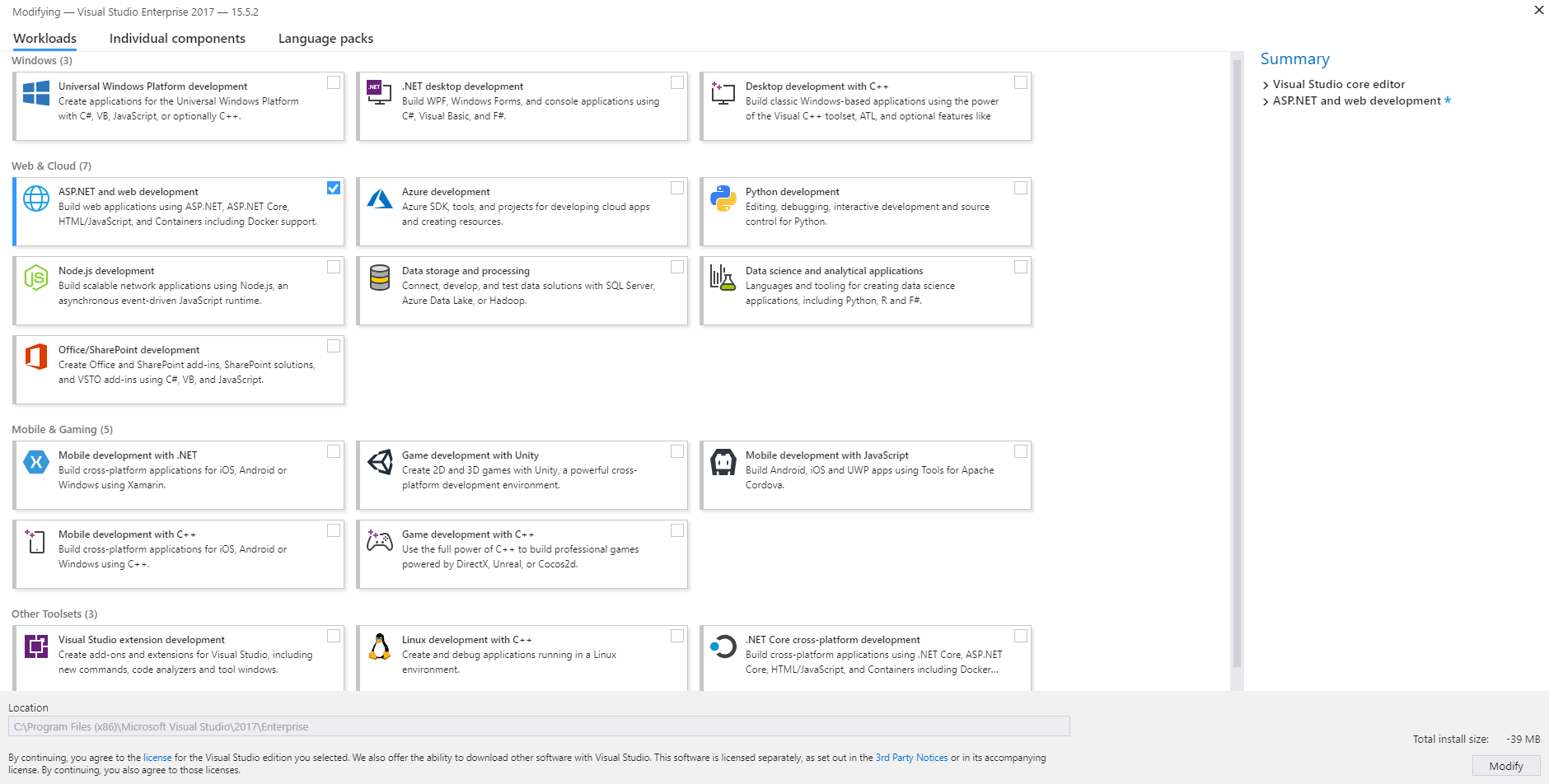
Instruction to setup Coral Time.

