

Oracle Data Modeling and Relational Database Design

Duration: 4 Days

What you will learn

This Oracle Data Modeling and Relational Database Design training covers the Data Modeling and Database Development process and the models that are used at each phase of the lifecycle. Learn from expert Oracle University instructors through interactive instruction and hands-on exercises.

Learn To:

Identify the types of models.

Develop a process model (Data Flow Diagram).

Use advanced data modeling techniques.

Create the Physical Model, add several Physical Model objects, and generate the DDL.

Use several real life examples to document business requirements.

Work with design rules that can be applied to check and enforce the integrity and consistency of your designs.

Work in a collaborative environment using Subversion.

Benefits to You

By taking this course, you will develop an understanding of the data modeling and database development process, as well as the models used in each phase of the development lifecycle. You'll develop the skills to model and understand the database development lifecycle based on real life examples, while mapping the objects and engineer the logical model to a relational model.

Validate Data Models

You will also learn techniques to validate these data models. Once the Relational Design has been validated, you can create physical models to add all physical properties and finally generate a DDL to create the database objects for your database. You will also better understand how you can work as a team on developing a model using Subversion.

Related Training

Suggested Prerequisites

Basic understanding of relational database concepts

Course Objectives

Create an Entity Relationship Diagram by identifying entities

attributes

relationships and constraints from a set of requirements

Normalize the Entity Relationship Diagram to third Normal form

Enhance the Entity Relationship Diagram to utilize several data modeling techniques

Create a Data Flow Diagram by identifying processes

external agents

information stores and information flows that show how the information flows and how it is being transformed

Engineer the Entity Relationship Model into an initial relational database design

Optimize the Relational Database Design

Complete the Physical Model and generate the DDL

Use Oracle SQL Developer Data Modeler to document all the concepts learned throughout the course

Course Topics

Understanding What to Model

Why Model?

Why Model: A Practical Example

Database and Application Development Life Cycle

Process Modeling

Logical Data Modeling

Database Design

Database Generation

Data Type Model

Documenting the Business Background

Documenting the Business Direction

Components of a Business Direction Statement

- Business Objectives
- Assumptions
- Critical Success Factors
- Key Performance Indicators
- Problems
- Devising Business Direction Objectives and Actions

Building a Process Model (Data Flow Diagram)

- What Is a Process Model?
- Why Create a DFD?
- Components of a Data Flow Diagram
- Events
- Analyzing Event Responses

Using Oracle SQL Developer Data Modeler to Create Your Process Model (Data Flow Diagram)

- Downloading and Installing Oracle SQL Developer Data Modeler
- Oracle SQL Developer Data Modeler Main Window Components
- Building a Data Flow Diagram
- Editing the Diagram Layout
- Adding and Reusing Process Events
- Saving Your Model
- Opening a Saved Model

Validating Your Process Model (Data Flow Diagram)

- DFD Rules
- Design Rules in Oracle SQL Developer Data Modeler
- Types of Processes
- Process Decomposition
- Decomposition Guidelines

Identifying Entities and Attributes

- What Is a Logical Data Model?
- Why Create an ERD?
- Components of an Entity Relationship Diagram
- Attributes
- Attribute Characteristics

Identify Relationships

- Relationships
- Components of a Relationship
- Relationships: Additional Examples
- Relationship Types
- Using a Relationship Matrix
- Determining a Relationship's Existence
- Naming the Relationship
- Determining the Relationship's Cardinality

Assign Unique Identifiers

- Unique Identifiers
- Unique Identifier Examples
- Identifying Relationships
- Identifying Relationships with Multiple Entities

- Non-Identifying Relationships
- Primary and Secondary Unique Identifiers
- Searching for Unique Identifiers

Using Oracle SQL Developer Data Modeler to Create the Entity Relationship Diagram

- Building an Entity Relationship Diagram
- Specifying Logical Model General Option
- Modifying Model Properties
- Notation Types
- Editing a Diagram Layout
- What Is a Subview?
- Creating a Subview
- What Is a Display?

Validating your Entity Relationship Diagram

- ERD Checklist
- Attribute Rules
- Distinguishing Attributes and Entities
- Attribute Optionality
- Adding Additional Information to the ERD
- Creating Reports

Normalizing your Data Model

- What Is Normalization?
- First Normal Form (1NF)
- Second Normal Form (2NF)
- Third Normal Form (3NF)
- Normalization Example

Validating Relationships

- Resolving M:M Relationships
- Modeling Hierarchical Data
- Examining Recursive Relationships
- Resolving a M:M Recursive Relationships
- Modeling Exclusive Relationships
- Creating an Exclusive Relationship in Oracle SQL Developer Data Modeler
- Entity Type Hierarchies
- Modeling Subtypes in Oracle SQL Developer Data Modeler

Adding and Using Data Types

- Attribute Data Types
- Logical Type
- Types Administration
- Domain
- Adding a Check Constraint to a Domain
- Adding Ranges or Value Lists to a Domain
- Preferred Logical Types and Domains
- Creating Domains from Logical Types

Put It All Together

- Build an ERD from a Case Study

Map Your Entity Relationship Diagram to a Relational Database Design

Why Create a Relational Model?

Review: Database Design

Relational Database Overview

Terminology Mapping

Naming Conventions

Naming Restrictions with Oracle

Ensuring That Your Logical Data Model Is Complete

Mapping Simple Entities

Engineering Your Entity Relationship Diagram to a Relational Database Design in Oracle SQL Developer Data Modeler

Relational Model and Relational Model Diagram Preferences

Reviewing Table Properties

Previewing the DDL for a Table

Preferences: Classification Types

Assigning a Classification Type to One Table

Changing the Color for Classified Tables

Changing the Prefix for Classified Tables

Assigning Classification Types to Multiple Tables

Defining Your Physical Model

What Is a Physical Model?

Creating a Physical Model

RDBMS Administration

RDBMS Administration: Changing the Default RDBMS Sites

Creating Physical Model Objects

Adding a User

Adding Segment Templates (Storage)

Associating Physical Objects with Your Table

Generating Your Database

Database Generation

Generating DDL

DDL Preferences

DDL/Migration General Options

Design Rules

Working With Rule Sets

Working With Custom Rules

Working With Libraries

Altering an Existing Design

Approaches to Modeling

Using Import to Create a Model

Importing an Existing Database

Importing Domains

Creating a Logical Data Model from Your Relational Model

Reviewing and Making Changes to Your Logical Model

Checking the Design Rules

Forward Engineering to a New Relational Model

Working in a Collaborative Environment

The Benefits of Version Control

Working With Data Modeler and Subversion
Pending Changes
Basic Workflow: Using Subversion with a Design
Maintaining Versions