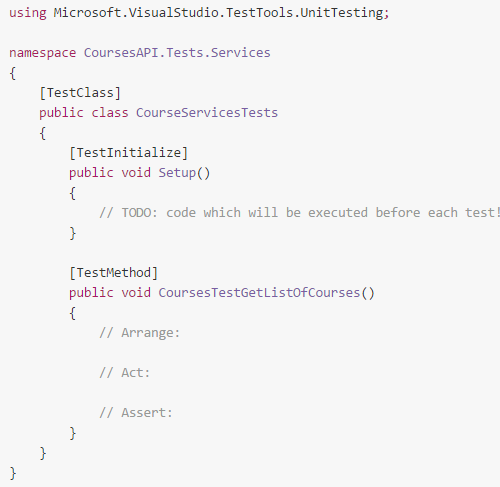
Business logic.. most important code to unit test.

In our **unit test project**.. have at least **ONE TEST CLASS FOR EACH UNIT WE‘RE TESTING**..

The **units** are **MOSTLY SERVICE CLASSES**..

A unit test has the following structure:



* Test class named whatsoever. **NEEDS [TestClass] attribute.**
* It may contain as many test methods as we wish.. called what we want... should **return void**, not accept any params. **NEED [TestMethod] attribute**...
* May declare a method with the **[TestInitialize]**.. will get called **BEFORE EACH TEST IN CLASS IS EXECUTED**
* Should have three As: **Arrange, Act, Assert**

**System Under Test**

Each unit test should test a **single unit**. This unit is sometimes referred to as „System Under Test“.

**Arrange/Act/Assert**

* **Arrange** – this section makes the necessary arrangements for the test.
* **Act** – This section is usually just a single line, **EXECUTE THE METHOD** being tested
* **Assert** – Asserts that all the POSTCONDITIONS ARE VALID.. after the method is executed

**Arrange usually:**

* CREATE AN INSTANCE OF THE SUT
* Create the NECESSARY TEST DATA
* Declare various params/constants which will be used in the Act section

**Test Data**

Most unit tests require some test data.. Test data usually just a collection of objects, which are then fed to a **MockUnitOfWork**



With large applications.. maybe need large amount of test data..

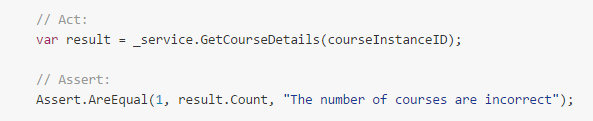
Can declare in several places:

* Declare in **Arrange** section (as seen above). Benefit.. test easier to read and understand.. if results in duplication of test data between tests.. might want to refactor
* Declare common data in the **[TestInitialize]** method. Readability of test could suffer..
* Declare test data in **external files**.. if used across multiple unit test classes

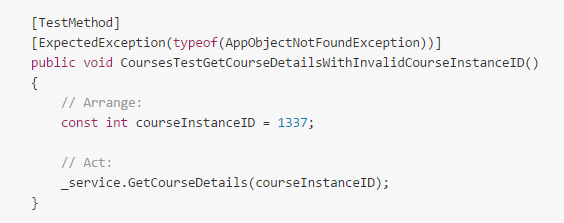
**Assert Section**

Assert class accessible in unit test calsses..

**AreEqual, AreNotEqual, IsNull, IsNotNull, isTrue**



**Exceptions**..





**AUTO TEST GENERATION**

CREATE A LOT OF DATA... **USE Nbuilder**