1. Install next soft:

1.1. **Visual Studio 2017** <https://www.visualstudio.com/downloads/>

Choose Next Options:



1.2. **NodeJS** (LTS Version) <https://nodejs.org/en/> (example: C:\Program Files\nodejs)

2. Download Repository by next Url: (Put link from GitHub after upload Project) <https://coralteq.visualstudio.com/DefaultCollection/_git/CoralTime>

3. Additional Setup:

3.1. Setup NodeJS:

In VS: Tools -> Options -> Projects and Solutions -> Web Package Management -> External Web Tools

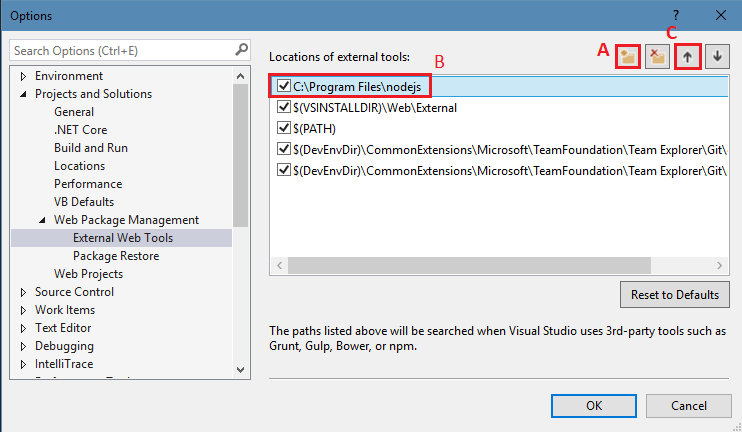
3.1.1 Put path where you install NodeJS (example: C:\Program Files\nodejs)

3.1.2. Set this path to the first place in this list:

A. Create New Path.

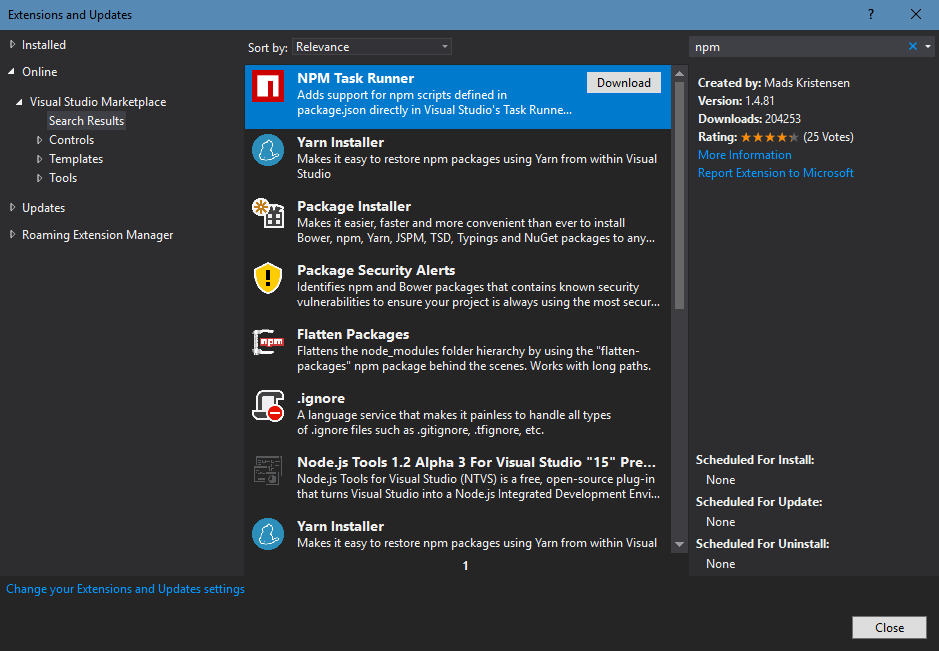
B. Fill path where you install NodeJS (example: C:\Program Files\nodejs)

