构	36
3.2.2 Windows 数据绑定原理	36
3.3 使用简单数据绑定	37
3.3.1 设计时直接绑定	37
3.3.2 通过代码建立绑定	41
3.4 使用复杂绑定	42
3.4.1 设计时直接绑定	42
3.4.2 通过代码建立绑定	43
3.5 使用 CurrencyManager 对象	44
3.5.1 BindingContext 对象说明	44
3.5.2 Binding 对象的说明	45
3.5.3 CurrencyManager 的属性说明	47
3.5.4 CurrencyManager 的方法说明	50
3.5.5 CurrencyManager 的事件说明	51
3.6 小结	52
3.7 思考解答	52

第二篇 应用开发篇

第4章 数据库的连接与浏览 54

- 4.1 数据库的连接 54
 - 4.1.1 创建连接 54
 - 4.1.2 Connection 对象的属性 58
 - 4.1.3 Connection 对象的方法 60
 - 4.1.4 Connection 对象的事件 62
 - 4.1.5 使用连接池 62
- 4.2 数据库的浏览 64
 - 4.2.1 创建 Command 对象 65
 - 4.2.2 通过编写代码实现数据浏览 67
 - 4.2.3 Command 对象的属性 68
 - 4.2.4 Command 对象的方法 73
 - 4.2.5 DataReader 对象介绍 75
 - 4.2.6 DataReader 对象的属性 75
 - 4.2.7 DataReader 对象的方法 76
- 4.3 小结 78
- 4.4 思考解答 79

第5章 数据适配器 DataAdapter 80

- 5.1 DataAdapter 简介 80
 - 5.1.1 概述 80
 - 5.1.2 DataAdapter 对象包含的内容 80
- 5.2 通过设计器创建 DataAdapter 对象 81
- 5.3 在代码中创建 DataAdapter 对象 84
 - 5.3.1 DataAdapter 的构造函数 84
 - 5.3.2 用 DataAdapter 填充数据集 85
 - 5.3.3 打开与关闭连接 86
 - 5.3.4 读取多个表 87
- 5.4 DataAdapter 对象的属性 87
- 5.5 DataAdapter 对象的方法 92
 - 5.5.1 Fill 方法 92
 - 5.5.2 FillSchema 方法 95
 - 5.5.3 Update 方法 95
- 5.6 DataAdapter 对象的事件 98
 - 5.6.1 FillError 事件 98
 - 5.6.2 RowUpdating 事件和 RowUpdated 事件 98
- 5.7 小结 100
- 5.8 思考解答 100

