\* **Definições**

* Teste unitário se refere a testar a menor unidade do código.

Menor trecho de código testado, por exemplo o método de uma classe

Os testes unitários têm por objetivo validar pequenas partes do software com base em suas entradas possíveis e saídas esperadas. As unidades usadas nesse tipo de teste são as menores partes testáveis de um sistema, normalmente funções, que recebem argumentos e retornam um determinado valor ou efetuam alguma ação cujo resultado pode ser analisado. (Garante que o software vai fazer o que foi programado pra fazer)

* O teste unitário é uma automatização de teste.
* Temos uma classe de teste para cada classe de produção, então depois do teste ser escrito ele pode ser executado diversas vezes, será executado pela suite de teste.
* Os framework e as bibliotecas de teste unitário possuem uma ferramenta de execução de teste.
* No corpo do teste são definidos os métodos de teste ou seja os teste que vão testar os métodos de produção dentro da classe de teste, o framework fará todo trabalho e só mostrará na tela os que resultados.
* Na prática, os testes unitários consistem de classes contendo métodos que testam partes pequenas e isoladas do sistema. Essas classes são criadas com o auxílio de um framework de testes (como o JUnit no Java, ou o NUnit ou o Microsoft Test no .Net). Esses testes podem então ser executados pela linha de comando, pela sua IDE ou mesmo por um serviço de build automatizado.

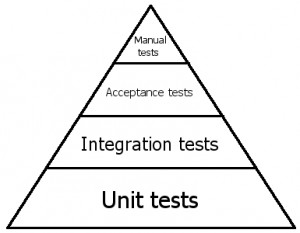
**Benefícios**

- Ajuda a prevenir a regressão de bugs. caso alguma alteração futura faça com que o bug retorne (“regresse”), o teste vai indicar isso.

Redução de custos

É até possível argumentar que os ciclos são praticamente os mesmos. Mas a grande vantagem dos testes unitários é que, uma vez escritos, eles estão automatizados para o resto da vida do projeto. Você “perde tempo” só uma vez. Você investe tempo e esforço no começo, para criar os testes, e colhe os benefícios por tempo indefinido.

**Pirâmide**:



Obs: a importância dos teste manuais.

**Junit**

JUnit é um framework que facilita o desenvolvimento e execução de testes unitários em código Java.

Ele Fornece uma completa API para construir os testes e aplicações gráficas em modo console para executar os testes criados.

O JUnit trabalha basicamente com anotações (Annotations). Essas anotações indicam se um método é de teste ou não, se um método deve ser executado antes da classe e/ou depois da classe, indicam também se o teste deve ou não ser ignorado e se a classe em questão é uma suíte de teste, ou seja, se a partir desta classe é disparada a execução das demais classes de teste, entre outras anotações menos utilizadas.

Como fazer

-Teste unitário para cada classe

- Criar um pacote para o teste unitário

- Criar junit com [nomedaclasse]Test

- Marcar o SetUp()