C. Put up your row with NodeJS path at the first position by means of an arrow up



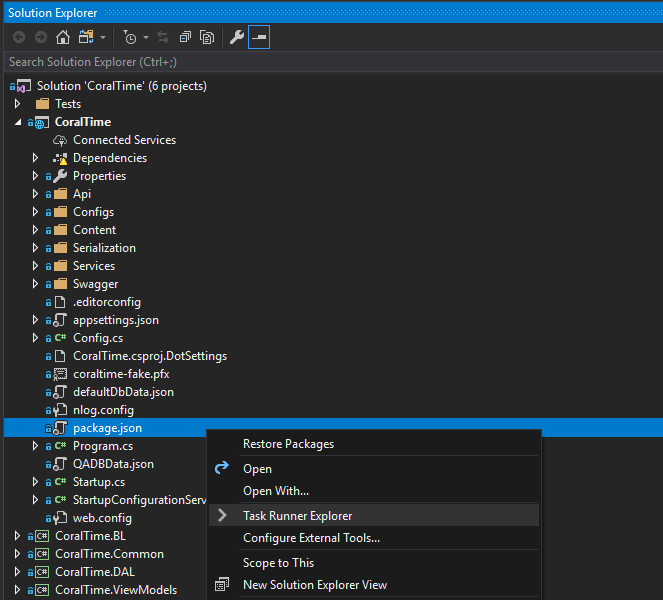
3.2. Install **NPM Task Runner** <https://marketplace.visualstudio.com/items?itemName=MadsKristensen.NPMTaskRunner>

In VS: Tools -> Extensions and Updates.



3.2.1. Open Projectin VSby next path: \backend\**CoralTime.sln**

3.2.2. Choose in Solution Explorer **package.json** by right mouse click and choose **Task Runner Explorer**.



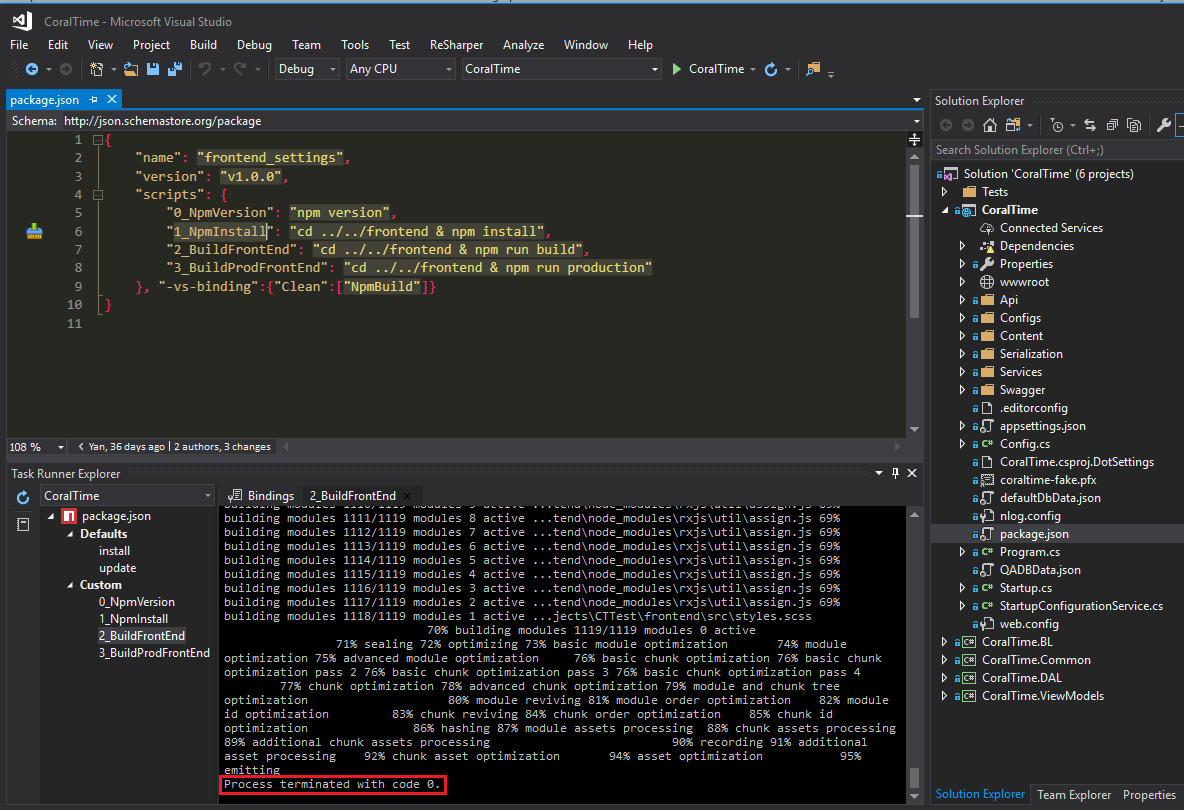
3.3. Install NPM into local frontend folder and build Frontend part (put all frontend files into backend\wwwroot folder)

In opened **Task Runner Explorer** window execute scripts in the following order:

