Workout Tracker

Instructions.

Database Objects Creation:

Prerequisites: Microsoft SQL Server Management Studio 2008 and above installation and Windows 7 and above machine.

Step 1: Unzip or extract Backend.zip file to local machine.

Step 2: Open installed version Microsoft SQL Server Management Studio then open 1\_DataBaseCreate SQL file from extracted location and execute it.

Step 3: Verify WorkoutTracker database created successfully.

Step 4: Open 2\_DDL SQL file from extracted location and execute it.

Step 5: Verify user, database, workout\_category, workout\_collection, workout\_active tables are created in WorkoutTracker database successfully.

Step 6: Open 3\_DML SQL file from extracted location and execute it.

Step 7: Verify data is inserted in user table successfully.

Configurations to run front-end application:

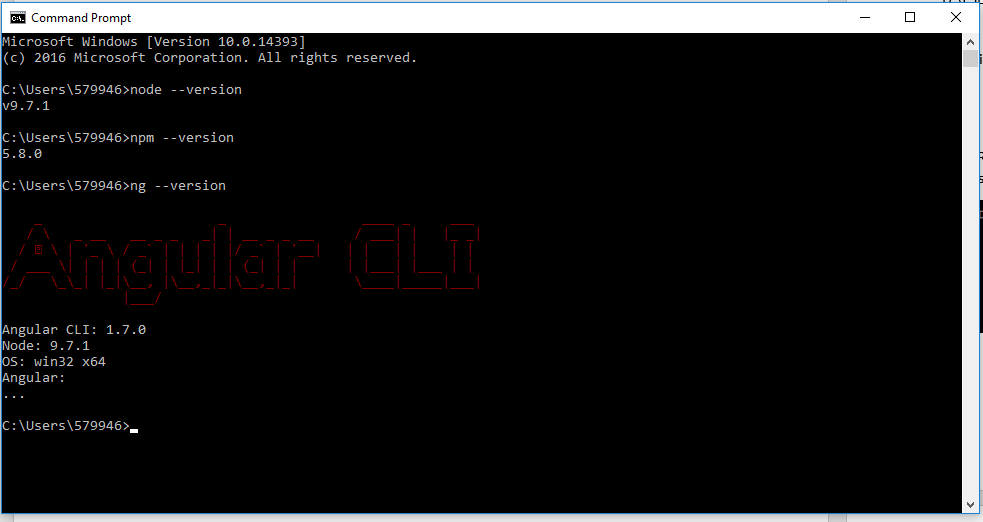
Prerequisites: Microsoft Visual Studio 2017 and above installation and Windows 7 and above machine, Node.js version 9.7.1, Node Package Manager(npm) version 5.8.0, Angular CLI version 1.7.0, Typescript version 2.5.3 installations.

Step 1: Unzip or extract Frontend.zip file to local machine.

Step 2: Find WorkoutTracker.sln solution from extracted location and open it in visual studio.

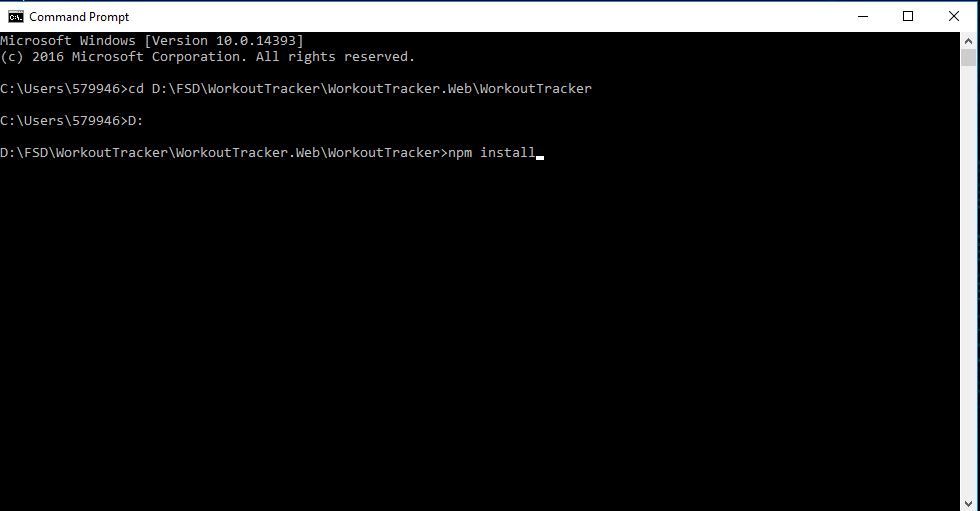
Step 3: Clean and re-build the solution and verify solution gets build without error.

