<table>
<thead>
<tr>
<th>Mon</th>
<th>Tue</th>
<th>Wed</th>
<th>Thu</th>
<th>Fri</th>
</tr>
</thead>
<tbody>
<tr>
<td>14</td>
<td></td>
<td>16</td>
<td>17</td>
<td>18</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>AH-3; Project Deliverables</strong></td>
<td></td>
<td><strong>Chap. 4 DB Design using Normalization</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Chap. 4 DB Design using Normalization</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td></td>
<td>23</td>
<td>24</td>
<td>25</td>
</tr>
<tr>
<td></td>
<td><strong>AH-5a; Project Deliverable</strong></td>
<td></td>
<td><strong>Chap. 5 Data Modeling with ER Model</strong></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Chap. 5 Data Modeling with ER Model</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
I. GETTING STARTED

II. DATABASE DESIGN & IMPLEMENTATION

III. SELECTED TOPICS

Introduction (C-1)
Introduction to Structured Query Language (C-2)
Rel. Model and Normalization (C-3)
Managing Multiuser Databases (C-9)
XML (C-13)

Database Design and Normalization

Data Modeling with the ER Model (C-5)
Transforming Data Models into Database Designs (C-6)

SQL for Database Construction (C-7)
MODULE II: DATABASE DESIGN & IMPLEMENTATION

Part 4: Database Design and Normalization (Text: Chapter 4)

NOTES FOR CLASS SESSION

STARTING POINTS

Questions to begin our deliberation

1. How do you apply the concepts the relational model and normalization concepts to the design of databases created from existing data?

Student Learning Objectives (Desired Outcomes)

At the end of in-class and outside-class work on this topic, you should be able to:

1. Design updateable databases to store data received from another source. [Synthesis]
2. Use SQL to access table structure. [Application]
3. Discuss the advantages and disadvantages of normalization. [Comprehension]
4. Describe denormalization. [Comprehension]
5. Design read-only databases to store data from updateable databases. [Synthesis]
6. Recognize and correct common design problems
[Comprehension/Analysis]
   a. The multi-value, multi-column problem.
   b. The inconsistent values problem.
   c. The missing values problem.
   d. The general-purpose remarks column problem

QUESTIONS TO EXPLORE READINGS

Q 1. What makes the design of an updateable database different from design of a read-only database?

Q 2. What is denormalization?

Q 3. Why should you fully normalize the tables while designing a database?

In-Class Exercise Set 4

Chapter: Database Design using Normalization

Points to consider (Please take time review all the points before starting the exercise):

1. If updateable database, we normally want tables in BCNF
2. If read-only database, we may not use BCNF tables
3. Advantages of Normalization:
   a. Eliminate modification Anomalies
   b. Reduce duplicated data
   c. Eliminate data integrity problems
d. Save file space

4. Disadvantages of Normalization:
   a. More complicated SQL required for multi-table sub-queries and joins
   b. Extra work for DBMS means slower applications

5. **Read-only databases** are non-operational databases using data extracted from operational databases. They are used for querying, reporting and data mining applications

6. NULL values for a column is ambiguous:
   a. May indicate that a value is inappropriate
   b. May indicate that a value is appropriate but unknown
   c. May indicate that a value is appropriate and known, but has never been entered

**Exercise Set:**

1. Georgia Pacific uses the following tables to identify its racers and teams. At some point they wish to make it as a read-only database. Please provide them the table structures for read-only databases.
   RACER_REGISTRATION (**RacerName, TeamName**, DateRacerRegistered, Paid)
   TEAM_REGISTRATION (**TeamName**, DateTeamRegistered)

2. Please provide the queries for moving the data from the normalized form to denormalized form.

3. Identify any multi-dependencies in the table structure provided in Question 1.

Take-Home Assignment Set 4

Chapter: Database Design using Normalization

This assignment is a continuation of the previous take-home assignment. Please read the solution for the previous assignment before starting this one. The following are four relations designed for little John’s bookstore.

