Telefónica

NTT DATA

Capacitación Pruebas Unitarias (xUnit)

18 Noviembre 2022 – Sesión 6

FUTURE AT HEART

Contenido

Modulo I – Conceptos Básicos

- ¿Qué es una prueba unitaria?
- Otros niveles de prueba
- Conociendo las pruebas unitarias
- Tecnicas de diseño de casos de prueba
- Principios FIRST

Modulo II – Test Driven Development

- Definición
- Desarrollo Ágil – Características
- Ciclo de desarrollo TDD
- Como escribir código que se pueda probar
- Cobertura de código
- Ventajas/Desventajas

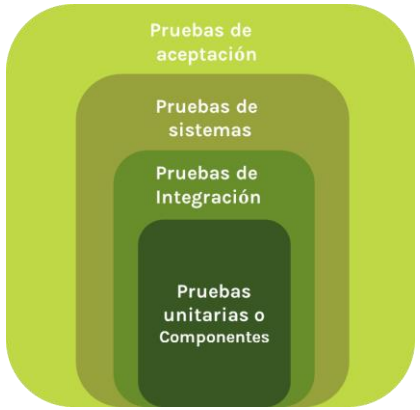
Modulo III – xUnit

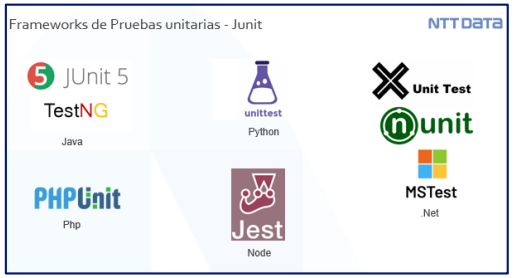
- ¿Qué es xUnit?
- ¿Por qué xUnit?
- ¿Cómo funciona xUnit?
- Conceptos básicos
- Escenarios con xUnit
- Patrones de nombramiento de los escenarios
- Simulaciones
- Aserciones

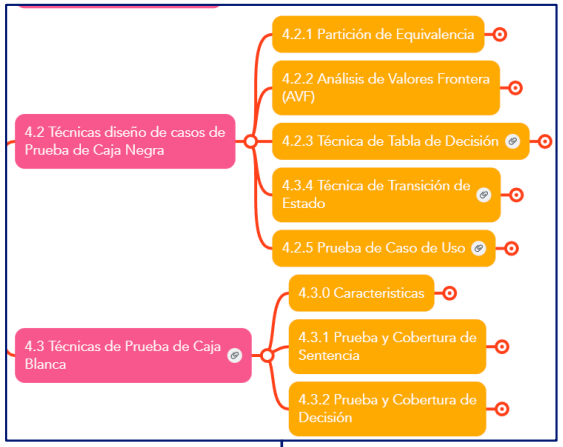
Modulo IV – Taller

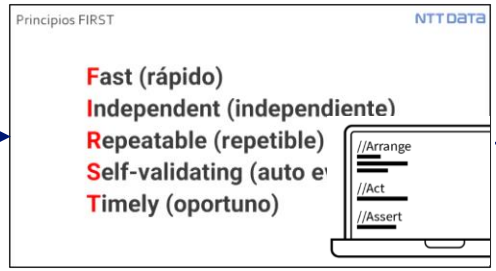
- Preparación del Entorno
- Ejemplo práctico
- Creando casos de prueba
- Ejecución de casos
- Mutation Testing
- Conclusiones/Recomendaciones

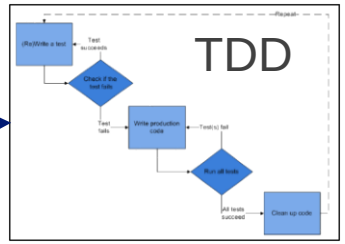
En capítulos anteriores...





