**Practical 10: Build websites to demonstrate the working of entity frameworks in dot net.**

**📌 Steps to Implement**

1. **Create a SQL Server Database & Table**
2. **Create an ASP.NET Web Application in Visual Studio**
3. **Install & Configure Entity Framework (EF) ORM**
4. **Use EF to perform CRUD operations**
5. **Display data in GridView & allow users to Add, Edit, Delete records**

**1️ SQL Server: Create a Database & Table**

Open **SQL Server Management Studio (SSMS)** and execute:

CREATE DATABASE EmployeeDB;

USE EmployeeDB;

CREATE TABLE Employees (

EmpID INT PRIMARY KEY IDENTITY(1,1),

Name NVARCHAR(100),

Department NVARCHAR(100),

Salary DECIMAL(10,2)

);

INSERT INTO Employees (Name, Department, Salary)

VALUES ('John Doe', 'IT', 60000), ('Jane Smith', 'HR', 55000);

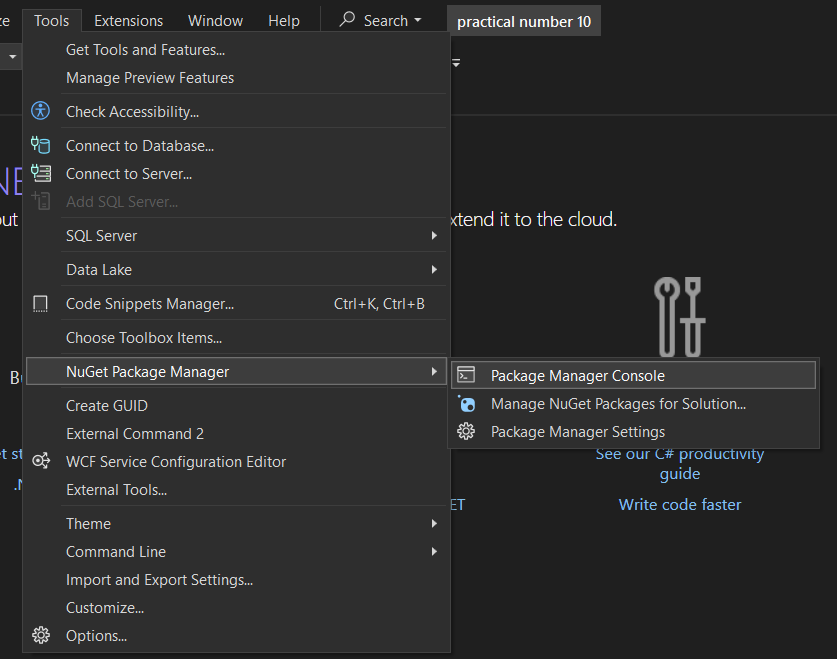
**2️ Create an ASP.NET Web Application**

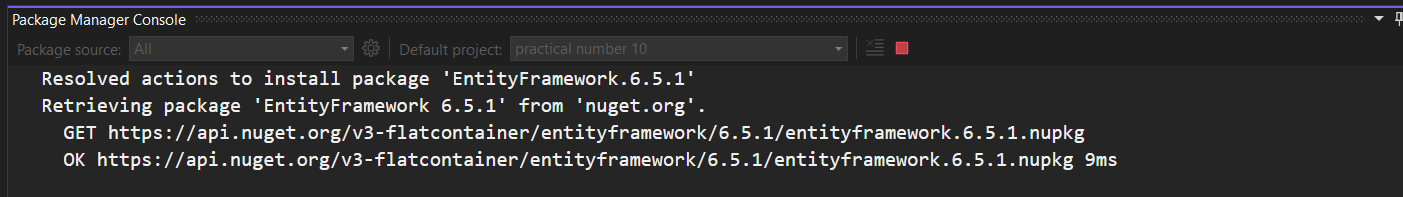
1. Open **Visual Studio** → Create a **New Project** → Choose **ASP.NET Web Application (.NET Framework)**
2. Select **Web Forms** and click **Create**

