

Table of Contents

| | | |
|----------|---|-----------|
| 1 | The Worlds of Database Systems | 1 |
| 1.1 | The Evolution of Database Systems | 1 |
| 1.1.1 | Early Database Management Systems | 2 |
| 1.1.2 | Relational Database Systems | 4 |
| 1.1.3 | Smaller and Smaller Systems | 5 |
| 1.1.4 | Bigger and Bigger Systems | 5 |
| 1.2 | The Architecture of a DBMS | 7 |
| 1.2.1 | Overview of DBMS Components | 7 |
| 1.2.2 | The Storage Manager | 9 |
| 1.2.3 | The Query Manager | 10 |
| 1.2.4 | The Transaction Manager | 11 |
| 1.2.5 | Client-Server Architecture | 14 |
| 1.3 | The Future of Database Systems | 14 |
| 1.3.1 | Types, Classes, and Objects | 14 |
| 1.3.2 | Constraints and Triggers | 18 |
| 1.3.3 | Multimedia Data | 18 |
| 1.3.4 | Data Integration | 19 |
| 1.4 | Outline of the Book | 20 |
| 1.4.1 | Design | 21 |
| 1.4.2 | Programming | 21 |
| 1.5 | Summary of Chapter 1 | 22 |
| 1.6 | References for Chapter 1 | 23 |
| 2 | Database Modeling | 25 |
| 2.1 | Introduction to ODL | 26 |
| 2.1.1 | Object-Oriented Design | 27 |
| 2.1.2 | Interface Declarations | 29 |
| 2.1.3 | Attributes in ODL | 29 |
| 2.1.4 | Relationships in ODL | 30 |
| 2.1.5 | Inverse Relationships | 31 |
| 2.1.6 | Multiplicity of Relationships | 33 |
| 2.1.7 | Types in ODL | 36 |
| 2.1.8 | Exercises for Section 2.1 | 38 |

| | | |
|-------|---|----|
| 2.2 | Entity-Relationship Diagrams | 40 |
| 2.2.1 | Multiplicity of E/R Relationships | 41 |
| 2.2.2 | Multiway Relationships | 42 |
| 2.2.3 | Roles in Relationships | 43 |
| 2.2.4 | Attributes on Relationships | 45 |
| 2.2.5 | Converting Multiway Relationships to Binary | 46 |
| 2.2.6 | Exercises for Section 2.2 | 48 |
| 2.3 | Design Principles | 50 |
| 2.3.1 | Faithfulness | 50 |
| 2.3.2 | Avoiding Redundancy | 51 |
| 2.3.3 | Simplicity Counts | 51 |
| 2.3.4 | Picking the Right Kind of Element | 52 |
| 2.3.5 | Exercises for Section 2.3 | 55 |
| 2.4 | Subclasses | 57 |
| 2.4.1 | Subclasses in ODL | 57 |
| 2.4.2 | Multiple Inheritance in ODL | 58 |
| 2.4.3 | Subclasses in Entity-Relationship Diagrams | 60 |
| 2.4.4 | Inheritance in the E/R Model | 60 |
| 2.4.5 | Exercises for Section 2.4 | 62 |
| 2.5 | The Modeling of Constraints | 63 |
| 2.5.1 | Keys | 64 |
| 2.5.2 | Declaring Keys in ODL | 66 |
| 2.5.3 | Representing Keys in the E/R Model | 67 |
| 2.5.4 | Single-Value Constraints | 67 |
| 2.5.5 | Referential Integrity | 68 |
| 2.5.6 | Referential Integrity in E/R Diagrams | 69 |
| 2.5.7 | Other Kinds of Constraints | 70 |
| 2.5.8 | Exercises for Section 2.5 | 71 |
| 2.6 | Weak Entity Sets | 72 |
| 2.6.1 | Causes of Weak Entity Sets | 72 |
| 2.6.2 | Requirements for Weak Entity Sets | 73 |
| 2.6.3 | Weak Entity Set Notation | 75 |