Step 4: Open Command prompt and command node --version, npm --version, ng --version to verify Node.js, NPM, Angular CLI installations.



Step 5: Open Command prompt and navigate to solution project path WorkoutTracker. Web\WorkoutTracker\.

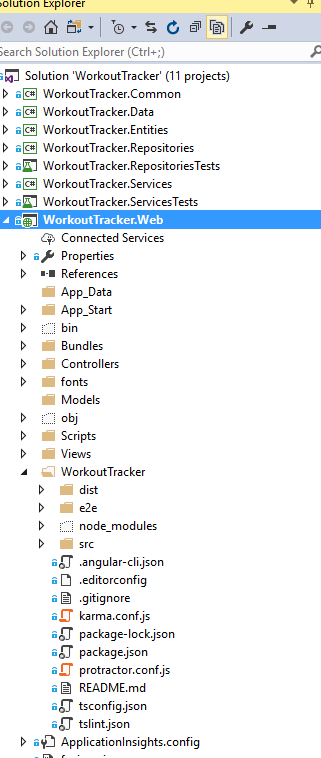
Example path to navigate D:\FSD\WorkoutTracker\WorkoutTracker.Web\WorkoutTracker\



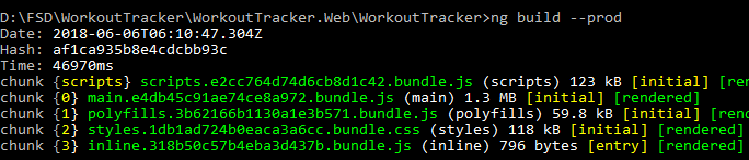
Step 6: Verify command prompt current directory is D:\FSD\WorkoutTracker\WorkoutTracker.Web\WorkoutTracker\.

Step 7: Then input the command npm install, it will install all node modules required and wait till installation gets complete.

Step 8: Verify all node modules created under node\_modules in above mentioned path.



Step 9: Then provide ng build --ec or ng build --prod command in command prompt it will build the application then Rebuild the entire solution again and verify solution gets build without error. Example successful build screenshot.



Step 10: Open App.config file in any editor for the below mentioned projects

1. WorkoutTracker.RepositoriesTests.
2. WorkoutTracker.ServicesTests.
3. WorkoutTracker.WebApiTests.

And find “WorkoutTrackerEntities” configuration in <connectionStrings></connectionStrings> part and replace below highlighted part

Example connection string.

<add name="WorkoutTrackerEntities" connectionString="metadata=res://\*/WorkoutTrackerModel.csdl|res://\*/WorkoutTrackerModel.ssdl|res://\*/WorkoutTrackerModel.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source={PlaceServerName};initial catalog=WorkoutTracker;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

With current SQL Server Management Studio name where WorkoutTracker database created in each projects one by one.

Step 11: Open web.config file in any editor for the below project

1. WorkoutTracker.WebApi.

And find “WorkoutTrackerEntities” configuration in <connectionStrings></connectionStrings> part and replace below highlighted part

Example connection string.

<add name="WorkoutTrackerEntities" connectionString="metadata=res://\*/WorkoutTrackerModel.csdl|res://\*/WorkoutTrackerModel.ssdl|res://\*/WorkoutTrackerModel.msl;provider=System.Data.SqlClient;provider connection string=&quot;data source={PlaceServerName};initial catalog=WorkoutTracker;integrated security=True;MultipleActiveResultSets=True;App=EntityFramework&quot;" providerName="System.Data.EntityClient" />

With current SQL Server Management Studio name where WorkoutTracker database created.

Step 12: Open web.config file in any editor for the below project.