第6章 数据集 DataSet 101

- 6.1 DataSet 概述 101
 - 6.1.1 数据集 101

- 6.1.2 DataSet 对象模型 102

- 6.1.3 处理脱机数据 104

- 6.1.4 类型化与非类型化 DataSet 104

- 6.1.5 XML 105

6.2 创建 DataSet 105

- 6.2.1 在设计器中创建 DataSet 对象 106

- 6.2.2 在运行时创建 DataSet 108

6.3 DataSet 属性 108

6.4 DataSet 对象的方法 112

6.5 数据表 DataTable 117

- 6.5.1 创建与添加数据表 117

- 6.5.2 DataTable 属性 118

- 6.5.3 DataTable 方法 122

6.6 DataRow 和 DataColumn 125

- 6.6.1 DataRow 简介 125

- 6.6.2 DataColumn 简介 125

- 6.6.3 DataRow 和 DataColumn 举例 126

6.7 小结 128

6.8 思考解答 128

第7章 数据库联合处理 129

7.1 约束 129

- 7.1.1 UniqueConstraint 约束 129

- 7.1.2 ForeignKeyConstraint 约束 130

7.2 DataRelation 对象介绍 132

7.3 创建 DataRelation 132

- 7.3.1 通过设计器创建 DataRelation 132

- 7.3.2 通过代码创建 DataRelation 134

7.4 数据库联合处理的用法 136

- 7.4.1 GetChildRows 方法实例 136

- 7.4.2 GetParentRows 方法实例 138

7.5 DataRelation 对象的属性 139

7.6 小结 140

7.7 思考解答 141

第8章 数据库管理 142

8.1 数据筛选与排序 142

- 8.1.1 DataView 简介 142

- 8.1.2 创建 DataView 142

- 8.1.3 DataView 对象的属性 144

- 8.1.4 DataView 实例 147

8.2 数据操作 149

- 8.2.1 行状态与行版本 149

8.2.2 数据查找	150	9.7 小结	193
8.2.3 数据添加	151	9.8 思考解答	193
8.2.4 数据更改	152	第10章 水晶报表的制作	194
8.2.5 数据删除	153	10.1 报表使用的基本概念	194
8.2.6 允许或拒绝数据集的修改	154	10.1.1 报表的实现	194
8.3 数据更新	154	10.1.2 水晶报表的实现	196
8.3.1 通过 DataAdapter 对象的 Update 方法	155	10.2 水晶报表的模式	198
8.3.2 通过 Command 对象进行 更新	156	10.2.1 使用 Pull 模式	198
8.4 数据操作与更新例子	157	10.2.2 使用 Push 模式	204
8.4.1 用数据集提供的方法进行 操作	157	10.3 设计水晶报表	207
8.4.2 用 Command 对象的 SQL 语句进行 操作	161	10.3.1 使用报表向导设计水晶 报表	207
8.5 小结	164	10.3.2 手工定制水晶报表	210
8.6 思考解答	164	10.4 导出报表	213
第9章 Web 应用程序	167	10.5 具体实例	214
9.1 ASP.NET 简介	167	10.5.1 Push 方法实现	214
9.1.1 ASP.NET 发展历史	167	10.5.2 通过手工定制实现水晶 报表	216
9.1.2 安装 ASP.NET 平台	168	10.6 小结	219
9.2 数据库连接	169	10.7 思考解答	219
9.2.1 Access 数据库连接	169	第11章 使用 XML 数据	220
9.2.2 SQL 数据库连接	170	11.1 XML 概述	220
9.3 在 Web 页面中显示与查询	171	11.2 XML 数据的读取与写入方法	220
9.3.1 创建简单的 Web 页面	171	11.2.1 读取 XML 文档	220
9.3.2 在 Web 页面中显示与查询	172	11.2.2 编写 XML 文档	225
9.4 数据缓存	178	11.3 XmlDataDocument 类	227
9.4.1 页面输出缓存	179	11.4 从 SQL Server 中获取 XML 数据	232
9.4.2 片段缓存和用户控件输出 缓存	180	11.4.1 使用 FOR XML	232
9.4.3 在 cache 中存储数据	180	11.4.2 SQL XML 托管类	234
9.4.4 缓存数据引用模式	182	11.5 类型化 DataSet	234
9.5 数据编辑	183	11.5.1 将批注用于类型化 DataSet	235
9.5.1 数据插入	183	11.5.2 强类型化的数据集	237
9.5.2 数据删除	185	11.6 XML 数据的操作	238
9.5.3 数据更新	187	11.6.1 DOM 树	238
9.6 具体实例	188	11.6.2 打开一个现有的文档	238
9.6.1 学生基本信息管理	189	11.6.3 在文档中搜索内容	239
9.6.2 添加学生信息	191	11.6.4 访问内容	239
9.6.3 修改学生信息	192	11.6.5 修改文档	240
9.6.4 删除学生信息	193	11.7 具体实例	242
		11.8 小结	244
		11.9 思考解答	244