| 2.6.4 | Exercises for Section 2.6 | 76 |
| 2.7 | Models of Historical Interest | 77 |
| 2.7.1 | The Network Model | 77 |
| 2.7.2 | Representing Network Schemas | 78 |
| 2.7.3 | The Hierarchical Model | 79 |
| 2.7.4 | Exercises for Section 2.7 | 80 |
| 2.8 | Summary of Chapter 2 | 82 |
| 2.9 | References for Chapter 2 | 83 |

| | | |
|----------|--|-----------|
| 3 | The Relational Data Model | 85 |
| 3.1 | Basics of the Relational Model | 85 |
| 3.1.1 | Attributes | 86 |
| 3.1.2 | Schemas | 86 |
| 3.1.3 | Tuples | 87 |
| 3.1.4 | Domains | 87 |
| 3.1.5 | Equivalent Representations of a Relation | 88 |
| 3.1.6 | Relation Instances | 89 |
| 3.1.7 | Exercises for Section 3.1 | 90 |
| 3.2 | From ODL Designs to Relational Designs | 91 |
| 3.2.1 | From ODL Attributes to Relational Attributes | 92 |
| 3.2.2 | Nonatomic Attributes in Classes | 92 |
| 3.2.3 | Representing Other Type Constructors | 96 |
| 3.2.4 | Representing Single-Valued Relationships | 97 |
| 3.2.5 | Representing Multivalued Relationships | 99 |
| 3.2.6 | What If There Is No Key? | 100 |
| 3.2.7 | Representing a Relationship and Its Inverse | 101 |
| 3.2.8 | Exercises for Section 3.2 | 102 |
| 3.3 | From E/R Diagrams to Relational Designs | 103 |
| 3.3.1 | From Entity Sets to Relations | 104 |
| 3.3.2 | From E/R Relationships to Relations | 106 |
| 3.3.3 | Handling Weak Entity Sets | 108 |
| 3.3.4 | Exercises for Section 3.3 | 111 |
| 3.4 | Converting Subclass Structures to Relations | 112 |
| 3.4.1 | Relational Representation of ODL Subclasses | 113 |
| 3.4.2 | Representing Isa in the Relational Model | 114 |
| 3.4.3 | Comparison of Approaches | 115 |
| 3.4.4 | Using Null Values to Combine Relations | 115 |
| 3.4.5 | Exercises for Section 3.4 | 116 |
| 3.5 | Functional Dependencies | 118 |
| 3.5.1 | Definition of Functional Dependency | 118 |
| 3.5.2 | Keys of Relations | 121 |
| 3.5.3 | Superkeys | 122 |
| 3.5.4 | Discovering Keys for Relations | 122 |
| 3.5.5 | Keys for Relations Derived from ODL | 124 |
| 3.5.6 | Exercises for Section 3.5 | 126 |
| 3.6 | Rules About Functional Dependencies | 126 |
| 3.6.1 | The Splitting/Combining Rule | 127 |
| 3.6.2 | Trivial Dependencies | 128 |
| 3.6.3 | Computing the Closure of Attributes | 129 |
| 3.6.4 | The Transitive Rule | 132 |
| 3.6.5 | Closing Sets of Functional Dependencies | 134 |
| 3.6.6 | Exercises for Section 3.6 | 135 |
| 3.7 | Design of Relational Database Schemas | 137 |
| 3.7.1 | Anomalies | 138 |

| | | |
|----------|--|------------|
| 3.7.2 | Decomposing Relations | 138 |
| 3.7.3 | Boyce-Codd Normal Form | 140 |
| 3.7.4 | Decomposition into BCNF | 142 |
| 3.7.5 | Projecting Functional Dependencies | 147 |
