Practical Database Programming with Java

Ying Bai
Department of Computer Science and Engineering
Johnson C. Smith University
Charlotte, North Carolina

IEEE PRESS

A John Wiley & Sons, Inc., Publication
Practical Database Programming with Java
IEEE Press
445 Hoes Lane
Piscataway, NJ 08854

IEEE Press Editorial Board
Lajos Hanzo, Editor in Chief

R. Abhari        M. El-Hawary        O. P. Malik
J. Anderson      B-M. Haemmerli     S. Nahavandi
G. W. Arnold     M. Lanzorotti      T. Samad
F. Canavero      D. Jacobson        G. Zobrist

Kenneth Moore, Director of IEEE Book and Information Services (BIS)
Practical Database Programming with Java

Ying Bai
Department of Computer Science and Engineering
Johnson C. Smith University
Charlotte, North Carolina

IEEE PRESS

WILEY
A John Wiley & Sons, Inc., Publication
This book is dedicated to my wife, Yan Wang, and to my daughter, Xue Bai.
## Contents

 Preface xxiii

Acknowledgments xxv

Chapter 1 Introduction 1

<table>
<thead>
<tr>
<th>What This Book Covers</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>How This Book Is Organized and How to Use This Book</td>
<td>3</td>
</tr>
<tr>
<td>How to Use the Source Code and Sample Databases</td>
<td>5</td>
</tr>
<tr>
<td>Instructor and Customer Support</td>
<td>6</td>
</tr>
<tr>
<td>Homework Solutions</td>
<td>7</td>
</tr>
</tbody>
</table>

Chapter 2 Introduction to Databases 9

<table>
<thead>
<tr>
<th>2.1 What Are Databases and Database Programs?</th>
<th>10</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1.1 File Processing System</td>
<td>10</td>
</tr>
<tr>
<td>2.1.2 Integrated Databases</td>
<td>11</td>
</tr>
<tr>
<td>2.2 Develop a Database</td>
<td>12</td>
</tr>
<tr>
<td>2.3 Sample Database</td>
<td>13</td>
</tr>
<tr>
<td>2.3.1 Relational Data Model</td>
<td>15</td>
</tr>
<tr>
<td>2.3.2 Entity–Relationship Model</td>
<td>16</td>
</tr>
<tr>
<td>2.4 Identifying Keys</td>
<td>17</td>
</tr>
<tr>
<td>2.4.1 Primary Key and Entity Integrity</td>
<td>17</td>
</tr>
<tr>
<td>2.4.2 Candidate Key</td>
<td>17</td>
</tr>
<tr>
<td>2.4.3 Foreign Keys and Referential Integrity</td>
<td>17</td>
</tr>
<tr>
<td>2.5 Define Relationships</td>
<td>18</td>
</tr>
<tr>
<td>2.5.1 Connectivity</td>
<td>18</td>
</tr>
<tr>
<td>2.6 ER Notation</td>
<td>21</td>
</tr>
<tr>
<td>2.7 Data Normalization</td>
<td>21</td>
</tr>
<tr>
<td>2.7.1 First Normal Form (1NF)</td>
<td>22</td>
</tr>
<tr>
<td>2.7.2 Second Normal Form (2NF)</td>
<td>23</td>
</tr>
<tr>
<td>2.7.3 Third Normal Form (3NF)</td>
<td>24</td>
</tr>
</tbody>
</table>
Contents

2.8 Database Components in Some Popular Databases 26
  2.8.1 Microsoft Access Databases 26
    2.8.1.1 Database File 27
    2.8.1.2 Tables 27
    2.8.1.3 Queries 27
  2.8.2 SQL Server Databases 27
    2.8.2.1 Data Files 28
    2.8.2.2 Tables 28
    2.8.2.3 Views 29
    2.8.2.4 Stored Procedures 29
    2.8.2.5 Keys and Relationships 29
    2.8.2.6 Indexes 30
    2.8.2.7 Transaction Log Files 30
  2.8.3 Oracle Databases 30
    2.8.3.1 Data Files 31
    2.8.3.2 Tables 31
    2.8.3.3 Views 31
    2.8.3.4 Stored Procedures 31
    2.8.3.5 Indexes 32
    2.8.3.6 Initialization Parameter Files 33
    2.8.3.7 Control Files 33
    2.8.3.8 Redo log Files 33
    2.8.3.9 Password Files 34

2.9 Create Microsoft Access Sample Database 34
  2.9.1 Create the LogIn Table 34
  2.9.2 Create the Faculty Table 36
  2.9.3 Create the Other Tables 37
  2.9.4 Create Relationships among Tables 39

2.10 Create Microsoft SQL Server 2008 Sample Database 44
  2.10.1 Create the LogIn Table 46
  2.10.2 Create the Faculty Table 48
  2.10.3 Create Other Tables 49
  2.10.4 Create Relationships among Tables 54
    2.10.4.1 Create Relationship between the LogIn and the Faculty Tables 54
    2.10.4.2 Create Relationship between the LogIn and the Student Tables 57
    2.10.4.3 Create Relationship between the Faculty and the Course Tables 58
    2.10.4.4 Create Relationship between the Student and the StudentCourse Tables 59
    2.10.4.5 Create Relationship between the Course and the StudentCourse Tables 60

2.11 Create Oracle 10g XE Sample Database 61
  2.11.1 Create an Oracle User Database 63
  2.11.2 Add New Data Tables into the Oracle User Database 64
    2.11.2.1 Create the LogIn Table 65
    2.11.2.2 Create the Faculty Table 69
    2.11.2.3 Create Other Tables 74
Chapter 3 JDBC API and JDBC Drivers