第三篇 项目实战篇

第 12 章 汽车销售管理系统	246	12.6.1 客户信息查询	257
12.1 系统设计	246	12.6.2 汽车信息查询	258
12.2 数据库的设计与实现	247	12.6.3 员工信息查询	260
12.2.1 数据库的需求分析	247	12.7 资料管理模块	261
12.2.2 数据库的逻辑设计	247	12.7.1 界面设计	262
12.3 系统主窗体设计	250	12.7.2 代码设计	262
12.3.1 页面设计	250	12.8 汽车销售管理模块	269
12.3.2 代码设计	251	12.8.1 销售主界面	269
12.4 连接与断开服务器设计	252	12.8.2 汽车预定管理	271
12.4.1 连接服务器	252	12.8.3 汽车销售管理	273
12.4.2 断开服务器连接	253	12.9 数据统计模块	275
12.5 用户管理	254	12.9.1 员工业绩统计	275
12.5.1 用户登录	254	12.9.2 产品销售情况统计	276
12.5.2 用户注销	256	12.10 合同管理模块	277
12.5.3 密码更改	256	12.10.1 预定订单报表	277
12.6 信息查询模块	257	12.10.2 销售订单报表	278
		12.11 小结	278

第一篇

基础知识篇



本章专家知识导学

本章旨在引导读者了解数据库的基础知识，掌握最基本的也是最常用的关系数据库理论，为以后学会如何设计与开发数据库应用系统打下坚实的基础。

SQL 是一种介于关系代数与关系演算之间的结构化查询语言。由于它是一个综合的、功能极强同时又是简单易学的语言，现已经成为数据库领域内的一个主流语言。学好 SQL 语言是学习 ADO.NET 控件的基础，有助于加深对 ADO.NET 的了解与认识，为读者动手实践编程打下了基础。

1.1 数据库基础

我们先了解一些与数据库技术有关的基本概念、数据库管理系统的控制功能、常用的数据模型以及数据库系统的三级模式结构。

1.1.1 数据库基本概念

(1) 数据库(Database, DB)是长期存储在计算机内、有组织的、统一管理的相关数据的集合。DB 能为各种用户共享，具有较小冗余度、数据间联系紧密而又有较高的数据独立性等特点。

(2) 数据库管理系统(Database Management System, DBMS)是指负责数据库存取、维护和管理的系统软件。它的基本功能有：数据定义功能、数据操作功能、数据库的运行管理功能、数据库的建立和维护功能等。

(3) 数据库系统(Database System, DBS)是指在计算机系统中引入数据库后的系统构成。一般由数据库、操作系统、数据库管理系统及其工具、应用系统、数据库管理人员和用户构成。

(4) 数据库管理员(Database Administrator, DBA)是指数据库建立、使用和维护等工作的专门职业人员。

1.1.2 数据库管理系统的控制功能

(1) 并发共享：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数据。

(2) 数据安全性保护：保护数据以防止因使用不合法而造成数据的泄密和破坏，用户只能对数据以规定的方式进行使用和处理。

(3) 数据完整性检查：将数据控制在有效的范围内，保证数据的正确性、有效性和兼容性。

(4) 并发控制：对多用户的并发操作加以控制和协调，以保证多用户同时存取、修改数据库时，不会发生相互干扰而得到错误的结果或使得数据库的完整性遭到破坏。

(5) 数据恢复：DBMS 能够将硬件故障、软件故障等错误状态恢复到某一正确的状态。

1.1.3 常用的数据模型

目前,数据库领域中最常用的数据模型有4种,它们是层次模型、网状模型、关系模型以及面向对象模型。其中层次模型和网状模型统称为非关系模型。

1. 层次模型

层次模型是数据库系统中最早出现的数据模型,用树形结构来表示各类实体及实体间的联系。层次模型的查询效率很高,曾得到广泛应用,但它只能表示1:n联系,对数据进行查询和更新操作时则很复杂,所以编写应用程序也很复杂。

2. 网状模型

用有向图结构表示实体类型以及实体间联系的数据模型称为网状模型。网状数据模型的特点是记录之间的联系通过指针来实现,它能表示M:N联系,查询效率较高,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,而且DDL和DML语言复杂,用户不容易使用。