| 3.7.6 | Recovering Information from a Decomposition | 148 |
| 3.7.7 | Third Normal Form | 151 |
| 3.7.8 | Exercises for Section 3.7 | 154 |
| 3.8 | Multivalued Dependencies | 155 |
| 3.8.1 | Attribute Independence and Its Consequent Redundancy | 156 |
| 3.8.2 | Definition of Multivalued Dependencies | 157 |
| 3.8.3 | Reasoning About Multivalued Dependencies | 159 |
| 3.8.4 | Fourth Normal Form | 161 |
| 3.8.5 | Decomposition into Fourth Normal Form | 162 |
| 3.8.6 | Relationships Among Normal Forms | 163 |
| 3.8.7 | Exercises for Section 3.8 | 164 |
| 3.9 | An Example Database Schema | 166 |
| 3.10 | Summary of Chapter 3 | 169 |
| 3.11 | References for Chapter 3 | 171 |
| 4 | Operations in the Relational Model | 173 |
| 4.1 | An Algebra of Relational Operations | 173 |
| 4.1.1 | Set Operations on Relations | 174 |
| 4.1.2 | Projection | 176 |
| 4.1.3 | Selection | 177 |
| 4.1.4 | Cartesian Product | 178 |
| 4.1.5 | Natural Joins | 179 |
| 4.1.6 | Theta-Joins | 180 |
| 4.1.7 | Combining Operations to Form Queries | 182 |
| 4.1.8 | Renaming | 185 |
| 4.1.9 | Dependent and Independent Operations | 185 |
| 4.1.10 | Exercises for Section 4.1 | 187 |
| 4.2 | A Logic for Relations | 194 |
| 4.2.1 | Predicates and Atoms | 194 |
| 4.2.2 | Arithmetic Atoms | 195 |
| 4.2.3 | Datalog Rules and Queries | 196 |
| 4.2.4 | Meaning of Datalog Rules | 197 |
| 4.2.5 | Extensional and Intensional Predicates | 200 |
| 4.2.6 | Exercises for Section 4.2 | 200 |
| 4.3 | From Relational Algebra to Datalog | 201 |
| 4.3.1 | Intersection | 201 |
| 4.3.2 | Union | 201 |
| 4.3.3 | Difference | 202 |
| 4.3.4 | Projection | 202 |
| 4.3.5 | Selection | 203 |
| 4.3.6 | Product | 205 |

| | | |
|----------|--|------------|
| 4.3.7 | Joins | 205 |
| 4.3.8 | Simulating Multiple Operations with Datalog | 207 |
| 4.3.9 | Exercises for Section 4.3 | 208 |
| 4.4 | Recursive Programming in Datalog | 209 |
| 4.4.1 | The Fixedpoint Operator | 211 |
| 4.4.2 | Computing the Least Fixedpoint | 211 |
| 4.4.3 | Fixedpoint Equations in Datalog | 213 |
| 4.4.4 | Negation in Recursive Rules | 218 |
| 4.4.5 | Exercises for Section 4.4 | 221 |
| 4.5 | Constraints on Relations | 223 |
| 4.5.1 | Relational Algebra as a Constraint Language | 223 |
| 4.5.2 | Referential Integrity Constraints | 224 |
| 4.5.3 | Additional Constraint Examples | 225 |
| 4.5.4 | Exercises for Section 4.5 | 227 |
| 4.6 | Relational Operations on Bags | 229 |
| 4.6.1 | Why Bags? | 229 |
| 4.6.2 | Union, Intersection, and Difference of Bags | 230 |
| 4.6.3 | Projection of Bags | 232 |
| 4.6.4 | Selection on Bags | 233 |
| 4.6.5 | Product of Bags | 234 |
| 4.6.6 | Joins of Bags | 235 |
