When unit testing using the Regression Model, a unit test would be all layers until hitting the db, then it becomes an integration test. Using the Regression Model, all layers (Model, View,…) would be a single ‘unit of work’

Each layer could be mocked.

Load testing does not test the code, but the infrastructure supporting the code to determine if the infrastructure can support the code.

Testing not included on the slide:

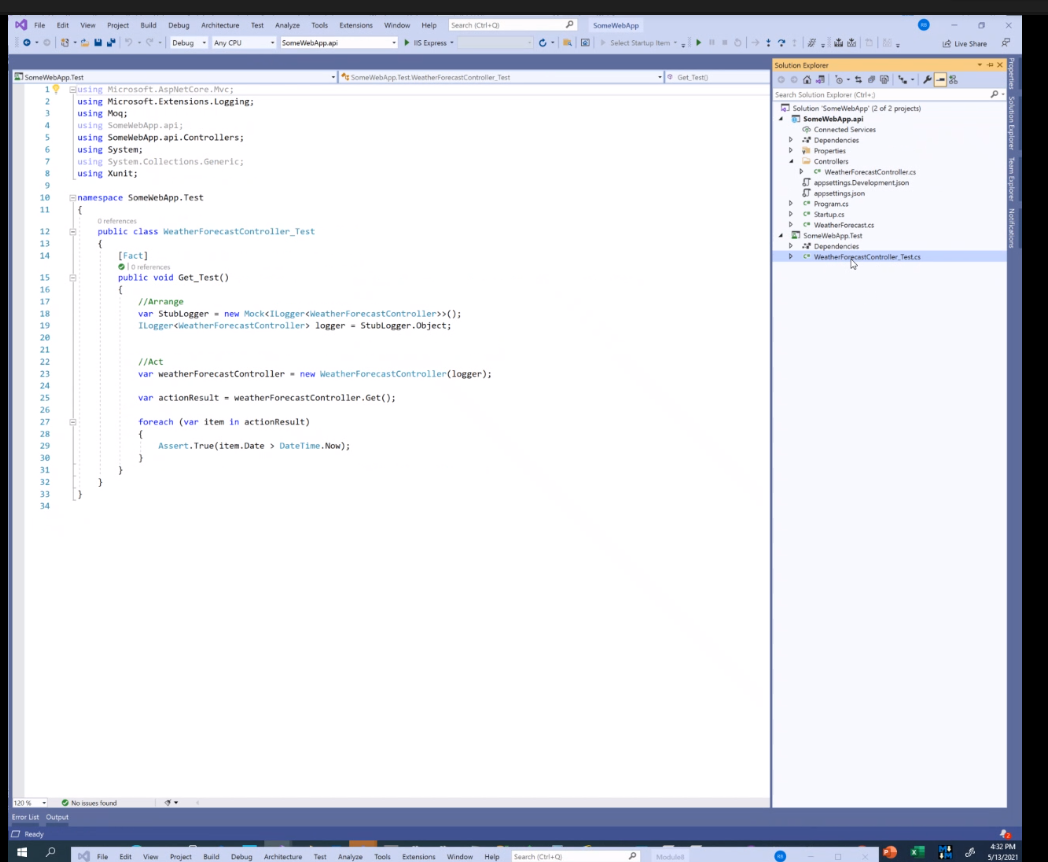
* Regression, Manual and User Acceptance Testing

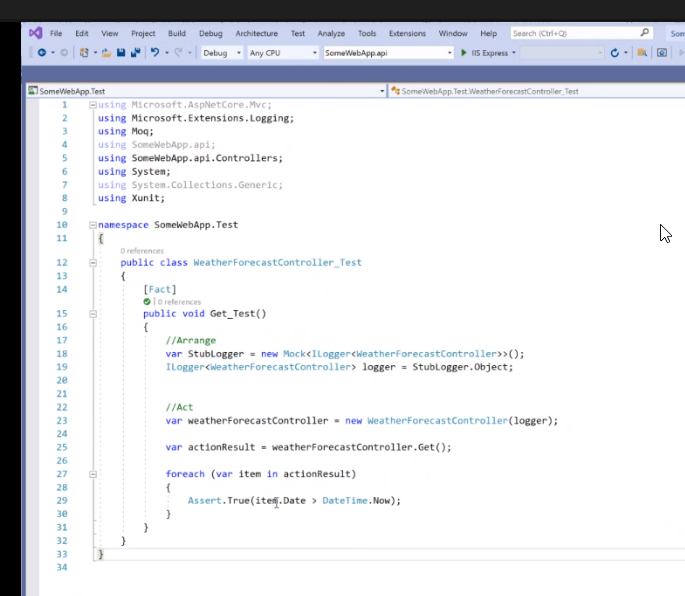
Unit Testing for .Net Core/ .Net 5 is XUnit