1. WorkoutTracker.WebApi.

<appSettings>

<add key="ErrorLogFileName" value="WorkoutTrackerLog.txt" />

<add key="ErrorLogFilePath" value="D:\FSD\WorkoutTracker\WorkoutTrackerLogs\" />

<add key="Audience" value="http://localhost:55804/" />

<add key="Issuer" value="http://localhost:50499/" />

<add key="JWTSecurityKey" value="401b09eab3c013d4ca54922bb802bec8fd5318192b0a75f201d8b3727429090fb337591abd3e44453b954555b7a0812e1081c39b740293f765eae731f5a65ed1" />

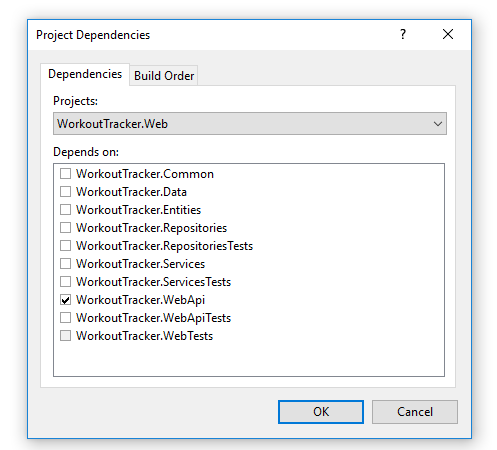
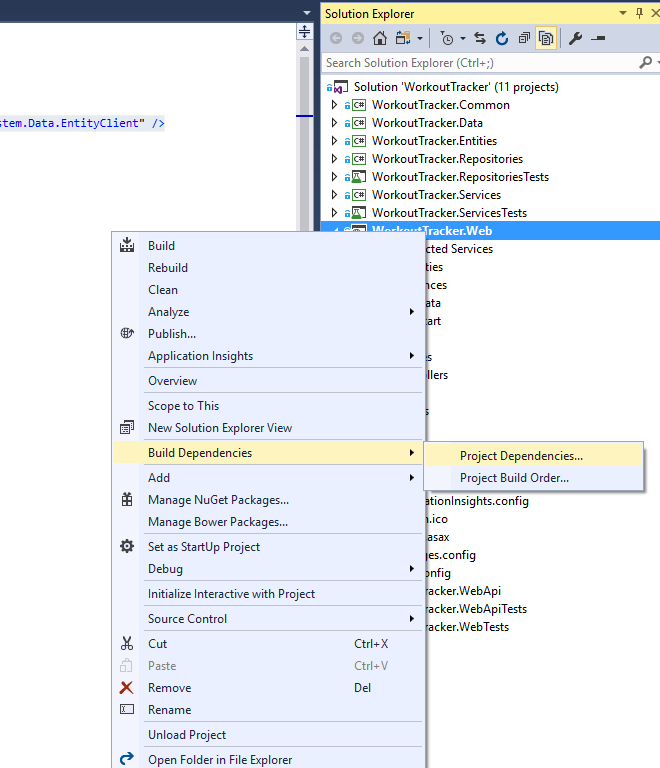
<add key="TokenExpiryInMinutes" value="30" />

</appSettings>

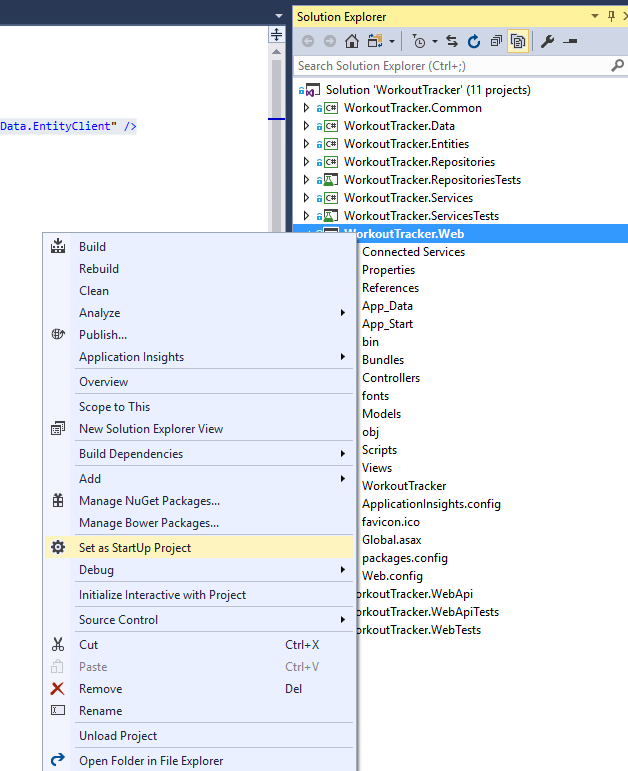
And change the value of highlighted parts according to the solution.

