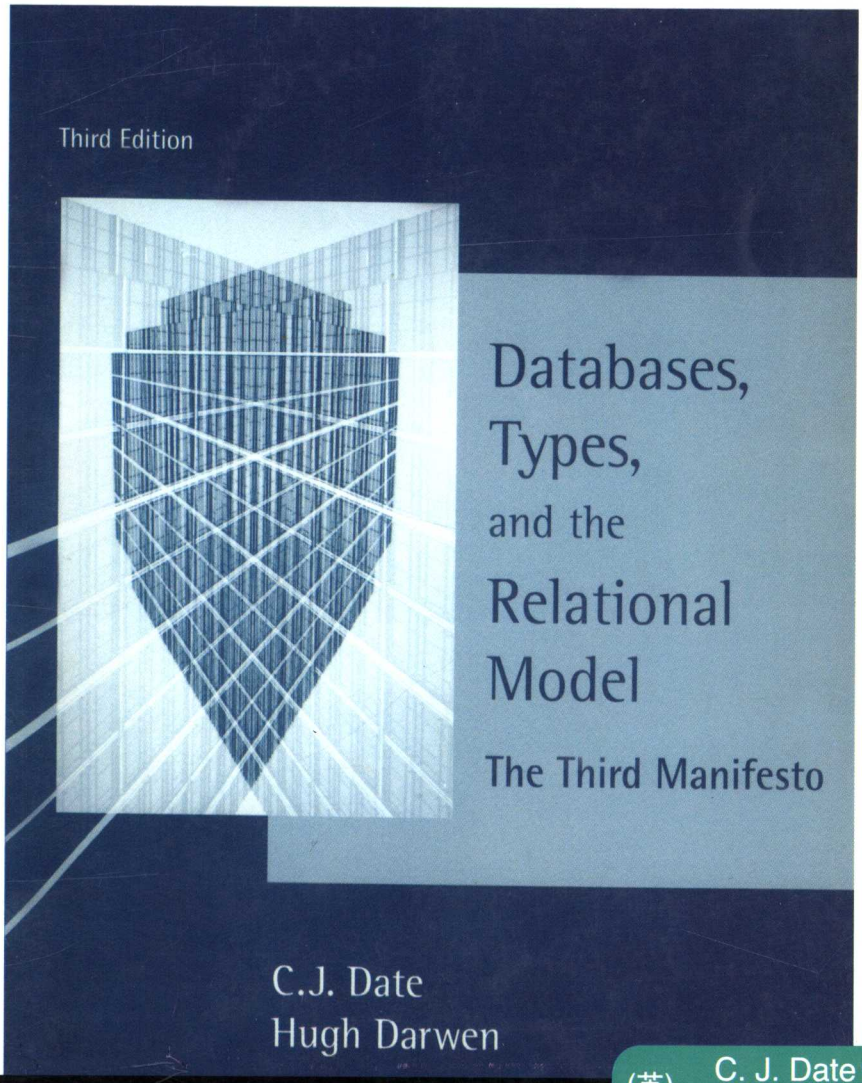
HZ BOOKS

经 典 原 版 书 库

PEARSON Addison Wesley

数据库、类型和关系模型

（英文版 · 第3版）



(英) C. J. Date Hugh Darwen 著

机械工业出版社
China Machine Press

经 典 原 版 书 库

数据库、类型和关系模型

（英文版 · 第3版）

Databases, Types, and the Relational Model

(Third Edition)

(英) C. J. Date
Hugh Darwen 著

B
E1060838
B13

机械工业出版社
China Machine Press

English reprint edition copyright © 2006 by Pearson Education Asia Limited and China Machine Press.

Original English language title: *Databases, Types, and the Relational Model, Third Edition* (ISBN 0-321-39942-0) by C. J. Date and Hugh Darwen, Copyright © 2007 by Pearson Education, Inc.

All rights reserved.

Published by arrangement with the original publisher, Pearson Education, Inc., publishing as Addison-Wesley.

For sale and distribution in the People's Republic of China exclusively (except Taiwan, Hong Kong SAR and Macau SAR).

