# Assignment 2 : Repository Pattern in .NET Web API (DB First Approach)

Topic : Develop a secure ASP.NET Core Web API for managing employees.  
You must:

* Use **Entity Framework (DB First)** to generate models from an existing database.
* Apply the **Repository Pattern** for data operations.
* Implement **JWT Token Authentication** to secure API endpoints.

**Database Structure:**

**Create a database named EmployeeDB in SQL Server with the following tables:**

**Table: Employees**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Constraints** |
| EmployeeId | int | Primary key,Identity |
| FullName | Nvarchar(100) | Not null |
| Department | Nvarchar(50) | Not null |
| Salary | Decimal(10,2) | Not null |
| email | Nvarchar(100) | Unique |

Table: Users

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Constraints** |
| UserId | int | Primary key ,Identity |
| UserName | Nvarchar(50) | Unique |
| Password | Nvarchar(100) | Not null |
| Role | Nvarchar(20) | Eg Admin Or User |

Note:

**Implement Repository Pattern**

**Create Repositories/IEmployeeRepository.cs**

using EmployeeManagementAPI.Models;

public interface IEmployeeRepository

{

Task<IEnumerable<Employee>> GetAllAsync();

Task<Employee?> GetByIdAsync(int id);

Task AddAsync(Employee employee);

Task UpdateAsync(Employee employee);

Task DeleteAsync(int id);

}

**Jwt Token Authentication**

"Jwt": {

"Key": "ThisIsASecretKeyForJWTAuthentication",

"Issuer": "https://localhost:5001",

"Audience": "https://localhost:5001"

}