**3️ Install Entity Framework**

1. Open **Package Manager Console** (Tools → NuGet Package Manager → Package Manager Console)
2. Run the command:

Install-Package EntityFramework

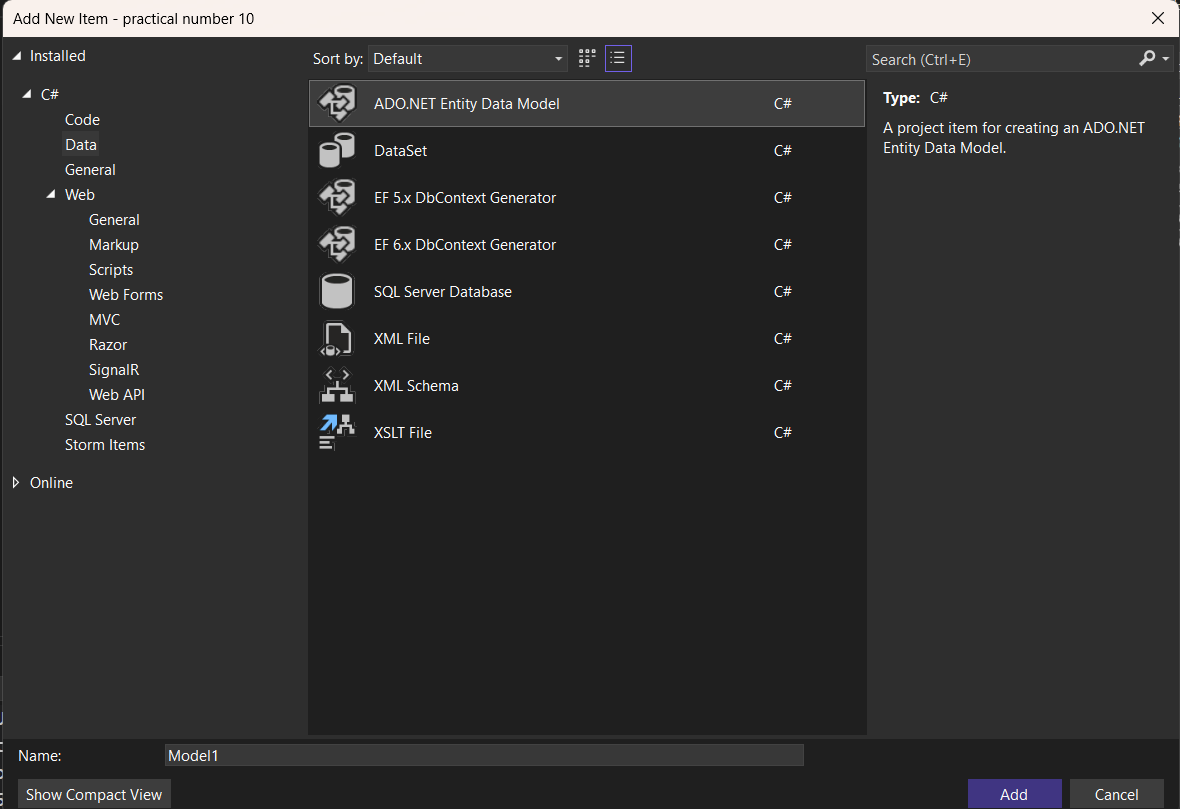


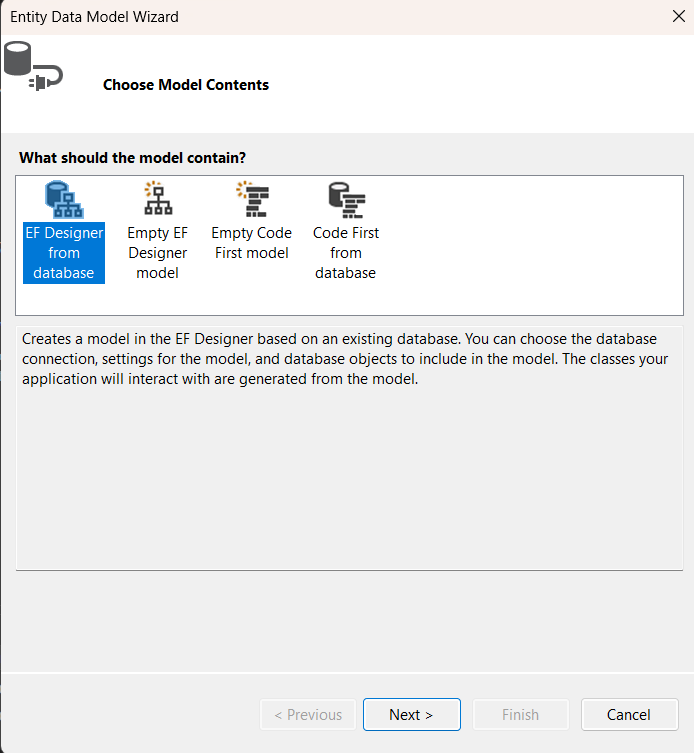


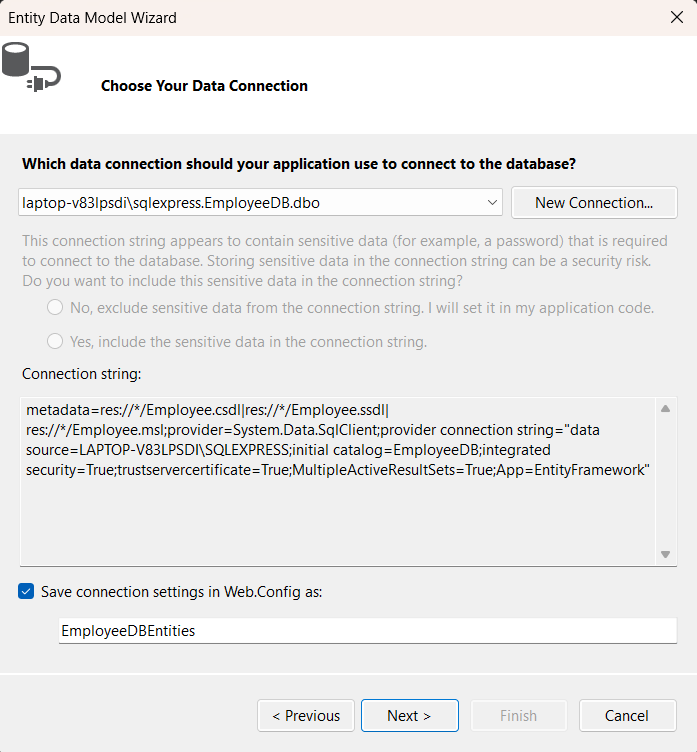
