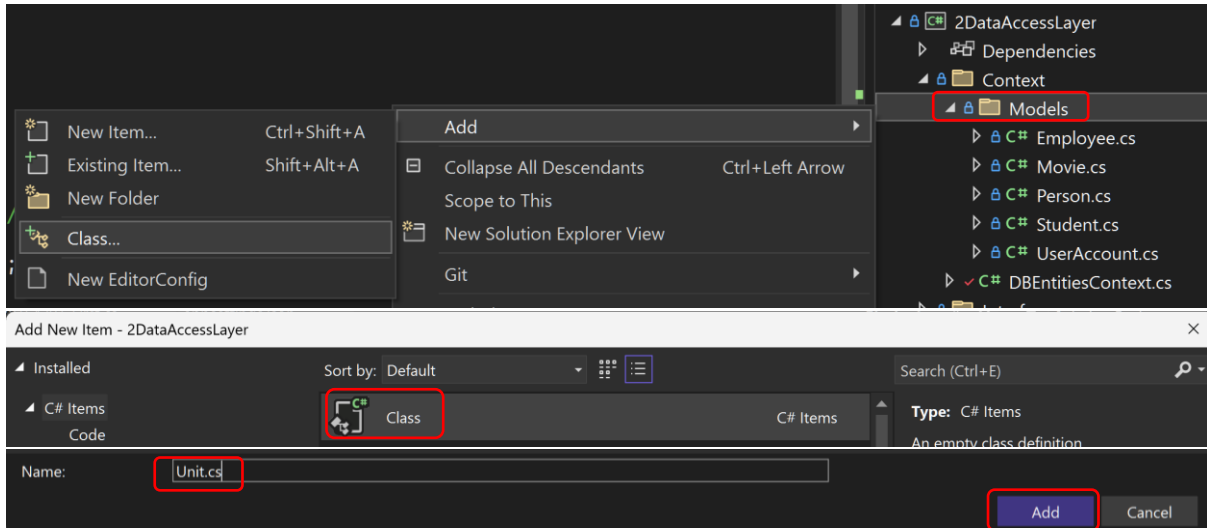


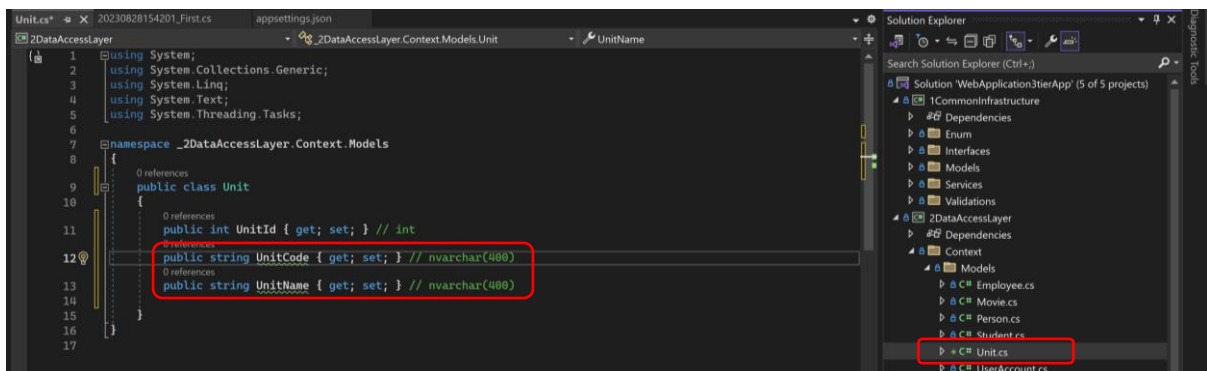
Steps to add end-to-end.

1. Add a new entity class in the Models folder under the Data Access Layer.
 - Right-click on the Models folder.
 - Add > **Class**
 - Change the name to “Unit.cs” and click **Add**.



