Dotnet Centric Computing: Lab4 BCSIT 6th Semester

Lab rules:

1. Submit your lab reports on your own, no copying is allowed. You can collaborate while working with the project for conceptual clarifications, but the report should be written individually. Write your own codes and descriptions.
2. Create a solution called Lab3 within your Labs folder (that was used for Lab1 and Lab2 before) and there we will be creating various ASP.Net core apps.
3. Write all the steps followed while accomplishing the project in your lab report. You can copy paste the screenshots of folder structures and the output.

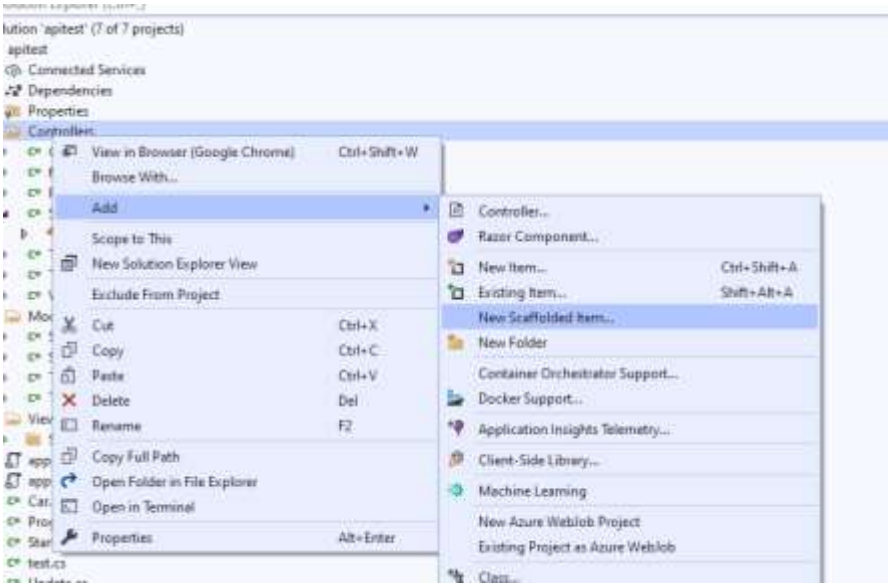
Lab3: ASP.NET Core Web app with MVC and API

1. Within Lab4 solution, add a new MVC Web project named MVCEmployeeCRUD and implement the employ and employeecontext models as well as CRUD operations. You can use the two model files (Employee and EmployeeContext) and the same database setup as done in the Lab3 razor page CRUD assignment. You can auto-generate the CRUD operations by adding a new controller with EntityFramework. Explain in brief all the action methods and corresponding views of the Employee controller. Also compare this with raxzor page models (Lab3 CRUD).

Employee Model Properties:

EmpId (PK), EmpName(40chars), Address(50chars), and age(int)

2. Now, add a new WebAPI project and setup the database and table for Employee as done in q1. Use the same model to add a new API controller with CRUD operations. You can auto-generate the CRUD operations for the Employee table API by Adding a new Scafolded Item and clicking on the Action with EntityFramework option (see the fig. below). Also see the StudentController.cs file for example in the WebAPI-test repository in the github.



Briefly comment on the WebAPI functions and show the results of this App.