**4️ Create Entity Framework Model**

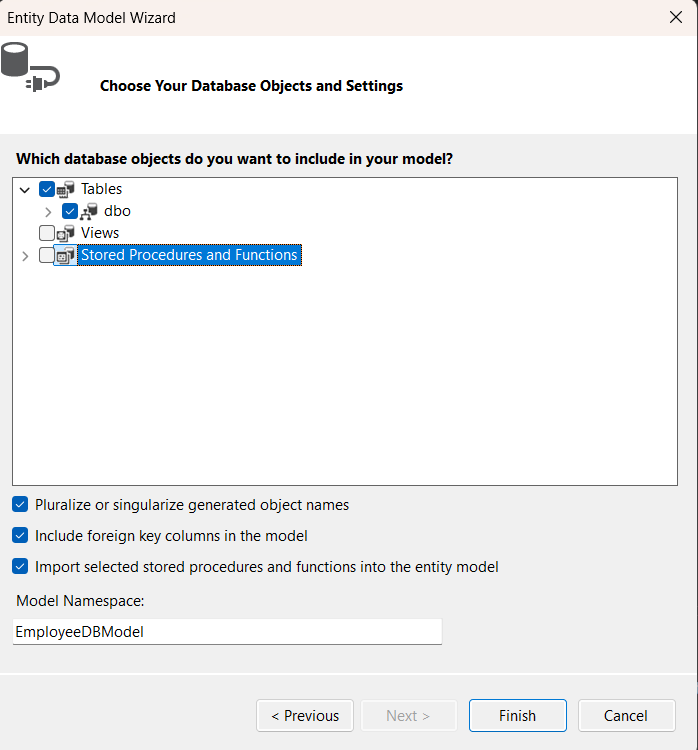
1. Right-click the project → Add → New Item → Select **ADO.NET Entity Data Model**
2. Choose **EF Designer from Database**
3. Select **"EmployeeDB"** as the database
4. Select the **Employees** table → Finish