Unit test is the statement of the problem domain, while the code is the solution to the problem

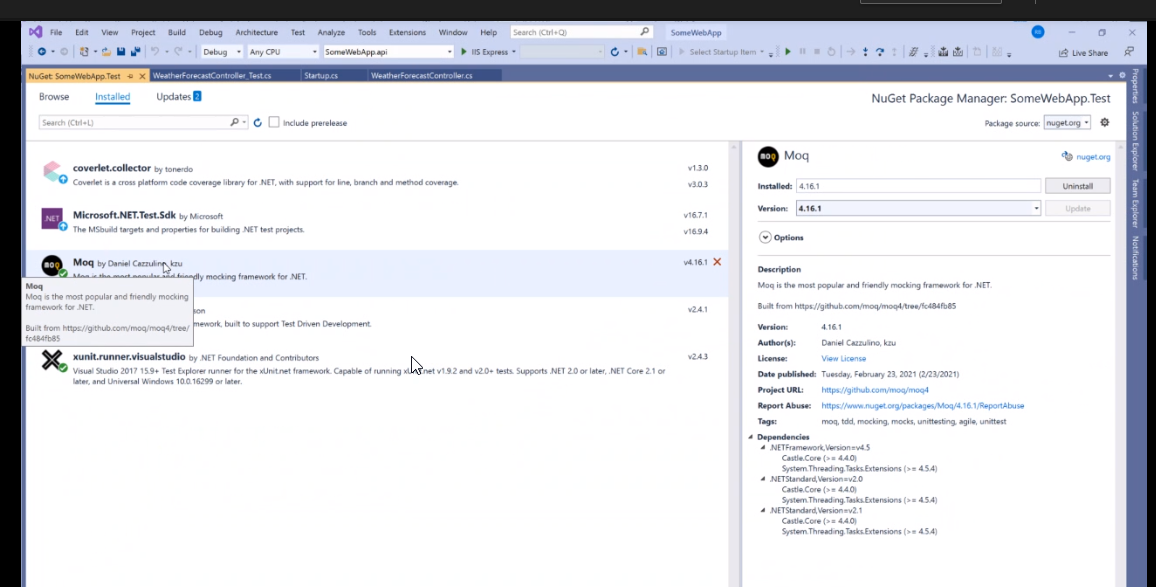
How to do TDD in a Web Application

1. Create a new solution
2. First create the API
3. Then create the controller
4. Add the test solution (like xunit or nunit)
5. Add the test class, which has the same name as the controller
   1. Name the test class the same as the controller
   2. Ex: WeatherForecastController 🡪 WeatherForecastController\_Test