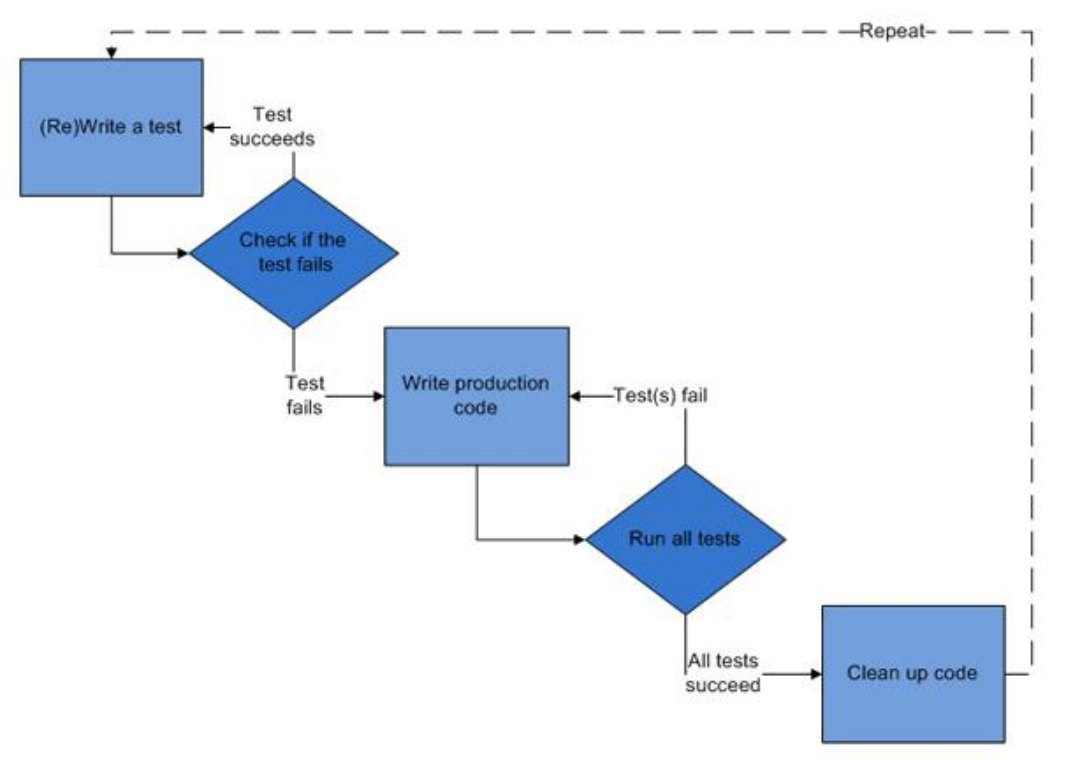


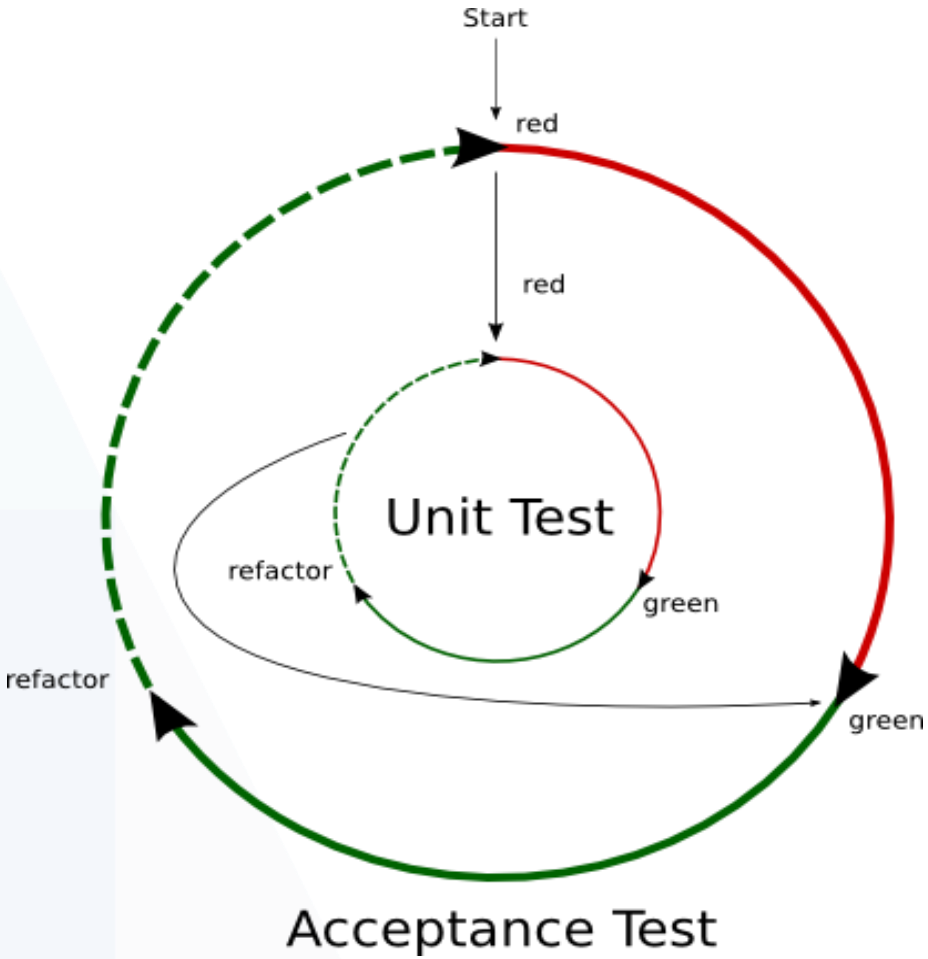




https://github.com/oacarrillo/curso-xUnit

TDD


Repeat
(Re)Write a test
Test succeeds
Check if the test fails
Test fails
Write production code
Test(s) fail
Run all tests
All tests succeed
Clean up code


Start
red
red
Unit Test
refactor
green
refactor
green
Acceptance Test