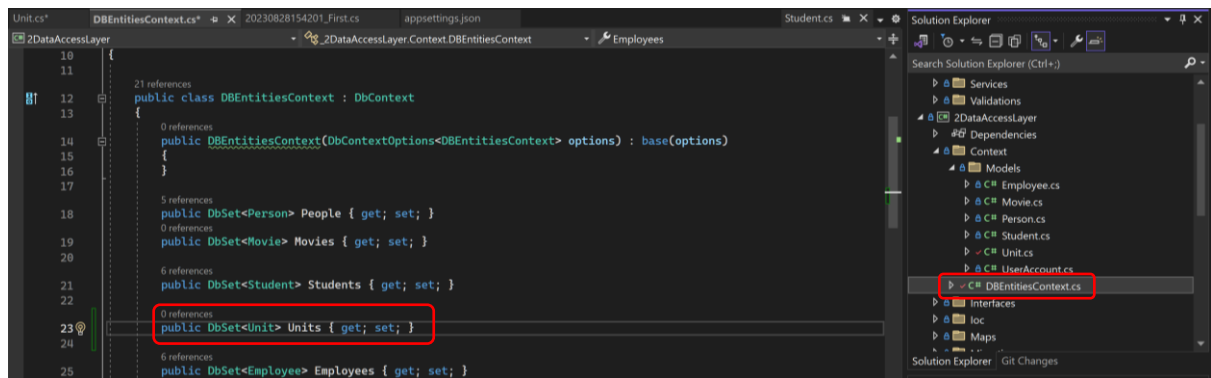
2. Add the following -

```
public class Unit
{
    public int UnitId { get; set; } // int
    public string UnitCode { get; set; } // nvarchar(400)
    public string UnitName { get; set; } // nvarchar(400)
}
```

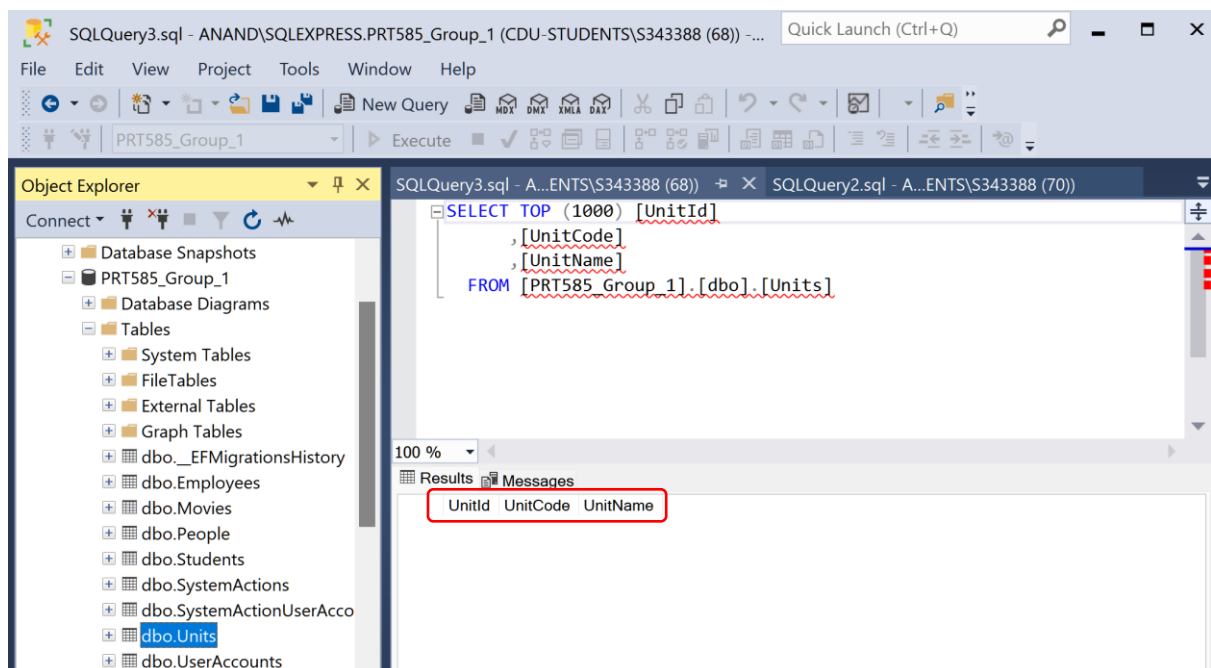
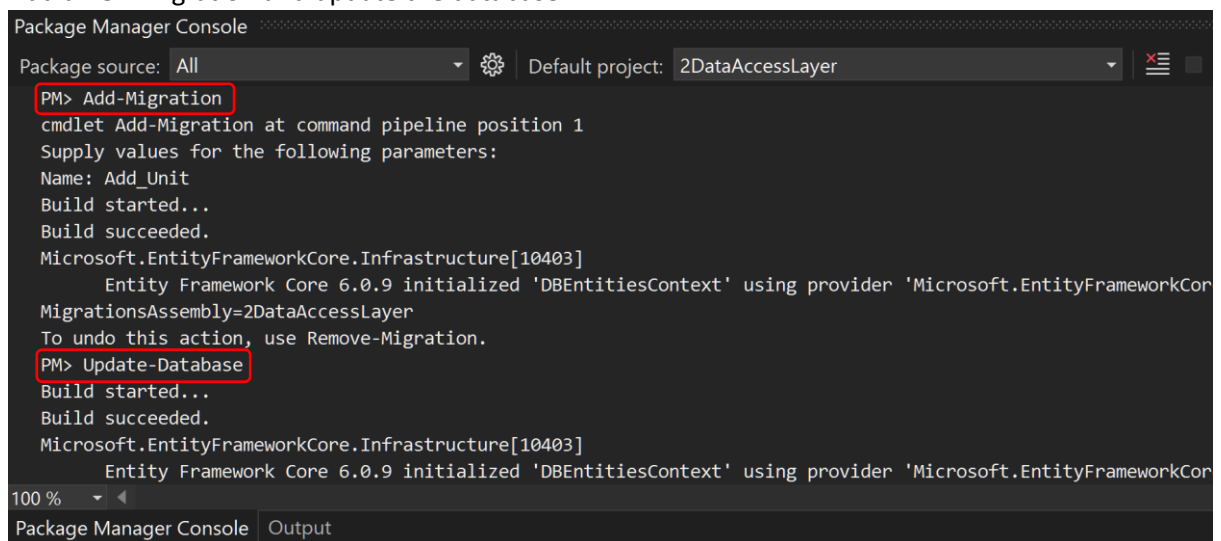


