So far we have seen MS Test in action. In this lecture, I'm going to show you how to use NUnit, which is my preferred testing framework. So NUnit is not part of visual studio, and if you want to use it you need to install a couple of new git packages. So open up package manager console, make sure to select testninja.unit tests, in the project dropdown lists,

now we need to install a couple of packages.

So, install-package, the first one is NUnit, and make sure to set the version to the same version that I'm going to use in this video, so you will have the same experience. So -Version 3.8 Beautiful. Now the second package. Install-Package. NUnit, 3TestAdapter. We need this package to run NUnit tests, inside Visual Studio, because by default, the test runner that is built into Visual Studio, only recognizes MS Tests.

So, set the version here as well, this one is 3.8.0. So it's not exactly the same version as NUnit, because these two packages are developed independently.

But these two versions are compatible. Okay, beautiful, now, with NUnit we have, different attributes, so instead of test class, we use Test Fixture, now we need to import this type, so press control, and period, and select the first item. Using NUnit.framework. Okay?

And instead of test method we have Test. Like this, and finally, we need to modify how we make assertions. So here we have a compilation error, because visual studio doesn't know whether you're referring to the Assert class in NUnit framework, or MSTest. So, I'm going to go on the top, and remove this using statement for importing Microsoft Unit Testing framework. So delete,

The compilation is gone, so you can see this Assert class has the same API as the Assert class in MSTest, but here we have a different way of writing Assertions, which is more readable, so, another way to write the same Assertion is like this, Assert that result, is True. It reads like plain English. Also, another way is to write Assert, that, result equals true. So we can add a boolean expression here. Which approach to choose is purely your personal preference, there is no technical differences between these three ways of making assertions.

So in this lecture, I'm going to use the second form. Delete, and delete. Alright now, let's run this test, so once again, on the top, test, run, all tests.

So, all tests are passing, beautiful,

So this is how you use NUnit inside Visual Studio.

Now if you use ReSharper, you don't need to install any

Unit Test adapter, because the test runner that comes with ReSharper

automatically takes NUnit tests, the same is true

if you use the writer ID.