Oracle Data Modelling & Database Design Course Content:35-40hours

Course Outline

Introduction to Modeling

- List the reasons why modeling is important
- Describe the phases of the Database and Application Development Lifecycle
- Identify which modeling approach to use for a given situation

Document the Business Background

- Define and identify business objectives,
- assumptions, critical success factors,
- key performance indicators and problems
- Establish Business Direction Objectives

Build a Process Model (Data Flow Diagram)

- List the reasons why process modeling is useful
- Describe the components of a Data Flow Diagram
- Build a Data Flow Diagram

Use SQL Developer Data Modeler to Create Your Data Flow Diagram

- Load and set the default options for Oracle SQL Developer Data Modeler
- Build a Data Flow Diagram using Oracle SQL Developer Data Modeler
- Edit the Layout of your Data Flow Diagram
- Opening and Saving the Process Model

Validate Your Data Flow Diagram

- Validate a DFD based on set of DFD Rules
- Identify different types of processes
- Decompose Processes into Primitive Processes

Identify Entities and Attributes

- Identify and Diagram Entities
- Identify and Diagram Attributes

Identify Relationships

- Create a relationship between two entities
- Model relationships using a relationship matrix
- Determining a Relationship’s Existence
- Naming the Relationship
- Determining the Relationship’s Cardinality
- Validating the Relationship

Assign Unique Identifiers

- Identify unique identifiers for entities and relationships
- Identifying Relationships with Multiple Entities
- Non-Identifying Relationships
- Primary and Secondary Unique Identifiers
- Searching for Unique Identifiers

Use Oracle SQL Developer Data Modeler to Create an Entity Relationship Diagram

- Examine the General Options for Logical Data Modeling
- Build an ERD in Oracle SQL Developer Data Modeler
- Edit the Layout of your ERD
- Create a Subview and Display

Validate Your Entity Relationship Diagram

- Apply Diagram Layout and Attribute Rules
- Distinguish Entities from Attributes
- Evaluate Attribute Optionality
- Supplement the ERD with Useful Information
- Create Reports

Normalize Your Data Model

- Define Normalization
- Normalize your ERD to third normal form

Validate Relationships

- Resolve M:M Relationships
- Model Hierarchical Data
- Examine Recursive Relationships
- Model Exclusive Relationships
- Model Entity Type Hierarchies
- Model Data Over Time

Add and Use Data Types

- Create different types of data types
- Build a Data Type model
- Analyze various relationships between structured types on your data type model
- Assign data types to the attributes in your logical data model

Put It All Together

- Build an ERD from a Case Study

Map Your Entity Relationship Diagram to a Relational Database Design

- Describe why a database design is needed
- Decide on naming conventions and rules
- Map the logical model into a relational model
- Map exclusive relationships to foreign keys
- Map subtypes to tables
- Apply general options and setting the Compare/Copy options
- View the mapping comparison and synchronize deleted objects
- Utilize the SQL Developer Data Modeler tool

Analyze your Relational Model

- Modify table properties according to requirements
- Discover foreign keys
- Create an index
- Use the Table to View wizard
- Use the View to Table wizard
- Specify volume properties
- Define spatial properties and column groups
- Create views

Denormalize Your Design to Increase Performance

- Recognize when denormalization techniques can be used in your relational model
- What Is Denormalization?
- Storing Derivable Values
- Pre-Joining Tables
- Hard-Coded Values
- Keeping Details with the Master Table
- Repeating Current Detail with the Master Table
- Hierarchy Level Indicator
- **Define Your Physical Model**
- Create objects in a physical model
- Refine relational model objects in the physical model
- Generate Your Database
- Generate DDL for your Database
- Selecting a Database, "Create" Selection, and DDL Script
- Assigned to Users, "Drop" Selection, and Name Substitution
- Including Table Scripts, Masking Oracle Errors, and Using Find
- DDL Preferences
- Design Rules
- Working With Rule Sets and Custom Rules
- Working With Libraries and Transformations

Alter an Existing Design

- Using Import to Create a Model
- Importing an Existing Database and Domains
- Creating a Logical Data Model from Your Relational Model
- Reviewing and Making Changes to Your Logical Model
- Forward Engineering to a New Relational Model
- Comparing Your Relational Model Changes With What is in the Database
- Synchronizing the Data Dictionary With Changes in a Model
- Running the Generated DDL Script in Oracle SQL Developer and Confirming the Change

Working in a Collaborative Environment

- Describe Version Control and its benefits
- Work With Data Modeler and Subversion
- Use Subversion With a Design