3. Add context.

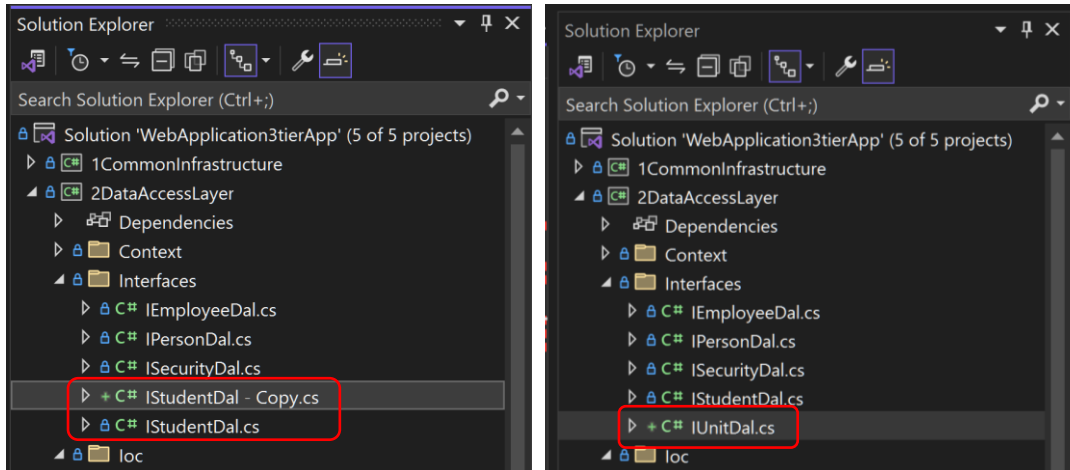
```
public DbSet<Unit> Units { get; set; }
```



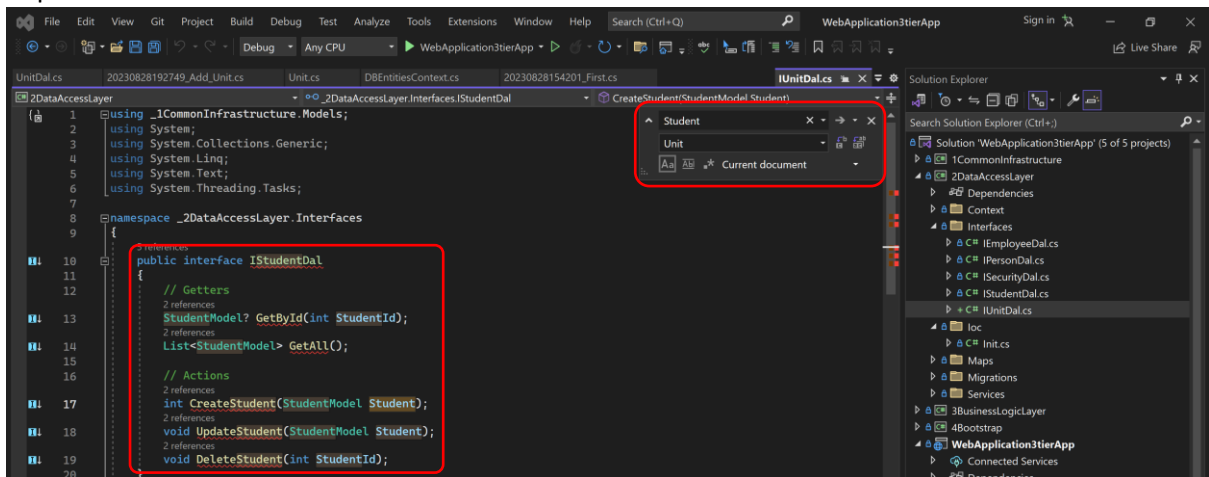
4. Add a new migration and update the database.