| 4.6.7 | Datalog Rules Applied to Bags | 235 |
| 4.6.8 | Exercises for Section 4.6 | 237 |
| 4.7 | Other Extensions to the Relational Model | 238 |
| 4.7.1 | Modifications | 238 |
| 4.7.2 | Aggregations | 238 |
| 4.7.3 | Views | 239 |
| 4.7.4 | Null Values | 239 |
| 4.8 | Summary of Chapter 4 | 240 |
| 4.9 | References for Chapter 4 | 241 |
| 5 | The Database Language SQL | 243 |
| 5.1 | Simple Queries in SQL | 244 |
| 5.1.1 | Projection in SQL | 245 |
| 5.1.2 | Selection in SQL | 247 |
| 5.1.3 | Comparison of Strings | 248 |
| 5.1.4 | Comparing Dates and Times | 251 |
| 5.1.5 | Ordering the Output | 251 |
| 5.1.6 | Exercises for Section 5.1 | 252 |
| 5.2 | Queries Involving More than One Relation | 254 |
| 5.2.1 | Products and Joins in SQL | 254 |
| 5.2.2 | Disambiguating Attributes | 255 |
| 5.2.3 | Tuple Variables | 256 |
| 5.2.4 | Interpreting Multirelation Queries | 257 |
| 5.2.5 | Union, Intersection, and Difference of Queries | 260 |

| | | |
|-------|--|-----|
| 5.2.6 | Exercises for Section 5.2 | 262 |
| 5.3 | Subqueries | 263 |
| 5.3.1 | Subqueries that Produce Scalar Values | 264 |
| 5.3.2 | Conditions Involving Relations | 265 |
| 5.3.3 | Conditions Involving Tuples | 266 |
| 5.3.4 | Correlated Subqueries | 267 |
| 5.3.5 | Exercises for Section 5.3 | 269 |
| 5.4 | Duplicates | 270 |
| 5.4.1 | Eliminating Duplicates | 271 |
| 5.4.2 | Duplicates in Unions, Intersections, and Differences | 271 |
| 5.4.3 | Exercises for Section 5.4 | 273 |
| 5.5 | Aggregation | 273 |
| 5.5.1 | Aggregation Operators | 273 |
| 5.5.2 | Grouping | 274 |
| 5.5.3 | HAVING Clauses | 276 |
| 5.5.4 | Exercises for Section 5.5 | 277 |
| 5.6 | Database Modifications | 279 |
| 5.6.1 | Insertion | 279 |
| 5.6.2 | Deletion | 281 |
| 5.6.3 | Updates | 283 |
| 5.6.4 | Exercises for Section 5.6 | 284 |
| 5.7 | Defining a Relation Schema in SQL | 285 |
| 5.7.1 | Data Types | 286 |
| 5.7.2 | Simple Table Declarations | 286 |
| 5.7.3 | Deleting Tables | 287 |
| 5.7.4 | Modifying Relation Schemas | 288 |
| 5.7.5 | Default Values | 288 |
| 5.7.6 | Domains | 289 |
| 5.7.7 | Indexes | 290 |
| 5.7.8 | Exercises for Section 5.7 | 292 |
| 5.8 | View Definitions | 294 |
| 5.8.1 | Declaring Views | 294 |
| 5.8.2 | Querying Views | 295 |
| 5.8.3 | Renaming Attributes | 297 |
| 5.8.4 | Modifying Views | 297 |
| 5.8.5 | Interpreting Queries Involving Views | 300 |
| 5.8.6 | Exercises for Section 5.8 | 303 |
| 5.9 | Null Values and Outerjoins | 304 |
| 5.9.1 | Operations on Nulls | 304 |
| 5.9.2 | The Truth-Value UNKNOWN | 306 |
| 5.9.3 | SQL2 Join Expressions | 307 |
| 5.9.4 | Natural Joins | 309 |
| 5.9.5 | Outerjoins | 309 |
| 5.9.6 | Exercises for Section 5.9 | 311 |
| 5.10 | Recursion in SQL3 | 313 |