3.1 What Are JDBC and JDBC API? 89
3.2 JDBC Components and Architecture 90
3.3 How Does JDBC Work? 92
   3.3.1 Establish a Connection 92
       3.3.1.1 Using DriverManager to Establish a Connection 92
       3.3.1.2 Using DataSource Object to Establish a Connection 93
   3.3.2 Build and Execute SQL Statements 94
   3.3.3 Process Results 94
       3.3.3.1 Using ResultSet Object 95
       3.3.3.2 Using RowSet Object 95
3.4 JDBC Driver and Driver Types 95
   3.4.1 Type I: JDBC-ODBC Bridge Driver 96
   3.4.2 Type II: Native-API-Partly-Java Driver 97
   3.4.3 Type III: JDBC-Net-All-Java Driver 97
   3.4.4 Type IV: Native-Protocol-All-Java Driver 98
3.5 JDBC Standard Extension API 99
   3.5.1 JDBC DataSource 99
       3.5.1.1 Java Naming and Directory Interface 100
       3.5.1.2 Deploy and Use a Basic Implementation of DataSource 100
   3.5.2 JDBC Driver-Based Connection Pooling 102
   3.5.3 Distributed Transactions 104
       3.5.3.1 Distributed Transaction Components and Scenarios 104
       3.5.3.2 The Distributed Transaction Process 105
   3.5.4 JDBC RowSet 106
       3.5.4.1 Introduction to Java RowSet Object 106
       3.5.4.2 Implementation Process of a RowSet Object 107
3.6 Chapter Summary 108
   Homework 109

Chapter 4 JDBC Application Design Considerations

4.1 JDBC Application Models 113
   4.1.1 Two-Tier Client-Server Model 113
   4.1.2 Three-Tier Client–Server Model 114
4.2 JDBC Applications Fundamentals 115
4.2.1 Loading and Registering Drivers 116
4.2.2 Getting Connected 117
   4.2.2.1 The DriverManager and Driver Classes 117
   4.2.2.2 Using the DriverManager.getConnection() Method 119
   4.2.2.3 Using the Driver.connect() Method 119
   4.2.2.4 The JDBC Connection URL 120
   4.2.2.5 Establish a Database Connection 120
4.2.3 Executing Statements 122
   4.2.3.1 Overview of Statement Objects and Their Execution Methods 123
   4.2.3.2 Using the Statement Object 126
   4.2.3.3 Using the PreparedStatement Object 127
   4.2.3.4 Using the CallableStatement Object 131
   4.2.3.5 More about the Execution Methods 135
   4.2.3.6 Creating and Executing SQL Statements 137
4.2.4 Retrieving Results 140
   4.2.4.1 The ResultSet Interface 141
   4.2.4.2 Getting and Processing the ResultSet Object 142
4.2.5 Using JDBC MetaData Interfaces 145
   4.2.5.1 Using the ResultSetMetaData Interface 145
   4.2.5.2 Using the DatabaseMetaData Interface 147
   4.2.5.3 Using the ParameterMetaData Interface 149
4.2.6 Closing the Connection and Statements 149
4.3 Chapter Summary 151
Homework 152

Chapter 5 Introduction to NetBeans IDE 155
5.1 Overview of the NetBeans IDE 6.8 156
   5.1.1 The NetBeans Platform 158
   5.1.2 The NetBeans Open Source IDE 159
5.2 Installing and Configuring the NetBeans IDE 6.8 161
5.3 Exploring NetBeans IDE 6.8 164
   5.3.1 An Overview of NetBeans IDE 6.8 GUI 165
   5.3.2 Build a New Java Project 167
      5.3.2.1 Build a Java Application Project 168
      5.3.2.2 Build a Java Desktop Application 178
      5.3.2.3 Build a Java Class Library 183
      5.3.2.4 Build a Java Project with Existing Sources 191
      5.3.2.5 Build a Java Free-Form Project 192
   5.3.3 Build a JavaFX Application Project 193
      5.3.3.1 Overview of JavaFX 193
      5.3.3.2 JavaFX SDK 194
      5.3.3.3 JavaFX Script Language 195
      5.3.3.4 Build a JavaFX Script Application 195
      5.3.3.5 Build a JavaFX Desktop Business Application 201
      5.3.3.6 Build JavaFX Mobile Business Application 214
Section II  Insert, Update and Delete Data Using Java Runtime Objects Method  

7.3 Perform Data Manipulations to SQL Server Database Using Java Runtime Object  
7.3.1 Perform Data Insertion to SQL Server Database Using Java Runtime Object  
7.3.1.1 Modify the FacultyFrame Window Form  
7.3.1.2 Develop the Codes for the Insert Button Event Handler  
7.3.1.3 Develop the Codes for the Validation of the Data Insertion  
7.3.1.4 Build and Run the Project to Test the Data Insertion  
7.3.2 Perform Data Updating to SQL Server Database Using Java Runtime Object  
7.3.2.1 Develop the Codes for the Update Button Event Handler  
7.3.2.2 Build and Run the Project to Test the Data Updating  
7.3.3 Perform Data Deleting to SQL Server Database Using Java Runtime Object  
7.3.3.1 Develop the Codes for the Delete Button Event Handler  
7.3.3.2 Build and Run the Project to Test the Data Deleting  

7.4 Perform Data Manipulations to Oracle Database Using Java Runtime Object  
7.4.1 Perform Data Insertion to Oracle Database Using Java Runtime Object  
7.4.1.1 Modify the FacultyFrame Window Form  
7.4.1.2 Develop the Codes for the Insert Button Event Handler  
7.4.2 Perform Data Updating to Oracle Database Using Java Runtime Object  
7.4.3 Perform Data Deleting to Oracle Database Using Java Runtime Object  

7.5 Perform Data Manipulations Using Updatable ResultSet  
7.5.1 Introduction to ResultSet Enhanced Functionalities and Categories  
7.5.2 Perform Data Manipulations Using Updatable ResultSet Object  
7.5.2.1 Insert a New Row Using the Updatable ResultSet  
7.5.2.2 Update a Row Using the Updatable ResultSet  
7.5.2.3 Delete a Row Using the Updatable ResultSet  