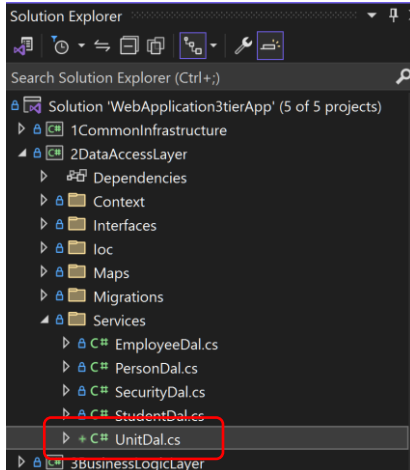
5. Create a copy of “IStudentDal.cs” in the Interfaces folder under DAL and rename it “IUnitDal.cs”.



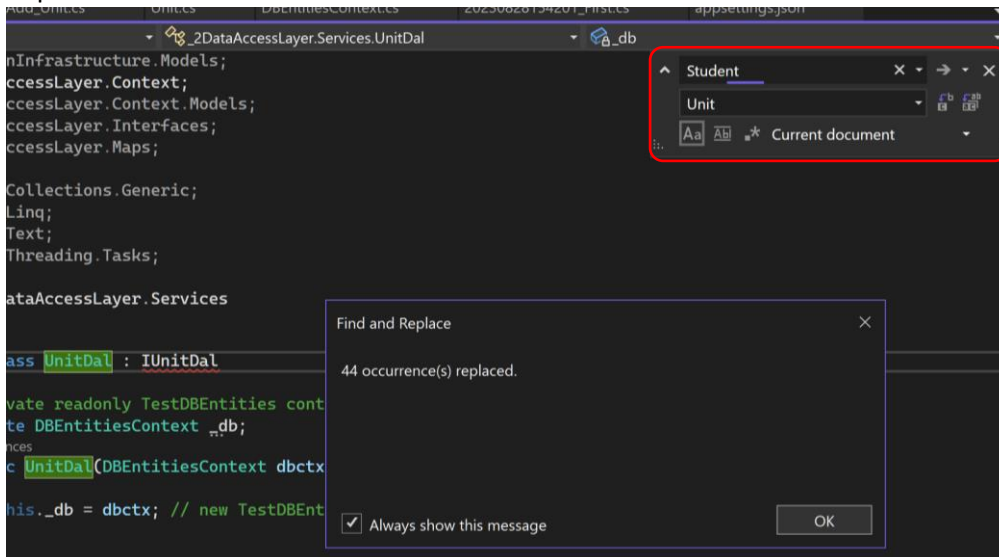
Replace all the words “Student” with “Unit”.



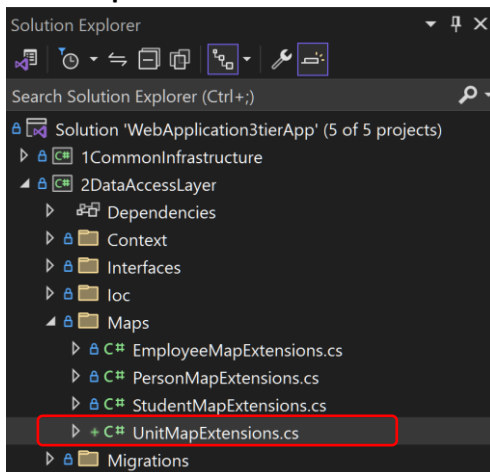
6. Create a copy of “**StudentDal.cs**” in the Services folder under DAL and rename it “**UnitDal.cs**”.



Replace all the words “**Student**” with “**Unit**”.



