

Programação Orientada a Objetos

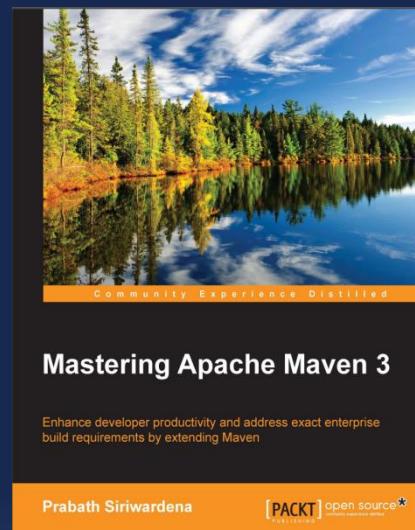
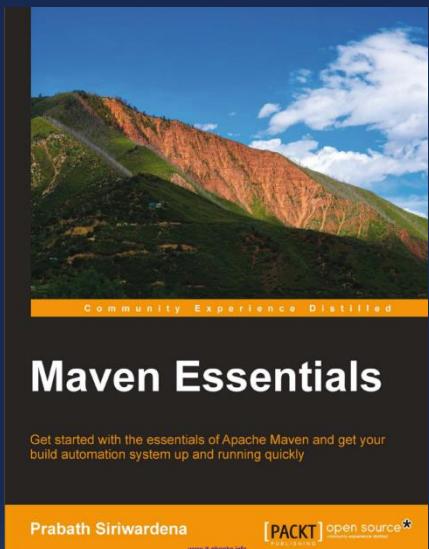
Unidade 9 - Desenvolvimento Web com Servlets

Criação de Projeto Java Web com Maven e git



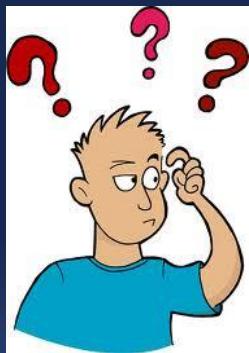
Prof. Aparecido V. de Freitas
Doutor em Engenharia
da Computação pela EPUSP
aparecido.freitas@prof.uscs.edu.br
aparecidovfreitas@gmail.com

Bibliografia



Bibliografia

O que é Maven ?





- Ferramenta de automação de build
- Gerencia o processo de desenvolvimento de produtos (artefatos) em Java (preferencialmente)
- Abordagem declarativa (diferentemente do Ant)
- Convenção sobre Configuração
- Desenvolvido pelo grupo Apache (suporte do projeto Avalon)

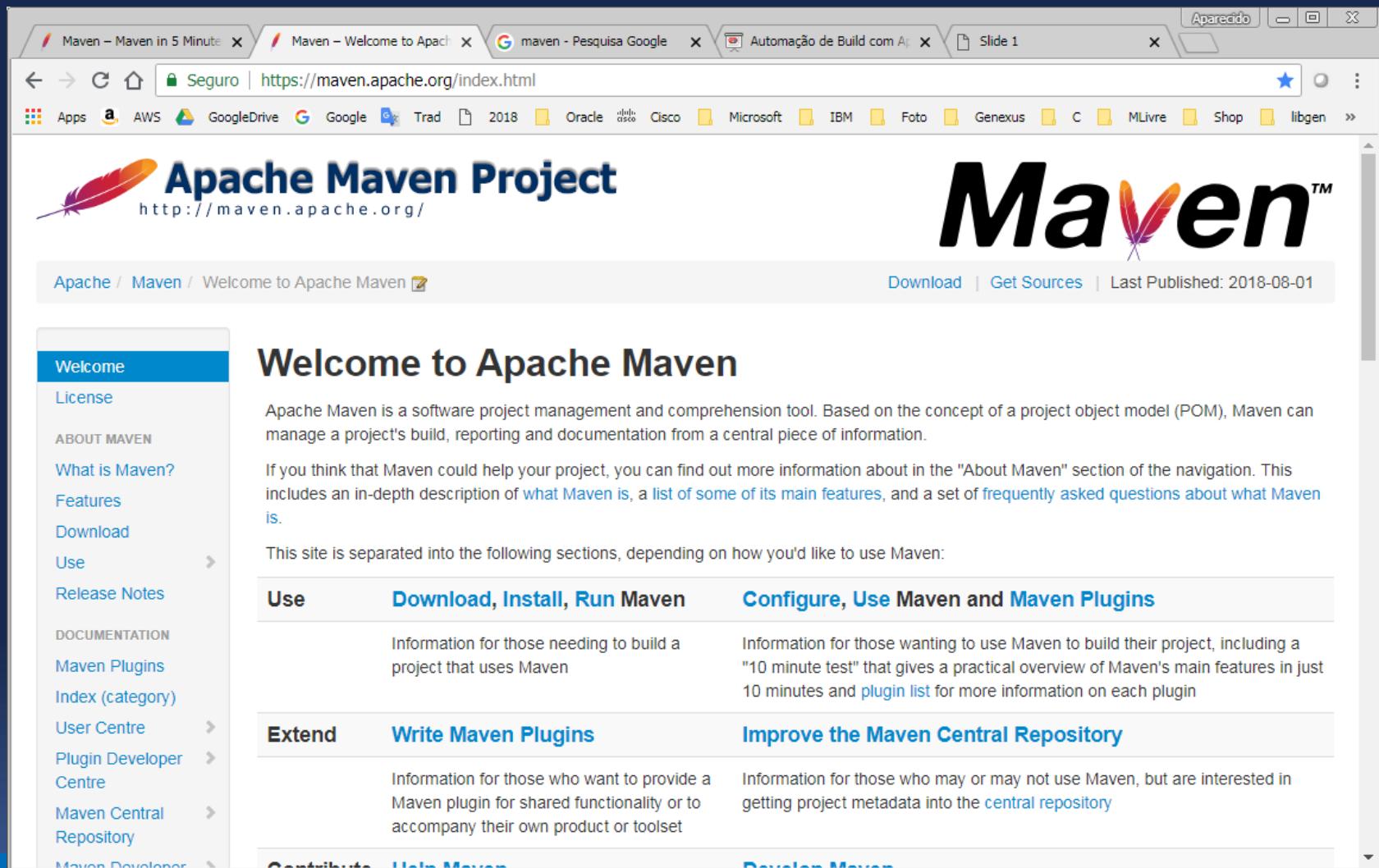
Porque usar Maven ?

- Padronização do processo de desenvolvimento
- Gerenciamento de dependências (bibliotecas)
- Compartilhamento de componentes
 - Diferentemente de compartilhamento de código/build
- Extensível através de plugins
- Testes facilitados
- Documentação facilitada
- Fácil integração com ferramentas de integração contínua e monitoramento da qualidade do código
- Integração com IDE

Instalação

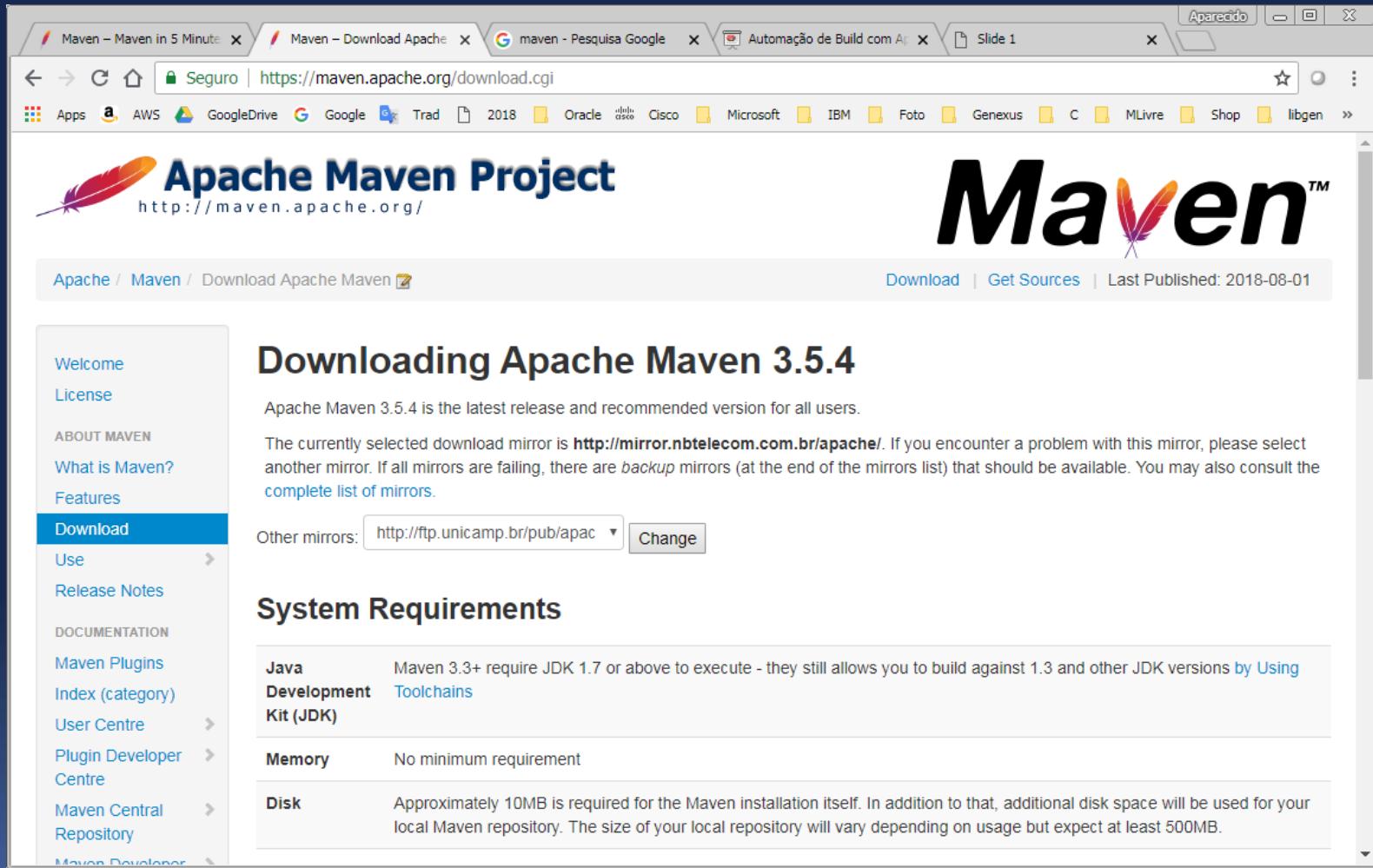
- Baixar distribuição
 - <http://maven.apache.org/download.html>
- Descompactar
- Configurar variáveis de ambiente
 - Windows
 - M2_HOME=C:\apache-maven-3.0.4 (opcional)
 - JAVA_HOME=<pasta de instalação do JDK>
 - PATH=%M2_HOME%\bin;%JAVA_HOME%\bin;%PATH%
- Testar instalação
 - PROMPT> mvn -version

```
Apache Maven 3.0.4 (r1232337; 2012-01-17 06:44:56-0200)
Maven home: C:\Program Files\Java\apache-maven-3.0.4
Java version: 1.7.0_03, vendor: Oracle Corporation
Java home: C:\Program Files\Java\jdk1.7.0_03-x64\jre
Default locale: pt_BR, platform encoding: Cp1252
OS name: "windows 7", version: "6.1", arch: "amd64", family: "windows"
```



The screenshot shows a web browser window with the URL <https://maven.apache.org/index.html>. The page title is "Apache Maven Project". The main content area features a large "Maven™" logo with a feather icon. The navigation bar includes links for "Download" and "Get Sources", with a note that the page was last published on 2018-08-01. The left sidebar contains a "Welcome" section with links to "License", "ABOUT MAVEN", "What is Maven?", "Features", "Download", "Use", "Release Notes", and sections for "DOCUMENTATION", "Maven Plugins", "Index (category)", "User Centre", "Plugin Developer Centre", and "Maven Central Repository". The main content area also includes sections for "Use", "Extend", "Download, Install, Run Maven", "Configure, Use Maven and Maven Plugins", "Write Maven Plugins", and "Improve the Maven Central Repository".