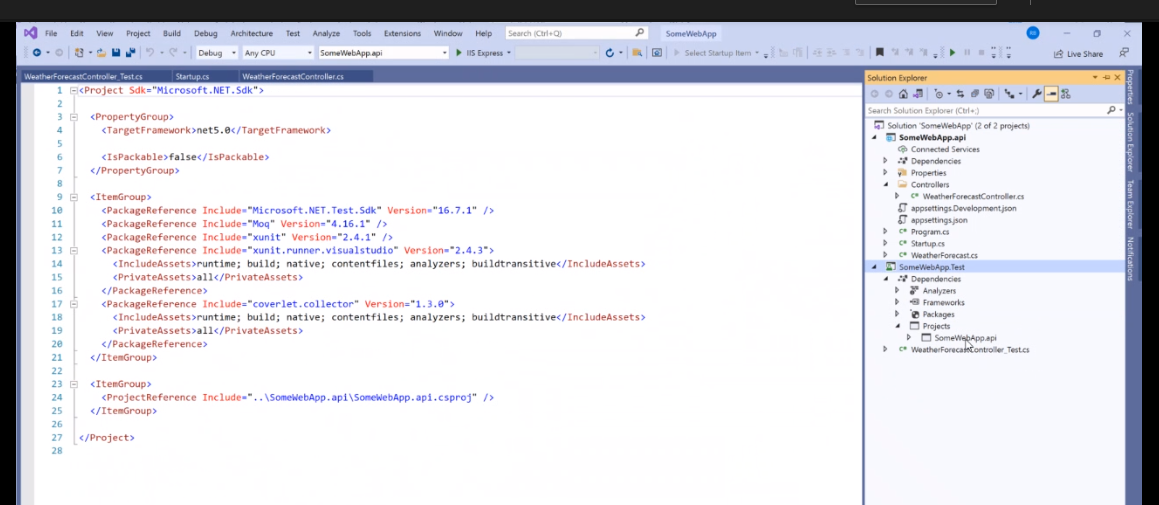




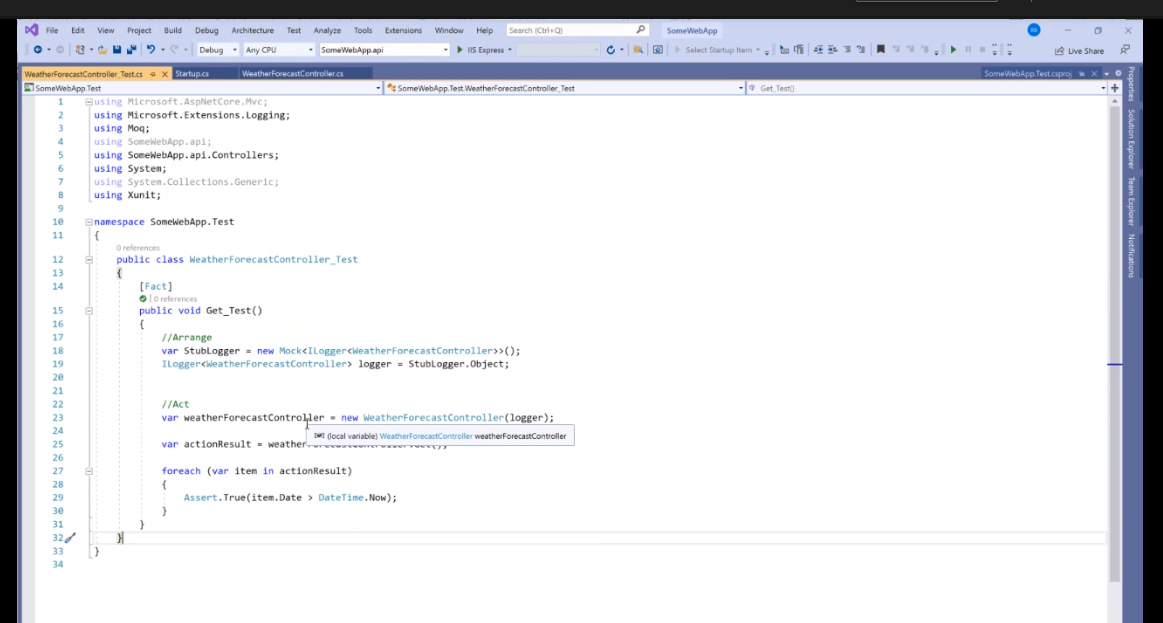
1. Add moc to the project so can use fakes/mocks/stubs…



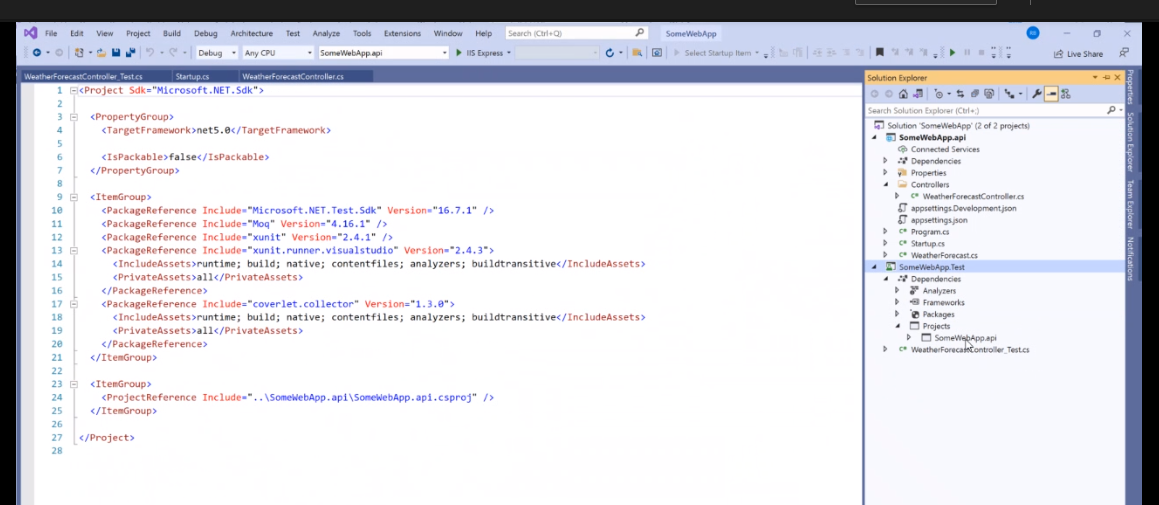
1. When creating the test project, include the target project in the test project



1. Adding to the test



1. Adding the logic in the project:



When debugging C#, put the brake point at the curly bracket {} if you are testing everything within the brackets

