

Entity Framework - Simplified

**A Quick Reference For
Developers**

This is the first edition : 2013.

www.manzoorthetrainer.com

Entity Framework - Simplified

Table of Content

About the Author.....	3
About the Technical Reviewers And Supporters.....	4
Chapter 1 : Introduction to Entity Framework.....	5
Chapter 2 : Creating an Entity Data Model.....	7
Chapter 3 : Exploring Entity Data Model	16
Chapter 4: Exploring Entity Data Model in Entity Framework 5.X	24
Chapter 5 : Performing an Insert Operation	26
Chapter 6 : Performing a Select Operation	31
Chapter 7 : Performing a Search Operation	34
Chapter 8 : Performing an Update Operation	36
Chapter 9 : Performing a Delete Operation	40
Chapter 10 : Working With Stored Procedures-I	44
Chapter 11 : Working With Stored Procedures-II	52
Chapter 12 : Working With Stored Procedures-III	57
Chapter 13 : Immediate Mode Vs Differed Mode	61
Chapter 14 : Working With Transactions	64
Chapter 15 : Working With Navigation Properties	73
Chapter 16 : Lazy Loading in Entity Framework	76
Chapter 17 : Asp.Net Dynamic Data Entity Website	91

About the Author



ManzoorTheTrainer – MCTS, MCPD and MCT. Planning for Ph.D in the domain of Middleware Technologies with over 14+ years of teaching and over 8+ years of Development Exp. Has an excellent track record as one of the best IT faculty in imparting technology. Makvin's Ace Trainer who believes in giving the real essence of technology to the learner this uniqueness has made me earn a strong foothold in the IT Training Industry.

[Microsoft Certified Professional Profile \(MCP\)](#)

Founder of [ManzoorTheTrainer](#)

www.manzoorthetrainer.com

Entity Framework - Simplified

About the Technical Reviewers And Supporters:

I am really thankful to Almighty and all the followers of my video training portal www.ManzoorTheTrainer.com and encouraging me to take an step of writing a technical book on the latest data access technology i.e., Entity Framework. I am also thankful to Mr. Satish Kumar Mandava, my colleague having 6+ yrs of .NET expertise for reviewing this book and giving feedbacks for betterment.

I am also thankful to my organization Efficacy System and my PM Mr.Srikanth (PMP) for motivating me always.

Extending my special thanks to Mr. Shivprasad Koirala Sir for encouraging and guiding me and to all the experts from whom I got inspired.

www.manzoorhetrainer.com

Entity Framework - Simplified

Chapter 1: Introduction to Entity Framework

Few Necessities Of New Data Access Technology Entity Framework:

- Want easy and automated way of mapping between Objects (Classes) and Relations (Tables).
- Want to eliminate the need for most of the data-access code that developers usually need to write.
- Want to write pure objects, less code and simplify the process of updating the database.
- Want layer of abstraction between your Application and Database.
- Want to write C# language syntaxed queries to work with any database with full intellisense support for columns and tables name. (Strongly Typed)
- Want to make database independent apps.

Introduction To Entity Framework:

- *The Microsoft ADO.NET Entity Framework is an Object/Relational Mapping (ORM) framework that enables developers to work with relational data as domain-specific objects, eliminating the need for most of the data access plumbing code that developers usually need to write.*
- *Using the Entity Framework, developers issue queries using LINQ, then retrieve and manipulate data as strongly typed objects. The Entity Framework's ORM implementation provides services like change tracking, identity resolution, lazy loading, and query translation so that developers can focus on their application-specific business logic rather than the data access fundamentals.*

- Definition of Entity Framework given by Microsoft

Entity Framework - Simplified

Various approaches to work with Entity Framework:

