

Entity Framework 6

EF6 - Version: 1

 4 days Course

Description:

Entity Framework is the new ORM and data access technology introduced by Microsoft. Entity framework provides an easy and efficient mapping from relational data into business entities. It is easy to use and it is integrated with many other Microsoft technologies. The Entity framework relies on the idea of modeling to provide the right level of abstraction. It describes the same data in two different models and provides the mapping between the two. The conceptual model describes the business entities and the storage model describes how the database store that data. The ADO.NET Entity Framework is designed to enable developers to create data access applications by programming against a conceptual application model instead of programming directly against a relational storage schema.

Intended audience:

Developers and Designers

Prerequisites:

.Net framework

C#

ADO.Net

LINQ

Objectives:

Topics:

Introduction

- About the course
- Storage trends
- Storing Application Data
- Relational DB (Pro / Con)
- NoSQL storage

Data Access Foundation (ADO.NET)

- ADO.NET
- Stored Procedures
- Transactions
- Isolation Levels

ORM Introduction

- Introduction
- What is Entity Framework?
- Why Do We Care?
- What Should We Expect?
- Object-Oriented VS RDBMS
- OO vs. Relational Database

Get Started with EF 6

- How to get started
- EF 6, How to start
- Attach database to Visual Studio
- Choosing the Development Work
- Entity Framework Concept
- EDM (Entity Data Model)
- EF Designer (EDM)
- Limitation

Code First Fundamentals

- When to use Code First
- Getting started
- Add DTOs (Data Transfer Objects)
- Add Context
- Start CRUD operation
- Default conversion
- Code First and EDM
- Entity Framework Power Tools
- Complex Type
- Add and Read data

Configuration

- Database Initialization
- Configuration options
- DbConfiguration discovery
- Connection Convention
- Choosing the Provider
- Handling Initialization
- Custom Initialization

Logging

- EF6 Logging API
- Advance Logging
- SQL Server Profiler

Entity Framework Tuning

- Unit of work
- Local cache
- Lazy
- Change Tracking Proxies
- Limitation
- Laziness and Serialization
- Unit of Work / Save

- Auto Detect Changes
- On-Demand Detect Changes
- Manual State Management
- Batch operation
- Query the tracking state
- Transaction
- Distributed Transaction
- Disable Laziness

Fine Tuning Configuration

- Limitations of Convention
- Schema definition
- Annotation
- Annotation Limitation
- Fluent API
- Fluent API Limitation
- Entity Framework Power Tools

Basic Interceptions

- DbContext Extensions

EF Designer

- Query and Save Changes
- Connection String
- Connection String (Best Practice)
- Model Browser
- Update Model from Database

T4

- Code Generation
- T4 template

- When to use T4
- Directives
- Hybrid Text and Code
- T4 Syntax
- T4 Tooling
- Reuse T4 across multiple files
- Tuning the EDM
- Disable proxies

Best practice

- Separation of Concern
- Multiple EDMs with cross-entities (same assembly)
- Separation of EDMs
- Entity Color
- Multiple-Diagram View
- EF Designer vs. Code First

Service-Oriented Architecture

- SOA orientation
- Distributed (SOA) scenario
- Batch loading
- Loading a graph with a single request
- Loading indirect, related entities
- Loading multiple related enti
- Include's Limitations
- Explicit loading of related entities
- Fine-tuning explicit loading

Code First Migrations

- What is Migration
- Code First Migration
- Trace schema changes
- Update the Database

- Migration Summary

Stored Procedures

- Entity Framework and Stored Procedures
- Insert, Update, Delete
- Code First
- EF Designer

Exception Handling

- How to

Concurrency

- Conflicts
- Common Concurrency Patterns
- Common Optimistic Patterns
- Code First
- Ef Designer

Spatial and Enum

- Enum support
- Spatial Types
- Geographic standard
- Parsing Geographical Data

Performance

- Paging
- Batch Update / Delete
- Bulk Insert
- Pre-Generated Mapping Views
- Default Mapping (Include)

- Multiple Result Set

Asynchronous

- Async (Introduction to I/O operations)
- Task is Data Structure
- Scheduling a Task using Factory
- Task<T>
- New async syntax
- async and UI
- Using
- Exception flow control
- IOCP
- Pitfall and Best Practice

Trouble shooting

- Deferred Execution
- Expression vs. Delegate
- Update From Database
- Namespace agnostic
- Code First and Shared DTOs
- Code First and Shared
- Configuration racing problem
- Migration

Unit Testing

- Unit Test for Entity Framework
- Isolating the context
- Mocking the Context
- Prerequisite

Appendix

- Inheritance: Table Per Type (TPT)
- Inheritance: Table Per Hierarchic (TPH)
- Map a Single Entity to Multiple Tables
- Table Splitting
- Trouble Shooting OO

◦ Summary