7. Create a copy of “**StudentMapExtensions.cs**” in the Maps folder under DAL and rename it “**UnitMapExtensions.cs**”.



Replace all the words “**Student**” with “**Unit**”. And add the following -
`dst.UnitCode = src.UnitCode;`

```
2 references
public static UnitModel ToUnitModel(this Unit src)
{
    var dst = new UnitModel();

    dst.UnitId = src.UnitId;
    dst.UnitCode = src.UnitCode;
    dst.UnitName = src.UnitName;

    return dst;
}

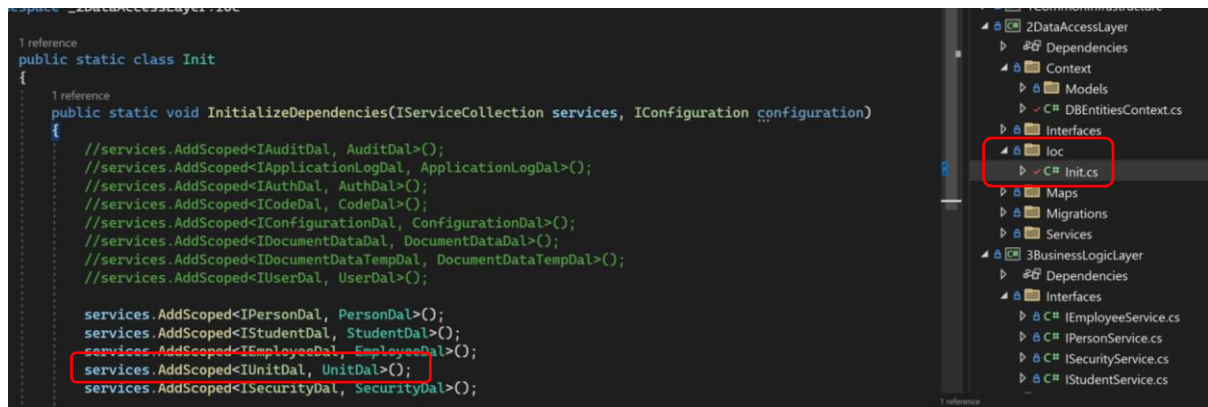
2 references
public static Unit ToUnit(this UnitModel src, Unit dst = null)
{
    if (dst == null)
    {
        dst = new Unit();
    }

    dst.UnitId = src.UnitId;
    dst.UnitCode = src.UnitCode;
    dst.UnitName = src.UnitName;

    return dst;
}
```

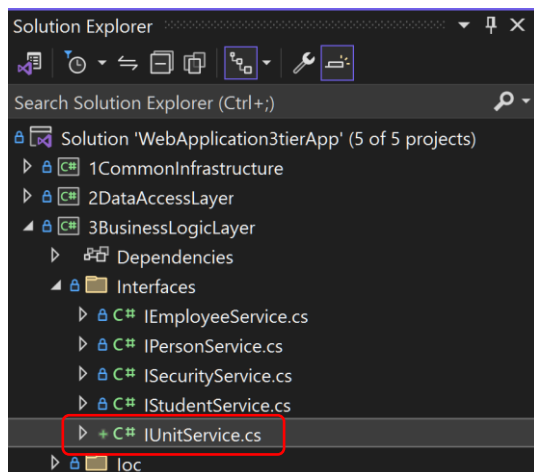
8. Open “Init.cs” from the loc folder under DAL. Add the following -

```
services.AddScoped<IUnitDal, UnitDal>();
```