本书英文影印版由Pearson Education Asia Ltd.授权机械工业出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

仅限于中华人民共和国境内（不包括中国香港、澳门特别行政区和中国台湾地区）销售发行。

本书封面贴有Pearson Education（培生教育出版集团）激光防伪标签，无标签者不得销售。

版权所有，侵权必究。

本书法律顾问　北京市展达律师事务所

本书版权登记号：图字：01-2006-5652

图书在版编目（CIP）数据

数据库、类型和关系模型（英文版·第3版）/（英）戴特（Date, C. J.）等著. -北京：机械工业出版社，2007.1

（经典原版书库）

书名原文：Databases, Types, and the Relational Model, Third Edition

ISBN 7-111-20168-X

Ⅰ.数… Ⅱ.戴… Ⅲ.数据库管理系统-英文 Ⅳ.TP311.13

中国版本图书馆CIP数据核字（2006）第124976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：迟振春

北京瑞德印刷有限公司印刷·新华书店北京发行所发行

2007年1月第1版第1次印刷

170mm×242mm · 35.75印张

定价：65.00元

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

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅擘划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall，Addison-Wesley，McGraw-Hill，Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum，Stroustrup，Kernighan，Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及庋藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专程为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京

邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业的教学度身订造的。其中许多教材均已为M. I. T.，Stanford，U.C. Berkeley，C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

电子邮件：hzjsj@hzbook.com
联系电话：（010）68995264
联系地址：北京市西城区百万庄南街1号
邮政编码：100037

此为试读，需要完整PDF请访问：www.ertongbook.com

专家指导委员会

（按姓氏笔画顺序）

尤晋元　王　珊　冯博琴　史忠植　史美林
石教英　吕　建　孙玉芳　吴世忠　吴时霖
张立昂　李伟琴　李师贤　李建中　杨冬青
邵维忠　陆丽娜　陆鑫达　陈向群　周伯生
周克定　周傲英　孟小峰　岳丽华　范　明
郑国梁　施伯乐　钟玉琢　唐世渭　袁崇义
高传善　梅　宏　程　旭　程时端　谢希仁
裘宗燕　戴　葵

All logical differences are big differences

—Ludwig Wittgenstein

Again I must remind you that
A Dog's a Dog—A CAT'S a CAT

—T. S. Eliot, demonstrating that
logical differences are not
everything, and nor is logic

We would like to dedicate this book to
philosophers and poets everywhere

Preface

This is a textbook on database management. It is based on an earlier book by the same authors, *Foundation for Future Database Systems: The Third Manifesto* (Addison-Wesley, 2000), but it has been thoroughly revised—indeed, completely rewritten—from start to finish. Part of our reason for wanting to revise it was to make it more suitable as a textbook (the earlier book, by contrast, was quite terse and formal and not very easy to read); in particular, most chapters now include a set of exercises, solutions to which can be found in the Solutions Manual at the Web site (see later in this preface). However, we have naturally taken the opportunity to incorporate many other changes as well, including numerous clarifications, a certain amount of restructuring, many more examples, and—we regret to have to say—quite a few corrections also.

Like its predecessor, the book is organized around the ideas of *The Third Manifesto*. As Chapter 1 explains in more detail, *The Third Manifesto*—the *Manifesto* for short—is a proposal for a foundation for data and database management systems (DBMSs); it can be seen as an abstract blueprint for the design of a DBMS and the language interface to such a DBMS. In particular, it serves as a basis for a model of type inheritance, also discussed in this book. The overall structure of the book is thus as follows. First of all, it is divided into four major parts:

I. Preliminaries
II. Formal Specifications
III. Informal Discussions and Explanations
IV. Subtyping and Inheritance

To elaborate briefly:

- Part I (three chapters) sets the scene by explaining in general terms what the *Manifesto* is all about and why we wrote it. It also contains an informal overview of both the relational model and a theory of types. We recommend that you read these chapters fairly carefully before moving on to later parts of the book.

- Part II (two chapters) is the most formal part. It contains the *Manifesto* proper and a grammar for a language based on *Manifesto* principles called **Tutorial D**. *Note:* This part of the book is provided mainly for reference purposes; it is not necessary, and probably not even a good idea, to study it exhaustively, at least not on a first reading. **Tutorial D** in particular is intended to be largely self-explanatory. Even though examples throughout the book are based on it, therefore, it should not be necessary to study the **Tutorial D** grammar in depth in order to understand those examples.

- Part III is the real heart of the book. It consists of six chapters, one for each of the six sections of the *Manifesto* as defined in Part II. Each chapter discusses the relevant section of the *Manifesto* in detail, with examples, and thereby explains the motivations and rationale behind the formal proposals of Part II.