Download



The screenshot shows a web browser window with multiple tabs open. The active tab is for the Apache Maven Project download page at <https://maven.apache.org/download.cgi>. The page title is "Apache Maven Project" and the main heading is "Maven™". A sidebar on the left contains links for Welcome, License, ABOUT MAVEN, What is Maven?, Features, Download (which is highlighted in blue), Use, Release Notes, DOCUMENTATION, Maven Plugins, Index (category), User Centre, Plugin Developer Centre, Maven Central Repository, and Maven Developers. The main content area starts with "Downloading Apache Maven 3.5.4". It states that Apache Maven 3.5.4 is the latest release and recommended version for all users. It mentions the currently selected download mirror is <http://mirror.nbtelecom.com.br/apache/>. Below this, there is a "Other mirrors:" dropdown set to <http://ftp.unicamp.br/pub/apac> with a "Change" button. The "System Requirements" section lists Java Development Kit (JDK) requirements (Java 3.3+ requiring JDK 1.7 or above), Memory requirements (No minimum requirement), and Disk requirements (Approximately 10MB required for installation and additional space for local repository, expecting at least 500MB). The bottom of the page includes a footer with links to Apache, Google, Microsoft, IBM, Foto, Genexus, C, MLivre, Shop, and Ibgem.

Processo de Instalação

- ✓ **Maven** é uma **Java Tool**, assim você deve ter Java instalado em seu computador;
- ✓ Assegure que a variável de ambiente **JAVA_HOME** está definida e apontada para a instalação **JDK**;
- ✓ Extrair a distribuição em qualquer diretório do sistema.

```
unzip apache-maven-3.5.4-bin.zip  
or  
tar xzvf apache-maven-3.5.4-bin.tar.gz
```

Processo de Instalação

- ✓ Adicione o diretório bin do diretório onde o **Maven** foi instalado na variável de ambiente **PATH**;
- ✓ Adicione a variável de ambiente **MAVEN_HOME** apontando para o diretório de instalação do **MAVEN**;
- ✓ Confirme com **mvn –version** no prompt de comandos;
- ✓ O resultado deve ser similar a:

```
Apache Maven 3.5.4 (1edded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T20:33:14+02:00)
Maven home: /opt/apache-maven-3.5.4
Java version: 1.8.0_45, vendor: Oracle Corporation
Java home: /Library/Java/JavaVirtualMachines/jdk1.8.0_45.jdk/Contents/Home/jre
Default locale: en_US, platform encoding: UTF-8
OS name: "mac os x", version: "10.8.5", arch: "x86_64", family: "mac"
```

mvn -version

```
C:\ Command Prompt
C:\Users\VB_Aparecido>mvn -version
Apache Maven 3.5.4 (1eedded0938998edf8bf061f1ceb3cfdeccf443fe; 2018-06-17T11:33:1
4-07:00)
Maven home: C:\apache-maven-3.5.4\bin\..
Java version: 1.8.0_171, vendor: Oracle Corporation, runtime: C:\Program Files\J
ava\jdk1.8.0_171\jre
Default locale: en_US, platform encoding: Cp1252
OS name: "windows 7", version: "6.1", arch: "amd64", family: "windows"
C:\Users\VB_Aparecido>
```

Funcionalidades

- Controla a transformação de itens fonte em itens derivados
- Faz a gestão de dependências dos módulos envolvidos
 - Evita compilações desnecessárias
 - Calcula transitivamente as dependências



Elementos Básicos

- **POM (*Project Object Model*)**
 - Descritor XML da estrutura de um projeto
- **Ciclo de vida (*lifecycle*)**
 - Processo de construção
 - Ex.: compile → test → package → install → deploy
- **Fase (*phase*)**
 - Passo do processo de construção
 - Ex.: compile
- **Plug-in**
 - Ferramenta utilizada no processo de construção
 - Ex.: scm
- **Meta (*goal*)**
 - Funcionalidade provida por uma ferramenta
 - Ex.: scm:checkin



Gestão da Estrutura do Projeto

- Problema: “Onde cada artefato do projeto deve ser colocado?”
- Maven permite gerar o **esqueleto do projeto (layout)** aderente às suas convenções
- Fornece 292 layouts, dentre eles:
 - Aplicação Java, Groovy, Ruby, Scala



Arquivo pom.xml

```
<project ...>
  <modelVersion>4.0.0</modelVersion>
  <groupId>uff</groupId>
  <artifactId>teste</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>jar</packaging>
  <name>teste</name>
  <url>http://maven.apache.org</url>
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  </properties>
  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
```



Convenção da Estrutura do Projeto

src/main/java	Código fonte da aplicação (no exemplo, código java)
src/main/resources	Recursos da aplicação (imagens, sons, etc.)
src/test/java	Código de teste (no exemplo, testes junit)
src/test/resources	Recursos de teste
src/site	Site do projeto
target	Diretório com arquivos gerados pelo processo de build
LICENSE.txt	Licença do projeto
README.txt	Visão geral do projeto
pom.xml	Descriptor Maven do projeto



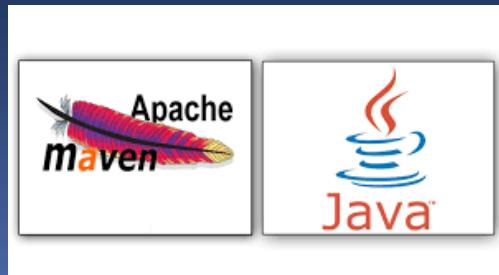
Gestão do Processo de Build

- Problema: “Como posso construir o projeto?”
- Maven permite utilizar um **processo padrão** para construção do projeto
- Processo composto das seguintes fases (entre outras):
 - Compilação
 - Testes
 - Empacotamento

Gestão do Processo de Build

(fases mais comuns)

- **validate**: verifica se o projeto está correto e se os dados estão disponíveis
- **compile**: compila o código do projeto
- **test**: executa testes de unidade
- **package**: empacota o código compilado em um formato apropriado (ex.: jar)
- **integration-test**: implanta o pacote em um ambiente apropriado e executa testes de integração
- **verify**: executa verificações de qualidade sobre o pacote
- **install**: instala o pacote no repositório local
- **deploy**: disponibiliza o pacote em um repositório remoto
- **clean**: remove artefatos criados por processos anteriores de construção
- **site**: gera o site do projeto



Gestão de Dependências do Projeto

- Problema: “Como lidar com situações onde o projeto depende de bibliotecas externas?”
- Maven permite definir dependências para um projeto
 - As dependências são definidas no pom.xml
 - O Maven calcula as dependências considerando transitividade
 - As dependências são baixadas de repositórios centrais por demanda



Gestão de Dependências do Projeto

(definindo dependência no pom.xml)

```
<project ...>
...
<dependencies>
    <dependency>
        <groupId>axis</groupId>
        <artifactId>axis</artifactId>
        <version>1.4</version>
        <scope>compile</scope>
    </dependency>
    <dependency>
        <groupId>junit</groupId>
        <artifactId>junit</artifactId>
        <version>3.8.1</version>
        <scope>test</scope>
    </dependency>
</dependencies>
</project>
```



Repositório Local

- ✓ O Maven utiliza um diretório local para baixar os artefatos da internet. O diretório padrão fica dentro pasta do usuário, na pasta **.m2**. Um exemplo no Windows é **c:\users\aparecido\.m2\repository**.
- ✓ Entretanto, pode-se mudar esse diretório para junto de outros arquivos de desenvolvimento.
- ✓ Para isso, basta editar o settings.xml, movendo a tag **<localRepository>** para fora do comentário e adicionando o caminho, por exemplo:

```
<localRepository>c:\develop\apache-maven-3.5.4\repo</localRepository>
```

- ✓ Não se esqueça de criar o diretório especificado caso o mesmo não exista.



Exemplo – Processo de Build



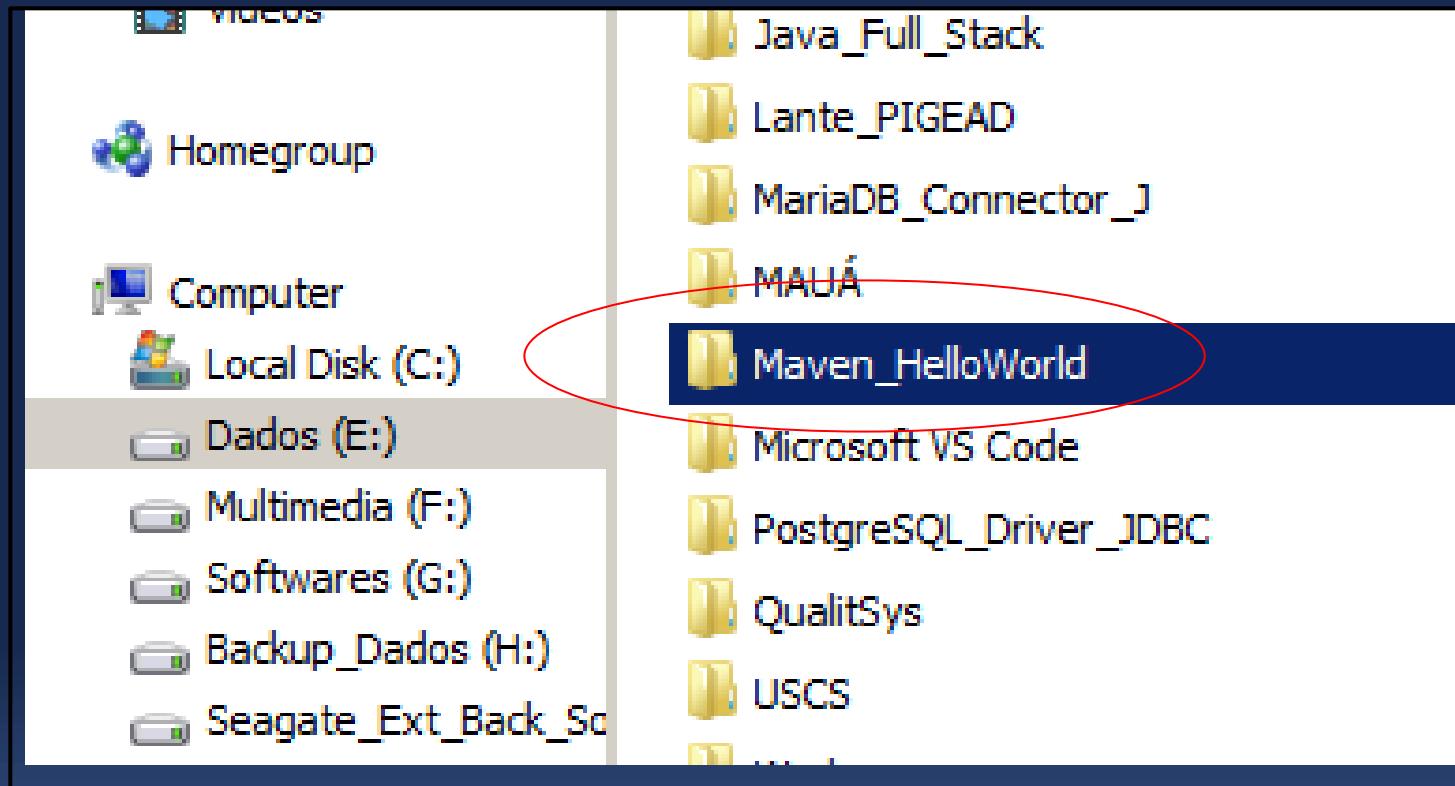
Exemplo de Geração – Passo 1

1. Vamos criar uma aplicação “Hello World” a partir da linha de comandos do Maven;
2. A pasta do projeto será: E:\Maven_HelloWorld
3. Estamos assumindo que o Maven está instalado e a variável de ambiente MAVEN_HOME está definida e apontando para a pasta de instalação do Maven.