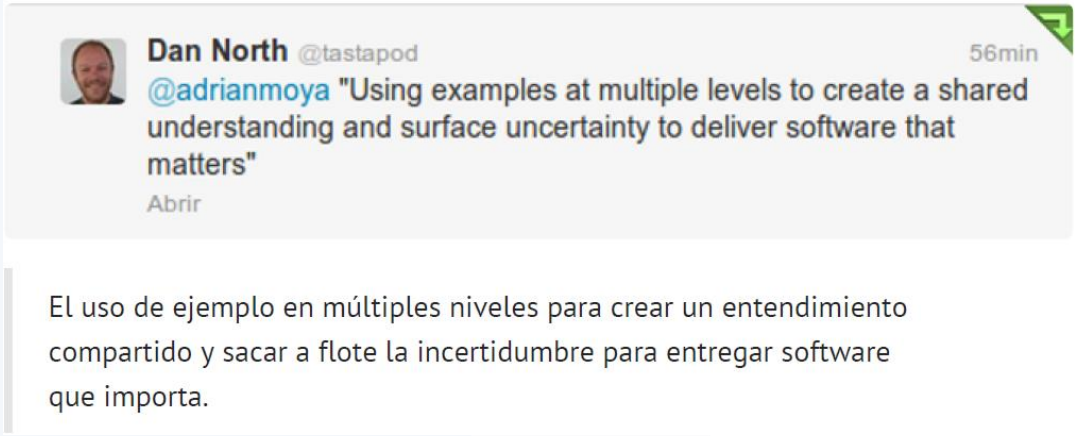
Los orígenes - BDD

Creador BDD

DAN NORTH & ASSOCIATES

"BDD es una metodología ágil de segunda generación, de afuera hacia adentro, basada en demanda, de múltiples interesados, múltiple escala y alta automatización. Describe un ciclo de interacciones con salidas bien definidas, resultando en la entrega de software funcionando y probado que importa."

