**Лабораторная работа №1**

**Технологии разработки программного**

**обеспечения: "разработка через тестирование"**

NUnit - открытая среда юнит-тестирования приложений для .NET, которая позволяет создавать автоматические тесты. Даный вид тестов обладает рядом преимуществ:

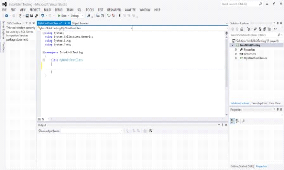
1. Высокое качество программы;
2. Снижение стоимости;
3. Безопасность регрессии сети

Чем выше качество программы, тем меньше средств затрачивается на устранение недостатков проекта. То есть, если найти недостатки в проекте на раннем этапе, решить их будет дешевле.

**№1**: Создание проекта библиотечного класса

Откройте Visual Studio.

Для создания проекта библиотечного класса сначала нажмите File -> New -> Project, выберите Visual C # в установленном шаблоне, выберите Class Library. Введите подходящее название (мы используем IntroNUnitTest), нажмите "OK".



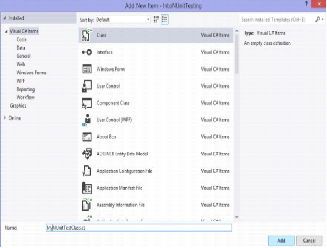
После того, как Вы кликнули кнопку "OK", Вы увидите Ваш проект в Solution Explorer (в правом верхнем углу Visual Studio).

**№2**: Добавьте ссылки к Вашему проекту.

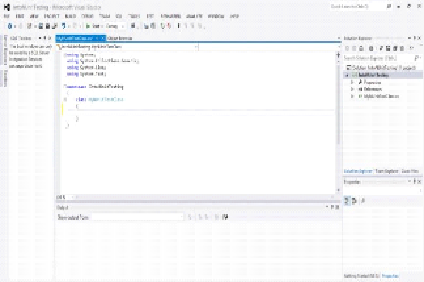
Чтобы добавить ссылку NUnit к Вашему проекту, кликните правой кнопкой мыши на ссылку, выберите Add reference-> Browse, затем nunit.framework.dll и «OK».

**№ 3**: Создание тестового класса.

Кликните правой кнопкой мыши на Вашем проекте, выберите Add -> Class, введите имя (используем MyNUnitTestClass) и далее "Add".



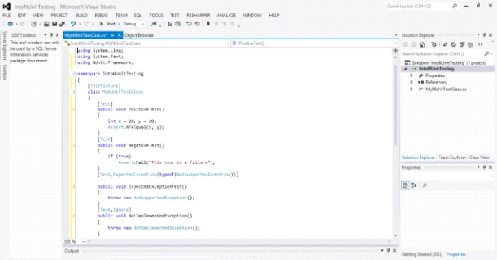
После того, как Вы кликнули на кнопку Add, Вы увидите класс.



После того, как Вы создали класс, добавьте using NUnit.Framework.

Следует соблюдать некоторые условия и понятия:

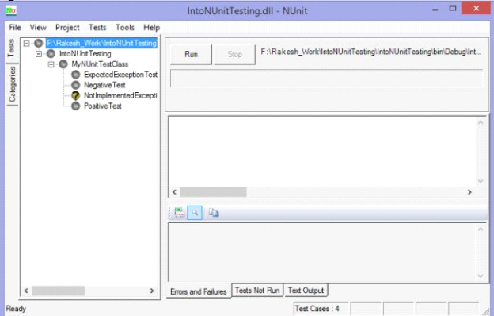
1. Каждый класс должен содержать атрибут [TestFixure] и должен быть общедоступен.
2. В каждом методе должен присутствовать атрибут [Test].
3. Оператор подтверждения об отсутствии ошибок: Булевские значения, описывающие, что должно быть ключевым словом, когда выполняется действие.
4. Ожидаемое исключение: один из типов исключения, который мы ожидаем во время выполнения тест-метода.
5. Установка: программа, которая запускается перед выполнением каждого тест-метода (например, регистрация в системе конкретного пользователя или инициализация одноэлементных классов).
6. Демонтаж: программа запускается после окончания каждого тест-метода (например, удаление строк из таблицы, которые были вставлены во время теста).



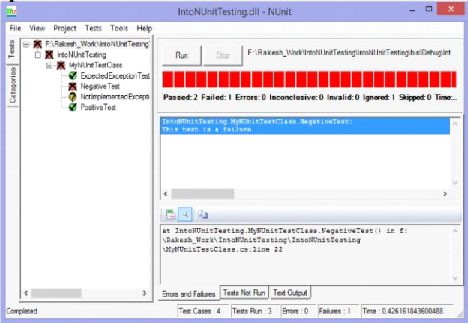
На примере мы постараемся сравнить две переменные х и у. Каждой переменной присвоили значение 20. В общей сложности написали 4 теста (PositiveTest, NegativeTest, ExpectedExceptionTest и NotImplementedException).

**Запуск тестов**

После написания всех тестов в тестовом классе необходимо запустить тесты, чтобы проверить, проходят ли они успешно. Для запуска тестов перейдите в папку NUnit, выберите NUnit Application (.exe) и дважды кликнике по ней, выберите команду File-> Open Project, выберите проект (в нашем случае это IntroNUnitTest.dll), затем кликаем Run.



Далее Вы увидите количество тестов, которые не сработали, и количество невыполненных тестов.



Если Вы хотите запустить какой-то определенный тест, кликните правой кнопкой мыши в тесте, который Вы хотите запустить, и выберите Run test(s).