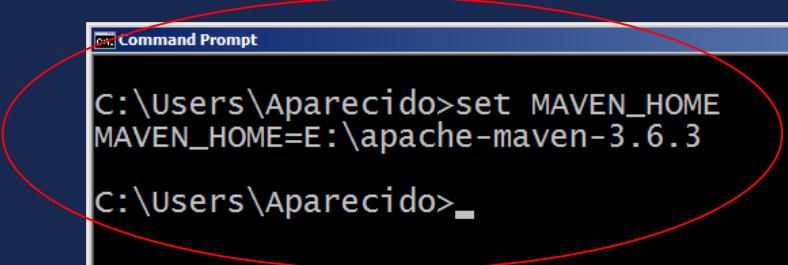
Exemplo de Geração – passo 1

4. Verificação da pasta criada para o projeto:



Exemplo de Geração – passo 1

5. Verificação da variável de ambiente do Maven (MAVEN_HOME)



```
cmd Command Prompt
C:\Users\Aparecido>set MAVEN_HOME
MAVEN_HOME=E:\apache-maven-3.6.3

C:\Users\Aparecido>_
```



Exemplo de Geração – passo 2

1. Por padrão, o Maven baixa todos os módulos especificados no POM.XML na pasta: \users\usuario\.m2\repository;
2. Pode-se configurar o Maven para baixar as bibliotecas em alguma pasta que pode ser definida pelo usuário;
3. Para isso, basta editar o arquivo settings.xml, movendo a tag <localRepository> para fora do comentário e adicionando o caminho da nova pasta usada para o Repositório, por exemplo:

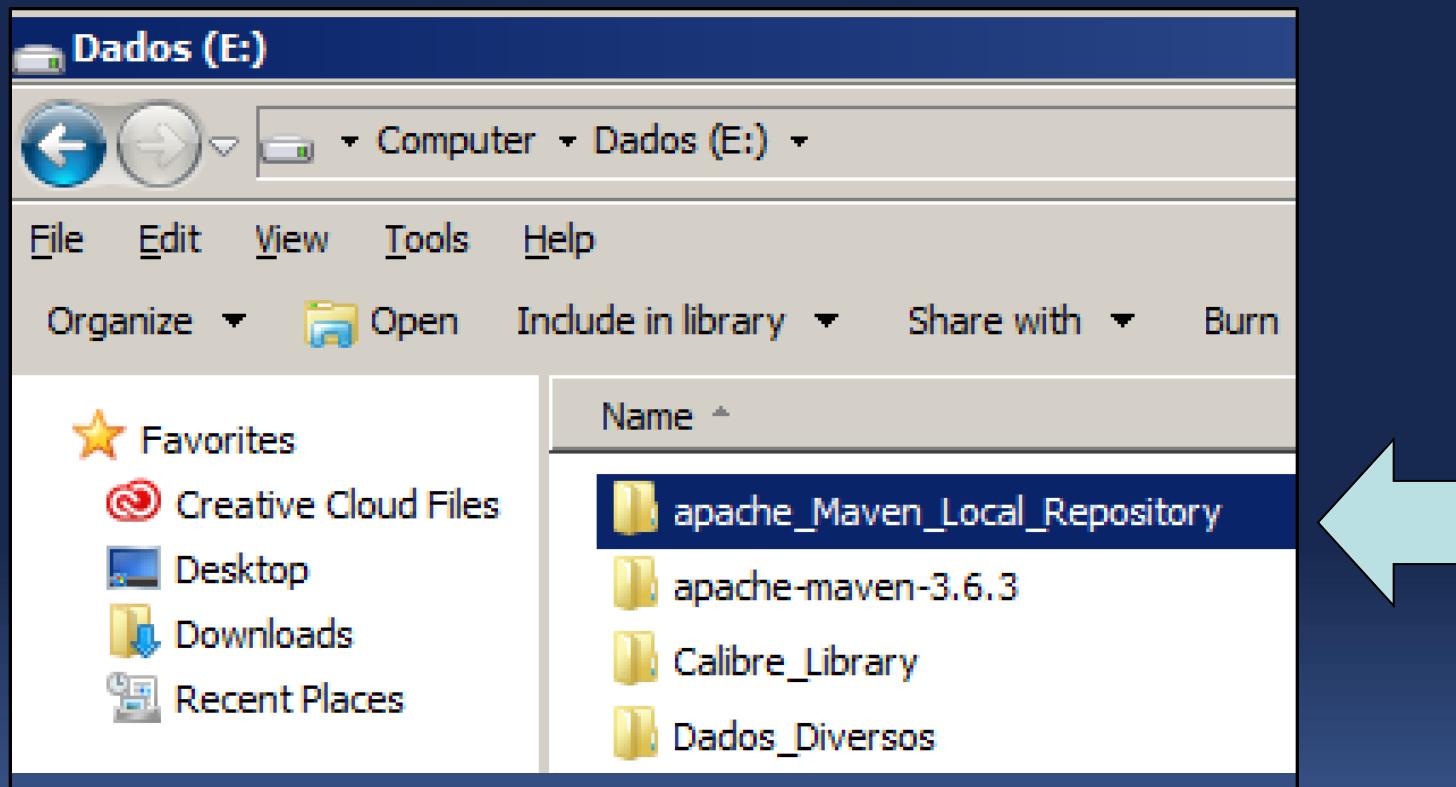


```
<!-- E:\apache_Maven_Local_Repository--><localRepository>
```



Exemplo de Geração – passo 2

4. Vamos então criar uma pasta para configurar o repositório local do Maven (C:\apache_Maven_Local_Repository).



Exemplo de Geração – passo 2

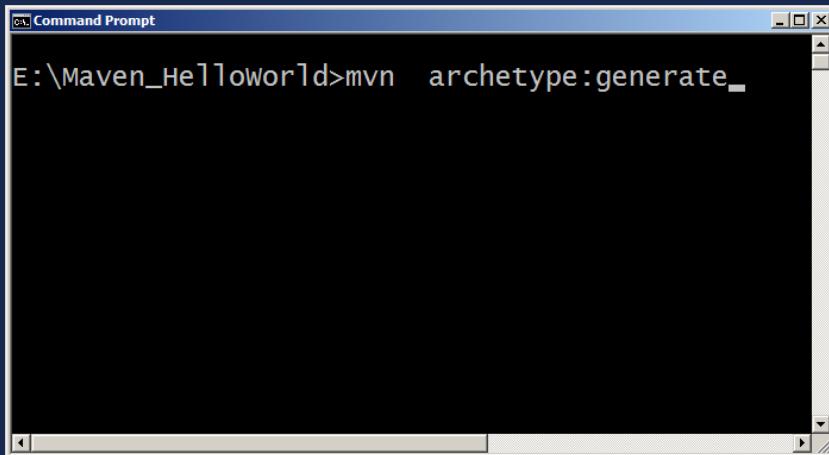
5. Na pasta conf do diretório de instalação do Maven, alterar o arquivo settings.xml informando a pasta que será usada como Repositório Local.

```
<!-- localRepository  
| The path to the local repository maven will use to store artifacts.  
|  
| Default: ${user.home}/.m2/repository  
<localRepository>/path/to/local/repo</localRepository>  
-->  
  
<localRepository>E:\apache_Maven_Local_Repository</localRepository>
```



Exemplo de Geração – passo 3

1. Entre na pasta do projeto recentemente criada:



2. Vamos agora criar o nosso primeiro projeto Maven pelo terminal;
Archetypes são templates de projetos pré-configurados.
3. Entre com o comando:

\$ mvn archetype:generate



Exemplo de Geração – passo 3

4. Maven irá iniciar o processo de download.

```
Command Prompt - mvn archetype:generate
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-install-plugin/2.4/maven-install-plugin-2.4.jar (27 kB at 71 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.pom (5.6 kB at 16 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-deploy-plugin/2.7/maven-deploy-plugin-2.7.jar (27 kB at 76 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-site-plugin/3.3/maven-site-plugin-3.3.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-site-plugin/3.3/maven-site-plugin-3.3.pom (21 kB at 54 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/24/maven-plugins-24.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/24/maven-plugins-24.pom (11 kB at 30 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven-parent/23/maven-parent-23.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven-parent/23/maven-parent-23.pom (33 kB at 91 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/apache/13/apache-13.pom
```



Exemplo de Geração – passo 3

5. Há uma lista muito grande de archetypes previamente definidos no Maven; Teclar <enter> para aceitar o default (1593).

```
C:\ Command Prompt - mvn archetype:generate
2675: remote -> uk.ac.ebi.gxa:atlas-archetype (Archetype for generating a custom
Atlas webapp)
2676: remote -> uk.ac.gate:gate-plugin-archetype (Maven archetype to create a new
GATE plugin project.)
2677: remote -> uk.ac.gate:gate-pr-archetype (Maven archetype to create a new GA
TE plugin project including a sample PR class (an empty LanguageAnalyser).)
2678: remote -> uk.ac.nactem.argo:argo-analysis-engine-archetype (An archetype w
hich contains a sample Argo (UIMA) Analysis Engine)
2679: remote -> uk.ac.nactem.argo:argo-reader-archetype (An archetype which cont
ains a sample Argo (UIMA) Reader)
2680: remote -> uk.ac.rdg.resc:edal-ncwms-based-webapp (-)
2681: remote -> uk.co.nemstix:basic-javaee7-archetype (A basic Java EE7 Maven ar
chetype)
2682: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RES
Tful spring services with an AngularJS frontend. Includes debian deployment)
2683: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)
2684: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)
2685: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)
2686: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)
2687: remote -> ws.osiris:osiris-archetype (Maven Archetype for Osiris)
2688: remote -> xyz.luan.generator:xyz-gae-generator (-)
2689: remote -> xyz.luan.generator:xyz-generator (-)
2690: remote -> za.co.absa.hyperdrive:component-archetype (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive co
ntains): 1593:
```