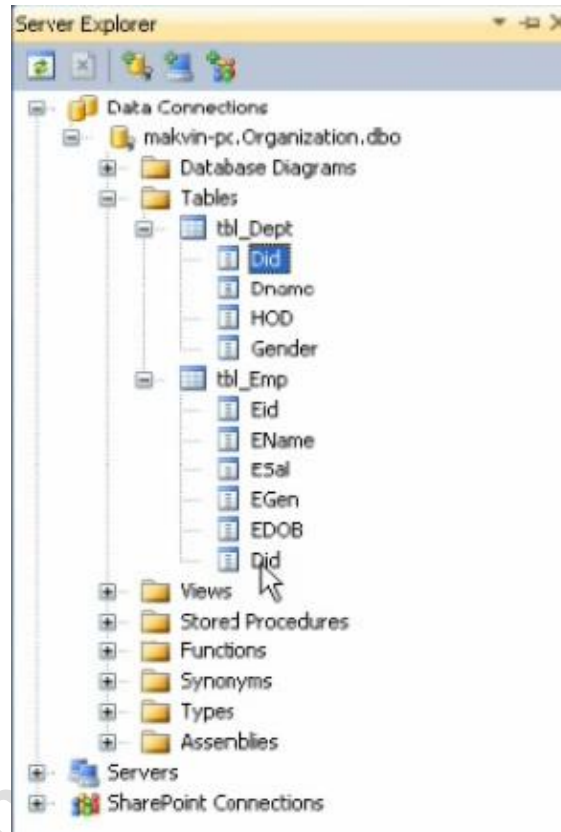
- Database First (This eBook is Focused on):
 - Creating the model from database (Generating Classes or Class Diagram from Database Tables).
- Model First
 - Creating the database from model (Generating Database Tables from Classes or Class Diagram).
- Code First (Some Time POCO)
 - Coding classes against the existing Database is Code First.
 - Creating the database from the coded classes(No class diagrams) is POCO.

www.manzoorthetrainer.com

Entity Framework - Simplified

Chapter 2: Creating an Entity Data Model

- In this chapter, I'm going to show you How to create your Entity Data Model.
- This is very simple. I've a database called as organization in which I've department table and employee table.
- I'm going to use the same database throughout our course on Entity Framework.

