Principles of Database Systems with Internet and Java Applications provides a concise and modern treatment of introductory database topics that is suitable for use in undergraduate database courses, applications-oriented courses on database interaction with the World Wide Web, and professional development. As part of this effort, it enlists Java and the Internet to add an applications perspective to the core DBMS theory.

This book covers the basic material related to information management, database systems, Java programming, and interaction with databases on the World Wide Web. It assumes that readers have a background in programming and helps them to improve their skills in the design and implementation of complex information systems.

Information management is the central theme of Principles of Database Systems. It motivates the development of data models and the representation of information in relational database systems. Readers learn how to define information content with Entity–Relationship models, as well as how to represent that content in relational database systems. Along the way, they become thoroughly familiar with SQL, the Structured Query Language, plus the advanced features of relational database systems. They learn exactly what is required to build high-quality, information-rich applications. In addition, they see how the Web and Java can work together to help them publish and collect information in the widest possible context.

Database and information systems material are covered extensively in this book. Topics include analyzing information requirements, developing conceptual data models, translating conceptual models into relational models, normalizing and improving relational schemas, writing queries in SQL, and developing database applications. Interesting examples are used to show students how to apply this material. Additional topics include object-oriented modeling and databases, database performance and optimization,
constraints and triggers, transactions, backup and recovery, file structures, indexing, and distributed object technology.

The thorough treatment of relational database systems includes the use of readily available database tools, such as Microsoft Access, SQL databases, and object-oriented design tools. Microsoft Access is used to illustrate the role of relational models in developing information applications. We demonstrate the use of SQL databases with ODBC interaction and with JDBC and Java. Instructors will be able to use the programming projects in Oracle or adapt them to other relational database systems as desired.

Java provides the basis for discussing how information is represented in an object-oriented language. The presentation of this programming language begins with a description of how it interacts with relational database systems. In addition, Java is employed in creating Web interactions, representing file structures and indexes, creating distributed applications, and designing object-oriented data models. A student who is familiar with an object-oriented language will be able to readily understand this material. In contrast, one who is not already a Java programmer will benefit by perusing the appendix on Java and may need to consult a language supplement.

The interaction between applications and databases is discussed and illustrated in the context of Web sites. The JDBC classes of Java provide a database- and platform-independent method of creating database applications. The book includes a thorough discussion of these classes, punctuated with abundant examples. After learning the fundamentals of HTML and CGI programming, students are asked to create their own Web sites using Java programs to service CGI requests and generate HTML responses. Additional topics include the use of Java servlets to replace CGI programs and the development of simple Java applets as database user interfaces.

The important principles of file structures are presented in a way that emphasizes software development. Students are shown an object-oriented style of representing information in files. The book includes class definitions for direct-access file I/O, buffer packing and unpacking, indexing, and B+ trees. The file structures section ends by discussing methods of implementing relational database systems.

## Database Applications and Examples

Among the unique features of this book are its emphasis on software development, its use of the Java programming language, and its presentation of the World Wide Web as a tool for database interaction. Java is used to first introduce object-oriented methods and then apply those methods to produce database applications. Its support for database interaction is the key to success in providing access to information on the Internet. Detailed discussions of the interaction between Java and the Web make it clear just what is required to create useful applications. As a consequence, this book will prove particularly appealing to instructors who want to make their database courses relevant to Internet activities.
Complex information applications are introduced early in the book and used throughout to illustrate the concepts and the details. BigHit VideoCompany, a (fictional) chain of video rental and sales stores, provides our major example. The information and application needs of this company are sufficient to construct applications that range from simple to complex. Each new topic is fully illustrated by showing how it can be used to satisfy the needs of specific information applications.

Supplementary materials that are available online include the database, SQL, and programming examples that are included in the book. In addition, full implementations of the Microsoft Access database examples are available to the instructors. These databases are carefully constructed to allow students to extend and enhance them for course projects. The SQL examples are suitable for use in Access or any other standard SQL database system. We have taken care to ensure that the SQL code used in both this book and in the supplements comes from a subset of standard SQL that is universally applicable.

A Book for Teaching Database Systems

The upper-division course in database systems for computer science is one of the most important courses in the curriculum, because it combines the needs of users, developers, and maintainers of software systems. Instructors must present a blend of topics that allows students to understand what is required to satisfy these diverse needs. Although students taking this course may already have a significant amount of expertise in computer programming, they still need to learn how to combine analysis, design, and programming skills to produce useful and effective software.