3. 关系模型

关系数据库系统采用关系模型作为数据库的组织方式。关系模型是指用表格形式结构表示实体类型以及实体间联系的模型。关系模型中数据的逻辑结构是一张二维表,它由行和列组成。关系模型的数据结构简单,容易被初学者接受。它是一个成熟的、有前途的模型,已得到广泛应用。

【专家提示】 关系模型是目前最重要的一种数据模型,学好关系模型是学好数据库系统设计的基础。

4. 面向对象模型

现实世界中存在着许多含有更复杂数据结构的实际应用领域,如CAD数据、图形数据等,加上人工智能研究的需要,就导致了面向对象的数据模型的出现。在面向对象的数据模型中最基本的概念为对象和类。面向对象的数据模型可完整地描述现实世界的数据结构,它比层次、网状、关系数据模型具有更加丰富的表达能力,能表达嵌套、递归的数据结构。

1.1.4 数据库系统的三级模式结构

数据库系统的三级模式结构是指数据库系统是由模式、外模式和内模式三级构成的。

1. 模式

模式也称概念模式或逻辑模式,是数据库中全体数据的逻辑结构和特征的描述,是所有用户的公共数据视图。模式实际上是数据库数据在逻辑级上的视图。定义模式时不仅要定义数据的逻辑结构,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。一个数据库只有一个模式。

2. 外模式

外模式也称子模式或用户模式,它是数据库用户能够看见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,是与某一应用有关的数据的逻辑表示。外模式通常是模式的子集。一个数据库可以有多个外模式。外模式是保证数据库安全性的一个有力措施。

3. 内模式

内模式也称存储模式或物理模式,一个数据库只有一个内模式。存储方式的描述是数据在

数据库内部的表示方式。

【专家提示】 模式在数据库系统中对应的是数据表，外模式对应的是视图，而内模式对应的是存储文件。

1.2 关系数据库理论

关系数据库系统是支持关系数据模型的数据库系统，也是现行最常用的数据库模型。几十年来，对关系数据库系统的研究取得了辉煌的成就。目前市场上已涌现出许多性能良好的商品化关系数据库管理系统，如 Oracle，SYBASE，SQL Server 等。