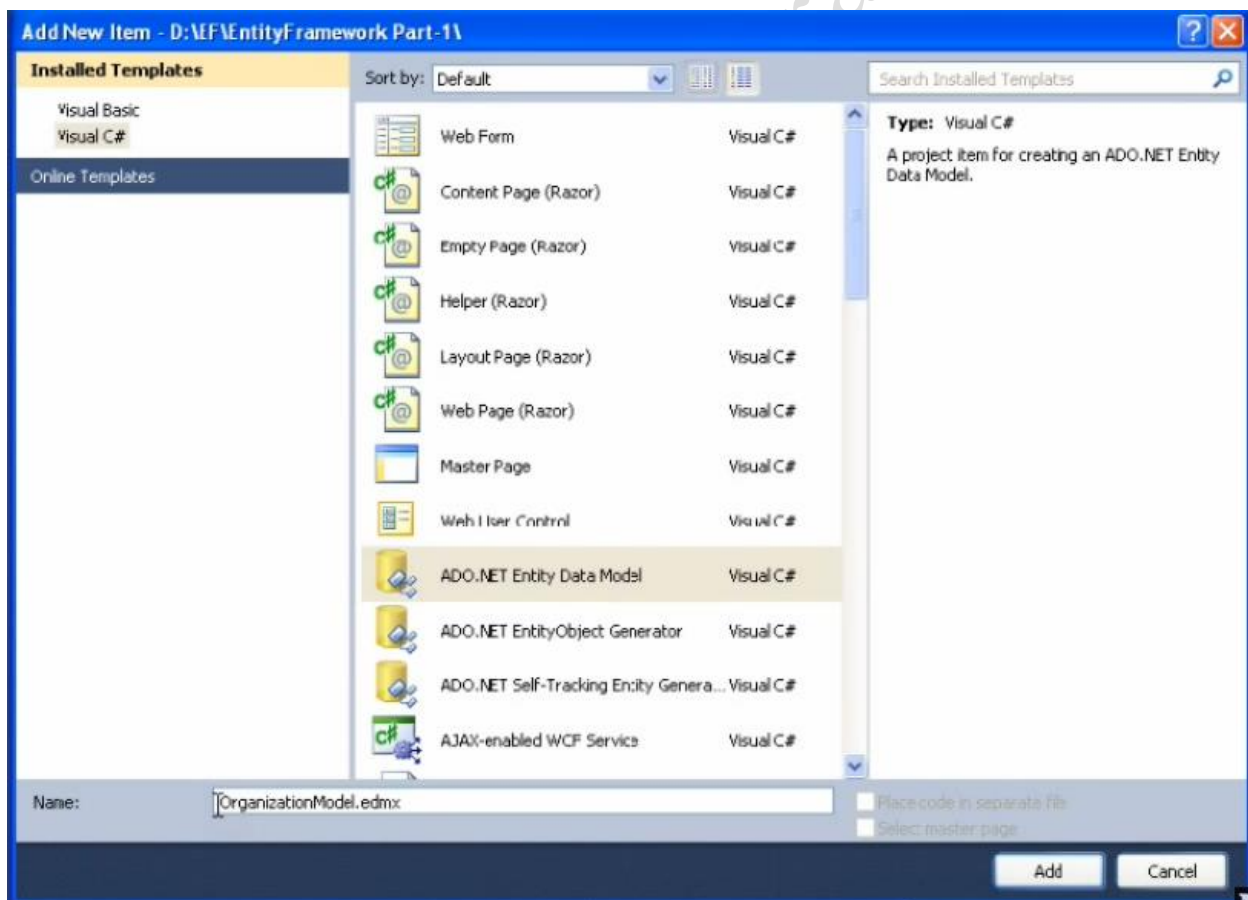


- So, In Department table I've Did(Department ID), Dname(Department Name), HOD and Gender in which Did is primary key.
- In Employee table I've Eid(Employee ID), Ename(Employee Name), Esal(Employee Salary), EGen(Employee Gender), EDOB(Employee Date Of Birth) and Did is the foreign key from department table.

Entity Framework - Simplified

Creating Entity Data Model:

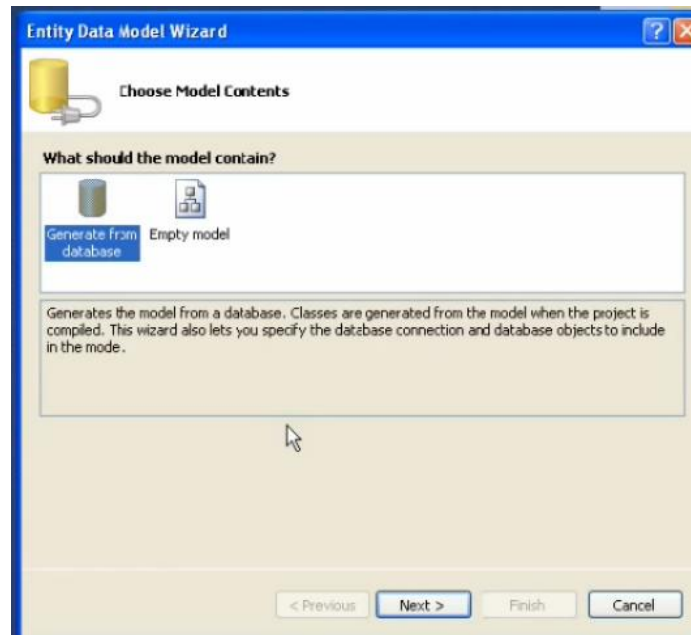
- Create a web Project say EntityFrameworkPart-1
- Lets create our entity data model(EDM) which is the heart of Entity Framework and it is very simple.
- I'll goto my Project what I've created as EntityFrameworkPart-1
- Right click the project
- Add new item
- Select ADO.NET Entity Data Model
- And I'll name it as OrganizationModel.edmx
- Click Add
- It is going to add in App_Code folder
- I'll say yes



- It'll fire a window with two options for me.

Entity Framework - Simplified

1. Generate Empty Database (Create a model from an existing database)
2. Empty Model (Create model first then create the database)

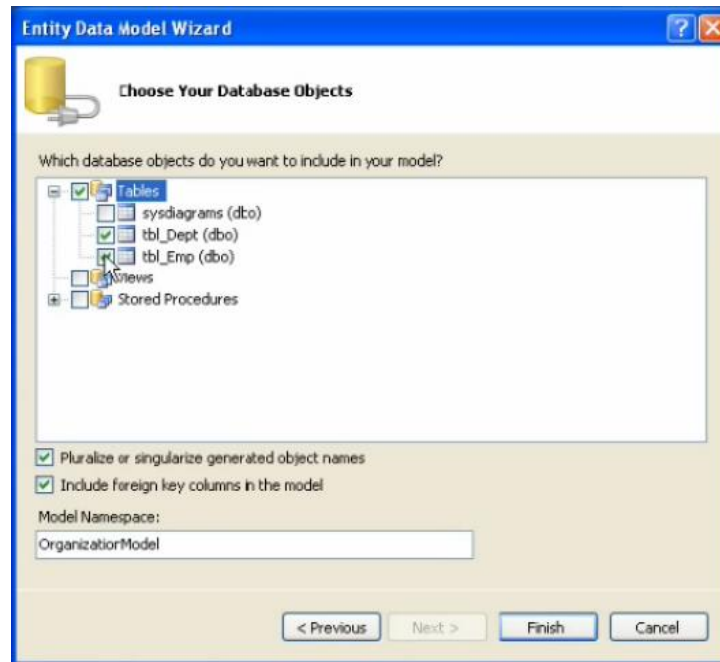


- As I want to generate Entity Data Model from an existing database. Select Generate Empty Database click Next.
- You'll see the database is organization and Entity connection string (same as your connection string which provides the information about the provider, DataSource, Initial Catalog , UserID, Password, Integrated Security....)

