

Tópicos Abordados

- A importância dos testes
- Tipos de testes
- JUnit
 - O que é
 - Como funciona
 - `@Test`, `@Ignore`, `@BeforeClass`, `@AfterClass`,
`@Before` e `@After`
 - Test Suites

A Importância dos Testes

- Parte mais importante do desenvolvimento
 - Código que não funciona não presta
- Quem testa
 - Desenvolvedor, equipe de testes, clientes
- Automatização de testes
 - Avaliam se outro programa funciona de acordo com o esperado

A Importância dos Testes

- Por que testar?
 - Como saber se algo funciona sem testar?
 - Como saber se algo ainda funciona após alterações terem sido feitas?
- Código testado é mais confiável
- Código testado pode ser alterado sem “medo”

Tipos de Testes

- Testes de unidade
 - Testam unidades de lógica
 - Em OO, normalmente testam métodos
 - Ignoram condições externas
- Testes de integração
 - Testam como unidades interagem entre si ou com o ambiente onde executam

Tipos de Testes

- Testes funcionais
 - Conhecidos como “caixa preta”
 - Testam casos de uso de uma aplicação
 - Validam a interface do usuário, operações requisitadas, etc.

O que é o JUnit

- Framework que facilita o desenvolvimento de **testes de unidade** em Java
- Site oficial
 - www.junit.org
- Open-source
- Integrado em diversas IDEs

Funcionamento do JUnit

- Existência de Test Cases
 - Uso de anotações em métodos de testes executados pelo JUnit
 - Comparação do dado retornado com o dado esperado para definir se o teste falhou ou não
 - `assertEquals()`
 - `assertTrue()`
 - `assertNotNull()`
 - etc.

Anotando Métodos: @Test

- Anotação usada para indicar quais métodos o JUnit deve executar
- O método deve ser public, retornar void e não ter parâmetros

```
@Test
public void metodoParaTestar() {
    //assertTrue(...);
    //assertEquals(...);
    //...
}
```

Anotando Métodos: @Test

- A anotação @Test pode receber dois parâmetros opcionais

Parâmetro	Descrição
expected	Se o método deve lançar uma exceção, especifica que exceção é essa. Se a exceção especificada não for lançada, o método falha.
timeout	Indica quanto tempo o método pode levar para executar no máximo. Se este tempo for ultrapassado, o teste falha.

Anotando Métodos: @Ignore

- Anotação usada para desabilitar um ou mais testes temporariamente
- Caso a anotação seja usada na classe, todos os métodos de teste são desabilitados

```
@Test
@Ignore
public void metodoParaTestar() {
    //assertTrue(...);
    //assertEquals(...);
    //...
}
```

Anotando Métodos: @BeforeClass

- Anotação usada no método que deve ser executado antes dos testes da classe serem executados
- Este método pode inicializar recursos necessários para realização dos testes
- O método deve ser public, static, void e sem parâmetros

```
@BeforeClass
public static void inicializar() {
    //inicialização de recursos
}
```

Anotando Métodos: @AfterClass

- Anotação usada no método que deve ser executado depois dos testes da classe serem executados
- Este método pode destruir recursos abertos pelo método anotado com @BeforeClass
- O método deve ser public, static, void e sem parâmetros

```
@AfterClass
public static void destruir() {
    //destruição de recursos
}
```

Anotando Métodos: @Before

- Métodos anotados com @Before são executados toda vez que um novo teste da classe está prestes a ser executado
- O método deve ser public, void e sem parâmetros

```
@Before
public void executarAntesDoTeste() {
    //inicialização antes de executar cada teste
}
```

Anotando Métodos: @After

- Métodos anotados com @After são executados toda vez que um teste da classe termina de ser executado
- O método deve ser public, void e sem parâmetros

```
@After
public void executarDepoisDoTeste() {
    //destruição depois de executar cada teste
}
```

Test Suites

- Test suites agrupam diversos test cases
- Facilitam a execução

```
@RunWith (Suite.class)
@SuiteClasses({ TestCase1.class, TestCase2.class })
public class MySuite {
}
```

Colocando em Prática...



Agora que você já aprendeu a teoria, acesse as vídeo-aulas práticas e pratique os assuntos abordados neste módulo!

[Clique aqui para acessar as vídeo-aulas práticas](#)