Dan North



BDD

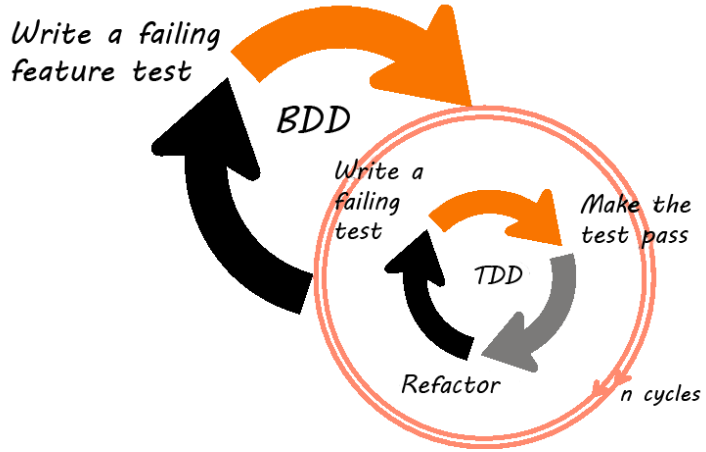
```
public class BuscadorDeClienteTest extends TestCase {
        testBuscarClienteUsandoId() {
                ...
        }
        testFallaCuandoEncuentraDosClientesIguales() {
                ...
        }
        ...
}
```

agiledox (http://agiledox.sourceforge.net/)

```
BuscadorDeClientes
   - Buscar cliente usando Id
   - falla cuando encuentra dos clientes iguales
   - ...
```

BDD

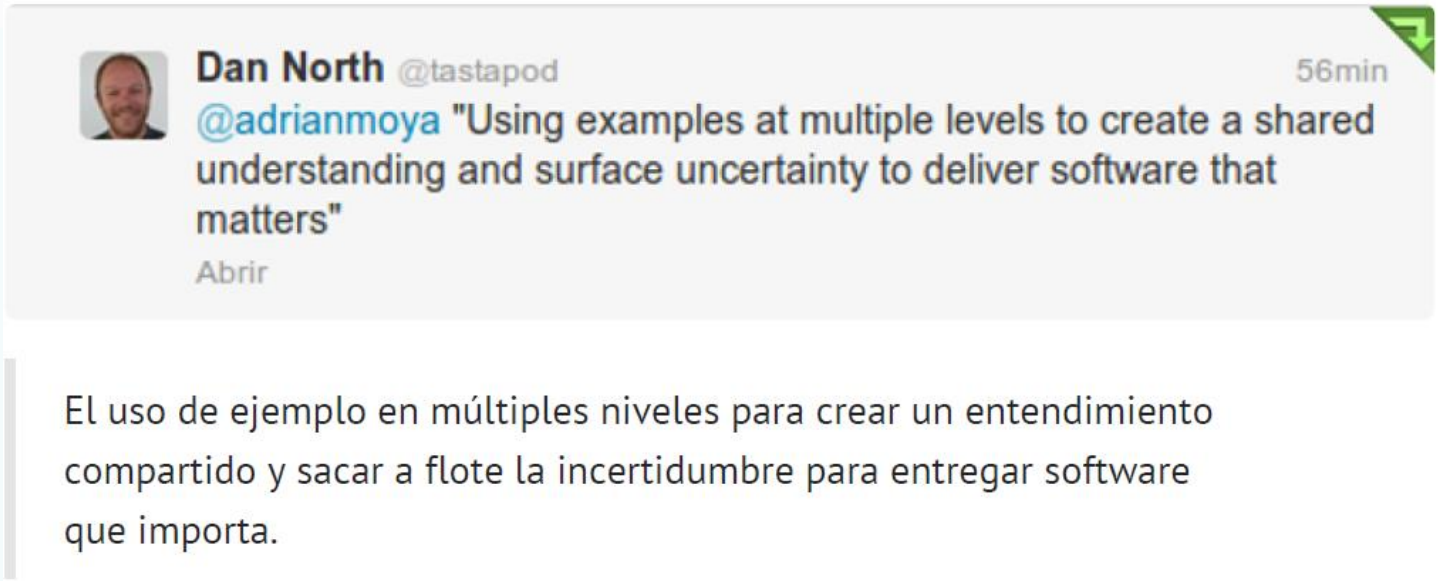
- Una prueba unitaria con buen nombre brilla
- Debería (Should) en vez de Test
- 2003 creación de Jbehave con enfoque de debería y no de test.
- En 2004 se da cuenta que los “deberías” estaban expresados como enunciados de análisis de sistema.
- Lenguaje Ubicuo “DDD”
- Nace el Given, When, Then (Dado que, Cuando, Entonces)
- Entonces los criterios de aceptación se pueden ejecutar
- Gherkin



