ASSESSMENT 6 - UNIT 8: ENTITY FRAMEWORK

Core concepts: Entity Framework

OVERVIEW

Implement a SQL table and connect it to an MVC project using entity. This table will be employee table with various info on each employee. You will use this data to determine what benefits employees receive upon retiring.

SETUP

Create a new ASP.NET Web Application (.NET Framework) named **Assessment6Mock**.

BUILD SPECIFICATIONS

The assessment is worth ten points, one for each of the test cases below. Pay special attention to the items in bold. **You must use these in your program exactly, including capitalization**, in order to get the points. Note: You'll be graded on what's in the **HomeController,** but you are allowed to scaffold out another controller to help with populating your table.

Within SSMS, create a database named **EmployeeDb** and a table named **Employee** with the following datatypes:

- **Id** (Primary Key) int, auto increment
- **FirstName**, nvarchar(40)
- **Age**, int
- **Salary** money

Fill in the following into the table:

Id	FirstName	Age	Salary
1	Jimmy	70	90000
2	Sandy	50	45000
3	Allen	25	30000

Now build the app:

1. Using Entity Framework, bring the table into your **Assessment6Mock** project. Call the item **EmployeeDb** when prompted.

Your web application will have the **Employees** view which will display the following:

2. An HTML table which will display each Employee in the database, displaying FirstName, Age, and Salary. This data should be passed in as a list of Employee Models.
    a. Syntax for this:
        i. In ActionResult: return View(listHere)
        ii. In View: @model List<Employees>
3. A form on the bottom of the page that allows the user to select an employee by id.
    a. The form must have a single input named **Id** which will be of type number. The form allows the user to select an employee by Id
    b. The form submits to the ActionResult **RetirementInfo**.
    c. The action **RetirementInfo** will take in an int **Id** as a parameter. Do the following in the action:
        i. Find the employee from the database with the matching Id (Assume you will get valid inputs here)
        ii. If the employee is older than 60 set **ViewBag.CanRetire** to **true**
        iii. Else set **ViewBag.CanRetire** to **false**
        iv. Next set **ViewBag.Benefits** to be **60%** of the Employee's **Salary**
4. In the **RetirementInfo** view display both **ViewBag.CanRetire** and **ViewBag.Benefits**

SUBMISSION

When finished, find your project in your File Explorer. Close Visual Studio. Delete the bin and obj folders in your project, compress the **Assessment6Mock** project as an archive file (ZIP) and upload it below.