1.2.1 关系模型的基本术语

(1) 关系：一个关系对应一个二维表，二维表名就是关系名。

(2) 属性：在二维表中的列(字段)，称为属性。属性的个数称为关系的元数，列的值称为属性值，属性值的取值范围称为值域。

(3) 元组：二维表中的一行(记录的值)，称为一个元组。关系模式和元组的集合通称为关系。

(4) 候选码：如果在一个关系中，存在多个属性(或同性组合)都能用来惟一标识该关系的元组，这些属性(或属性组合)都称为该关系的候选码。

(5) 主码：在一个关系的若干个候选码中指定一个用来惟一标识该关系的元组，这个被指定的候选码称为该关系的主码。

(6) 外码：当关系中的某个属性(或属性组)虽然不是该关系的主码或是主码的一部分，但却是另一个关系的主码时，称该属性(或属性组)为这个关系的外码。

(7) 关系模式：在二维表中的行定义，即对关系的描述称为关系模式，一般表示为：关系名(属性1，属性2，…，属性n)。

(8) 主属性：关系中包含在任何一个候选码中的属性称为主属性或码属性。

1.2.2 关系数据库对关系的限定

(1) 每一个属性是不可分解的。

(2) 每一个关系模式中属性的数据类型以及属性的个数是固定的，并且每一个关系模式中，属性名必须是不同的。

(3) 每一个关系仅仅有一种记录类型，即一种关系模式。

(4) 在关系中元组的顺序(即行序)是无关紧要的。

(5) 在关系中属性的顺序可任意交换，交换时应连同属性名起交换。

(6) 同一个关系中不允许出现完全相同的元组。

1.2.3 关系模型的完整性约束

数据完整性由完整性规则来定义，关系模型的完整性规则是对关系的某种约束条件。模型中可以有三类完整性约束：实体完整性、参照完整性和用户定义的完整性。

1. 实体完整性规则

若属性 A 是基本关系 R 的主属性，则属性 A 不能取空值。实体完整性规则规定基本关系的所有主属性都不能取空值。

2. 参照完整性规则

若属性(或属性组) F 是基本关系 R 的外码,它与基本关系 S 的主码 K_s 相对应(基本关系 R 和 S 不一定是不同的关系),则对于 R 中每个元组在 F 上的值必须为空值(F 的每个属性值均为空值)或者等于 S 中某个元组的主码值。

3. 用户定义的完整性

用户定义的完整性规则是针对某一应用环境的完整性约束条件,它反映了某一具体应用所涉及的数据应满足的要求。用户定义的完整性也称为域完整性规则,包括数据类型、精度、取值范围、是否允许空值等。

1.2.4 关系代数

关系代数是关系操纵语言的一种传统表示方式,它是以集合代数为基础发展起来的,但它的运算对象和运算结果都是关系。同时,关系代数是一种抽象的查询语言,通过对关系运算来表达查询。关系代数的运算可分为两类:传统的集合运算和专门的关系运算。

1. 传统的集合运算

传统的集合运算是二目运算,包括并、差、交、广义笛卡尔积四种运算。

(1)并:设关系 R 和关系 S 具有相同的目 n (即都有 n 个属性),且相应的属性来自同一个域,则关系 R 与 S 的并是由属性 R 或属性 S 的元组组成的,结果仍为 n 目关系,记作 $R \cup S = \{t \mid t \in R \vee t \in S\}$, t 是元组变量。

(2)差:设关系 R 和关系 S 具有相同的目 n ,且相应的属性取自同一个域,则关系 R 与关系 S 的差是由属于 R 而不属于 S 的所有元组组成的,结果仍为 n 目关系,记作 $R - S = \{t \mid t \in R \wedge t \notin S\}$, t 是元组变量。

(3)交:设关系 R 和关系 S 具有相同的目 n ,且相应的属性取自同一个域,则关系 R 与关系 S 的交是由既属于 R 又属于 S 的所有元组组成的,结果仍为 n 目关系,记作 $R \cap S = \{t \mid t \in R \wedge t \in S\}$, t 是元组变量。关系的交可以用差来表示,即 $R \cap S = R - (R - S)$ 。

(4)广义笛卡尔积:设关系 R 和 S 的元数分别是 r 和 s 。定义 R 和 S 的笛卡尔积是一个 $(r+s)$ 元元组的集合,每一个元组的前 r 个分量来自 R 的一个元组,后 s 个分量来自 S 的一个元组。若 R 有 m 个元组, S 有 n 个元组,则关系 R 和 S 的广义笛卡尔积有 $m \times n$ 个元组,记作 $R \times S = \{t \mid t = \langle t_r, t_s \rangle \wedge t_r \in R \wedge t_s \in S\}$ 。

2. 专门的关系运算

专门的关系运算包括:对单个关系进行垂直分解(投影操作)或水平分解(选择操作)和对多个关系进行结合(连接操作)等。

(1)选择:选择又称为限制,是在关系中选择满足给定条件的元组,记做 $\delta_F(R) = \{t \mid t \in R \wedge F(t) = \text{'真'}\}$,其中 F 表示选择条件,它是一个逻辑表达式。

【专家提示】 选择运算实际上是从关系 R 中选取使逻辑表达式 F 为真的元组。这是从行的角度进行的运算。

(2)投影:关系 R 上的投影是从 R 中选择出若干属性列组成新的关系,记做 $\pi_A(R) = \{t[A] \mid t \in R\}$,其中 A 为 R 中的属性列。

【专家提示】 投影操作实际上是从关系中选取某些列,即从列的角度进行的运算。

(3)连接:连接是从两个关联的笛卡尔积中选取属性间满足一定条件的元组。在连接运算