Exemplo de Geração – passo 3

6. Teclar <enter> para quickstart version (8)

```
Command Prompt - mvn archetype:generate
2680: remote -> uk.ac.rdg.resc:edal-ncwms-based-webapp (-)
2681: remote -> uk.co.nemstix:basic-javaee7-archetype (A basic Java EE7 Maven ar-
chetype)
2682: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RES-
Tful spring services with an AngularJS frontend. Includes debian deployment)
2683: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)
2684: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)
2685: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)
2686: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)
2687: remote -> ws.osiris:osiris-archetype (Maven Archetype for osiris)
2688: remote -> xyz.luan.generator:xyz-gae-generator (-)
2689: remote -> xyz.luan.generator:xyz-generator (-)
2690: remote -> za.co.absa.hyperdrive:component-archetype (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive co-
ntains): 1593:
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:
1: 1.0-alpha-1
2: 1.0-alpha-2
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
Choose a number: 8:
```



Exemplo de Geração – passo 3

7. Entre com o groupId, que representa a Organização (br.uscs)

```
Command Prompt - mvn archetype:generate
1: 1.0-alpha-1
2: 1.0-alpha-2
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
Choose a number: 8:
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.6 kB at 2.0 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom (4.5 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.1 kB at 21 kB/s)
Define value for property 'groupId': br.uscs
```



Exemplo de Geração – passo 3

8. Entre com o artifactId, que representa a nossa aplicação. Entraremos com MavenHelloWorld.

```
C:\> Command Prompt - mvn archetype:generate
2: 1.0-alpha-2
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
Choose a number: 8:
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.6 kB at 2.0 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom (4.5 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.1 kB at 21 kB/s)
Define value for property 'groupId': br.uscs
Define value for property 'artifactId': MavenHelloWorld
```



Exemplo de Geração – passo 3

9. Tecle <Enter> para aceitar a versão: 1.0-SNAPSHOT

```
Command Prompt - mvn archetype:generate
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
Choose a number: 8:
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.6 kB at 2.0 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom (4.5 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.1 kB at 21 kB/s)
Define value for property 'groupId': br.uscs
Define value for property 'artifactId': MavenHelloWorld
Define value for property 'version' 1.0-SNAPSHOT: :
```



Exemplo de Geração – passo 3

10. Tecle <Enter> para aceitar o package definido (br.uscs)

```
Command Prompt - mvn archetype:generate
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
Choose a number: 8:
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.6 kB at 2.0 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom (4.5 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.1 kB at 21 kB/s)
Define value for property 'groupId': br.uscs
Define value for property 'artifactId': MavenHelloWorld
Define value for property 'version' 1.0-SNAPSHOT: :
Define value for property 'package' br.uscs: :
```



Exemplo de Geração – passo 3

11. Tecle <Enter> para aceitar as configurações (Y)

```
Command Prompt - mvn archetypegenerate
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.pom (1.6 kB at 2.0 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-bundles/1.4/maven-archetype-bundles-1.4.pom (4.5 kB at 14 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/archetypes/maven-archetype-quickstart/1.4/maven-archetype-quickstart-1.4.jar (7.1 kB at 21 kB/s)
Define value for property 'groupId': br.uscs
Define value for property 'artifactId': MavenHelloWorld
Define value for property 'version' 1.0-SNAPSHOT: :
Define value for property 'package' br.uscs: :
Confirm properties configuration:
groupId: br.uscs
artifactId: MavenHelloWorld
version: 1.0-SNAPSHOT
package: br.uscs
Y: :
```

Exemplo de Geração – passo 3

12. Processo de Build concluído (Projeto Criado)

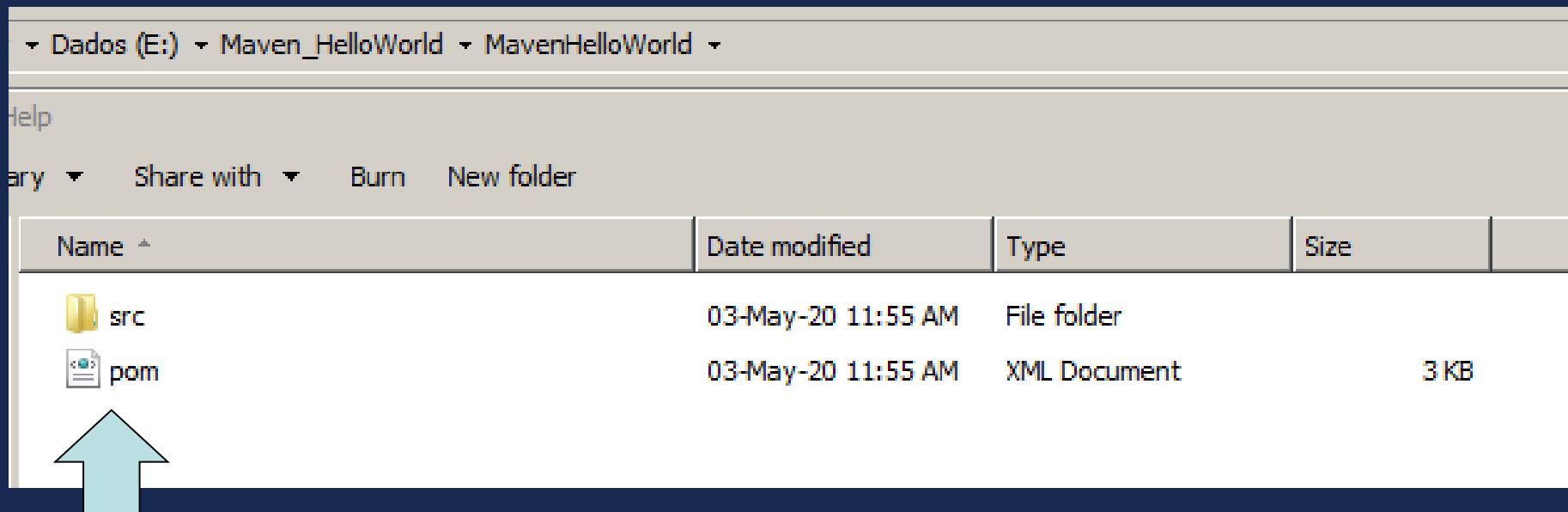
```
C:\> Command Prompt
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: maven-archetype-quickstart:1.4
[INFO] -----
[INFO] Parameter: groupId, value: br.uscs
[INFO] Parameter: artifactId, value: MavenHelloWorld
[INFO] Parameter: version, value: 1.0-SNAPSHOT
[INFO] Parameter: package, value: br.uscs
[INFO] Parameter: packageInPathFormat, value: br/uscs
[INFO] Parameter: package, value: br.uscs
[INFO] Parameter: version, value: 1.0-SNAPSHOT
[INFO] Parameter: groupId, value: br.uscs
[INFO] Parameter: artifactId, value: MavenHelloWorld
[INFO] Project created from Archetype in dir: E:\Maven_Helloworld\MavenHelloWorld
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 39:41 min
[INFO] Finished at: 2020-05-03T11:55:24-07:00
[INFO] -----
```

E:\Maven_Helloworld>



Exemplo de Geração – passo 3

13. Verificando o projeto criado . O arquivo POM.XML também foi criado.



Name	Date modified	Type	Size
src	03-May-20 11:55 AM	File folder	
pom	03-May-20 11:55 AM	XML Document	3 KB



Exemplo de Geração – passo 3

14. Consultando o arquivo POM.XML criado no projeto.



```
File Edit Selection View Run Terminal Help pom.xml - Visual Studio Code — □ ×
```

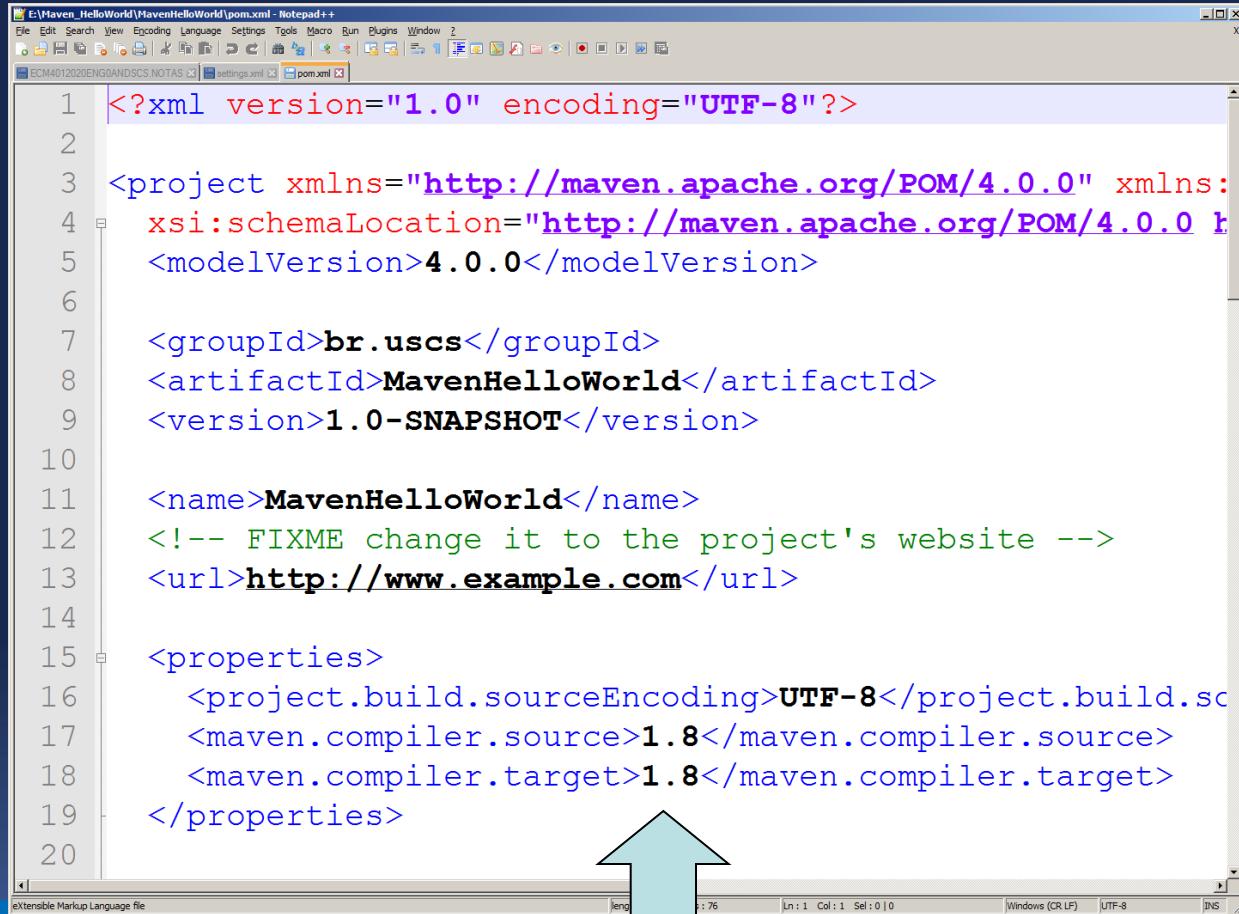
E: > Maven_HelloWorld > MavenHelloWorld > pom.xml

```
1<?xml version="1.0" encoding="UTF-8"?>
2
3<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
4    <modelVersion>4.0.0</modelVersion>
5
6    <groupId>br.uscs</groupId>
7    <artifactId>MavenHelloWorld</artifactId>
8    <version>1.0-SNAPSHOT</version>
9
10   <name>MavenHelloWorld</name>
11   <!-- FIXME change it to the project's website -->
12   <url>http://www.example.com</url>
13
14   <properties>
15       <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
16       <maven.compiler.source>1.7</maven.compiler.source>
17       <maven.compiler.target>1.7</maven.compiler.target>
18   </properties>
19
20   <dependencies>
21       <dependency>
22           <groupId>junit</groupId>
```