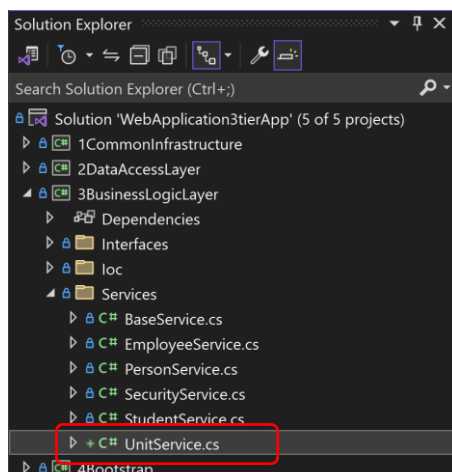


9. Repeat the steps in the Business Logic Layer (BLL).

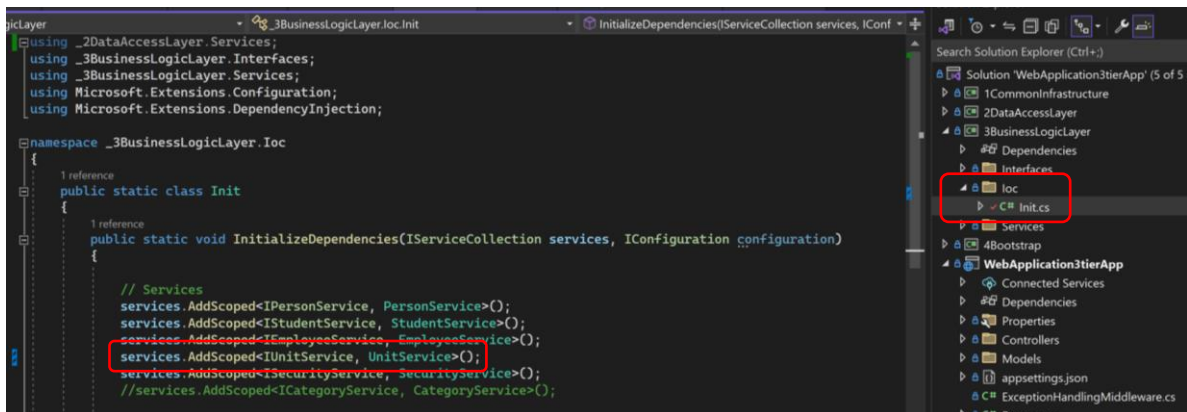
- Create a copy of “IStudentService.cs” in the Interfaces folder under BLL and rename it “IUnitService.cs”.
- Replace all the words “Student” with “Unit”.



- Create a copy of “StudentService.cs” in the Services folder under BLL and rename it “UnitService.cs”.
- Replace all the words “Student” with “Unit”.

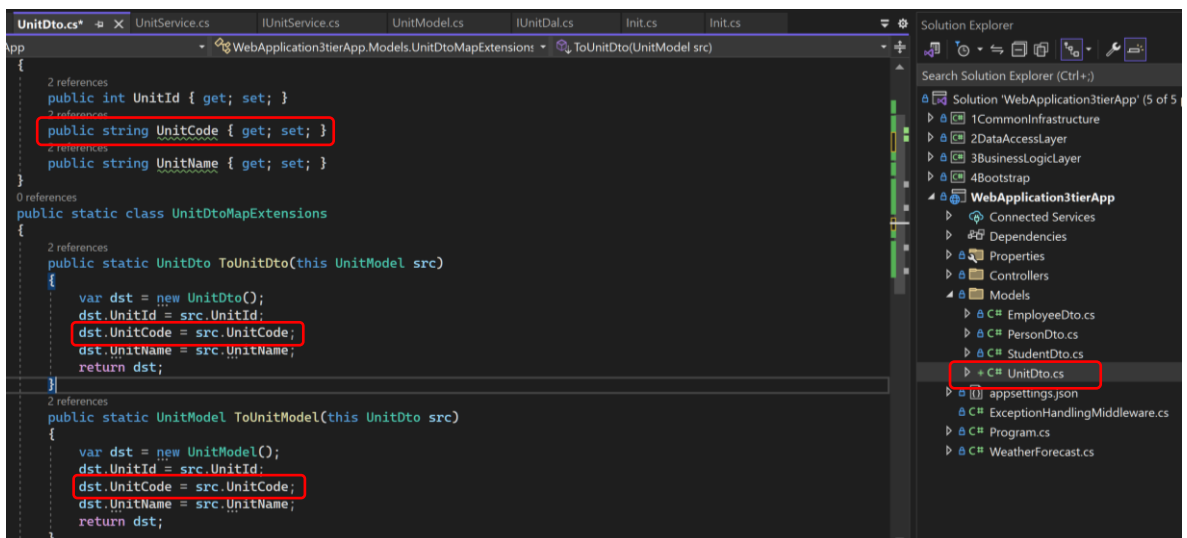


- Open “Init.cs” from the loc folder under BLL. Add the following -
`services.AddScoped<IUnitService, UnitService>();`



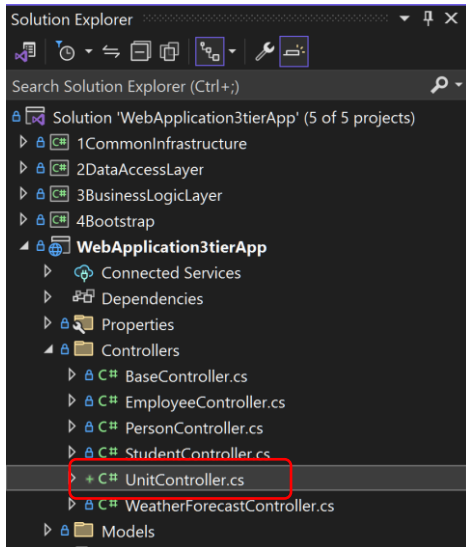
10. Create a copy of “StudentDto.cs” in the Models folder under WebApplication3tierApp and rename it “UnitDto.cs”.

