

Atualizando as dependências do projeto

No próximo exercício, vamos criar um teste de unidade com uma versão mais nova do JUnit do que a declarada no `pom.xml`.

Você pode sempre encontrar as informações sobre as bibliotecas do mundo Java e como adicioná-las ao seu projeto no `mvnrepository`. No caso do JUnit, as informações encontram-se [aqui \(http://mvnrepository.com/artifact/junit/junit\)](http://mvnrepository.com/artifact/junit/junit).

Em projetos com muitas dependências, com o passar do tempo elas podem ficar desatualizadas, caso você queira atualizar as dependências para suas versões mais atuais, poderá utilizar o seguinte comando:

```
mvn versions:use-latest-versions
```

Você pode encontrar a documentação desse *goal* [aqui \(http://www.mojohaus.org/versions-maven-plugin/use-latest-versions-mojo.html\)](http://www.mojohaus.org/versions-maven-plugin/use-latest-versions-mojo.html). Tente executar esse comando no seu projeto e perceba que o JUnit foi atualizado para a versão mais recente no `pom.xml`.

Perceba que o comando modifica o `pom.xml` atualizando as versões de todas as dependências. Isso pode ser perigoso, especialmente se no projeto não existirem bons [testes de unidade \(https://www.alura.com.br/curso-online-tdd\)](https://www.alura.com.br/curso-online-tdd) e [integração \(https://www.alura.com.br/curso-online-selenium\)](https://www.alura.com.br/curso-online-selenium), pois se algo deixar de funcionar isso pode ser descoberto apenas em produção.

Existe um outro *goal* que verifica por atualizações sem de fato alterar o `pom.xml`:

```
mvn versions:display-dependency-updates
```

Ele mostrará as dependências que possuem novas versões, mas não atualizará para você:

```
lucas@lucas-Inspiron-5458:~/Projects/produtos$ mvn versions:display-dependency-updates
[INFO] Scanning for projects...
[INFO]
[INFO] -----
[INFO] Building produtos 1.0-SNAPSHOT
[INFO] -----
[INFO] --- versions-maven-plugin:2.2:display-dependency-updates (default-cli) @ produtos ---
[INFO] The following dependencies in Dependencies have newer versions:
[INFO]   junit:junit ..... 3.18 -> 4.12
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 1.221 s
[INFO] Finished at: 2016-04-04T10:33:26-03:00
[INFO] Final Memory: 12M/150M
[INFO] -----
lucas@lucas-Inspiron-5458:~/Projects/produtos$
```

Os dois *goals* fazem parte do plugin Versions, cuja documentação pode ser encontrada [aqui \(http://www.mojohaus.org/versions-maven-plugin/\)](http://www.mojohaus.org/versions-maven-plugin/).