- Finally, Part IV (five chapters) does for type inheritance what Parts I-III do for the *Manifesto* proper.

In addition to the foregoing, there are ten appendixes, including one in particular (Appendix J) that gives an annotated and consolidated list of references for the entire book. *Note:* References throughout the book take the form of numbers in square brackets. For example, the reference "[3]" refers to the third item in the list in Appendix J: viz., a paper by Malcolm P. Atkinson and O. Peter Buneman entitled "Types and Persistence in Database Programming Languages," published in *ACM*

Computing Surveys, Volume 19, No. 2, in June 1987.

We defer discussion of exactly who the book is aimed at and what readers are expected to know to Chapter 1.

TO THE INSTRUCTOR

The Solutions Manual is available only to qualified instructors. Please visit Addison-Wesley's Instructor Resource Center (*www.aw.com/irc*) or contact your local Addison-Wesley Sales Representative to access the solutions.

THE *MANIFESTO* WEB SITE

The *Manifesto* Web site *www.thethirdmanifesto.com* (available to all readers of this book) contains a wealth of relevant additional material, including at the time of writing:

- Articles and papers by the authors
- Information about forthcoming seminars and presentations by C. J. Date and others
- Copies of handouts for various lectures given by Hugh Darwen
- A grammar, suitable for driving a mechanical parser, for the language **Tutorial D**
- Links to related Web sites, in particular those for certain projects related to implementation of the ideas of the *Manifesto*

The projects just mentioned include:

- *Rel:* A prototype implementation of **Tutorial D** by Dave Voorhis, of Armchair Airlines Computer Services Inc. in Canada and the University of Derby, England. We recommend Rel for use by teachers and students of relational theory. Rel is available as Open Source from *dbappbuilder.sourceforge.net*.
- *Duro:* A project by René Hartmann to create a relational database library based on the *Manifesto*, written in C. It supports a relational algebra and transactions, and is implemented using Berkeley DB. Duro is available as Open Source from *duro.sourceforge.net*.
- *D4:* D4 is the language used in Dataphor, a commercial product from Alphora, of Provo, Utah. Dataphor is the first known attempt at a commercial implementation of *The Third Manifesto*. It uses syntax similar to that of **Tutorial D**. *Note:* Because the product provides a front end to SQL, D4 deviates from the *Manifesto* by including support for SQL-style nulls, finding a conflict here between the *Manifesto* and the primary Dataphor design goal of automating application development. Nevertheless, the language does conform in many other respects and its authors say that the ideas of the *Manifesto* "have helped us achieve a level of automation that we never dreamed possible when we first set out to build the product." For further information visit *www.alphora.com*.
- *D^b ("D flat"):* An amusingly named project undertaken by Peter Nicol in his final year as a student at the University of Manchester Institute of Science and Technology (UMIST) in the United Kingdom. A description is available at *www.thethirdmanifesto.com*.

There is also an e-mail discussion forum: E-mails to *ttm@thethirdmanifesto.com* are distributed to all registered participants. To register and receive information about this forum and how to use it, send an empty e-mail to *ttm-subscribe@thethirdmanifesto.com*.

ACKNOWLEDGMENTS

The number of people who have supported us over the years in our work on the *Manifesto* in general and this book in particular has now grown so large that it is no longer feasible to list them all here. With apologies to other supporters, therefore, we would just like to mention the following: Shai Berger, Peter Elderon, Maurice Gittens, Jonathan Leffler, David McGoveran, Åke Persson, Bryn Rhodes, and Andrew Sieber. All of these helped in one way or another to ensure that this book finally saw the light of day in its present form. We would also like to thank Doug Inkster, David McGoveran, Jeremy Peel, Sudha Ram of the University of Arizona, Greg Speegle of Baylor University, Junping Sun of Nova Southeastern University, Vassilis Tsotras of UC Riverside, and especially Jonathan Leffler again, for their careful and constructive comments on earlier versions of the manuscript. Finally, we would like once again to thank our wives, Lindy Date and Lindsay Darwen, for their support throughout this project, and to acknowledge our copyeditor, Elisabeth Beller, for her assistance and usual high standards of professionalism.