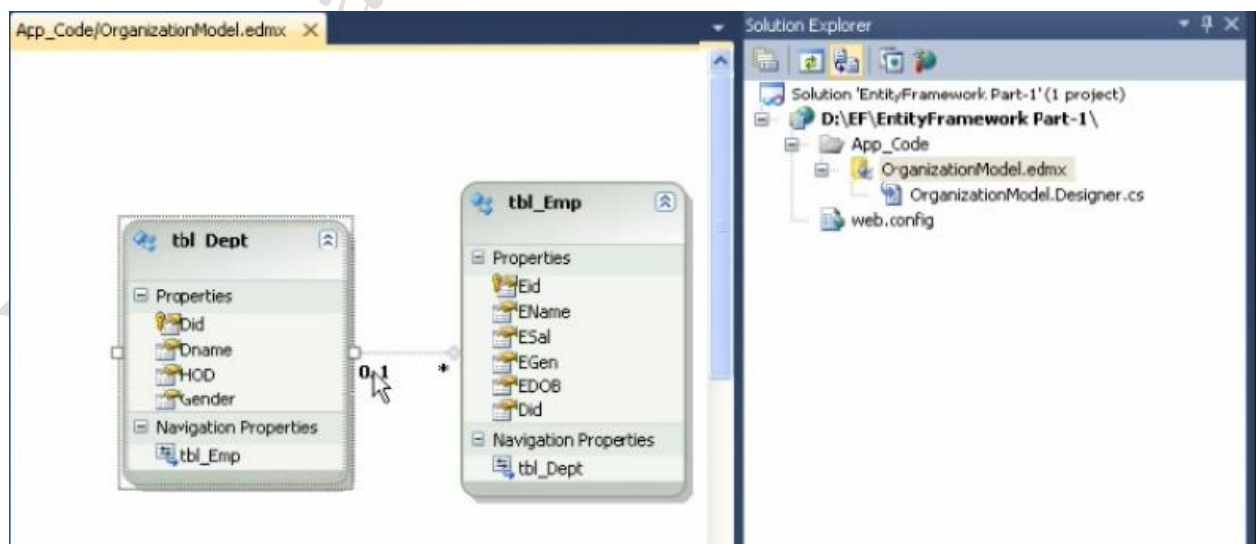


Entity Framework - Simplified

- The connection string also contains some extra information as metadata about your Entity Framework. Save entity connection settings in Web.Config as OrganizationEntities click next.



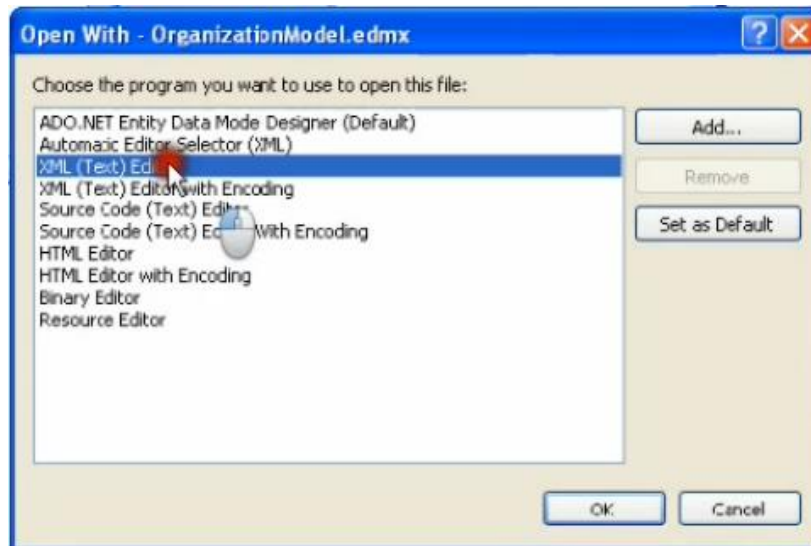
- At this point It'll retrieve the database information (tables, stored procedures..) As of now I'm going to work with two tables "tbl_Dept" and "tbl_Emp". I'm going to add these two tables to my Entity Data Model
- Leave the options checked for Pluralize or Singularize generated object names and Include foreign key columns in the model click on Finish.
- Here's my Entity Data Model



- As you might have seen, in a single department I can have n number of employees.