- Replace all the words “Student” with “Unit”.
- Add the following in `public class UnitDto`
`public string UnitCode { get; set; }`
- Add the following in `public static class UnitDtoMapExtensions`
`dst.UnitCode = src.UnitCode;`

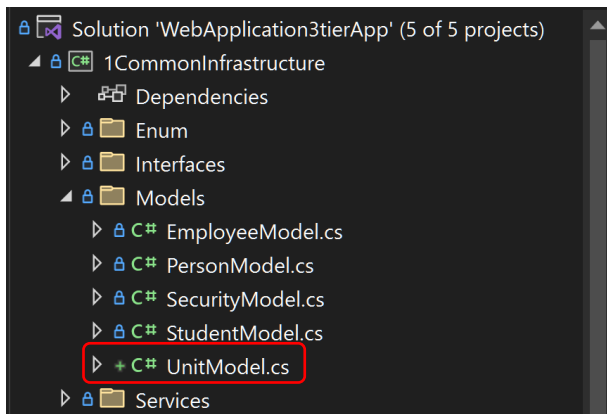


11. Create a copy of “**StudentController.cs**” in the Controllers folder under WebApplication3tierApp and rename it “**UnitController.cs**”.

- Replace all the words “**Student**” with “**Unit**”.

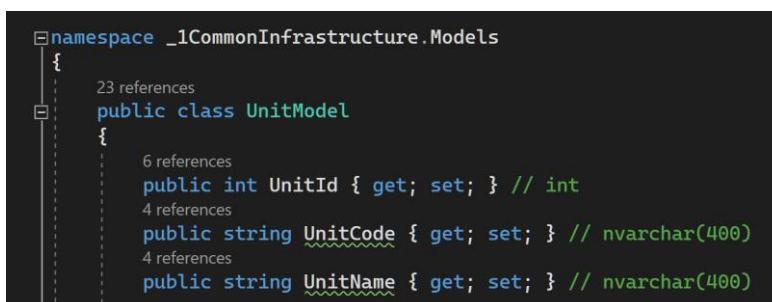


12. Create a copy of “**StudentModel.cs**” in the Models folder under the Common Infrastructure Layer and rename it “**UnitModel.cs**”.

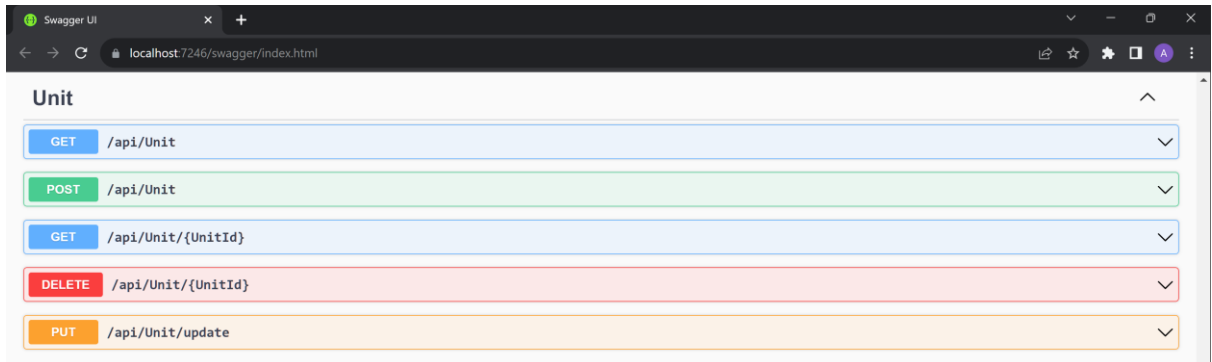
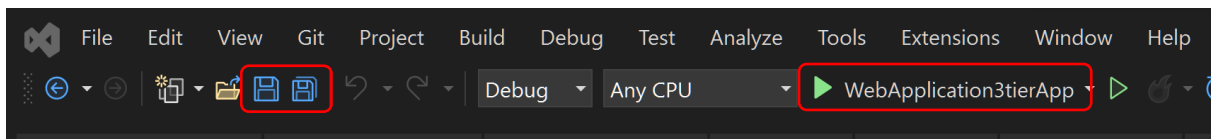


Add the following –

```
public class UnitModel
{
    public int UnitId { get; set; } // int
    public string UnitCode { get; set; } // nvarchar(400)
    public string UnitName { get; set; } // nvarchar(400)
}
```