7.6 Perform Data Manipulations Using Callable Statements  
7.6.1 Perform Data Manipulations to SQL Server Database Using Callable Statements  
7.6.1.1 Insert Data to SQL Server Database Using Callable Statements  
7.6.1.2 Update Data to SQL Server Database Using Callable Statements  
7.6.1.3 Delete Data from SQL Server Database Using Callable Statements  
7.6.2 Perform Data Manipulations to Oracle Database Using Callable Statements  
7.6.2.1 Modify the CourseFrame Form Window  
7.6.2.2 Build Three Oracle Stored Procedures  
7.6.2.3 Build and Run the Project to Test the Data Manipulations
### PART I  Building Two-Tier Client–Server Applications  317

#### Chapter 6  Query Data from Databases  319

##### Section I  Query Data Using Java Persistence API Wizards  319

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1</td>
<td>Java Persistence APIs</td>
<td>319</td>
</tr>
<tr>
<td>6.1.1</td>
<td>Features of JPA</td>
<td>320</td>
</tr>
<tr>
<td>6.1.2</td>
<td>Advantages of JPA</td>
<td>320</td>
</tr>
<tr>
<td>6.1.3</td>
<td>Architecture and Function of JPA</td>
<td>320</td>
</tr>
<tr>
<td>6.2</td>
<td>Query Data Using Java Persistence API Wizards (JPA)</td>
<td>321</td>
</tr>
<tr>
<td>6.2.1</td>
<td>Connect to Different Databases and Drivers Using JPA Wizards</td>
<td>322</td>
</tr>
<tr>
<td>6.2.1.1</td>
<td>Connect to the Microsoft Access Database CSE_DEPT</td>
<td>322</td>
</tr>
<tr>
<td>6.2.1.2</td>
<td>Connect to the Microsoft SQL Server 2008 Express Database CSE_DEPT</td>
<td>323</td>
</tr>
<tr>
<td>6.2.1.3</td>
<td>Connect to the Oracle Database 10g Express Edition CSE_DEPT</td>
<td>333</td>
</tr>
<tr>
<td>6.2.2</td>
<td>Create a Java Application Project to Query SQL Server Database</td>
<td>338</td>
</tr>
<tr>
<td>6.2.3</td>
<td>Use Java JPA Wizards to Query the LogIn Table</td>
<td>340</td>
</tr>
<tr>
<td>6.2.4</td>
<td>Use Java Persistence API to Build Entity Classes from Databases</td>
<td>341</td>
</tr>
<tr>
<td>6.2.5</td>
<td>Add LogIn Entity Manager and JPA Components into the Project</td>
<td>344</td>
</tr>
<tr>
<td>6.2.5.1</td>
<td>Entity Classes Mapping Files</td>
<td>345</td>
</tr>
<tr>
<td>6.2.5.2</td>
<td>Use Java Persistence Query Language Statement</td>
<td>346</td>
</tr>
<tr>
<td>6.2.5.3</td>
<td>Static and Dynamic JPA Query API</td>
<td>346</td>
</tr>
<tr>
<td>6.2.5.4</td>
<td>Positional Parameters and Named Parameters</td>
<td>348</td>
</tr>
<tr>
<td>6.2.5.5</td>
<td>Use Entity Classes to Build a Query to Perform the Login Process</td>
<td>349</td>
</tr>
<tr>
<td>6.2.5.6</td>
<td>Use a JDialog as a MessageBox</td>
<td>351</td>
</tr>
<tr>
<td>6.2.6</td>
<td>Use Java JPA Wizards to Create Selection Window</td>
<td>354</td>
</tr>
<tr>
<td>6.2.6.1</td>
<td>Add a New JFrame as the SelectionFrame Form</td>
<td>354</td>
</tr>
<tr>
<td>6.2.6.2</td>
<td>Modify Codes to Coordinate Operations in SelectionFrame and LogInFrame</td>
<td>358</td>
</tr>
<tr>
<td>6.2.7</td>
<td>Use Java JPA Wizards to Query the Faculty Table</td>
<td>360</td>
</tr>
<tr>
<td>6.2.7.1</td>
<td>Create a New FacultyFrame Class and Add It into Our Project</td>
<td>360</td>
</tr>
<tr>
<td>6.2.7.2</td>
<td>Add Faculty Entity Manager and JPA Components into the Project</td>
<td>362</td>
</tr>
<tr>
<td>6.2.7.3</td>
<td>Use Entity Classes to Perform Data Query from the Faculty Table</td>
<td>363</td>
</tr>
</tbody>
</table>
### Section II  Query Data Using Java Runtime Objects Method  383