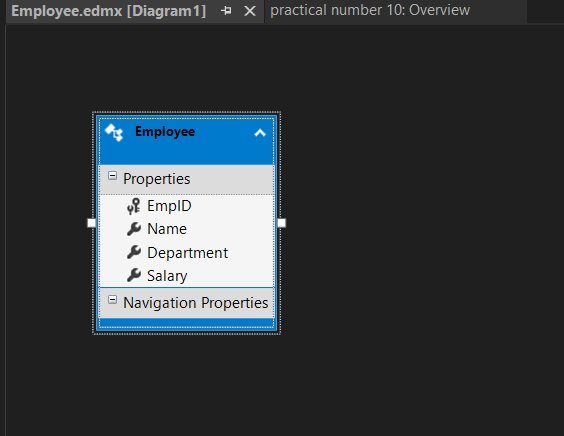
This generates the **Employee.cs model class**.











**5️ Create an ASP.NET Web Form (WebForm1.aspx)**

Modify WebForm1.aspx to include a **GridView** to display records and a **Form** for adding new employees.

<%@ Page Language="C#" AutoEventWireup="true" CodeBehind="WebForm1.aspx.cs" Inherits="practical\_number\_10.WebForm1" %>

<!DOCTYPE html>

<html lang="en">

<head runat="server">

<title>Entity Framework CRUD Demo</title>

</head>

<body>

<form id="form1" runat="server">

<div>

<h2>Employee Management (Entity Framework)</h2>

<!-- Add Employee Form -->

<asp:Label runat="server" Text="Name:"></asp:Label>

<asp:TextBox ID="txtName" runat="server" style="margin-left: 46px"></asp:TextBox>

<br />

<asp:Label runat="server" Text="Department:"></asp:Label>

<asp:TextBox ID="txtDepartment" runat="server"></asp:TextBox> <br />

<asp:Label runat="server" Text="Salary:"></asp:Label>

<asp:TextBox ID="txtSalary" runat="server" style="margin-left: 41px"></asp:TextBox>

<br />

<br />

<br />

<asp:Button ID="btnAdd" runat="server" Text="Add Employee" OnClick="btnAdd\_Click" />

<br /><br />

<!-- Display Employees -->

<asp:GridView ID="gvEmployees" runat="server" AutoGenerateColumns="False" DataKeyNames="EmpID"

OnRowEditing="gvEmployees\_RowEditing" OnRowUpdating="gvEmployees\_RowUpdating"

OnRowCancelingEdit="gvEmployees\_RowCancelingEdit" OnRowDeleting="gvEmployees\_RowDeleting" Height="233px" Width="677px">