Entity Framework - Simplified

- So, we've one-to-many relationship and all the columns that I've in the database are turned into properties for both the tables.
- There are two extra properties called as Navigation Properties. That means whenever I've relation between two tables for navigation from either tbl_Dept to tbl_Emp or tbl_Emp to tbl_Dept .
- So, this is my Entity Data Model.
- Right click on Entity Data Model
- Open with XML Editor Click Ok then Yes for Prompt window



- This is the XML file that is generated automatically for Entity Data Model

```

App_Code/OrganizationModel.edmx X
<?xml version="1.0" encoding="utf-8"?>
<edmx:Edmx Version="2.0" xmlns:edmx="http://schemas.microsoft.com/ado/2008/10/edmx">
  <!-- EF Runtime content -->
  <edmx:Runtime>
    <!-- $DL content -->
    <edmx:StorageModels>...</edmx:StorageModels>
    <!-- L3DL content -->
    <edmx:ConceptualModels>...</edmx:ConceptualModels>
    <!-- C-S mapping content -->
    <edmx:Mappings>...</edmx:Mappings>
  </edmx:Runtime>
  <!-- EF Designer content (DO NOT EDIT MANUALLY BELOW HERE) -->
  <Designer xmlns="http://schemas.">...</Designer>
</edmx:Edmx>

```

- Let us have a look into this XML file and let us see what it contains
 1. Runtime tag `<edmx:Runtime>` and
 2. Designer tag `<Designer>`

Entity Framework - Simplified

- The Designer tag contains some UI related to our Entity Data Model (Graphical table data)
- The Runtime tag consists of 3 parts
 1. SSDL (Storage Schema Definition Language) [Info about database, tables, columns.. for SQL data]

```

<edmx:StorageModels>
  <Schema Namespace="OrganizationModel.Store" Alias="Self" Provider="System.Data.SqlClient"
  ProviderManifestToken="2008"
  xmlns:store="http://schemas.microsoft.com/ado/2007/12/edm/EntityStoreSchemaGenerator"
  xmlns="http://schemas.microsoft.com/ado/2009/02/edm/ssdl">
    <EntityContainer Name="OrganizationModelStoreContainer">
      <EntitySet Name="tbl_Dept" EntityType="OrganizationModel.Store.tbl_Dept"
      store:Type="Tables" Schema="dbo" />
      <EntitySet Name="tbl_Emp" EntityType="OrganizationModel.Store.tbl_Emp" store:Type="Tables"
      Schema="dbo" />
      <AssociationSet Name="FK_tbl_Emp_tbl_Dept"
      Association="OrganizationModel.Store.FK_tbl_Emp_tbl_Dept">
        <End Role="tbl_Dept" EntitySet="tbl_Dept" />
        <End Role="tbl_Emp" EntitySet="tbl_Emp" />
      </AssociationSet>
    </EntityContainer>
    <EntityType Name="tbl_Dept">
      <Key>
        <PropertyRef Name="Did" />
      </Key>
      <Property Name="Did" Type="int" Nullable="false" StoreGeneratedPattern="Identity" />
      <Property Name="Dname" Type="varchar" MaxLength="50" />
      <Property Name="HOD" Type="varchar" Nullable="false" MaxLength="50" />
      <Property Name="Gender" Type="varchar" Nullable="false" MaxLength="50" />
      <Property Name="Active" Type="bit" />
    </EntityType>
    <EntityType Name="tbl_Emp">
      <Key>
        <PropertyRef Name="Eid" />
      </Key>
      <Property Name="Eid" Type="int" Nullable="false" StoreGeneratedPattern="Identity" />
      <Property Name="ENAME" Type="varchar" Nullable="false" MaxLength="50" />
      <Property Name="ESal" Type="float" Nullable="false" />
      <Property Name="EGen" Type="varchar" Nullable="false" MaxLength="10" />
      <Property Name="EDOB" Type="datetime" Nullable="false" />
      <Property Name="Did" Type="int" />
    </EntityType>
    <Association Name="FK_tbl_Emp_tbl_Dept">
      <End Role="tbl_Dept" Type="OrganizationModel.Store.tbl_Dept" Multiplicity="0..1" />
      <End Role="tbl_Emp" Type="OrganizationModel.Store.tbl_Emp" Multiplicity="*" />
      <ReferentialConstraint>
        <Principal Role="tbl_Dept">
          <PropertyRef Name="Did" />
        </Principal>
        <Dependent Role="tbl_Emp">
          <PropertyRef Name="Did" />
        </Dependent>
      </ReferentialConstraint>
    </Association>
    <Function Name="SP_GetAllEmployee" Aggregate="false" BuiltIn="false"
    NiladicFunction="false" IsComposable="false"
    ParameterTypeSemantics="AllowImplicitConversion" Schema="dbo" />
    <Function Name="SP_GetEmployeesByDid" Aggregate="false" BuiltIn="false" NiladicFunction="false"
    IsComposable="false" ParameterTypeSemantics="AllowImplicitConversion" Schema="dbo">
      <Parameter Name="Did" Type="int" Mode="In" />
    </Function>
  </Schema>