6.3  Introduction to Runtime Object Method  383

6.4  Create a Java Application Project to Access the SQL Server Database  384

6.4.1  Create Graphic User Interfaces  384

6.4.2  Perform the Data Query for the LogIn Table  388

6.4.2.1  Load and Register Database Drivers  389

6.4.2.2  Connect to Databases and Drivers  393

6.4.2.3  Create and Manage Statement Object  394

6.4.2.4  Use PreparedStatement Object to Perform Dynamic Query  395

6.4.2.5  Use ResultSet Object  398

6.4.3  Develop the Codes for the SelectionFrame Form  399

6.4.3.1  Modify Codes to Coordinate between SelectionFrame and LogInFrame  402

6.4.4  Perform the Data Query for the Faculty Table  403

6.4.4.1  Add Java Package and Coding for the Constructor  403

6.4.4.2  Query Data using JDBC MetaData Interface  404

6.4.4.3  Query Data Using the execute() Method to Perform a Query-Related Action  410

6.4.4.4  Query Data Using the CallableStatement Method  412

6.4.5  Perform the Data Query for the Course Table  412

6.4.5.1  Import Java Packages and Coding for the CourseFrame Constructor  413

6.4.5.2  Query Data from Course Table Using CallableStatements  414

6.4.5.3  Coding for the Select Button Click Event Handler to Perform CallableStatement Query  420

6.4.5.4  Build the SQL Stored Procedure dbo.FacultyCourse  421

6.4.5.5  Coding for the CourseList Box to Display Detailed Information for the Selected Course  427

6.4.5.6  Coding for the Back Button Click Event Handler  429

6.4.6  Query Data from the Student Table Using the Java RowSet Object  430

6.4.6.1  Introduction to Java RowSet Object  430

6.4.6.2  The Operational Procedure of Using the JDBC RowSet Object  432

6.4.6.3  Build a Graphical User Interface StudentFrame Form  433

6.4.6.4  Coding for the Constructor of the StudentFrame Class  435

6.4.6.5  Coding for the Select Button Event Handler to Query Data Using the CachedRowSet  436

6.4.6.6  Add and Display a Student Picture for the Selected Student  439

---

6.2.8  Use Java JPA Wizards to Query the Course Table  372

6.2.8.1  Create a New CourseFrame Class and Add It into Our Project  372

6.2.8.2  Add Course Entity Manager and JPA Components into the Project  373

6.2.8.3  Use Entity Classes to Perform Data Query from the Course Table  374

6.2.9  Use Java JPA Wizards to Query Oracle Database  381
Section II  Insert, Update and Delete Data Using Java Runtime Objects Method  488

7.3 Perform Data Manipulations to SQL Server Database Using Java Runtime Object  488
  7.3.1 Perform Data Insertion to SQL Server Database Using Java Runtime Object  488
    7.3.1.1 Modify the FacultyFrame Window Form  489
    7.3.1.2 Develop the Codes for the Insert Button Event Handler  490
    7.3.1.3 Develop the Codes for the Validation of the Data Insertion  492
    7.3.1.4 Build and Run the Project to Test the Data Insertion  493
  7.3.2 Perform Data Updating to SQL Server Database Using Java Runtime Object  496
    7.3.2.1 Develop the Codes for the Update Button Event Handler  496
    7.3.2.2 Build and Run the Project to Test the Data Updating  497
  7.3.3 Perform Data Deleting to SQL Server Database Using Java Runtime Object  499
    7.3.3.1 Develop the Codes for the Delete Button Event Handler  499
    7.3.3.2 Build and Run the Project to Test the Data Deleting  500

7.4 Perform Data Manipulations to Oracle Database Using Java Runtime Object  502
  7.4.1 Perform Data Insertion to Oracle Database Using Java Runtime Object  503
    7.4.1.1 Modify the FacultyFrame Window Form  503
    7.4.1.2 Develop the Codes for the Insert Button Event Handler  504
  7.4.2 Perform Data Updating to Oracle Database Using Java Runtime Object  507
  7.4.3 Perform Data Deleting to Oracle Database Using Java Runtime Object  509

7.5 Perform Data Manipulations Using Updatable ResultSet  510
  7.5.1 Introduction to ResultSet Enhanced Functionalities and Categories  510
  7.5.2 Perform Data Manipulations Using Updatable ResultSet Object  512
    7.5.2.1 Insert a New Row Using the Updatable ResultSet  512
    7.5.2.2 Update a Row Using the Updatable ResultSet  517
    7.5.2.3 Delete a Row Using the Updatable ResultSet  520

7.6 Perform Data Manipulations Using Callable Statements  522
  7.6.1 Perform Data Manipulations to SQL Server Database Using Callable Statements  523
    7.6.1.1 Insert Data to SQL Server Database Using Callable Statements  523
    7.6.1.2 Update Data to SQL Server Database Using Callable Statements  530
    7.6.1.3 Delete Data from SQL Server Database Using Callable Statements  536
  7.6.2 Perform Data Manipulations to Oracle Database Using Callable Statements  540
    7.6.2.1 Modify the CourseFrame Form Window  541
    7.6.2.2 Build Three Oracle Stored Procedures  542
    7.6.2.3 Build and Run the Project to Test the Data Manipulations  547
## Contents