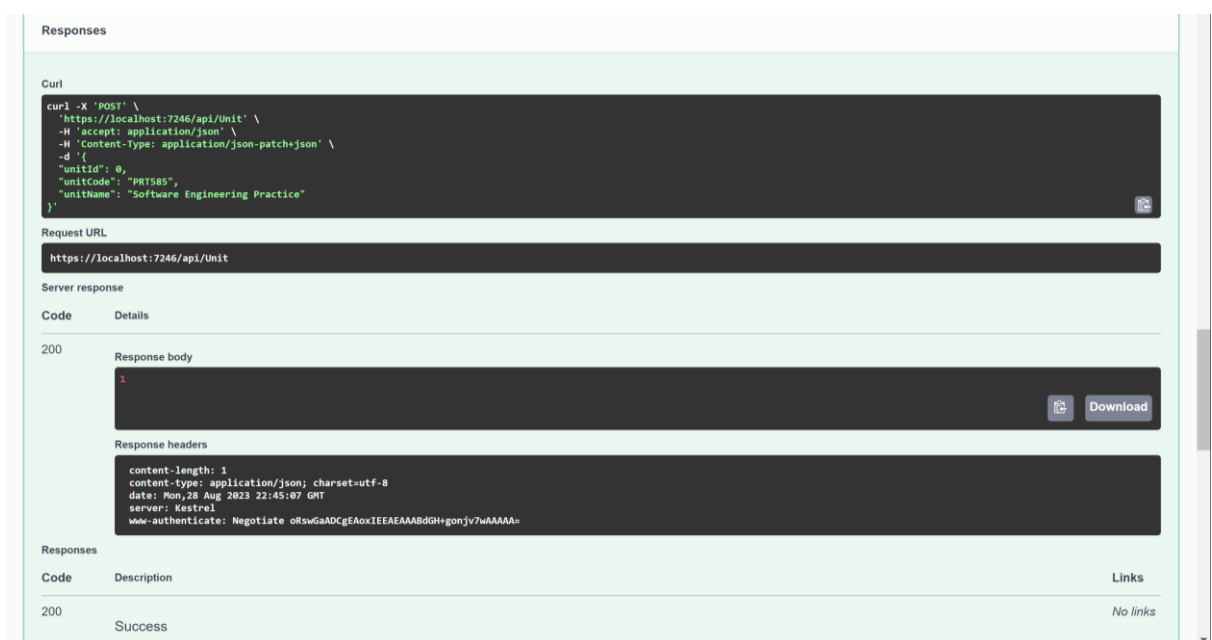
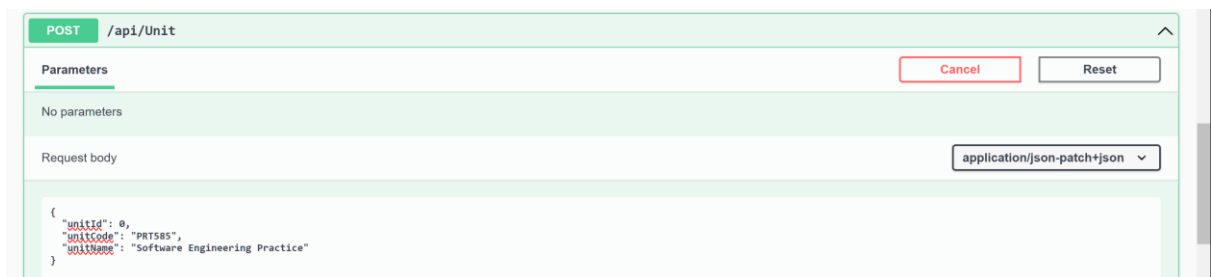


13. Save the project and run the app.



14. Make a request.

- Expand the **POST Student** endpoint.
- Click **Try it out**.
- Add unit code and unit name.
- Execute.



15. Check the database to see the new unit.

The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure for 'PRT585_Group_1', including tables like 'dbo.Units'. The central query window shows the following SQL query:

```
SELECT TOP (1000) [UnitId]  
                , [UnitCode]  
                , [UnitName]  
FROM [PRT585_Group_1].[dbo].[Units]
```

The Results pane at the bottom displays the query output in a table format. The first row is highlighted with a red box:

UnitId	UnitCode	UnitName
1	PRT585	Software Engineering Practice

The status bar at the bottom indicates the connection to 'ANAND\SQLEXPRESS (16.0 RTM) CDU-STUDENTS\S343388 (71) PRT585_Group_1' and shows '1 rows'.

16. **Next step** – Use Angular on the client side to perform CRUD operations.

<https://www.youtube.com/watch?v=CdE6rVfPJ9I&t=130s>