Ln 1, Col 1 Spaces: 2 UTF-8 CRLF XML ⚙️ 🔍

Exemplo de Geração – passo 3

15. Estamos trabalhando com Java 1.8, Vamos então atualizar o POM.XML gerado.



```
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>

  <groupId>br.uscs</groupId>
  <artifactId>MavenHelloWorld</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>MavenHelloWorld</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

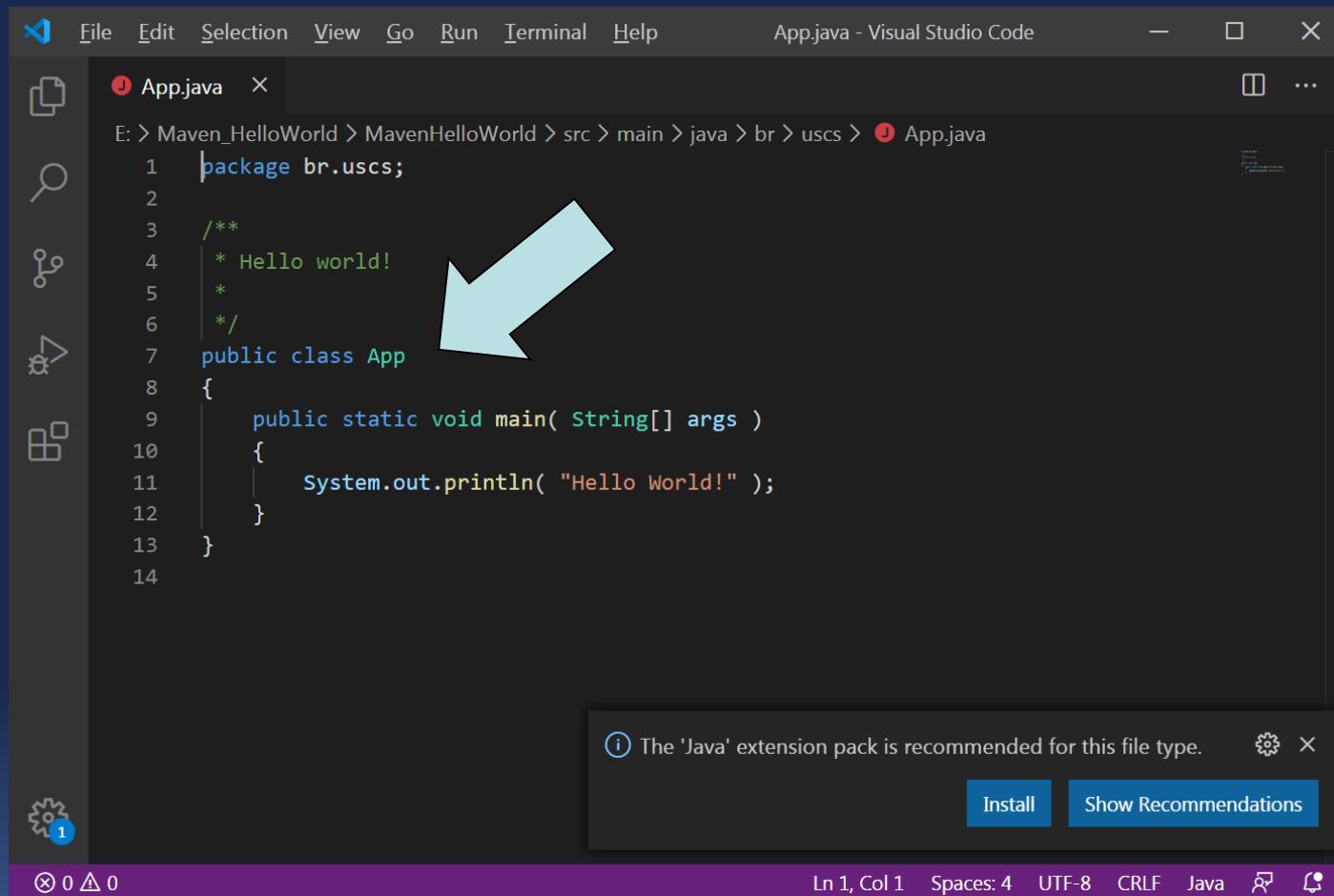
  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
  </properties>

```



Exemplo de Geração – passo 3

16. A classe App com o método main() foi criada.



```
File Edit Selection View Go Run Terminal Help App.java - Visual Studio Code
App.java ×
E: > Maven_HelloWorld > MavenHelloWorld > src > main > java > br > uscs > App.java
1 package br.uscs;
2
3 /**
4 * Hello world!
5 *
6 */
7 public class App
8 {
9     public static void main( String[] args )
10    {
11        System.out.println( "Hello World!" );
12    }
13 }
14
```

The 'Java' extension pack is recommended for this file type.
Install Show Recommendations

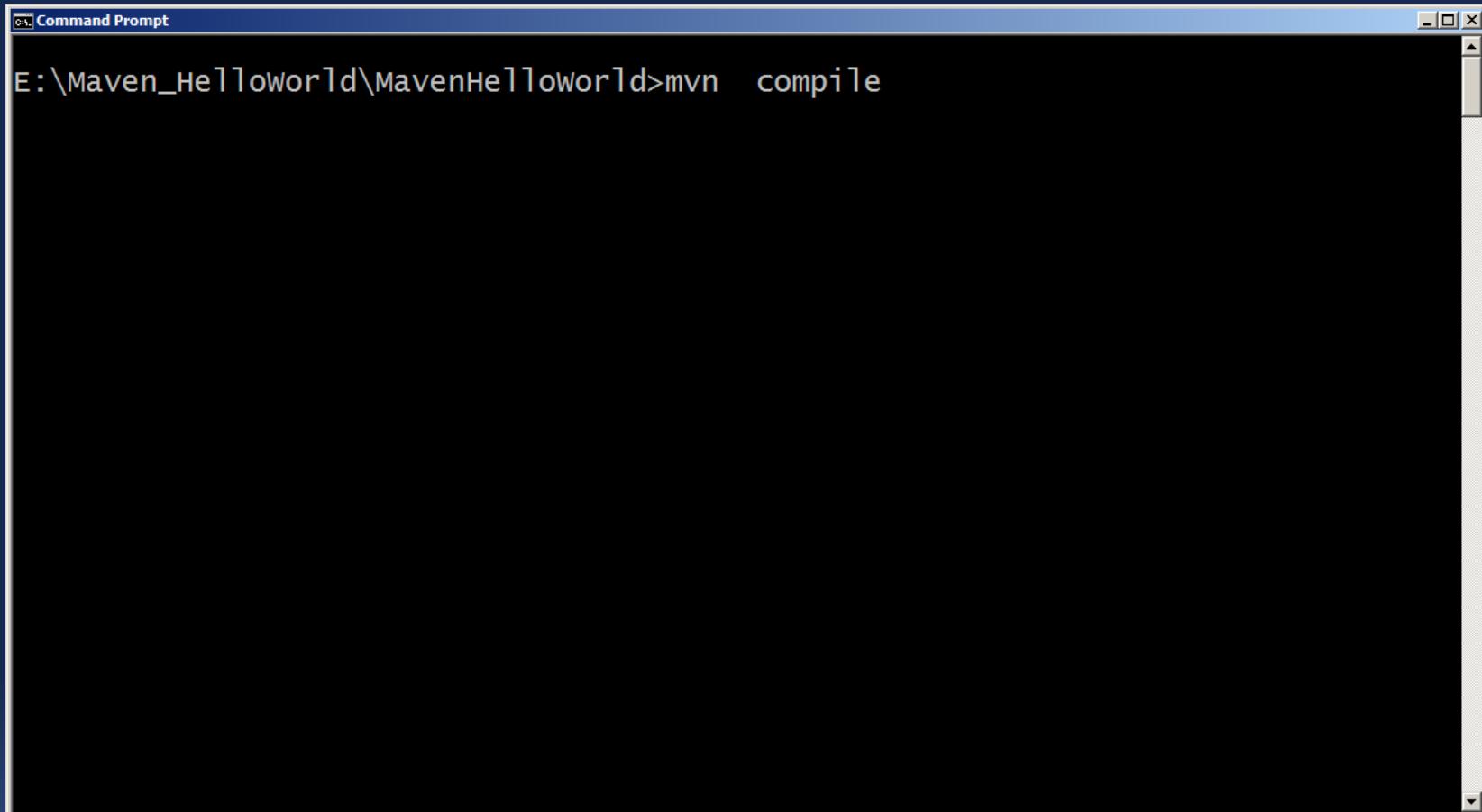
⊗ 0 ▲ 0

Ln 1, Col 1 Spaces: 4 UTF-8 CRLF Java ⚙️ 🔍



Exemplo de Geração – passo 4

1. Vamos compilar a aplicação: **\$ mvn compile** (na pasta da aplicação)

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window has a black background and white text. In the title bar, it says "C:\ Command Prompt". The main area shows the command: "E:\Maven_Helloworld\MavenHelloWorld>mvn compile". The window has standard window controls (minimize, maximize, close) at the top right.

