

# Oracle Data Modeling and Relational Database Design Ed 2.1

**Course Duration:32 Hours**

**Course code: Ed 2.1**

## 1. Course Overview

This course provides participants with the skills to design efficient, scalable, and well-structured relational databases using Oracle Data Modeler and industry-standard modeling techniques. Learners will explore conceptual, logical, and physical data modeling, normalization, ER modeling, relational design principles, and best practices for database implementation. Practical exercises emphasize creating and refining models that align with business requirements while ensuring performance, flexibility, and data integrity.

## 2. What you'll learn?

**By the end of this course, you should be able to:**

- Describe the role of data modeling in database design
- Create conceptual, logical, and physical models using Oracle Data Modeler
- Apply normalization rules to design efficient relational schemas
- Use Entity-Relationship (ER) modeling techniques
- Define keys, constraints, and relationships for integrity and performance
- Generate physical database designs from logical models
- Manage model versioning and reporting with Oracle Data Modeler
- Apply best practices in relational database design

## 3. Target Audience

- Database designers and architects
- Application developers working with relational databases
- Data modelers and analysts
- DBAs who need to understand database design principles

## 4. Pre-Requisites

### Familiarity with:

- Basic relational database concepts
- SQL fundamentals
- General understanding of data structures

## 5. Course content

### 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
- Copyright © 2017, Oracle. All rights reserved. Page 2
- 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
- Copyright © 2017, Oracle. All rights reserved. Page 3
- 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