C. J. Date, *Healdsburg, California*
Hugh Darwen, *Shrewley, England*
2005

Contents

Preface *vii*

PART I PRELIMINARIES

Chapter 1 **Background and Overview** **3**

The Third Manifesto 4
Back to the relational future 8
Some guiding principles 10
Some crucial logical differences 11
Concluding remarks 13
Exercises 14

Chapter 2 **A Survey of the Relational Model** **17**

The running example 17
Tuples 19
Relations 21
Relation variables 24
Relvars, relations, and predicates 28
Integrity constraints 32
Relational operators 34
Virtual relvars 46
The relational model 47
Exercises 48

Chapter 3 **Toward a Theory of Types** **55**

Values are typed 57
Types vs. representations 60
Scalar vs. nonscalar types 60
Possible representations 61
Selectors and THE_ operators 62
System-defined types 68
Operators 69
Type generators 73
Concluding remarks 75
Exercises 75

此为试读,需要完整PDF请访问: www.ertongbook.com

PART II FORMAL SPECIFICATIONS

Chapter 4 ***The Third Manifesto*** **81**

RM Prescriptions 81
RM Proscriptions 88
OO Prescriptions 88
OO Proscriptions 89
RM Very Strong Suggestions 89
OO Very Strong Suggestions 90
Recent *Manifesto* changes 90

Chapter 5 **Tutorial D 93**

Common constructs 96
Scalar definitions 98
Tuple definitions 100
Relational definitions 101
Scalar operations 102
Tuple operations 105
Relational operations 110
Relations and arrays 117
Statements 118
Recent language changes 120
A remark on syntax 122
Exercises 123

PART III INFORMAL DISCUSSIONS AND EXPLANATIONS

Chapter 6 **RM Prescriptions 129**

RM Prescription 1: Scalar types 129
RM Prescription 2: Scalar values are typed 132
RM Prescription 3: Read-only vs. update operators 133
RM Prescription 4: Physical vs. possible representations 139
RM Prescription 5: Expose possible representations 144
RM Prescription 6: Type generator TUPLE 147
RM Prescription 7: Type generator RELATION 152
RM Prescription 8: Equality 155
RM Prescription 9: Tuples 157
RM Prescription 10: Relations 158
RM Prescription 11: Scalar variables 159

RM Prescription 12: Tuple variables 160
RM Prescription 13: Relation variables (relvars) 161
RM Prescription 14: Kinds of relvars 162
RM Prescription 15: Candidate keys 165
RM Prescription 16: Databases 166
RM Prescription 17: Transactions 167
RM Prescription 18: Relational algebra 167
RM Prescription 19: Relvar names, relation selectors, and recursion 172
RM Prescription 20: User-defined tuple and relational operators 176
RM Prescription 21: Assignments 177
RM Prescription 22: Comparisons 180
RM Prescription 23: Integrity constraints 181
RM Prescription 24: Total database constraints 186
RM Prescription 25: Catalog 187
RM Prescription 26: Language design 187
Exercises 188

Chapter 7 **RM Proscriptions 191**

RM Proscription 1: No attribute ordering 191
RM Proscription 2: No tuple ordering 193
RM Proscription 3: No duplicate tuples 193
RM Proscription 4: No nulls 193
RM Proscription 5: No nullological mistakes 194
RM Proscription 6: No internal-level constructs 194
RM Proscription 7: No tuple-level operations 195
RM Proscription 8: No composite attributes 196
RM Proscription 9: No domain check override 196
RM Proscription 10: Not SQL 196
Exercises 197

Chapter 8 **OO Prescriptions 199**

OO Prescription 1: Compile-time type checking 199
OO Prescription 2: Type inheritance (conditional) 199
OO Prescription 3: Computational completeness 200
OO Prescription 4: Explicit transaction boundaries 201
OO Prescription 5: Nested transactions 202
OO Prescription 6: Aggregate operators and empty sets 203
Exercises 204

Chapter 9 **OO Proscriptions 205**

OO Proscription 1: Relvars are not domains 205
OO Proscription 2: No object IDs 209
Exercises 212

Chapter 10 **RM Very Strong Suggestions 215**

RM Very Strong Suggestion 1: System keys 215
RM Very Strong Suggestion 2: Foreign keys 218
RM Very Strong Suggestion 3: Candidate key inference 219
RM Very Strong Suggestion 4: Transition constraints 220
RM Very Strong Suggestion 5: Quota queries 221
RM Very Strong Suggestion 6: Generalized transitive closure 223
RM Very Strong Suggestion 7: User-defined generic operators 226
RM Very Strong Suggestion 8: SQL migration 227
Exercises 238