A database course for computer science majors must include the basics of information systems design and implementation, relational database systems, and SQL. It must improve the general skills and maturity of the students as they seek to become software professionals. Finally, it must motivate students to work hard—a goal easily accomplished by a particularly interesting course.

The premise of this book is that combining databases, Java, and the World Wide Web will produce a course that is extremely effective in meeting these challenges. It will be particularly appealing to students because of its presentation of up-to-date and interesting material.

The most appealing and exciting part of a database course may be the development of database applications that support Web sites. In *Principles of Database Systems*, the object-oriented programming theme is extended to show how application programs can interact with databases using SQL. The book covers the basics of HTML, HTML forms, and CGI programming. It even includes the use of Java servlets as an alternative to CGI.

Students do not need extensive experience with Java before using this book. Here, the coverage of Java starts from very basic classes and progressively includes more complex structures as required. Students who know any object-oriented language will understand the programming examples and exercises.
Possible Course Outlines

At least three approaches to teaching database systems can benefit from this book: a traditional database systems course, an object-oriented database applications course, and an applications-oriented database course. All courses will cover the introduction and the chapters on data modeling, Entity–Relationship modeling and its extensions, relational schemas, manipulation of relational information, and SQL. My experience is that these common topics take approximately one-half semester. This material is covered in:

- Chapter 1: Information Management and Database Systems;
- Part 1: Chapters 2–5, Information Models and Relational Databases; and
- Part 2: Chapters 6–7, Manipulating Relational Data.

In a traditional database systems course, the greatest emphasis in the remainder of the course is placed on ways that relational database systems contribute to information systems. The major emphasis focuses on how relational database systems are built on a foundation of file structures and indexing, how queries are executed and optimized, and how concurrency, transactions, security, and backup and recovery are supported. This material is covered in the following chapters:

- Part 4: Chapters 11–12, Physical Characteristics of Databases, and

In contrast, an object-oriented database applications course will place the greatest emphasis in the remainder of the course on the development of information-rich applications using Java as the primary programming language. The focus in this case is on the interactions between applications and relational database systems, database applications’ support for Web sites, distributed information systems, and object-oriented and object-relational database systems. This material is covered in the following chapters:

- Part 3: Chapters 8–10, Database Applications and the World Wide Web; and
- Part 6: Chapters 16–17, Object-Oriented and Distributed Information Systems.

I prefer to teach an applications-oriented database course that combines these approaches. In the first half of the course, we cover the standard material and the students complete a sophisticated application and user interface using Microsoft Access. We begin the second half of the course by considering relational database applications in Java (Chapter 8) and database–Web interaction (Chapter 9). At that point, the students begin working on creating a Web site using Java and an SQL database. The course continues with a brief treatment of physical databases (Part 4) and query processing (Chapter 13) and finishes with a detailed treatment of transactions, security, and backup and recovery (Chapters 14 and 15).
Database Applications Programming in Java

The applications developed in this book rely heavily on the Java programming language. For readers who are unfamiliar with Java, introduction to Java is included as an appendix. I have been teaching Java in undergraduate database classes at Florida State University for several years. Most of the students are competent C++ programmers, but have had no previous experience with Java. I always assure them that the Java programming required to create database applications and Web sites is very straightforward. They are often worried, but always successful in the end. Indeed, they are pleased to find that our programming projects use almost none of the features of Java that people find most difficult. In particular, no graphical user interface programming is required!

Many people think of Java as a language for creating lively Web pages. It’s true that it is a reasonable tool for creating user interfaces that run in Web browsers. But Java is also the best available programming tool for developing the kinds of database applications found in this book.

The true strength of Java derives from its ability to work behind the Web server and between applications. It’s ability to interact with any database server on any platform is crucial to the quality development of dynamic Web sites, and its distributed object capabilities are truly amazing.

The code examples in the book provide a roadmap for developing Java applications and for becoming a skilled Java database programmer. For most of the projects, students can adapt one or more included Java classes to create a solution. A few changes here and there, and a completely different application emerges. My students find that they can readily understand the Java code because they already know C++. As they complete one assignment after another, they begin to understand the capabilities of the language and become increasingly more competent. After graduating, they often find that Java and database programming are their most marketable skills.