Note: Audience denotes WorkoutTracker.Web project url, and Issuer denotes WorkoutTracker.WebApi project url which will get it from project properties. And also change the ErrorLogFilePath according to the solution path.

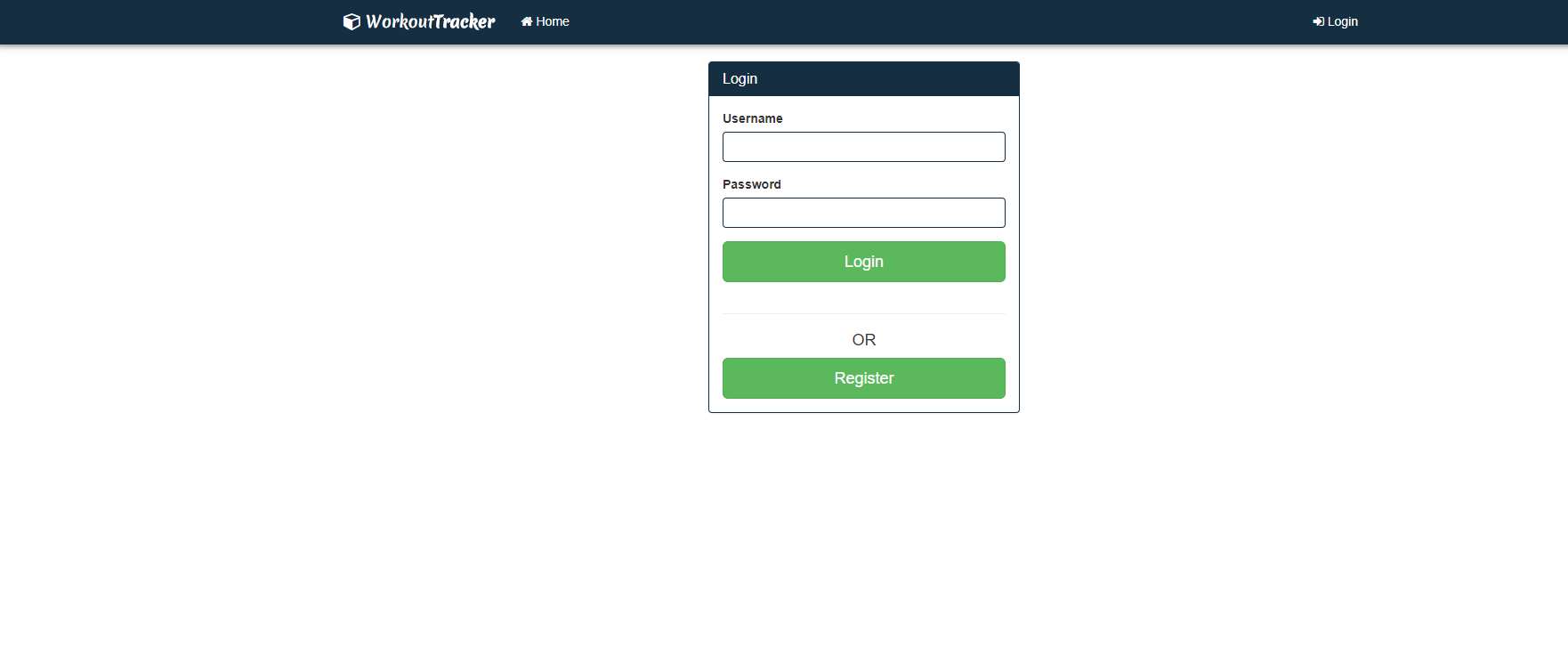
Step 13: Then Right click on WorkoutTracker.Web project then select Build Dependencies -> Project Dependencies it will show list of projects in the solution, then check WorkoutTracker.WebApi project and give ok.



Step 14: Then Right click on WorkoutTracker.Web select Set as Startup project.



Step 15: Then click F5 to run the application and application opens in browser and default page is user login page .



Step 16: Demo user credentials to login

User name: demouser1

Password: demopassword

Or user can also register themselves by navigate Register page by clicking Register button.