7.7 Chapter Summary 550  
Homework 551

### PART II Building Three-Tier Client–Server Applications 555

#### Chapter 8 Developing Java Web Applications to Access Databases 557

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.1</td>
<td>A Historical Review about Java Web Application Development</td>
<td>557</td>
</tr>
<tr>
<td>8.1.1</td>
<td>Using Servlet and HTML Web Pages for Java Web Applications</td>
<td>558</td>
</tr>
<tr>
<td>8.1.2</td>
<td>Using JavaServer Pages (JSP) Technology for Java Web Applications</td>
<td>560</td>
</tr>
<tr>
<td>8.1.3</td>
<td>Using Java Help Class Files for Java Web Applications</td>
<td>564</td>
</tr>
<tr>
<td>8.1.4</td>
<td>Using Java Persistence APIs for Java Web Applications</td>
<td>569</td>
</tr>
<tr>
<td>8.1.5</td>
<td>Using the JSP Implicit Object Session for Java Web Applications</td>
<td>572</td>
</tr>
<tr>
<td>8.1.5.1</td>
<td>Modify the FacultyPage JSP File to Use the Session Object</td>
<td>572</td>
</tr>
<tr>
<td>8.1.5.2</td>
<td>Build the Transaction JSP File FacultyQuery.jsp</td>
<td>574</td>
</tr>
<tr>
<td>8.1.5.3</td>
<td>Build the Help Class FacultyBean</td>
<td>575</td>
</tr>
<tr>
<td>8.1.6</td>
<td>Using Java Beans Technology for Java Web Applications</td>
<td>578</td>
</tr>
<tr>
<td>8.1.6.1</td>
<td>Modify the Help Class FacultyBean to Make it a Java Bean Class</td>
<td>580</td>
</tr>
<tr>
<td>8.1.6.2</td>
<td>Build a New Starting Web Page FacultyBeanPage</td>
<td>583</td>
</tr>
<tr>
<td>8.1.7</td>
<td>Using JavaServer Faces Technology for Java Web Applications</td>
<td>585</td>
</tr>
<tr>
<td>8.1.7.1</td>
<td>The Application Configuration Resource File</td>
<td>586</td>
</tr>
<tr>
<td></td>
<td>faces-config.xml</td>
<td></td>
</tr>
<tr>
<td>8.1.7.2</td>
<td>Sample JavaServer Face Page Files</td>
<td>587</td>
</tr>
<tr>
<td>8.1.7.3</td>
<td>The Java Bean Class File</td>
<td>590</td>
</tr>
<tr>
<td>8.1.7.4</td>
<td>The Web Deployment Descriptor File web.xml</td>
<td>591</td>
</tr>
<tr>
<td>8.1.7.5</td>
<td>A Complete Running Procedure of JSF Web Applications</td>
<td>591</td>
</tr>
<tr>
<td>8.2</td>
<td>Java EE Web Application Model</td>
<td>597</td>
</tr>
<tr>
<td>8.2.1</td>
<td>Java EE Web Applications with and without EJB</td>
<td>598</td>
</tr>
<tr>
<td>8.3</td>
<td>The Architecture and Components of Java Web Applications</td>
<td>599</td>
</tr>
<tr>
<td>8.3.1</td>
<td>Java EE Containers</td>
<td>600</td>
</tr>
<tr>
<td>8.3.2</td>
<td>Java EE 6 APIs</td>
<td>601</td>
</tr>
<tr>
<td>8.3.2.1</td>
<td>EJBs API Technology</td>
<td>602</td>
</tr>
<tr>
<td>8.3.2.2</td>
<td>Java Servlet API Technology</td>
<td>602</td>
</tr>
<tr>
<td>8.3.2.3</td>
<td>JSP API Technology</td>
<td>603</td>
</tr>
<tr>
<td>8.3.2.4</td>
<td>JavaServer Faces API Technology</td>
<td>604</td>
</tr>
<tr>
<td>8.3.2.5</td>
<td>Java Persistence API</td>
<td>606</td>
</tr>
<tr>
<td>8.3.2.6</td>
<td>Java Transaction API</td>
<td>606</td>
</tr>
<tr>
<td>8.3.2.7</td>
<td>Java Message Service API</td>
<td>607</td>
</tr>
<tr>
<td>8.3.3</td>
<td>Java Web Application Life Cycle</td>
<td>607</td>
</tr>
<tr>
<td>8.3.4</td>
<td>Java Web Modules</td>
<td>607</td>
</tr>
<tr>
<td>8.3.5</td>
<td>Java Web Frameworks</td>
<td>609</td>
</tr>
<tr>
<td>8.4</td>
<td>Getting Started with Java Web Applications Using NetBeans IDE</td>
<td>611</td>
</tr>
<tr>
<td>8.4.1</td>
<td>Create a Java Web Project</td>
<td>611</td>
</tr>
<tr>
<td>8.4.2</td>
<td>Create the Entity Classes from the Database</td>
<td>613</td>
</tr>
</tbody>
</table>
8.4.3 Create Five Web Pages Using Microsoft Office Publisher 2007  614
  8.4.3.1 Create the LogIn Page  615
  8.4.3.2 Create the Selection Page  617
  8.4.3.3 Create the Faculty Page  619
  8.4.3.4 Create the Course Page  622
  8.4.3.5 Create the Student Page  625

8.5 Build Java Web Project to Access SQL Server Database  625
  8.5.1 Access and Query the LogIn Table Using JSP and Help Class Files  626
    8.5.1.1 Modify the LogIn.jsp Page and Create LogInQuery.jsp File  627
    8.5.1.2 Create the Java Help Class File LogInQuery.java  629
    8.5.1.3 Create a Dialog Box as the Message Box  630
    8.5.1.4 Develop the Codes for the Help Class File  632
    8.5.1.5 Add the JDBC Driver for the SQL Server Database into the Project  635
  8.5.2 Build the Selection Page  637
  8.5.3 Query the Faculty Table Using JSP and JSP Implicit Session Object  640
    8.5.3.1 Modify the Faculty.jsp Page  641
    8.5.3.2 Create the FacultyProcess.jsp Page  642
    8.5.3.3 Create the Help Class File FacultyQuery.java  645
  8.5.4 Insert New Records to the Faculty Table Using JSP and Java Beans  650
    8.5.4.1 Modify the Java Help Class FacultyQuery to Make it Java Bean Class  651
    8.5.4.2 Modify the FacultyProcess.jsp Page to Handle Faculty Data Collection and Insertion  652
  8.5.5 Update and Delete Data from the Faculty Table Using JSP and Java Beans Techniques  656
    8.5.5.1 Create a New Java Session Bean Class  656
    8.5.5.2 Modify the FacultyProcess Page to Handle Faculty Data Updating  659
    8.5.5.3 Add a Method to the Session Bean to Perform Faculty Data Deleting  663
    8.5.5.4 Modify the FacultyProcess Page to Handle Faculty Data Deleting  664
  8.5.6 Query Data from the Course Table Using JavaServer Faces and Java Beans  668
    8.5.6.1 Modify the Course Page to Make it JavaServer Face Page  668
    8.5.6.2 Build the JavaServer Face Managed Bean CourseBean  672
    8.5.6.3 Build the Session Bean for Entity Classes CourseFacade  676
    8.5.6.4 Set Up Calling Relationship between the JSF Bean and the Session Bean  680
    8.5.6.5 Build and Run the Project to Test the Course Information Query Functions  681
  8.5.7 Update Records from the Course Table Using JavaServer Faces and Java Beans  682
    8.5.7.1 Create Codes for the Update() Method in the JSF Managed Bean  683
## Contents