Understanding the code examples of this book requires careful reading and access to the complete implementations that are found on the book’s Web site. The code samples in the book are often fragmentary and have limited explanatory comments. In contrast, the code found on the Web site is complete and includes more extensive documentation.

Supplementary Materials

Instructors and students have a variety of supplementary materials available online through the Addison-Wesley Web sites. You can find links to this material at the book’s official Web site (http://www.awl.com/cseng/authors/riccardi/databases). Although much of the material is freely available, the instructor’s manual, sample exams, answers to exercises, and some extensive applications are available only to instructors through contact with their Addison-Wesley local sales representatives.
All of the database schemas, SQL statements, and applications programs that appear in the book are available online. This material includes Microsoft Access databases, the SQL code required to create and populate tables, the complete Java code required to produce applications and Web sites, and the code required to integrate applications with the Web for both Unix and Windows systems.

Java packages that support the interaction between Web browsers, Web servers, and database applications are provided at the Web site in two versions. One version supports Microsoft Windows platforms using Microsoft Access, ODBC, and the personal Web server that is freely distributed by Microsoft. With this version, the Web site examples are fully implemented in a way that makes it easy for students and instructors to install them on any Windows platform. The other version of Web support uses a Unix platform with the Apache Web server and an SQL database with JDBC support. All examples have been tested using the freely available MySql database system and the Oracle8 database system. Java database application programs can be run on either Unix or Windows systems.

An online instructor’s manual contains course lecture notes, answers to exercises, and sample tests. This material is available exclusively to instructors who are teaching with this text. One way that the book will be kept up-to-date is by the occasional addition of new exercises and extensive projects to the Web site. It is available through your Addison-Wesley sales representative or by sending an e-mail message to aw.cse@awl.com.

Please feel free to contact me by e-mail (Greg.Riccardi@acm.org) to report or ask about errors and to get the latest supplements.

Acknowledgements

I would like to thank the editorial and production teams at the Computer Science Group of Addison-Wesley for their support and encouragement. Acquisitions editor Maite Suarez-Rivas insisted that it was possible for me to write an excellent book, and that she could get me to do it. Katherine Harutunian, the project editor, has now worked with me on two books, and I could not ask for a more helpful and pleasant editor. The production process greatly benefited from the work of Pat Mahtani and Diane Freed.

This book was extensively and thoroughly reviewed by Professors Munindar P. Singh of North Carolina State University, Henry A. Etlinger of Rochester Institute of Technology, Salih Yurttas of Texas A&M University, Arijit Sengupta of Georgia State University, and Le Gruenwald of the University of Oklahoma. Their suggestions were invaluable. Professors Suzanne Dietrich of Arizona State University, Bill Grosky of Wayne State University, and Junping Sun of Nova Southeastern University also contributed with their reviews of selected chapters.

My experiences teaching database systems and working with both students and faculty at Florida State University have been most helpful to me. I am particularly thankful to Lawrence Dennis, Bryon Ehlmann, Charles W. Ford, Jr., Lois Wright Hawkes,
Samuel J. Eaves II, Dmitriy Blaginin, Troy Cochran, and Shanmaguraja Ramaswamy. Students in COP 4710 Databases, at Florida State University consented to use preliminary versions of this book as their textbook. My thanks go out to all of those students, and especially to Ahmed Moussa, Gabrielle Reed, Tilak Madan, Andres Naranjo A., Huey Ling Toh, Ronie Robinson, Owen Chang, William Maher, Ron Steedman, and Solomon Williams.

My greatest debts are owed to my wife Ann and daughters Mary, Christina, and Elizabeth. They have put up with my many absences, both physical and mental, as I struggled with this project. I owe all of my successes to their support and encouragement.
Greg Ricardi is a professor in the Department of Computer Science at Florida State University, a faculty associate of the Florida State University School of Computational Sciences and Information Technology, and a member of the Hall B Collaboration at the Thomas Jefferson National Accelerator Facility in Newport News, Virginia. Professor Ricardi’s research interests include scientific databases, scientific computation, and distributed object computing. He received the President’s Award for Excellence in Undergraduate Teaching in 1997 from Florida State University.