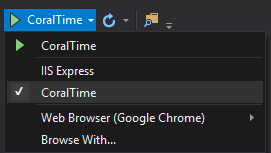
3.3.1. Install NPM into local frontend folder**: 1\_NpmInstall**

3.3.2. Build Frontend part: **2\_BuildFrontEnd**

\* If all the scripts have successfully completed, you will see the next message **«Process terminated with code 0.»**



3.4. In VS choose **CoralTime** as type of start of the project (instead default IIS Express)

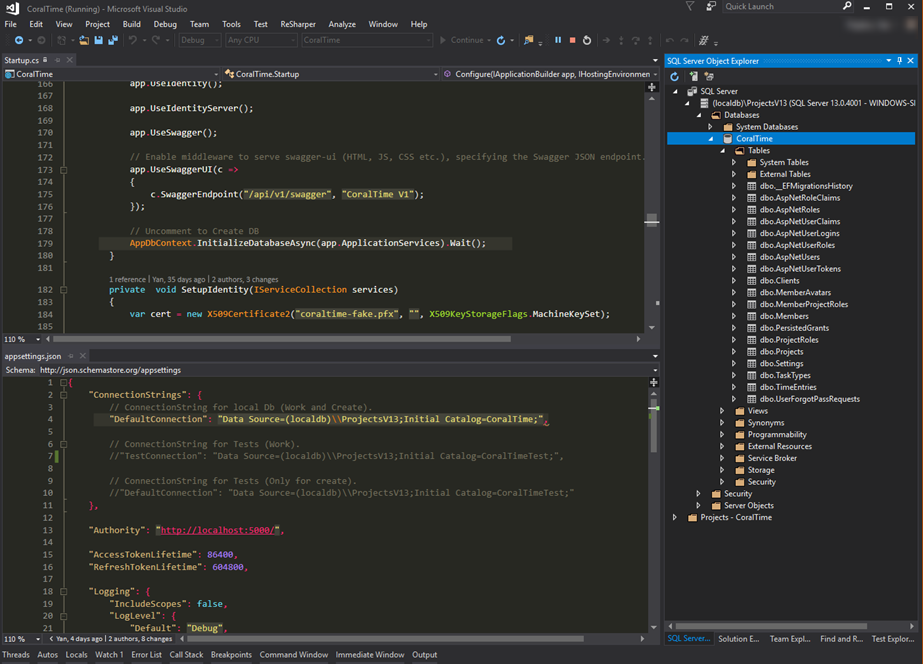


4. Create DBs.

4.1.1. Create **CoralTime** DB.

Uncomment string for create DB in **Startup.cs** and check **DefaultConnection** in **appsettings.json**.

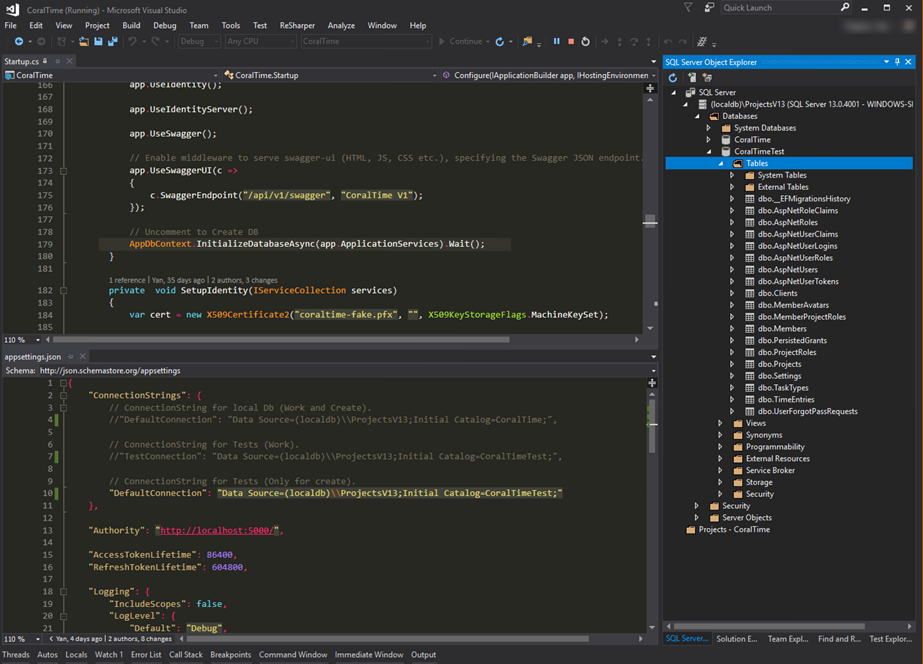
Run solution and wait for creation of the database **CoralTime** and check it!