Exemplo de Geração – passo 4

2. Serão feitos downloads dos módulos requisitados.

```
C:\ Command Prompt
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-javac/2.8.4/plexus-compiler-javac-2.8.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-manager/2.8.4/plexus-compiler-manager-2.8.4.jar (4.7 kB at 5.2 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-api/2.8.4/plexus-compiler-api-2.8.4.jar (27 kB at 29 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-compiler-javac/2.8.4/plexus-compiler-javac-2.8.4.jar (21 kB at 19 kB/s)

Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/2.0.4/plexus-utils-2.0.4.jar (222 kB at 184 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/qdox/qdox/2.0-M9/qdox-2.0-M9.jar (317 kB at 232 kB/s)
[INFO] Changes detected - recompiling the module!
[INFO] Compiling 1 source file to E:\Maven_Helloworld\MavenHelloWorld\target\classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  35.168 s
[INFO] Finished at: 2020-05-03T12:13:41-07:00
[INFO] -----
```

E:\Maven_Helloworld\MavenHelloWorld>



Exemplo de Geração – passo 4

3. Será também criada a pasta target com os classes geradas.

Name	Date modified	Type	Size
src	03-May-20 11:55 AM	File folder	
target	03-May-20 12:13 PM	File folder	
pom	03-May-20 12:08 PM	XML Document	3 KB



Exemplo de Geração – passo 4

4. Visualizando também o repositório Local

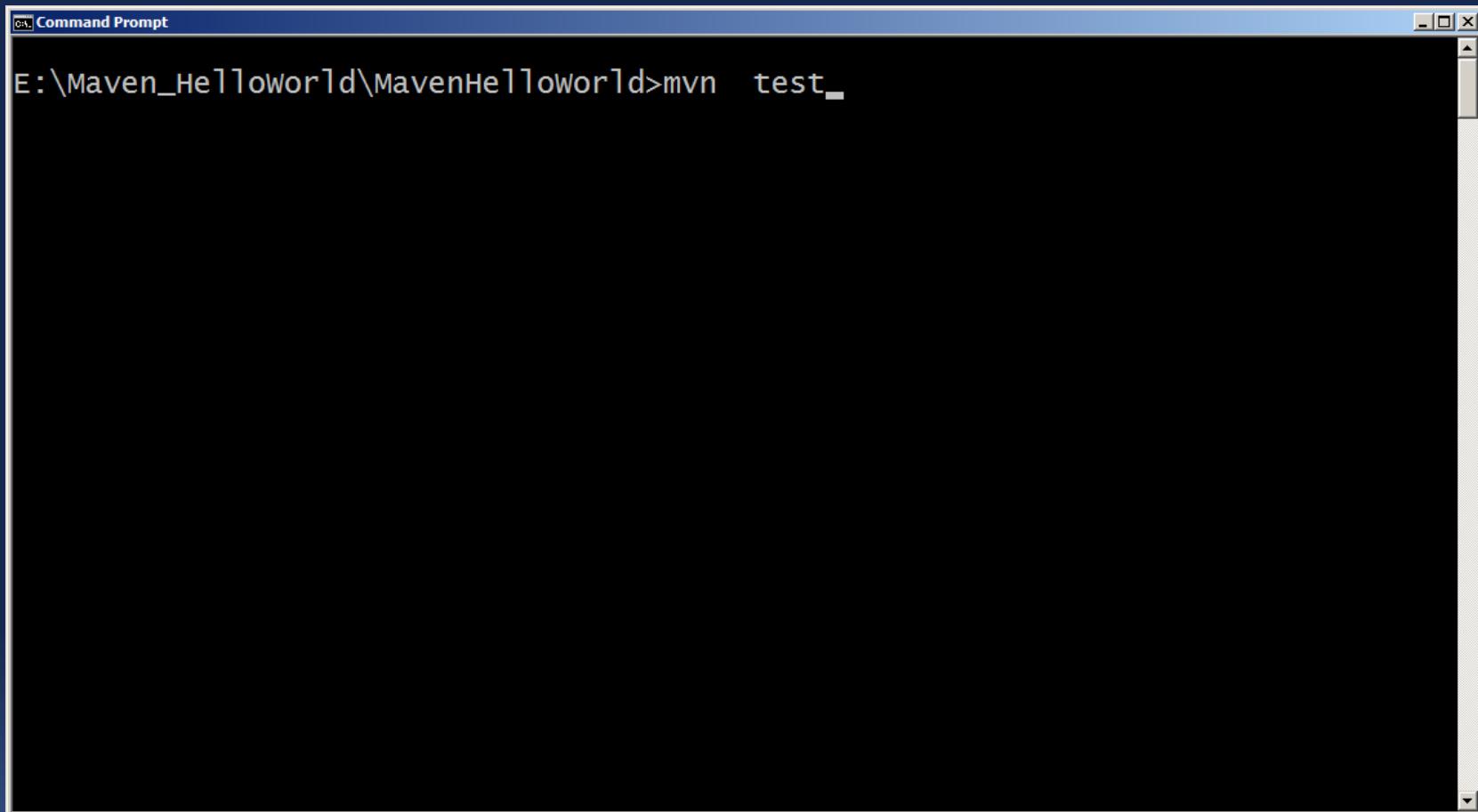


Dados (E:) \ apache_Maven_Local_Repository			
	Name	Date modified	Type
	antlr	03-May-20 11:16 AM	File folder
	asm	03-May-20 11:16 AM	File folder
	backport-util-concurrent	03-May-20 12:13 PM	File folder
	com	03-May-20 12:13 PM	File folder
	commons-codec	03-May-20 11:16 AM	File folder
	commons-collections	03-May-20 11:16 AM	File folder
	commons-io	03-May-20 11:16 AM	File folder
	commons-lang	03-May-20 11:16 AM	File folder
	jdom	03-May-20 11:16 AM	File folder
	junit	03-May-20 12:13 PM	File folder
	net	03-May-20 11:16 AM	File folder
	org	03-May-20 12:13 PM	File folder



Exemplo de Geração – passo 5

1. Gerando ambiente de teste: **\$ mvn test** (na pasta do projeto)

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the command "mvn test" being typed at the prompt. The path "E:\Maven_Helloworld\MavenHelloWorld>" is visible before the command. The rest of the window is blank, indicating the command has not yet been executed.

Exemplo de Geração – passo 5

2. Efetuando o BUILD de teste

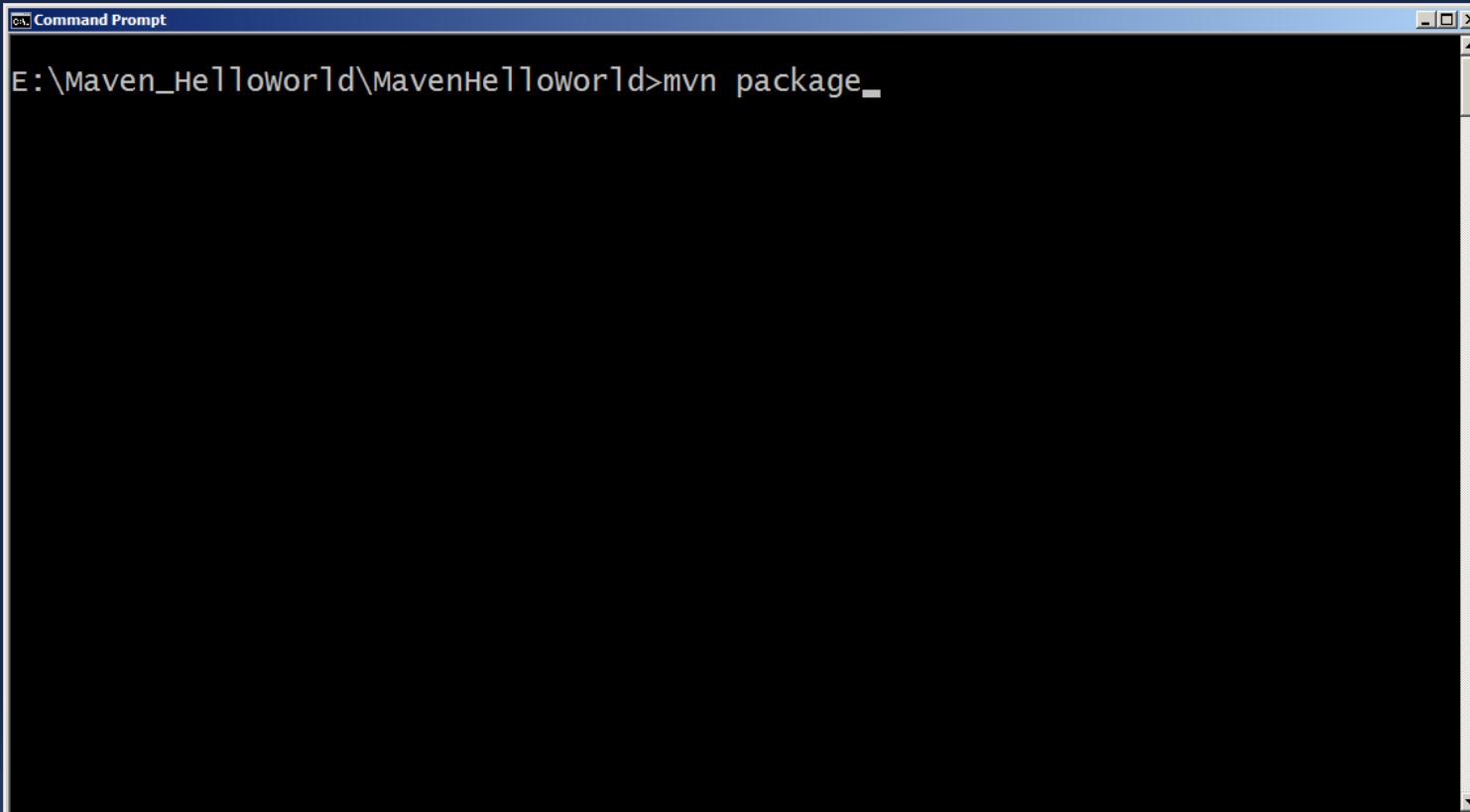
```
C:\ Command Prompt
B/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/
surefire/surefire-junit4/2.22.1/surefire-junit4-2.22.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/s
urefire/surefire-junit4/2.22.1/surefire-junit4-2.22.1.jar (85 kB at 221 kB/s)
[INFO] -----
[INFO] T E S T S
[INFO] -----
[INFO] Running br.uscs.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.038 s -
in br.uscs.AppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 22.914 s
[INFO] Finished at: 2020-05-03T12:21:03-07:00
[INFO] -----
```

E:\Maven_Helloworld\MavenHelloWorld>



Exemplo de Geração – passo 6

1. Empacotando a aplicação: **mvn package** (geração .jar)

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the command "E:\Maven_Helloworld\MavenHelloWorld>mvn package" entered at the prompt. The rest of the window is blank, indicating no output has been generated yet.

Exemplo de Geração – passo 6

2. .jar gerado

```
Command Prompt
nappy/0.4/snappy-0.4.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/iq80/snappy/snappy/0.4/snappy-0.4.jar (58 kB at 46 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/tukaani/xz/1.5/xz-1.5.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/2.7.1/plexus-io-2.7.1.jar (86 kB at 57 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-shared-utils/3.0.1/maven-shared-utils-3.0.1.jar (154 kB at 81 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/tukaani/xz/1.5/xz-1.5.jar (100 kB at 47 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/3.4/plexus-archiver-3.4.jar (187 kB at 78 kB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.11/commons-compress-1.11.jar (426 kB at 164 kB/s)
[INFO] Building jar: E:\Maven_Helloworld\MavenHelloWorld\target\MavenHelloWorld-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 10.666 s
[INFO] Finished at: 2020-05-03T12:25:24-07:00
[INFO] -----
```

E:\Maven_Helloworld\MavenHelloWorld>



Exemplo de Geração – passo 6

3. Verificando na pasta target o .jar criado

```
Command Prompt
1 File(s)          2,736 bytes
4 Dir(s)  365,494,022,144 bytes free

E:\Maven_Helloworld\MavenHelloWorld>cd target

E:\Maven_Helloworld\MavenHelloWorld\target>dir
volume in drive E is Dados
volume Serial Number is 14D9-25F0

Directory of E:\Maven_Helloworld\MavenHelloWorld\target

03-May-20  12:25 PM    <DIR>      .
03-May-20  12:25 PM    <DIR>      ..
03-May-20  12:13 PM    <DIR>      classes
03-May-20  12:13 PM    <DIR>      generated-sources
03-May-20  12:20 PM    <DIR>      generated-test-sources
03-May-20  12:25 PM    <DIR>      maven-archiver
03-May-20  12:13 PM    <DIR>      maven-status
03-May-20  12:25 PM          2,585 MavenHelloWorld-1.0-SNAPSHOT.jar
03-May-20  12:21 PM    <DIR>      surefire-reports
03-May-20  12:20 PM    <DIR>      test-classes
               1 File(s)          2,585 bytes
               9 Dir(s)  365,494,022,144 bytes free

E:\Maven_Helloworld\MavenHelloWorld\target>
```