**USER_TABLE**

(FirstName, LastName, Email, CreditCardNumber, PhoneNumber, ShippingAddress1, ShippingAddress2, ShippingCity, ShippingState, ShippingZip, Password)

<table>
<thead>
<tr>
<th>FirstName</th>
<th>LastName</th>
<th>Email</th>
<th>CreditCardNumber</th>
<th>PhoneNumber</th>
<th>ShippingAddress1</th>
<th>ShippingAddress2</th>
<th>ShippingCity</th>
<th>ShippingState</th>
<th>ShippingZip</th>
<th>Password</th>
</tr>
</thead>
<tbody>
<tr>
<td>Saravanaraj</td>
<td>Duraisamy</td>
<td>sduraisamy111222233334440</td>
<td>12334567890123456</td>
<td>12345678901234</td>
<td>12345678901234</td>
<td>12345678901234</td>
<td>12345678901234</td>
<td>12345678901234</td>
<td>12345678901234</td>
<td>12345678901234</td>
</tr>
<tr>
<td>Roger Federer</td>
<td>rfederer1</td>
<td>rfederer22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
<td>22223334445550</td>
</tr>
<tr>
<td>Rafael Nadal</td>
<td>rnal1</td>
<td>rnal1333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
<td>3333444455556666</td>
</tr>
<tr>
<td>John Travolta</td>
<td>jtravolta</td>
<td>jtravolta4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
<td>4444555566667777</td>
</tr>
</tbody>
</table>

**BOOK_TABLE**

(ISBN, Title, Price, PublicationYear, Description, AvailableCount, AuthorFirstName, AuthorLastName, Category, Publisher)

<table>
<thead>
<tr>
<th>ISBN</th>
<th>Title</th>
<th>Price</th>
<th>PublicationYear</th>
<th>Description</th>
<th>AvailableCount</th>
<th>AuthorFirstName</th>
<th>AuthorLastName</th>
<th>Category</th>
<th>Publisher</th>
</tr>
</thead>
<tbody>
<tr>
<td>0439136950</td>
<td>Harry Potter and the Prisoner of Azkaban</td>
<td>17.99</td>
<td>1999</td>
<td>Novel</td>
<td>25</td>
<td>Joanne</td>
<td>Rowling</td>
<td>Fantasy</td>
<td>Scholastic</td>
</tr>
</tbody>
</table>

**ORDER_TABLE**

(Email, OrderNumber, OrderDate, TotalCost, Status)

<table>
<thead>
<tr>
<th>OrderNumber</th>
<th>Email</th>
<th>OrderDate</th>
<th>TotalCost</th>
<th>Status</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>sduraisamy1@st...</td>
<td>2009-01-01</td>
<td>7.79</td>
<td>Shipped</td>
</tr>
<tr>
<td>2</td>
<td>rfederer1@stud...</td>
<td>2009-01-02</td>
<td>33.57</td>
<td>NotShipped</td>
</tr>
</tbody>
</table>
ORDER_ITEM_TABLE
(OrderNumber, ISBN, Quantity, TotalCost, OrderItemNumber)

<table>
<thead>
<tr>
<th>OrderNumber</th>
<th>ISBN</th>
<th>Quantity</th>
<th>TotalCost</th>
<th>OrderItemNumber</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0439785960</td>
<td>1</td>
<td>7.79</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0439785960</td>
<td>2</td>
<td>15.58</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>0439136350</td>
<td>1</td>
<td>17.99</td>
<td>5</td>
</tr>
</tbody>
</table>

1. Identify all the constraints between the tables:
2. Little John wants to make sure that there are no NULL values in the existing table structure. Please provide him the queries to display the number of rows with null values.

Study Guide Chapter 4 (SG-Ch4)

http://wps.prenhall.com/bp_kroenke_database_10/30/7927/2029554.cw/index.html

Self-Test 4 (ST4)

Available at the uLearn site for the course.