4.1.2. Create **CoralTimeTest** DB.

Change **DefaultConnection** in **appsettings.json** and solution.

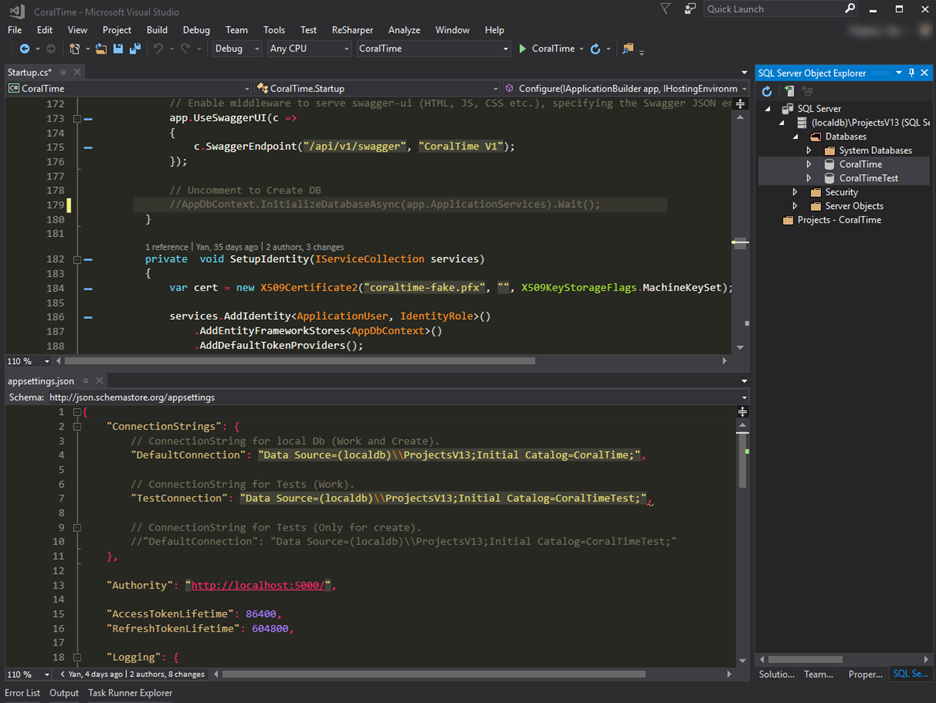
Restart solution and wait for creation of the database **CoralTimeTest** and check it!



4.1.3. Set work settings.

Stop Solution.

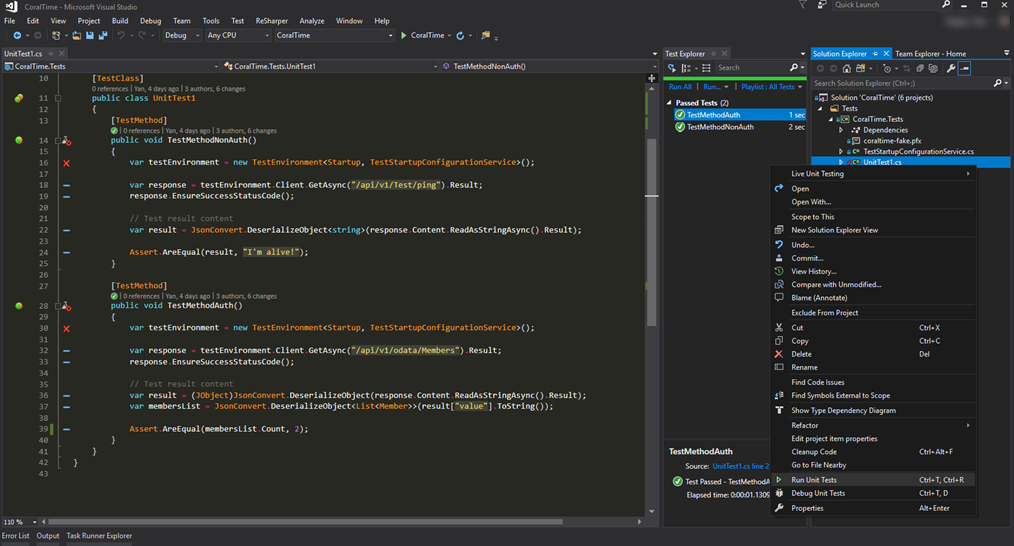
Comment string for create DB in **Startup.cs** and set **DefaultConnection** and **TestConnection** in **appsettings.json**.