8.5.7.2 Create Codes for the UpdateCourse() Method in the Session Bean 684
8.5.8 Delete Records from the Course Table Using JavaServer Faces and Java Beans 687
  8.5.8.1 Build Codes for the Delete() Method in the JSF Managed Bean 687
  8.5.8.2 Build Codes for the DeleteCourse() Method in the Session Bean 688

8.6 Build Java Web Project to Access and Manipulate Oracle Database 690
  8.6.1 Create a Java Web Application Project 691
  8.6.2 Modify the Hibernate Configuration File 692
  8.6.3 Create Hibernate Utility Files and Mapping Files 694
    8.6.3.1 Create the HibernateUtil.java Helper File 694
    8.6.3.2 Generate Hibernate Mapping Files and Java Classes 695
  8.6.4 Query the LogIn Table Using JSF Pages and Java Beans 697
    8.6.4.1 Modify the LogIn.jsp Page to Make it JSF Page 698
    8.6.4.2 Create and Build the Java Managed Bean LogInBean Class 700
  8.6.5 Build the SelectionPage and the SelectionBean Class 703
  8.6.6 Build the ErrorPage to Display any Error Information 707
  8.6.7 Set Up the Navigation Rules for Existing Web Pages 708
  8.6.8 Query the Faculty Table Using JavaServer Faces and Java Beans 711
    8.6.8.1 Modify the Faculty.jsp to Make it Our JSF Page FacultyPage.jsp 711
    8.6.8.2 Build the Java Session Bean FacultySessionBean to Handle Data Actions 715
    8.6.8.3 Build the Java Managed Bean FacultyMBean to Manage Data Actions 719
    8.6.8.4 Run the Project to Test the Faculty Information Query 723
    8.6.8.5 Modify the faces-config.xml File to Run Project in a Web Pages Sequence 724
    8.6.8.6 Add Codes to the Project to Display a Selected Faculty Image 725
    8.6.8.7 Run the Entire Project to Test the Faculty Information Query 727
  8.6.9 Insert New Records to the Faculty Table Using JavaServer Faces and Java Beans 728
    8.6.9.1 Add the Codes to the Java Managed Bean to Manage Data Insertions 728
    8.6.9.2 Build the InsertFaculty() Method for the Session Bean to Perform Data Insertions 730
    8.6.9.3 Run the Project to Test the New Faculty Record Insertion 730
  8.6.10 Update and Delete Records from the Faculty Table Using JSF Page and Java Bean 732
    8.6.10.1 Add the Codes to the Java Managed Bean to Manage Data Updating 732
    8.6.10.2 Build the UpdateFaculty() Method in the Session Bean to Perform Data Updating 734
    8.6.10.3 Run the Project to Test the Faculty Record Updating Action 735
    8.6.10.4 Add the Codes to the Java Managed Bean to Manage Data Deleting 737
xx Contents

9.5 Build Java Web Service Projects to Access SQL Server Database 787
9.5.1 Create a New Java Web Application Project WebServiceSQLApp 787
9.5.2 Create a New Java SOAP-Based Web Service Project WebServiceSQL 788
9.5.3 Add Desired Operations to the Web Service 789
9.5.4 Add New Operations to Our Web Services to Perform Data Query 790
9.5.5 Build the User-Defined Method DBConnection() 792
9.5.6 Deploy the Web Service Project and Test the Data Query Function 793

9.6 Build a Windows-Based Web Client Project to Consume the Web Service 795
9.6.1 Copy the FacultyFrame and MsgDislog Components as GUls 795
9.6.2 Create a Web Service Reference for Our Windows-Based Client Project 797
9.6.3 Develop the Codes to Call Our Web Service Project 799
9.6.4 Build and Run Our Client Project to Query Faculty Data via Web Service 801

9.7 Build a Web-Based Client Project to Consume the Web Service 801
9.7.1 Create a Web-Based Client Project WebClientSQL 802
9.7.2 Create a Java Managed Bean FacultyMBean and Add the JDialog Class MsgDialog 803
9.7.3 Create a Web Service Reference for Our Web-Based Client Project 804
9.7.4 Build the Codes to Call the Web Service to Perform Data Query 805
9.7.5 Build and Run Our Client Project to Query Faculty Data via Web Service 808

9.8 Build Java Web Service to Insert Data into the SQL Server Database 808
9.8.1 Add a New Operation InsertFaculty() into Our Web Service Project 809
9.8.2 Deploy the Web Service Project 811

9.9 Build a Windows-Based Web Client Project to Consume the Web Service 811
9.9.1 Refresh the Web Service Reference for Our Windows-Based Client Project 812
9.9.2 Develop the Codes to Call Our Web Service Project 812
9.9.3 Build and Run Our Client Project to Insert Faculty Data via Web Service 814

9.10 Build a Web-Based Client Project to Consume the Web Service 815
9.10.1 Refresh the Web Service Reference for Our Web-Based Client Project 816
9.10.2 Develop the Codes to Call Our Web Service Project 816
9.10.3 Build and Run Our Client Project to Insert Faculty Data via Web Service 818

9.11 Build Java Web Service to Update and Delete Data from the SQL Server Database 819
9.11.1 Add a New Operation UpdateFaculty() to Perform the Faculty Data Updating 820
9.11.2 Add a New Operation DeleteFaculty() to Perform the Faculty Data Deleting 822
9.11.3 Deploy and Test the Web Service Project 824

9.12 Build a Windows-Based Web Client Project to Consume the Web Service 827
9.12.1 Refresh the Web Service Reference for Our Windows-Based Client Project 827
9.12.2 Develop the Codes to Call Our Web Service Project 827
  9.12.2.1 Build the Codes to Call the UpdateFaculty() Operation 827
  9.12.2.2 Build the Codes to Call the DeleteFaculty() Operation 830
9.12.3 Build and Run Our Client Project to Update and Delete Faculty Record via Web Service 831