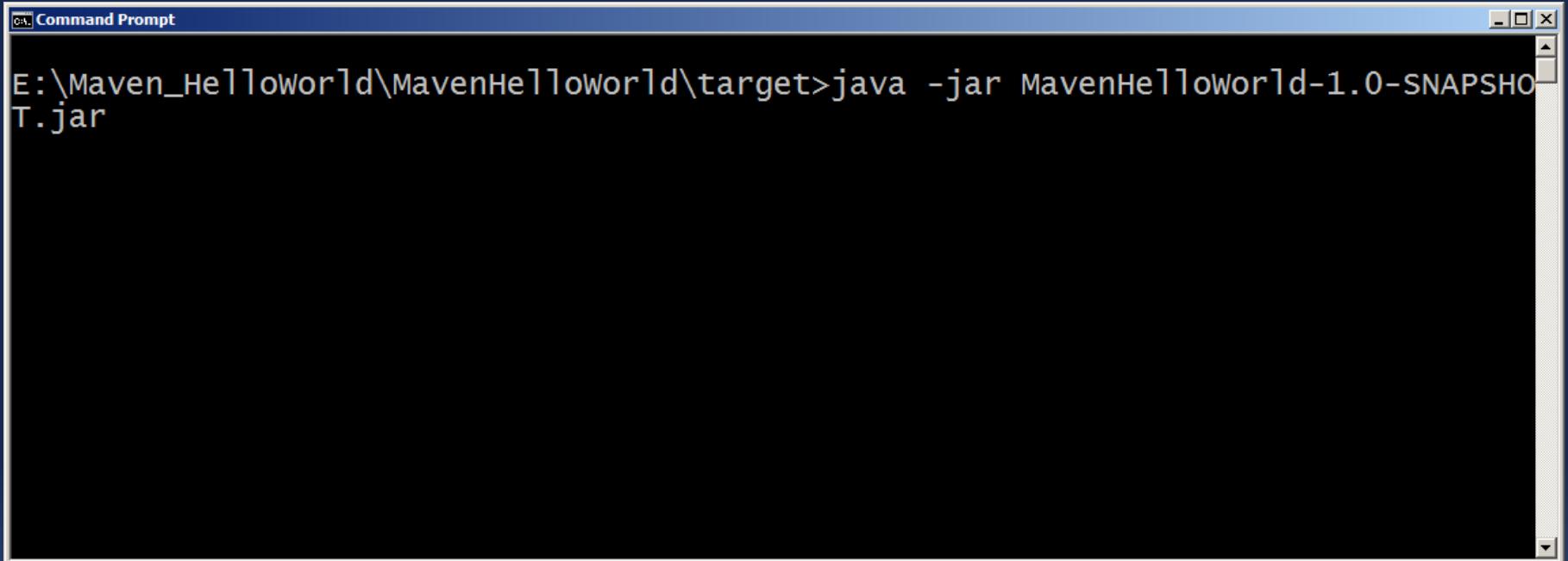


Exemplo de Geração – passo 7



1. Na pasta target do projeto vamos executar o .jar

Comando : **java -jar MavenHelloWorld-1.0-SNAPSHOT.jar**

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the command "E:\Maven_Helloworld\MavenHelloWorld\target>java -jar MavenHelloWorld-1.0-SNAPSHOT.jar" being typed. The window has a standard blue title bar and a black body with white text.

Exemplo de Geração – passo 7



2. Na hora da execução o interpretador Java retorna erro informando que não encontrou o arquivo com atributos Manifest



A screenshot of a Windows Command Prompt window titled "Command Prompt". The path shown is "E:\Maven_Helloworld\MavenHelloWorld\target>". The user has typed "java -jar MavenHelloWorld-1.0-SNAPSHOT.jar" and is receiving an error message: "no main manifest attribute, in MavenHelloWorld-1.0-SNAPSHOT.jar". The text "E:\Maven_Helloworld\MavenHelloWorld\target>" is highlighted with a red house-shaped selection tool.

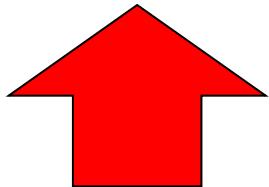
```
E:\Maven_Helloworld\MavenHelloWorld\target>java -jar MavenHelloWorld-1.0-SNAPSHOT.jar
no main manifest attribute, in MavenHelloWorld-1.0-SNAPSHOT.jar
E:\Maven_Helloworld\MavenHelloWorld\target>-
```



Exemplo de Geração – passo 8

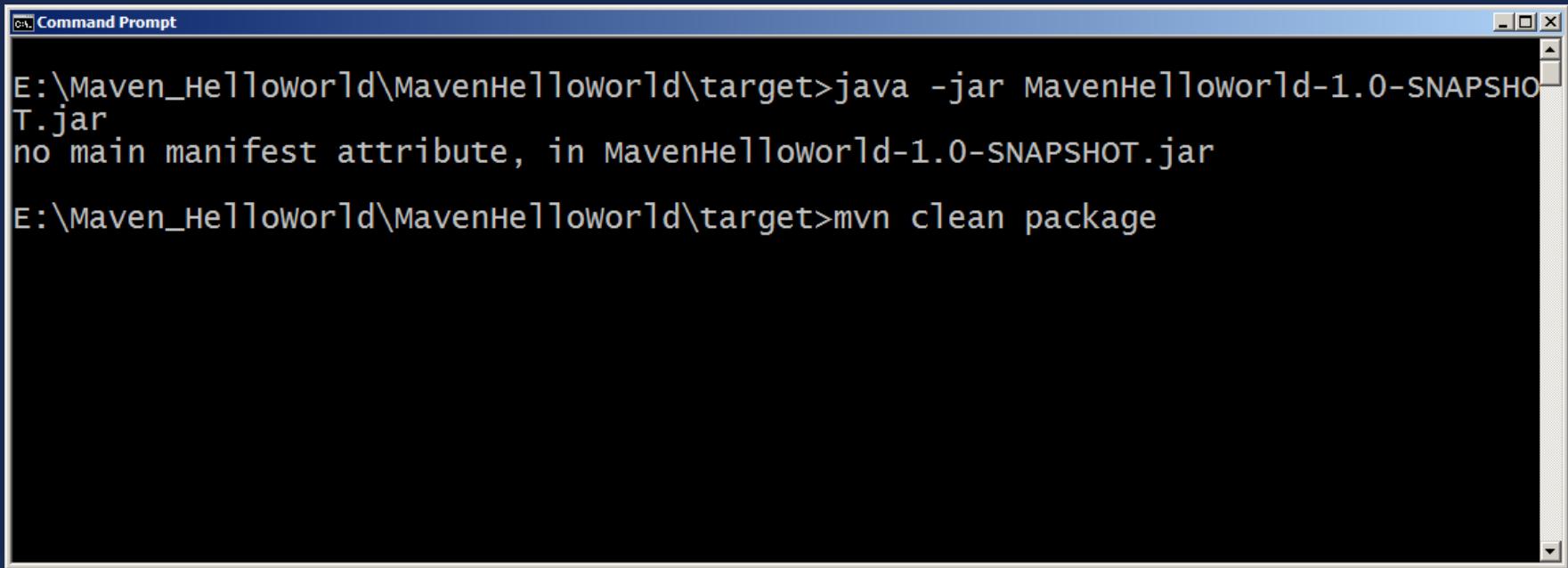
1. Vamos incluir no POM.XML o código XML para gerar o Manifest.

```
→ <plugin>
    <!-- Build an executable JAR -->
    <groupId>org.apache.maven.plugins</groupId>
    <artifactId>maven-jar-plugin</artifactId>
    <version>2.4</version>
    <configuration>
        <archive>
            <manifest>
                <mainClass>br.uscs.App</mainClass>
            </manifest>
        </archive>
    </configuration>
</plugin>
```



Exemplo de Geração – passo 9

1. Vamos gerar um novo build da aplicação : mvn clean package

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the following Maven commands and their outputs:

```
E:\Maven_Helloworld\MavenHelloWorld\target>java -jar MavenHelloWorld-1.0-SNAPSHOT.jar
no main manifest attribute, in MavenHelloWorld-1.0-SNAPSHOT.jar

E:\Maven_Helloworld\MavenHelloWorld\target>mvn clean package
```

The window has a standard Windows title bar and a scroll bar on the right side.

Exemplo de Geração – passo 9

2. Build recriado...

```
Command Prompt
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ MavenHelloWorld ---
[INFO] Building jar: E:\Maven_Helloworld\MavenHelloWorld\target\MavenHelloWorld-
1.0-SNAPSHOT.jar
[INFO] -
[INFO] BUILD SUCCESS
[INFO] -
[INFO] Total time: 3.207 s
[INFO] Finished at: 2020-05-03T12:44:35-07:00
[INFO] -
```

E:\Maven_Helloworld\MavenHelloWorld>



Exemplo de Geração – passo 9

3. Executando o arquivo .jar criado, na pasta target :

Comando : `java -jar MavenHelloWorld-1.0-SNAPSHOT.jar`



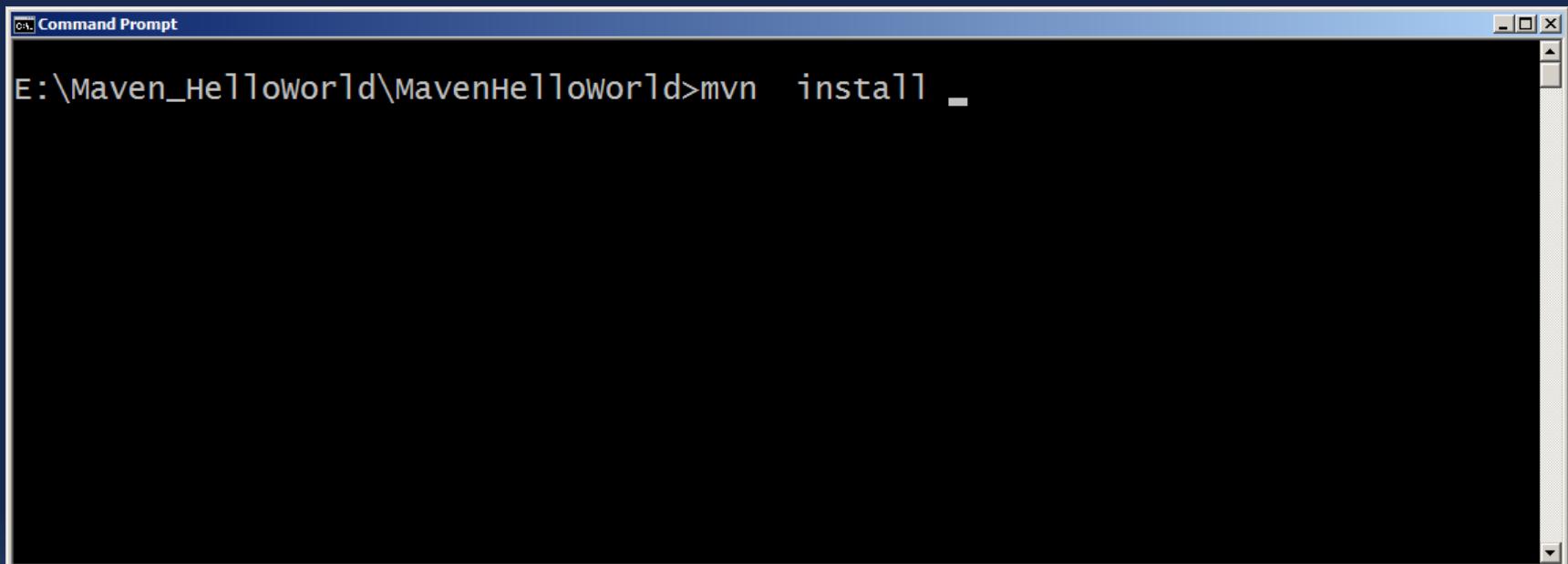
```
Command Prompt  
E:\Maven_Helloworld\MavenHelloWorld\target>java -jar MavenHelloWorld-1.0-SNAPSHOT.jar  
Hello world!  
E:\Maven_Helloworld\MavenHelloWorld\target>
```



Exemplo de Geração – passo 10

1. Gerando a aplicação (instalando) no Repositório Local

Comando : mvn install (na pasta do projeto)

A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the command "E:\Maven_Helloworld\MavenHelloWorld>mvn install -" entered by the user. The rest of the window is blank, indicating that the command has not yet been executed or is still running.