| | | |
|----------|---|------------|
| 5.10.1 | Defining IDB Relations in SQL3 | 313 |
| 5.10.2 | Linear Recursion | 316 |
| 5.10.3 | Use of Views in With-Statements | 317 |
| 5.10.4 | Stratified Negation | 318 |
| 5.10.5 | Problematic Expressions in Recursive SQL3 | 319 |
| 5.10.6 | Exercises for Section 5.10 | 322 |
| 5.11 | Summary of Chapter 5 | 324 |
| 5.12 | References for Chapter 5 | 326 |
| 6 | Constraints and Triggers in SQL | 327 |
| 6.1 | Keys in SQL | 328 |
| 6.1.1 | Declaring Keys | 328 |
| 6.1.2 | Enforcing Key Constraints | 330 |
| 6.1.3 | Exercises for Section 6.1 | 331 |
| 6.2 | Referential Integrity and Foreign Keys | 331 |
| 6.2.1 | Declaring Foreign-Key Constraints | 331 |
| 6.2.2 | Maintaining Referential Integrity | 333 |
| 6.2.3 | Exercises for Section 6.2 | 335 |
| 6.3 | Constraints on the Values of Attributes | 336 |
| 6.3.1 | Not-Null Constraints | 337 |
| 6.3.2 | Attribute-Based CHECK Constraints | 337 |
| 6.3.3 | Domain Constraints | 339 |
| 6.3.4 | Exercises for Section 6.3 | 340 |
| 6.4 | Global Constraints | 341 |
| 6.4.1 | Tuple-Based CHECK Constraints | 341 |
| 6.4.2 | Assertions | 342 |
| 6.4.3 | Exercises for Section 6.4 | 346 |
| 6.5 | Modification of Constraints | 348 |
| 6.5.1 | Giving Names to Constraints | 348 |
| 6.5.2 | Altering Constraints on Tables | 349 |
| 6.5.3 | Altering Domain Constraints | 350 |
| 6.5.4 | Altering Assertions | 351 |
| 6.5.5 | Exercises for Section 6.5 | 351 |
| 6.6 | Triggers in SQL3 | 352 |
| 6.6.1 | Triggers and Constraints | 352 |
| 6.6.2 | SQL3 Triggers | 353 |
| 6.6.3 | Assertions in SQL3 | 356 |
| 6.6.4 | Exercises for Section 6.6 | 357 |
| 6.7 | Summary of Chapter 6 | 359 |
| 6.8 | References for Chapter 6 | 360 |
| 7 | System Aspects of SQL | 361 |
| 7.1 | SQL in a Programming Environment | 361 |
| 7.1.1 | The Impedance Mismatch Problem | 362 |
| 7.1.2 | The SQL/Host Language Interface | 363 |

| | | |
|----------|--|------------|
| 7.1.3 | The DECLARE Section | 364 |
| 7.1.4 | Using Shared Variables | 365 |
| 7.1.5 | Single-Row Select Statements | 366 |
| 7.1.6 | Cursors | 367 |
| 7.1.7 | Modifications by Cursor | 370 |
| 7.1.8 | Cursor Options | 370 |
| 7.1.9 | Ordering Tuples for Fetching | 371 |
| 7.1.10 | Protecting Against Concurrent Updates | 372 |
| 7.1.11 | Scrolling Cursors | 373 |
| 7.1.12 | Dynamic SQL | 374 |
| 7.1.13 | Exercises for Section 7.1 | 375 |
| 7.2 | Transactions in SQL | 378 |
| 7.2.1 | Serializability | 378 |
| 7.2.2 | Atomicity | 380 |
| 7.2.3 | Transactions | 382 |
| 7.2.4 | Read-Only Transactions | 384 |
| 7.2.5 | Dirty Reads | 385 |
| 7.2.6 | Other Isolation Levels | 387 |
| 7.2.7 | Exercises for Section 7.2 | 388 |