9.13 Build a Web-Based Client Project to Consume the Web Service 834
  9.13.1 Refresh the Web Service Reference for Our Web-Based Client Project 834
  9.13.2 Develop the Codes to Call Our Web Service Operation UpdateFaculty() 835
  9.13.3 Develop the Codes to Call Our Web Service Operation DeleteFaculty() 837
  9.13.4 Build and Run Our Client Project to Update and Delete Faculty Record via Web Service 838

9.14 Build Java Web Service Projects to Access Oracle Databases 840
  9.14.1 Create a New Java Web Application Project WebServiceOracleApp 841
  9.14.2 Create a New Java SOAP-Based Web Service Project WebServiceOracle 842
  9.14.3 Add a JDialog Class into the Web Services Project 843
  9.14.4 Add Java Persistence API and Entity Classes from Database 843
  9.14.5 Add Java Session Beans for Entity Classes 845
  9.14.6 The Organization of Web Service Operations and Session Bean Methods 848
  9.14.7 Add the Session Bean Classes CourseFacade into Our Web Service 849
  9.14.8 Create and Build the Session Bean Methods and Web Service Operations 849
    9.14.8.1 Create and Build Session Bean Method getCourseID() 850
    9.14.8.2 Create and Build Web Service Operation QueryCourseID() 852
    9.14.8.3 Build and Run the Web Service to Test the course_id Query Function 854
    9.14.8.4 Create and Build Session Bean Method getCourse() 855
    9.14.8.5 Create and Build Web Service Operation QueryCourse() 857
    9.14.8.6 Build and Run the Web Service to Test the Course Query Function 858
    9.14.8.7 Create and Build Session Bean Method newCourse() 860
    9.14.8.8 Create and Build Web Service Operation InsertCourse() 862
    9.14.8.9 Build and Deploy the Web Service Project 863
    9.14.8.10 Create and Build Session Bean Method setCourse() 864
    9.14.8.11 Create and Build Web Service Operation UpdateCourse() 867
    9.14.8.12 Build and Deploy the Web Service Project 868
    9.14.8.13 Create and Build Session Bean Method removeCourse() 868
    9.14.8.14 Create and Build Web Service Operation DeleteCourse() 870
    9.14.8.15 Build and Test the Web Service Project 871

9.15 Build a Windows-Based Web Client Project to Consume the Web Service 873
  9.15.1 Create a New Windows-Based Web Client Project WinClientOracle 873
  9.15.2 Copy the CourseFrame and MsgDislog Components as GUIs 874
  9.15.3 Create a Web Service Reference for Our Windows-Based Client Project 875
Databases have become an integral part of our modern day life. We are an information-driven society. Database technology has a direct impact on our daily lives. Decisions are routinely made by organizations based on the information collected and stored in databases. A record company may decide to market certain albums in selected regions based on the music preference of teenagers. Grocery stores display more popular items at the eye level and reorders are based on the inventories taken at regular intervals. Other examples include patients’ records in hospitals, customers’ account information in banks, book orders by the libraries, club memberships, auto part orders, winter cloth stock by department stores, and many others.

In addition to database management systems, in order to effectively apply and implement databases in real industrial or commercial systems, a good graphic user interface (GUI) is needed to enable users to access and manipulate their records or data in databases. NetBeans IDE is an ideal candidate to be selected to provide this GUI functionality. Unlike other programming languages, Java is a kind of language that has advantages, such as easy to earn and easy to be understood, with little learning curves. Beginning from Java 1.0, Sun has integrated a few programming languages, such as C++, JavaFX, and PHP, with some frameworks into dynamic models that make Internet and Web programming easy and simple, and any language integrated in this model can be used to develop professional and efficient Web applications that can be used to communicate with others via Internet.

This book is mainly designed for college students and software programmers who want to develop practical and commercial database programming with Java and relational databases, such as Microsoft Access, SQL Server 2008, and Oracle Database 10g XE. The book provides a detailed description about the practical considerations and applications in database programming with Java and authentic examples and detailed explanations. More important, a new writing style is developed and implemented in this book, combined with real examples, to provide readers with a clear picture as how to handle the database programming issues in NetBeans IDE environment.

The outstanding features of this book include but are not limited to the following.

1. A novel writing style is adopted to attract students or beginning programmers who are interested in learning and developing practical database programs, with the hope of avoiding the headaches caused by huge blocks of codes found in traditional database programming books.

2. A real completed sample database, CSE_DEPT, with three versions (Microsoft Access 2007, SQL Server 2008, and Oracle Database 10g XE Release 2), is provided and used for the entire book. A step-by-step, detailed description about how to design and build a practical relational database is provided.
Preface

3. Both fundamental and advanced database programming techniques are covered, for the convenience of both beginning students and experienced programmers.

4. Updated Java database programming techniques, such as Java Persistence API, Java Enterprise Edition 6, JavaServer Pages, JavaServer Faces, and Enterprise Java Beans, are discussed and analyzed with real projects to enable readers to have a clear picture and easy-to-learn path for Java database applications.

5. More than 30 real sample database programming projects are covered, with detailed illustrations and explanations to help students to understand key techniques and programming technologies.

6. Three types of popular databases are covered and discussed in detail with practical sample examples: Microsoft Access, SQL Server 2008, and Oracle Database 10g Express Edition (XE).

7. The various actual JDBC APIs and JDBC drivers are discussed and presented with real example coding explanations. The working structure and principle of using a JDBC driver to establish a valid database connection, build an SQL statement, and process the query results are introduced in detail with example codes. JDBC RowSet, a useful tool, is also discussed and analyzed with some example codes.

8. Problems and selected solutions are provided for each chapter to strengthen and improve understanding of the topics.

9. Power Point teaching slides are also provided to help instructors.

I sincerely hope that this book will be useful to all who adopt it, as a textbook for college students, as well as a reference book for programmers, software engineers, and academic researchers. I would be more than happy to know that you have been able to develop and build professional and practical database applications with the help of this book.

