

DB Design and Modeling Course Content  
[www.crystal spiders .com](http://www.crystal spiders .com)

**Introduction to Database Concepts**

Database Design Phases

- Conceptual
- Logical
- Implementation
- Physical

Relational Data Structures

- Database and Schema
- Tables, Rows, and Columns
- The Information Principle
- Domains
- Metadata
- Keys
- Missing Values (NULLs)

Relationship between Entities

Data Access Language (SQL)

Understanding Dependencies

- Functional Dependency
- Determinant

Summary

**The Language of Data Modeling**

Introduction to Data Modeling

Entities

Attributes

- Primary Keys
- Alternate Keys
- Foreign Keys
- Domains
- Naming

Relationships

- Identifying Relationships
- Nonidentifying Relationships
- Role Names
- Relationship Cardinality
- Verb Phrases (Relationship Names)

Descriptive Information

Alternative Modeling Methodologies

- Information Engineering
- Chen ERD
- Visio
- Management Studio

Database Diagrams

Best Practices

Summary

**Conceptual Data Modeling**

Understanding the Requirements

Documenting the Process

Requirements Gathering

- Client Interviews
- Questions to Be Answered
- Existing Systems and Prototypes
- Other Types of Documentation

Identifying Objects and Processes

- Identifying Entities
- Relationships between Entities
- Identifying Attributes and Domains

Identifying Business Rules and Processes

- Identifying Business Rules
- Identifying Fundamental Processes

Finishing the Conceptual Model

- Identifying Obvious Additional Data Needs
- Review with the Client
- Repeat Until the Customer Agrees with Your Model

Best Practices

DB Design and Modeling Course Content  
[www.crystal spiders .com](http://www.crystal spiders .com)

---

### The Normalization Process

#### Why Normalize?

- Eliminating Duplicated Data
- Avoiding Unnecessary Coding
- Keeping Tables Thin
- Maximizing Clustered Indexes
- Lowering the Number of Indexes per Table

#### How Far to Normalize?

#### The Process of Normalization

#### Entity and Attribute Shape: First Normal Form

- All Attributes Must Be Atomic
- All Instances in an Entity Must Contain the Same Number of Values
- All Occurrences of an Entity Type in an Entity Must Be Different
- Programming Anomalies Avoided by First Normal Form
- Clues That an Existing Design Is Not in First Normal Form

#### Relationships between Attributes

- Second Normal Form
- Third Normal Form
- Boyce-Codd Normal Form

#### Multivalued Dependencies in Entities

- Fourth Normal Form
- Fifth Normal Form

#### Denormalization

#### Best Practices

### Implementing the Base Table Structures

#### Reviewing the Logical Design

#### Transforming the Design

- Choosing Names
- Dealing with Subtypes
- Determining Tree Implementation
- Choosing Key Implementation
- Determining Domain Implementation
- Setting up Schemas
- Reviewing the "Final" Implementation Model

#### Implementing the Design

- Creating the Basic Table Structures
- Adding Uniqueness Constraints
- Building Default Constraints
- Adding Relationships (Foreign Keys)
- Dealing with Collations and Sorting
- Computed Columns
- Implementing User-Defined Data types
- Documenting Your Database
- Working with Dependency Information

#### Best Practices

### Protecting the Integrity of Your Data

#### Automatic Data Protection

- Declarative Data Protection
- CHECK Constraints Based on Simple Expressions

**DB Design and Modeling Course Content**  
[www.crystal spiders .com](http://www.crystal spiders .com)

- 
- CHECK Constraints Based on Functions
  - Errors Caused by Constraints
  - DML Triggers
  - Handling Errors from Triggers and Constraints
- Manual Data Protection
- Stored Procedures
  - Client Code
- Best Practices
- Patterns and Query Techniques**
- Precalculated Values
- Sequence Tables
  - Calculations with Dates
- Binary Large Valued Objects (BLOB)
- Storing User-Specified Data
- Big Old List of Generic Columns
  - Entity-Attribute-Value (EAV)
  - Adding Columns to a Table
- Commonly Implemented Objects
- Anti-patterns
- One-Size-Fits-All Key Domain
  - Generic Key References
  - Overusing Unstructured Data
- Securing Access to the Data**
- Principals and Securables
- Database Security Overview
- Impersonation
  - Permissions
  - Controlling Access to Objects
  - Roles
  - Schemas
- Controlling Object Access via T-SQL
- Coded Objects
- Stored Procedures and Scalar Functions
  - Impersonation within Objects
  - Crossing Database Lines
  - Different Server (Distributed Queries)
- Views and Table-Valued Functions
- General Usage
  - Implementing Configurable Row-Level Security with Views
- Obfuscating Data
- Monitoring and Auditing
- Server and Database Audit
  - Watching Table History Using DML Triggers
  - DDL Triggers
  - Logging with Profiler
- Best Practices
- Table Structures and Indexing**
- Physical Database Structure
- Files and File groups
  - Extents and Pages
  - Data on Pages
  - Partitioning
- Indexes Overview
- Basic Index Structure
- Index Types
- Clustered Indexes
  - Nonclustered Indexes
  - Nonclustered Indexes on Clustered Tables
- Basics of Index Creation
- Basic Index Usage Patterns
- Using Clustered Indexes
  - Using Nonclustered Indexes

**DB Design and Modeling Course Content**  
[www.crystal spiders .com](http://www.crystal spiders .com)

- Using Unique Indexes
- Advanced Index Usage Scenarios
- Foreign Key Indexes
  - Indexed Views
- Best Practices

- Guidelines for choosing T-SQL
  - Guidelines for choosing .NET
  - CLR Object Types
- Best Practices

**Coding for Concurrency**

- What Is Concurrency?  
Query Optimization Basics  
OS and Hardware Issues  
Transactions
- Transaction Syntax
  - Compiled SQL Server Code
- SQL Server Concurrency Controls
- Locks
  - Isolation Levels
- Coding for Integrity and Concurrency
- Pessimistic Locking
  - Implementing a Single Threaded Code Block
  - Optimistic Locking
  - Row-Based Locking
  - Logical Unit of Work
- Best Practices

**Codd's 12 Rules for an RDBMS**

- The Information Rule  
Guaranteed Access Rule  
Systematic Treatment of NULL Values  
Dynamic Online Catalog Based on the Relational Model  
Comprehensive Data Sublanguage Rule  
View Updating Rule  
High-Level Insert, Update, and Delete  
Physical Data Independence  
Logical Data Independence  
Integrity Independence  
Distribution Independence  
Non-Subversion Rule

**Considering Data Access Strategies**

- Ad Hoc SQL
- Advantages
  - Pitfalls
- Stored Procedures
- Encapsulation
  - Dynamic Procedures
  - Security
  - Performance
  - Pitfalls
- T-SQL and the CLR