| 7.3 | The SQL Environment | 389 |
| 7.3.1 | Environments | 389 |
| 7.3.2 | Schemas | 391 |
| 7.3.3 | Catalogs | 392 |
| 7.3.4 | Clients and Servers in the SQL Environment | 392 |
| 7.3.5 | Connections | 393 |
| 7.3.6 | Sessions | 394 |
| 7.3.7 | Modules | 395 |
| 7.4 | Security and User Authorization in SQL2 | 395 |
| 7.4.1 | Privileges | 396 |
| 7.4.2 | Creating Privileges | 397 |
| 7.4.3 | The Privilege-Checking Process | 398 |
| 7.4.4 | Granting Privileges | 399 |
| 7.4.5 | Grant Diagrams | 401 |
| 7.4.6 | Revoking Privileges | 402 |
| 7.4.7 | Exercises for Section 7.4 | 407 |
| 7.5 | Summary of Chapter 7 | 408 |
| 7.6 | References for Chapter 7 | 410 |
| 8 | Object-Oriented Query Languages | 411 |
| 8.1 | Query-Related Features of ODL | 412 |
| 8.1.1 | Operations on ODL Objects | 412 |
| 8.1.2 | Declaring Method Signatures in ODL | 412 |
| 8.1.3 | The Extent of a Class | 415 |
| 8.1.4 | Exercises for Section 8.1 | 415 |
| 8.2 | Introduction to OQL | 417 |

| | | |
|-------|---|-----|
| 8.2.1 | An Object-Oriented Movie Example | 419 |
| 8.2.2 | The OQL Type System | 419 |
| 8.2.3 | Path Expressions | 420 |
| 8.2.4 | Select-From-Where Expressions in OQL | 421 |
| 8.2.5 | Eliminating Duplicates | 423 |
| 8.2.6 | Complex Output Types | 423 |
| 8.2.7 | Subqueries | 424 |
| 8.2.8 | Ordering the Result | 425 |
| 8.2.9 | Exercises for Section 8.2 | 426 |
| 8.3 | Additional Forms of OQL Expressions | 427 |
| 8.3.1 | Quantifier Expressions | 427 |
| 8.3.2 | Aggregation Expressions | 428 |
| 8.3.3 | Group-By Expressions | 428 |
| 8.3.4 | HAVING Clauses | 431 |
| 8.3.5 | Set Operators | 432 |
| 8.3.6 | Exercises for Section 8.3 | 433 |
| 8.4 | Object Assignment and Creation in OQL | 434 |
| 8.4.1 | Assigning Values to Host-Language Variables | 434 |
| 8.4.2 | Extracting Elements of Collections | 434 |
| 8.4.3 | Obtaining Each Member of a Collection | 435 |
| 8.4.4 | Creating New Objects | 436 |
| 8.4.5 | Exercises for Section 8.4 | 438 |
| 8.5 | Tuple Objects in SQL3 | 439 |
| 8.5.1 | Row Types | 439 |
| 8.5.2 | Declaring Relations with a Row Type | 440 |
| 8.5.3 | Accessing Components of a Row Type | 441 |
| 8.5.4 | References | 441 |
| 8.5.5 | Following References | 442 |
| 8.5.6 | Scopes of References | 444 |
| 8.5.7 | Object Identifiers as Values | 445 |
| 8.5.8 | Exercises for Section 8.5 | 448 |
| 8.6 | Abstract Data Types in SQL3 | 449 |
| 8.6.1 | Defining ADT's | 450 |
| 8.6.2 | Defining Methods for ADT's | 453 |
| 8.6.3 | External Functions | 456 |
| 8.6.4 | Exercises for Section 8.6 | 456 |
| 8.7 | A Comparison of the ODL/OQL and SQL3 Approaches | 458 |
| 8.8 | Summary of Chapter 8 | 459 |
| 8.9 | References for Chapter 8 | 460 |