```

Entity Framework - Simplified

```

    <Function Name="SP_GetEmpNameAndSalaryByEid" Aggregate="false" BuiltIn="false"
      NiladicFunction="false" IsComposable="false"
      ParameterTypeSemantics="AllowImplicitConversion" Schema="dbo">
    <Parameter Name="Eid" Type="int" Mode="In" />
    <Parameter Name="ENAME" Type="varchar" Mode="InOut" />
    <Parameter Name="ESal" Type="float" Mode="InOut" />
    </Function>
</Schema></edmx:StorageModels>

```

2. CSDL (Conceptual Schema Definition Language) [Properties for each table and each column.. for C# data]

```

<edmx:ConceptualModels>
  <Schema Namespace="OrganizationModel" Alias="Self"
    xmlns:annotation="http://schemas.microsoft.com/ado/2009/02/edm/annotation"
    xmlns="http://schemas.microsoft.com/ado/2008/09/edm">
    <EntityContainer Name="OrganizationEntities" annotation:LazyLoadingEnabled="true">
      <EntityType Name="tbl_Dept">
        <Key>
          <PropertyRef Name="Did" />
        </Key>
        <Property Name="Did" Type="Int32" Nullable="false"
          annotation:StoreGeneratedPattern="Identity" />
        <Property Name="Dname" Type="String" MaxLength="50" Unicode="false" FixedLength="false" />
        <Property Name="HOD" Type="String" Nullable="false" MaxLength="50" Unicode="false"
          FixedLength="false" />
        <Property Name="Gender" Type="String" Nullable="false" MaxLength="50" Unicode="false"
          FixedLength="false" />
        <NavigationProperty Name="tbl_Emp" Relationship="OrganizationModel.FK_tbl_Emp_tbl_Dept"
          FromRole="tbl_Dept" ToRole="tbl_Emp" />
        <Property Type="Boolean" Name="Active" />
      </EntityType>
      <EntityType Name="tbl_Emp">
        <Key>
          <PropertyRef Name="Eid" />
        </Key>
        <Property Name="Eid" Type="Int32" Nullable="false"
          annotation:StoreGeneratedPattern="Identity" />
        <Property Name="ENAME" Type="String" Nullable="false" MaxLength="50" Unicode="false"
          FixedLength="false" />
        <Property Name="ESal" Type="Double" Nullable="false" />
        <Property Name="EGen" Type="String" Nullable="false" MaxLength="10" Unicode="false"
          FixedLength="false" />
        <Property Name="EDOB" Type="DateTime" Nullable="false" />
        <Property Name="Did" Type="Int32" />
      </EntityType>
      <FunctionImport Name="SP_GetAllEmployee"
        ReturnType="Collection(OrganizationModel.SP_GetAllEmployee_Result)" />
      <FunctionImport Name="SP_GetEmployeesByDid"
        ReturnType="Collection(OrganizationModel.SP_GetEmployeesByDid_Result)">
        <Parameter Name="Did" Mode="In" Type="Int32" />
      </FunctionImport>
      <FunctionImport Name="SP_GetEmpNameAndSalaryByEid">
        <Parameter Name="Eid" Mode="In" Type="Int32" />
        <Parameter Name="ENAME" Mode="InOut" Type="String" />
        <Parameter Name="ESal" Mode="InOut" Type="Double" />
      </FunctionImport>
    </EntityContainer>
  </Schema>