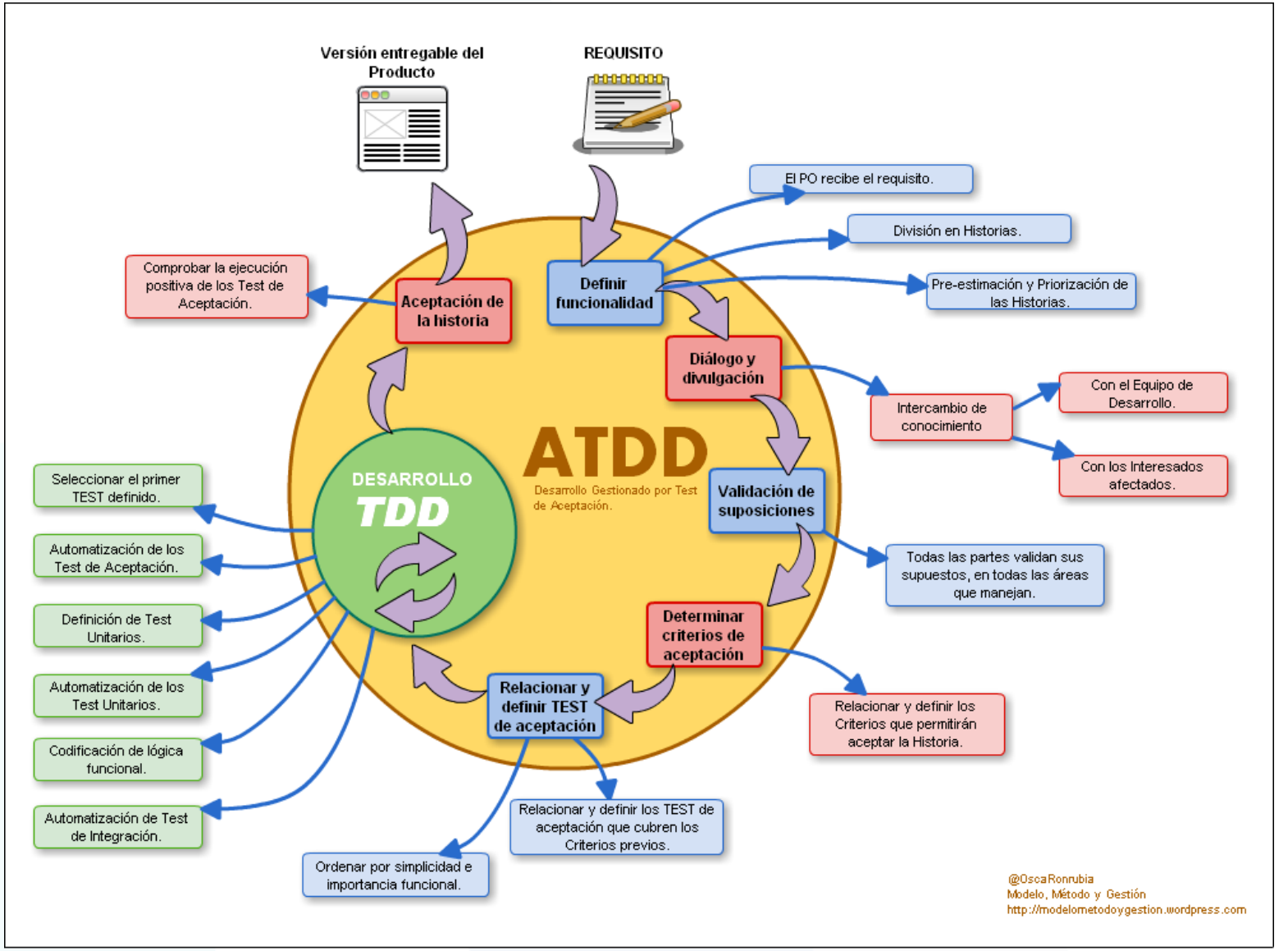
BDD

“BDD es una metodología ágil de segunda generación, de afuera hacia adentro, basada en demanda, de múltiples interesados, múltiple escala y alta automatización. Describe un ciclo de interacciones con salidas bien definidas, resultando en la entrega de software funcionando y probado que importa.”