4.2. Tests.

Before fully running Solution, you should check the **CoralTime.Tests.**

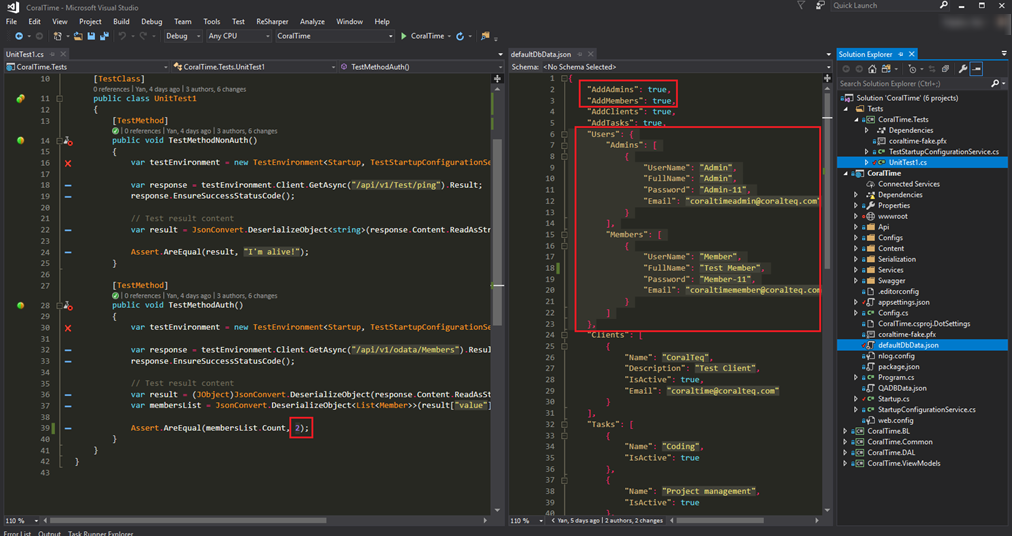
Try to run Tests by **TextExplorer** and check their results. Test results must be successfully.

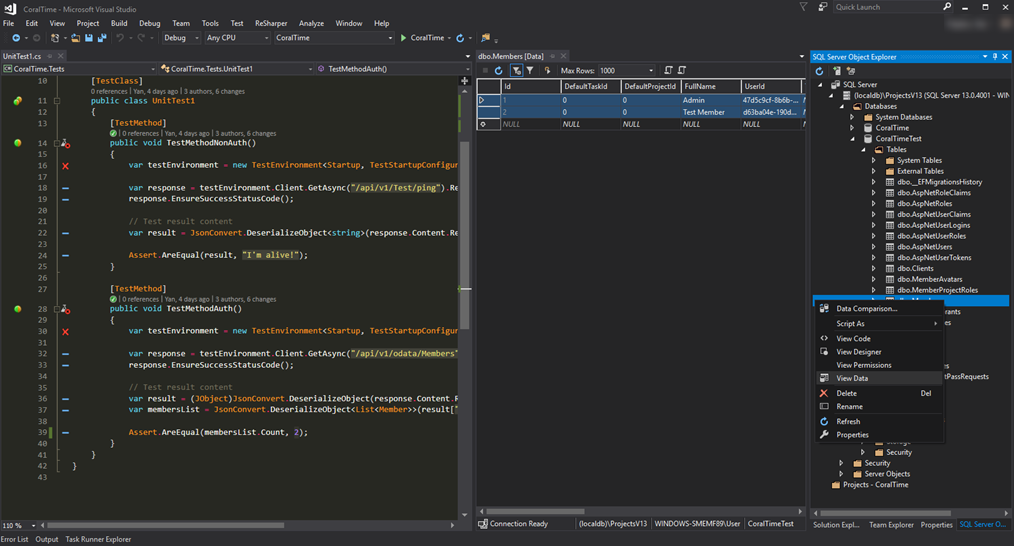


About Tests:

1. «TestMethodNonAuth» try to check ping to your **CoralTimeTests** Db.