```

Entity Framework - Simplified

```

        <NavigationProperty Name="tbl_Dept"
            Relationship="OrganizationModel.FK_tbl_Emp_tbl_Dept" FromRole="tbl_Emp" ToRole="tbl_Dept"
        />
    </EntityType>
    <Association Name="FK_tbl_Emp_tbl_Dept">
        <End Role="tbl_Dept" Type="OrganizationModel.tbl_Dept" Multiplicity="0..1" />
        <End Role="tbl_Emp" Type="OrganizationModel.tbl_Emp" Multiplicity="*" />
        <ReferentialConstraint>
            <Principal Role="tbl_Dept">
                <PropertyRef Name="Did" />
            </Principal>
            <Dependent Role="tbl_Emp">
                <PropertyRef Name="Did" />
            </Dependent>
        </ReferentialConstraint>
    </Association>
    <ComplexType Name="SP_GetAllEmployee_Result">
        <Property Type="Int32" Name="Eid" Nullable="false" />
        <Property Type="String" Name="ENAME" Nullable="false" MaxLength="50" />
        <Property Type="Double" Name="ESal" Nullable="false" />
        <Property Type="DateTime" Name="EDOB" Nullable="false" Precision="23" />
        <Property Type="String" Name="Dname" Nullable="true" MaxLength="50" />
        <Property Type="String" Name="HOD" Nullable="false" MaxLength="50" />
    </ComplexType>
    <ComplexType Name="SP_GetEmployeesByDid_Result">
        <Property Type="Int32" Name="Eid" Nullable="false" />
        <Property Type="String" Name="ENAME" Nullable="false" MaxLength="50" />
        <Property Type="Double" Name="ESal" Nullable="false" />
        <Property Type="String" Name="EGen" Nullable="false" MaxLength="10" />
        <Property Type="DateTime" Name="EDOB" Nullable="false" Precision="23" />
        <Property Type="Int32" Name="Did" Nullable="true" />
    </ComplexType>
</Schema>
</edmx:ConceptualModels>

```

3. C-S Mapping [Column Mappings]

```

<edmx:Mappings>
    <Mapping Space="C-S" xmlns="http://schemas.microsoft.com/ado/2008/09/mapping/cs">
        <EntityContainerMapping StorageEntityContainer="OrganizationModelStoreContainer"
            CdmEntityContainer="OrganizationEntities">
            <EntityTypeMapping Name="tbl_Dept"><EntityTypeMapping
                TypeName="OrganizationModel.tbl_Dept"><MappingFragment StoreEntitySet="tbl_Dept">
                    <ScalarProperty Name="Active" ColumnName="Active" />
                    <ScalarProperty Name="Did" ColumnName="Did" />
                    <ScalarProperty Name="Dname" ColumnName="Dname" />
                    <ScalarProperty Name="HOD" ColumnName="HOD" />
                    <ScalarProperty Name="Gender" ColumnName="Gender" />
                </MappingFragment></EntityTypeMapping></EntityTypeMapping>
            <EntityTypeMapping Name="tbl_Emp"><EntityTypeMapping
                TypeName="OrganizationModel.tbl_Emp"><MappingFragment StoreEntitySet="tbl_Emp">
                    <ScalarProperty Name="Eid" ColumnName="Eid" />
                    <ScalarProperty Name="ENAME" ColumnName="ENAME" />
                    <ScalarProperty Name="ESal" ColumnName="ESal" />
                    <ScalarProperty Name="EGen" ColumnName="EGen" />
                    <ScalarProperty Name="EDOB" ColumnName="EDOB" />
                    <ScalarProperty Name="Did" ColumnName="Did" />
                </MappingFragment></EntityTypeMapping></EntityTypeMapping>
            <FunctionImportMapping FunctionImportName="SP_GetAllEmployee"
                FunctionName="OrganizationModel.Store.SP_GetAllEmployee">
                <ResultMapping>
                    <ComplexTypeMapping TypeName="OrganizationModel.SP_GetAllEmployee_Result">
                        <ScalarProperty Name="Eid" ColumnName="Eid" />
                        <ScalarProperty Name="ENAME" ColumnName="ENAME" />
                        <ScalarProperty Name="ESal" ColumnName="ESal" />
                        <ScalarProperty Name="EDOB" ColumnName="EDOB" />
                    </ComplexTypeMapping>
                </ResultMapping>
            </FunctionImportMapping>
        </EntityContainerMapping>
    </Mapping>
</edmx:Mappings>