<Columns>

<asp:BoundField DataField="EmpID" HeaderText="EmpID" ReadOnly="True" />

<asp:BoundField DataField="Name" HeaderText="Name" />

<asp:BoundField DataField="Department" HeaderText="Department" />

<asp:BoundField DataField="Salary" HeaderText="Salary" />

<asp:CommandField ShowEditButton="True" ShowDeleteButton="True" />

</Columns>

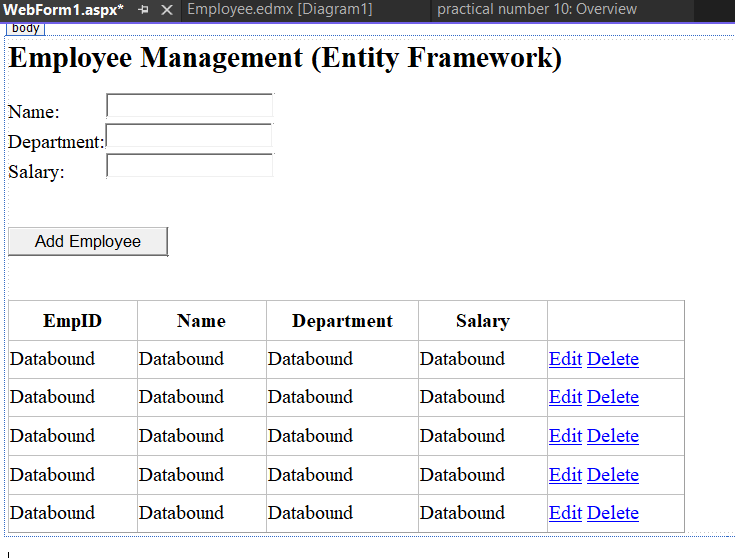
</asp:GridView>

</div>

</form>

</body>

</html>



**6️ Backend Code (**WebForm1**.aspx.cs)**

Modify WebForm1.aspx.cs to implement CRUD operations using **Entity Framework**.

using System;

using System.Linq;

using System.Web.UI;

using System.Web.UI.WebControls;

namespace practical\_number\_10

{

public partial class WebForm1 : System.Web.UI.Page

{

EmployeeDBEntities1 db = new EmployeeDBEntities1 ();

protected void Page\_Load(object sender, EventArgs e)

{

if (!IsPostBack)

{

LoadEmployees();

}

}

private void LoadEmployees()

{

gvEmployees.DataSource = db.Employees.ToList();

gvEmployees.DataBind();

}

// Add Employee

protected void btnAdd\_Click(object sender, EventArgs e)

{

Employee emp = new Employee

{

Name = txtName.Text,

Department = txtDepartment.Text,

Salary = Convert.ToDecimal(txtSalary.Text)

};

db.Employees.Add(emp);

db.SaveChanges();

LoadEmployees();

}

// Edit Employee

protected void gvEmployees\_RowEditing(object sender, GridViewEditEventArgs e)

{

gvEmployees.EditIndex = e.NewEditIndex;

LoadEmployees();

}

// Update Employee

protected void gvEmployees\_RowUpdating(object sender, GridViewUpdateEventArgs e)

{

int empID = Convert.ToInt32(gvEmployees.DataKeys[e.RowIndex].Value);

Employee emp = db.Employees.Find(empID);

TextBox txtName = (TextBox)gvEmployees.Rows[e.RowIndex].Cells[1].Controls[0];

TextBox txtDepartment = (TextBox)gvEmployees.Rows[e.RowIndex].Cells[2].Controls[0];

TextBox txtSalary = (TextBox)gvEmployees.Rows[e.RowIndex].Cells[3].Controls[0];

emp.Name = txtName.Text;

emp.Department = txtDepartment.Text;

emp.Salary = Convert.ToDecimal(txtSalary.Text);

db.SaveChanges();

gvEmployees.EditIndex = -1;

LoadEmployees();

}

// Cancel Edit

protected void gvEmployees\_RowCancelingEdit(object sender, GridViewCancelEditEventArgs e)

{

gvEmployees.EditIndex = -1;

LoadEmployees();

}

// Delete Employee

protected void gvEmployees\_RowDeleting(object sender, GridViewDeleteEventArgs e)

{

int empID = Convert.ToInt32(gvEmployees.DataKeys[e.RowIndex].Value);

Employee emp = db.Employees.Find(empID);

db.Employees.Remove(emp);

db.SaveChanges();

LoadEmployees();

}

}

}