Chapter 11 **OO Very Strong Suggestions 241**

OO Very Strong Suggestion 1: Type inheritance 241
OO Very Strong Suggestion 2: Types and operators unbundled 241
OO Very Strong Suggestion 3: Single-level store 242
Exercises 243

PART IV SUBTYPING AND INHERITANCE

Chapter 12 **Preliminaries 247**

Toward a type inheritance model 248
Single vs. multiple inheritance 249
Scalars, tuples, and relations 249
The running example 250
Concluding remarks 258
Exercises 259

Chapter 13 **The Inheritance Model 261**

IM Prescriptions 261
Recent inheritance model changes 267

Chapter 14 **Single Inheritance with Scalar Types 269**

IM Prescription 1: Types are sets 269
IM Prescription 2: Subtypes are subsets 269
IM Prescription 3: "Subtype of" is reflexive 271
IM Prescription 4: Proper subtypes 272
IM Prescription 5: "Subtype of" is transitive 272
IM Prescription 6: Immediate subtypes 272
IM Prescription 7: Root types disjoint 273

IM Prescription 8: Scalar values with inheritance 273
IM Prescription 9: Scalar variables with inheritance 274
IM Prescription 10: Specialization by constraint 279
IM Prescription 11: Assignment with inheritance 282
IM Prescription 12: Equality etc. with inheritance 285
IM Prescription 13: Join etc. with inheritance 286
IM Prescription 14: TREAT 288
IM Prescription 15: Type testing 292
IM Prescription 16: Read-only operator inheritance and value substitutability 2
IM Prescription 17: Operator signatures 298
IM Prescription 18: Read-only parameters to update operators 305
IM Prescription 19: Update operator inheritance and variable substitutability 3(
IM Prescription 20: Union, dummy, and maximal and minimal types 309
Exercises 315

Chapter 15 **Multiple Inheritance with Scalar Types 317**

An introductory example 317
Type graphs 323
Least specific types unique 325
Most specific types unique 326
Remarks on operator inheritance 333
Exercises 334

Chapter 16 **Inheritance with Tuple and Relation Types 337**

IM Prescription 21: Tuple/relation subtypes and supertypes 337
IM Prescription 22: Tuple/relation values with inheritance 341
IM Prescription 23: Maximal and minimal tuple/relation types 343
IM Prescription 24: Tuple/relation most specific types 346
IM Prescription 25: Tuple/relation variables with inheritance 351
Some implications of IM Prescriptions 21-25 353
Exercises 357

APPENDIXES

Appendix A **A New Relational Algebra 361**

Motivation and justification 361
◂REMOVE▸, ◂RENAME▸, and ◂COMPOSE▸ 364
Treating operators as relations 366
Formal definitions 369
How **Tutorial D** builds on **A** 371

Appendix B **A Design Dilemma? 377**

Encapsulation 377
Discussion 378
Further considerations 379

Appendix C **Types and Units 381**

Type definition 381
Selectors 382
THE_ operators 383
Computational operators 384
Display operators 385
Type constraints 386
A more complex example 386

Appendix D **What Is a Database? 389**

Updating the database 389
Databases vs. tuples 390

Appendix E **View Updating 393**

Date's appoach 393
Assumptions and notation 394
A closer look at relational assignment 395
A model of updating 398
A closer look at constraints 399
Updating restrictions 403
Updating intersections 405
Updating unions 408
Updating differences 410
Nested updates 412
Updating extensions 417
Updating joins 418
Updating projections 424
Some remarks on orthogonality 428
A remark on multiple assignment 430
Summary 431
Darwen's approach 432

Appendix F **A Closer Look at Specialization by Constraint 443**

The 3 out of 4 "rule" 443
What does inheritance really mean? 448
Benefits of S by C 449
What about objects? 452
Implementation considerations 455

Appendix G **A Closer Look at Structural Inheritance 461**

An introductory example 461
Tuple types, values, and variables 463
Subtables and supertables 467
Scalar types revisited 471
Structural inheritance with fewer tears 472

Appendix H **A Comparison with SQL 479**

RM Prescriptions 480
RM Proscriptions 490
OO Prescriptions 491
OO Proscriptions 492
RM Very Strong Suggestions 493
OO Very Strong Suggestions 493
IM Prescriptions 494

Appendix I **A Grammar for Tutorial D 499**

Appendix J **References and Bibliography 513**

Index 545

此为试读,需要完整PDF请访问: www.ertongbook.com