```

Entity Framework - Simplified

```

    <ScalarProperty Name="Dname" ColumnName="Dname" />
    <ScalarProperty Name="HOD" ColumnName="HOD" />
  </ComplexTypeMapping>
</ResultMapping>
</FunctionImportMapping>
  <FunctionImportMapping FunctionImportName="SP_GetEmployeesByDid"
    FunctionName="OrganizationModel.Store.SP_GetEmployeesByDid">
  <ResultMapping>
    <ComplexTypeMapping TypeName="OrganizationModel.SP_GetEmployeesByDid_Result">
      <ScalarProperty Name="Eid" ColumnName="Eid" />
      <ScalarProperty Name="ENAME" ColumnName="ENAME" />
      <ScalarProperty Name="ESal" ColumnName="ESal" />
      <ScalarProperty Name="EGen" ColumnName="EGen" />
      <ScalarProperty Name="EDOB" ColumnName="EDOB" />
      <ScalarProperty Name="Did" ColumnName="Did" />
    </ComplexTypeMapping>
  </ResultMapping>
</FunctionImportMapping>
  <FunctionImportMapping FunctionImportName="SP_GetEmpNameAndSalaryByEid"
    FunctionName="OrganizationModel.Store.SP_GetEmpNameAndSalaryByEid" />
</EntityContainerMapping>
</Mapping>
</edmx:Mappings>

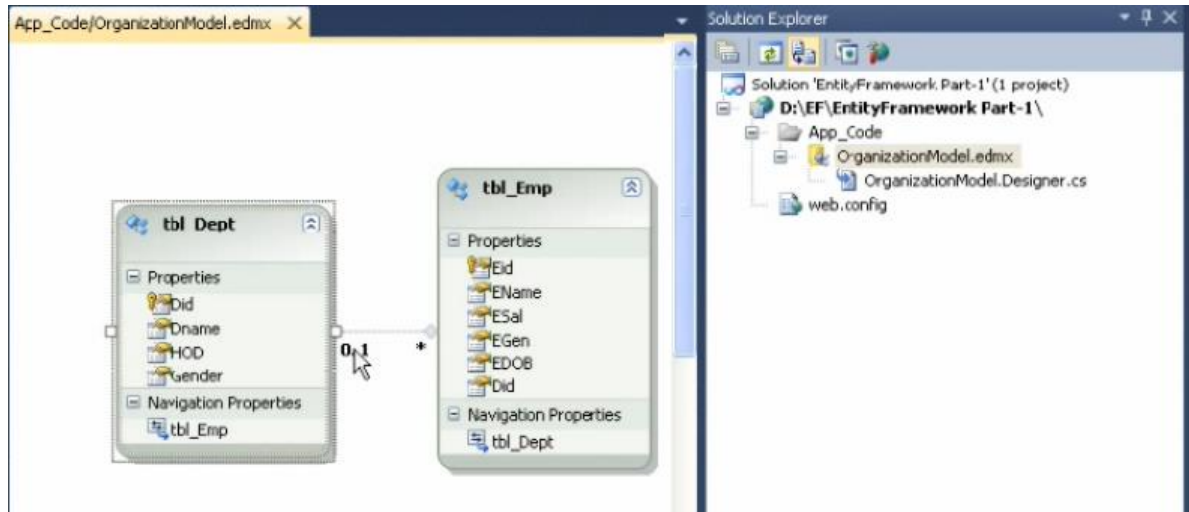
```

- CLR generates 3 different files for Runtime tag at runtime.

Entity Framework - Simplified

Chapter 3: Exploring Entity Data Model

- In this chapter we are going to explore the class that gets generated whenever we create Entity Data Model



- The class is OrganizationModel.Designer.cs and I'll double click this file.

```

//-----
// <auto-generated>
// This code was generated from a template.
//
// Manual changes to this file may cause unexpected behavior in your application.
// Manual changes to this file will be overwritten if the code is regenerated.
// </auto-generated>
//-----

using System;
using System.Data.Objects;
using System.Data.Objects.DataClasses;
using System.Data.EntityClient;
using System.ComponentModel;
using System.Xml.Serialization;
using System.Runtime.Serialization;

[assembly: EdmSchemaAttribute()]
EDM Relationship Metadata

namespace OrganizationModel
{
    Contexts
    Entities
}

```

- Now, there are two important things that you need to remember from this file

Thank You for previewing this eBook

You can read the full version of this eBook in different formats:

- HTML (Free /Available to everyone)
- PDF / TXT (Available to V.I.P. members. Free Standard members can access up to 5 PDF/TXT eBooks per month each month)
- Epub & Mobipocket (Exclusive to V.I.P. members)

To download this full book, simply select the format you desire below