Dan North



Ciclo ATDD


Versión entregable del Producto
REQUISITO
El PO recibe el requisito.
División en Historias.
Pre-estimación y Priorización de las Historias.
Definir funcionalidad
Diálogo y divulgación
Intercambio de conocimiento
Con el Equipo de Desarrollo.
Con los Interesados afectados.
Validación de suposiciones
Todas las partes validan sus supuestos, en todas las áreas que manejan.
Determinar criterios de aceptación
Relacionar y definir los Criterios que permitirán aceptar la Historia.
Relacionar y definir TEST de aceptación
Relacionar y definir los TEST de aceptación que cubren los Criterios previos.
Ordenar por simplicidad e importancia funcional.
Aceptación de la historia
Comprobar la ejecución positiva de los Test de Aceptación.
ATDD
Desarrollo Gestionado por Test de Aceptación.
DESARROLLO TDD
Seleccionar el primer TEST definido.
Automatización de los Test de Aceptación.
Definición de Test Unitarios.
Automatización de los Test Unitarios.
Codificación de lógica funcional.
Automatización de Test de Integración.
@OscaRonrubia
Modelo, Método y Gestión
http://modelometodoygestion.wordpress.com

Diferencias

“TDD se ocupa solo a nivel unitario o una porción pequeña de la aplicación en desarrollo, BDD se va a ocupar de que las pruebas sobre la integración de esas unidades a un nivel de negocio para que no solo sean pruebas, sino que también sean documentación viva de la aplicación al alcance de cualquier usuario y en su mismo idioma (términos que usa comúnmente). Por último, ATDD pasa a participar a nivel de la creación de la US, asegurándose que estamos construyendo lo que el cliente realmente necesita en esta etapa de desarrollo”

Beneficios BDD

- Propiedad de los clientes
- Escrito en conjunto con los clientes, desarrolladores y analistas de prueba. Entendimiento compartido.
- Sobre el **Qué** y no sobre el **Cómo**
- Expresada en lenguaje de dominio del problema
- Conciso, preciso y sin ambigüedades
- Documentación en Vivo
- Las pruebas definen el comportamiento del sistema
- Rápido desarrollo

Gherkin (Jbehave (Java) - Cucumber(Java, Ruby, JS) - SpecFlow (C#))

NTT DATA


```
#language:es
@DEMO
Característica: Prueba de concepto SpringBoot + Cucumber - GOOGLE

  @GOOGLE_SEARCH
  Escenario: caso1-Busqueda en google
    Dado que abro la pagina de google
    Cuando escribo la busqueda de: "cantantes de salsa"
    Entonces se muestran los resultados en pantalla
```

```
@Given("que abro la pagina de google")
public void queAbroLaPaginaDeGoogle() { manager.navigateTo(urlGoogle); }
```

Ejercicio TDD

Cree una calculadora que reciba como parámetro una cadena de carácter, tal cual como lo muestra la siguiente firma de método:

int Add(string numbers)

El método puede tomar hasta dos números, separados por coma y devolverá su suma. Por ejemplo "" o "1" o "1,2" como entradas. (Para una cadena vacía devolverá 0, Para una cadena de un solo digito devolverá en mismo numero, para una cadena de 2 números devolverá su suma)

Recomendaciones:

Reciba cualquier cantidad de parámetros y devuelva la suma

- Comience con el caso de prueba más simple que es el de la cadena vacía y pase a un numero y luego a dos números.
- Recuerda resolver las cosas de la manera más sencilla posible.
- Recuerde refactorizar después de cada prueba. Luego de hacer el refactor vuelva a ejecutar las pruebas para validar que no se rompió nada.

https://osherove.com/tdd-kata-1/


NTT DATA
GRACIAS
FUTURE
AT HEART