Ying Bai
Acknowledgments

First, I thank my wife, Yan Wang, in particular. I could not have finished this book without her sincere encouragement and support.

I also thank Satish Bhalla, who made important contributions to Chapter 2. Dr. Bhalla is a specialist in database programming and management, in particular, in SQL Server, Oracle, and DB2. Dr. Bhalla spent much time preparing materials for the first part of Chapter 2, and this is gratefully acknowledged.

Many thanks also go to Mary Mann at Wiley, who helped to make this book available to the public, and for her deep perspective and hard work. The same thanks are extended to the editorial team, without whose contributions the book would not have been published.

Thanks should also be extended to the following book reviewers for their important feedback on the manuscript:

• Dr. Jifeng Xu, Research Scientist, Boeing Company
• Dr. Xiaohong Yuan, Associate Professor, Department of Computer Science, North Carolina A&T State University
• Dr. Daoxi Xiu, Application Analyst Programmer, North Carolina Administrative Office of the Courts
• Dr. Dali Wang, Assistant Professor, Department of Physics and Computer Science, Christopher Newport University

Finally, thanks should be given to all of the people who supported me in the completion of this book.
Chapter 1

Introduction

For years while teaching database programming, I found it difficult to find a good textbook for this topic, so I had to combine a few different books together in order to teach the course. Most of those books are designed for programmers or software engineers, which cover a lot of programming strategies and huge blocks of coding, a headache to college students or beginning programmers who are new to programming. I dreamed that one day I would find a good textbook that is suitable for college students or beginning programmers, and that would help them to learn and master database programming easily and conveniently. Finally, I decided to realize this dream myself.

Another reason to write this book was the job market. Most companies in the United States, such as manufacturers, retailers, banks, and hospitals, use database applications extensively. The majority need professionals to develop and build database-related applications, but not necessarily database management and design. To enable our students to be good candidates for those jobs, a book such as this one is needed.

Unlike most database programming books on the current market, which discuss and present database programming techniques with huge blocks of programming codes from the first page to the last page, this book uses a new writing style to show readers, especially the college students, how to develop professional and practical database programs with Java, by using Java Persistence API (JAPI), Java Enterprise Edition (J2EE), Enterprise Java Beans (EJB), and plug-in tools related to NetBeans IDE, and to apply codes that are autogenerated by using those tools. Thus, the huge blocks of programming codes can be removed, and, instead, a simple and easy way to create database programs using plug-in tools can be developed to attract students’ interests, and furthermore to enable students to build professional and practical database programming in more efficient and interesting ways.

To meet the needs of some experienced or advanced students or software engineers, the book contains two programming methods: the interesting fundamental database programming method (JAPI and plug-in tools method) and the advanced database programming method (runtime object method). In the second method, all database-related objects are created and applied during or when your project is running by utilizing quite a few blocks of codes.
WHAT THIS BOOK COVERS

The contents of each chapter can be summarized as follows. Chapter 1 provides an introduction to the book. Chapter 2 provides a detailed discussion and analysis of the structure and components about relational databases. Some key technologies in developing and designing databases are also given and discussed. The procedure and components used to develop a practical relational database with three database versions, such as Microsoft Access, SQL Server 2008, and Oracle Database 10g XE, are analyzed in detail with some real data tables in our sample database CSE_DEPT.

Chapter 3 provides an introduction to JDBC APIs and JDBC drivers. A detailed introduction to components and architecture of JDBC is given with step-by-step illustrations. Four popular types of JDBC drivers are discussed and analyzed with their advantages and disadvantages in actual database applications. The working structure and operational principle of using JDBC drivers to establish a valid database connection, build a SQL statement, and process the query results are discussed and presented in detail. JDBC RowSet, a useful tool, is also discussed and analyzed with some example codes.

Chapter 4 provides a detailed discussion and analysis of JDBC design and actual application considerations. The fundamentals of using JDBC to access and manipulate data against databases are discussed and introduced with example codes. Different JDBC interfaces, including the ResultSet, ResultSetMetaData, DatabaseMetaData, and ParameterMetaData, are introduced and discussed with example codes.

Chapter 5 provides a detailed description of the NetBeans IDE, including the components and architecture. This topic is necessary for college students who have no knowledge of NetBeans IDE. Starting with an introduction to installing NetBeans IDE, this chapter goes through each aspect of NetBeans IDE, including the NetBeans Platform, NetBeans Open Source, and all plug-in tools. Different projects built with NetBeans IDE are discussed and presented in detail with 14 example projects.

Starting with Chapter 6, the real database programming techniques with Java, query data from database, are provided and discussed. Two parts are covered in this chapter: Part I contains detailed descriptions of how to develop professional data-driven applications with the help of the JAPI and plug-in tools with some real projects, and this part contains a lot of hiding codes that are created by NetBeans IDE automatically when using those tools and wizards. Therefore, the coding for this part is very simple and easy. Part II covers an advanced technique, the runtime object method, in developing and building professional data-driven applications. Detailed discussions and descriptions of how to build professional and practical database applications using this runtime method are provided combined with two real projects. In addition to basic query techniques, advanced query methods, such as PreparedStatement, CallableStatement, and stored procedure, are also discussed and implemented in this chapter with some real sample projects.

Chapter 7 provides detailed discussions and analyses of how to insert, update, and delete data from three popular databases: Microsoft Access, SQL Server 2008, and Oracle. This chapter is also divided into two parts: In Part I, JAPI and plug-in tools to perform data manipulations are discussed. Part II covers the technique to manipulate data in our sample database using the runtime object method. Four real projects illustrate how to perform the data manipulations against three different databases: Microsoft Access, SQL Server 2008, and Oracle Database 10g XE. Professional and practical data validation