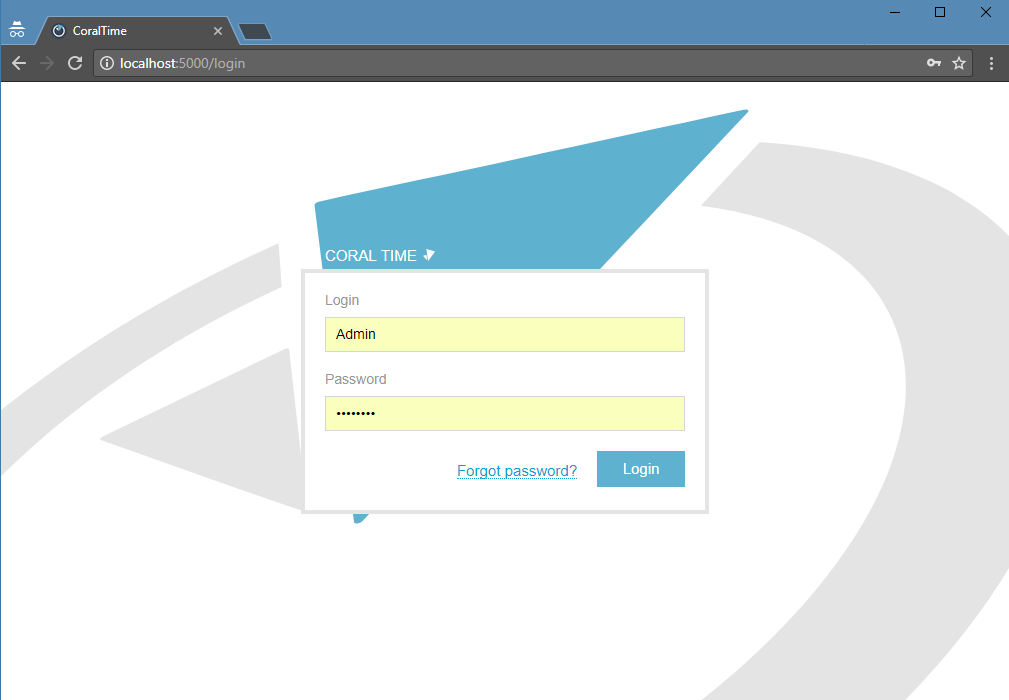
2. «TestMethodAuth» try to compare count of Members in the **CoralTimeTests** Db in table **Members** and your test custom value (2).

\*If you create database by next settings from **defaultDBData.json**:

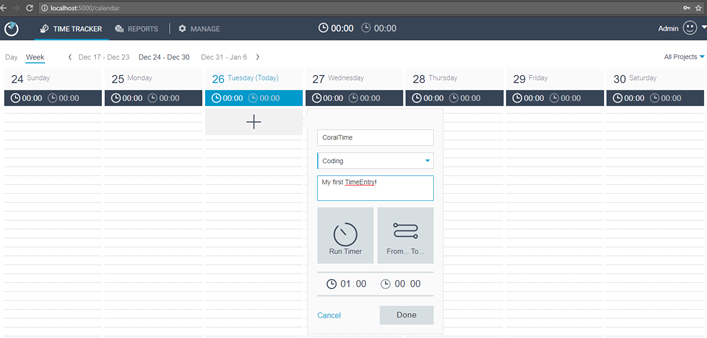
That you have next values in your **CoralTime.Tests** Db in **Members** table:

4.3. Run Solution.

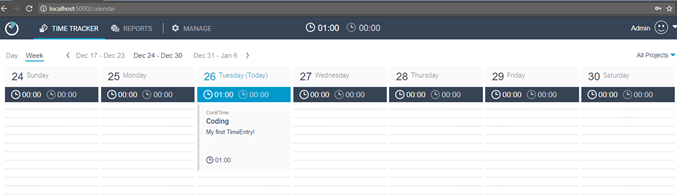
4.3.1. Runsolution and try to login using Login/Password for your User from **defaultDBData.json**:



4.3.2. If your login is successful, you can track your first TimeEntry!



4.3.3. After successful track, you can look:



5. That’s all!