Exemplo de Geração – passo 10

2. Build concluído

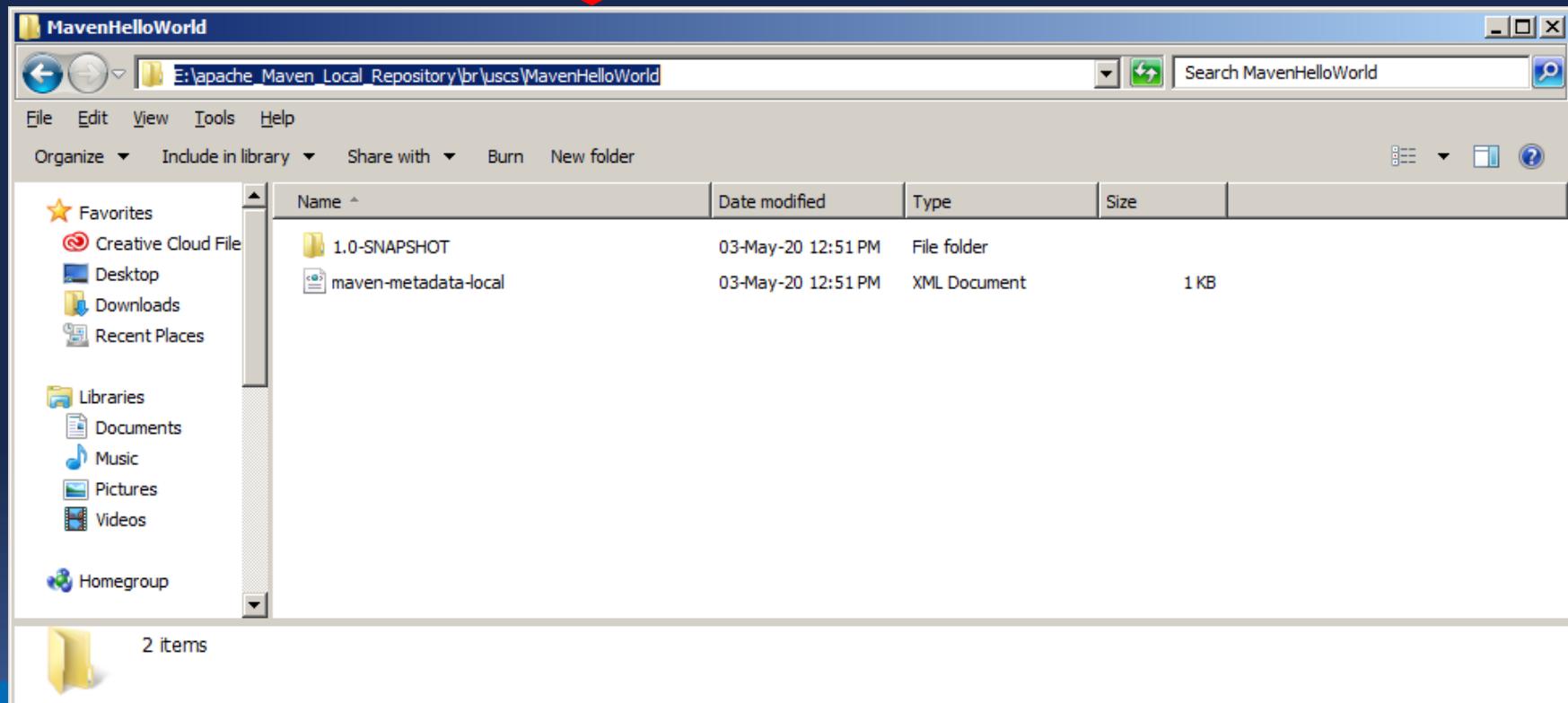
```
Command Prompt
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/3.0.15/plexus-utils-3.0.15.jar (239 kB at 184 kB/s)
[INFO] Installing E:\Maven_Helloworld\MavenHelloWorld\target\MavenHelloWorld-1.0-SNAPSHOT.jar to E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT\MavenHelloWorld-1.0-SNAPSHOT.jar
[INFO] Installing E:\Maven_Helloworld\MavenHelloWorld\pom.xml to E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT\MavenHelloWorld-1.0-SNAPSHOT.pom
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 8.682 s
[INFO] Finished at: 2020-05-03T12:51:48-07:00
[INFO] -----
```

E:\Maven_Helloworld\MavenHelloWorld>



Exemplo de Geração – passo 10

3. Checando a aplicação no Repositório Local



MavenHelloWorld

E:\apache Maven Local Repository\br\uscs\MavenHelloWorld

File Edit View Tools Help

Organize Include in library Share with Burn New folder

Name	Date modified	Type	Size
1.0-SNAPSHOT	03-May-20 12:51 PM	File folder	
maven-metadata-local	03-May-20 12:51 PM	XML Document	1 KB

Favorites

- Creative Cloud File
- Desktop
- Downloads
- Recent Places

Libraries

- Documents
- Music
- Pictures
- Videos

Homegroup

2 items

Exemplo de Geração – passo 10

4. Processando a aplicação a partir do Repositório Local



```
Windows Command Prompt  
E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT>java -jar  
MavenHelloWorld-1.0-SNAPSHOT.jar
```



Exemplo de Geração – passo 10

5. Processando a aplicação a partir do Repositório Local

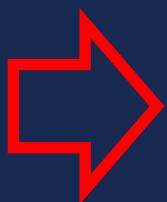


```
Command Prompt  
E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT>java -jar  
MavenHelloWorld-1.0-SNAPSHOT.jar  
Hello World!  
E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT>
```



Exemplo de Geração – passo 10

5. Processando a aplicação a partir do Repositório Local



```
Command Prompt  
E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT>java -jar  
MavenHelloWorld-1.0-SNAPSHOT.jar  
Hello world!  
E:\apache_Maven_Local_Repository\br\uscs\MavenHelloWorld\1.0-SNAPSHOT>
```



Integração do Maven com Eclipse



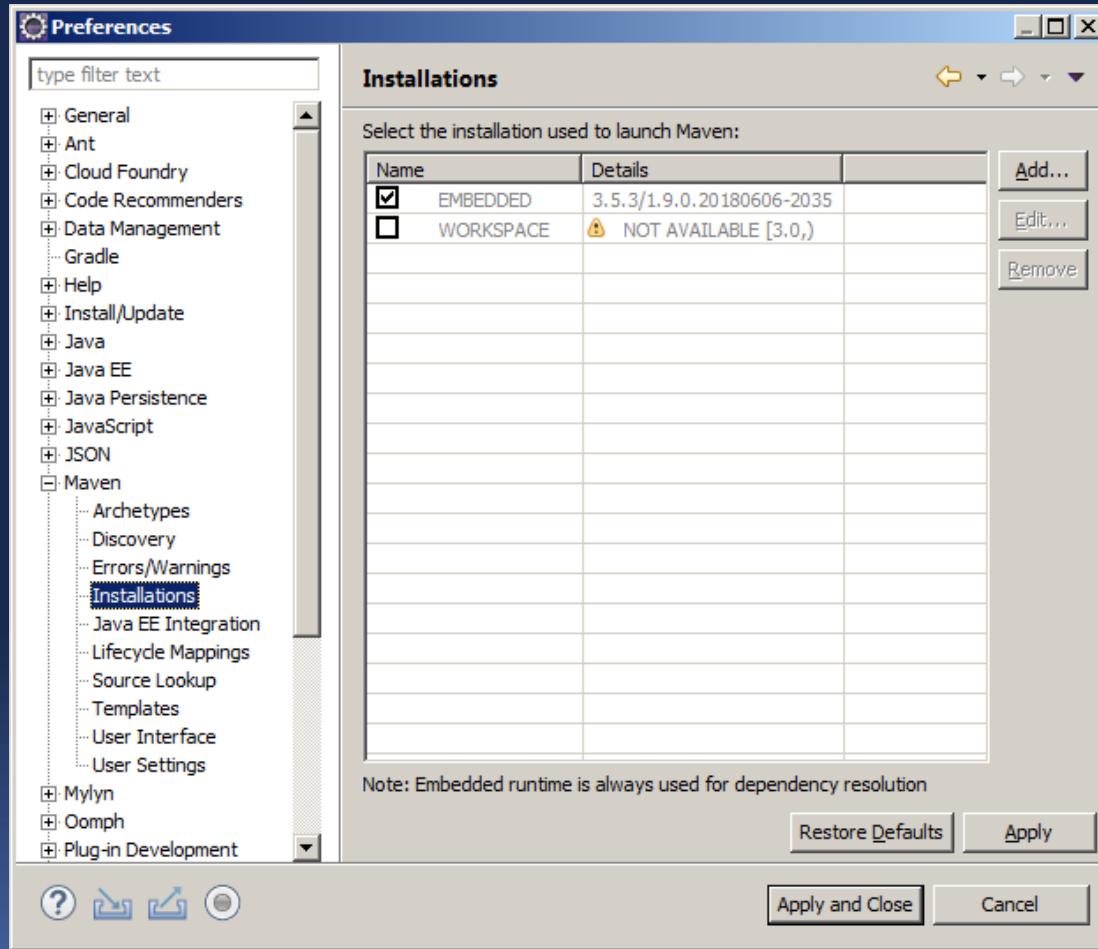
Integração do Maven com Eclipse

- ✓ Para a integração do **Maven** com o **Eclipse** recomenda-se utilizar o plugin **M2E**;
- ✓ **M2E** é um plugin do Eclipse que faz a integração com o **Maven**;
- ✓ A distribuição **Eclipse for JEE Developers** já vem com o plugin **M2E** e uma instalação interna do **Maven**;
- ✓ Caso você esteja com uma versão diferente do Eclipse, use o menu Help > Eclipse Marketplace ... , pesquise por **M2E** e instale o plugin;



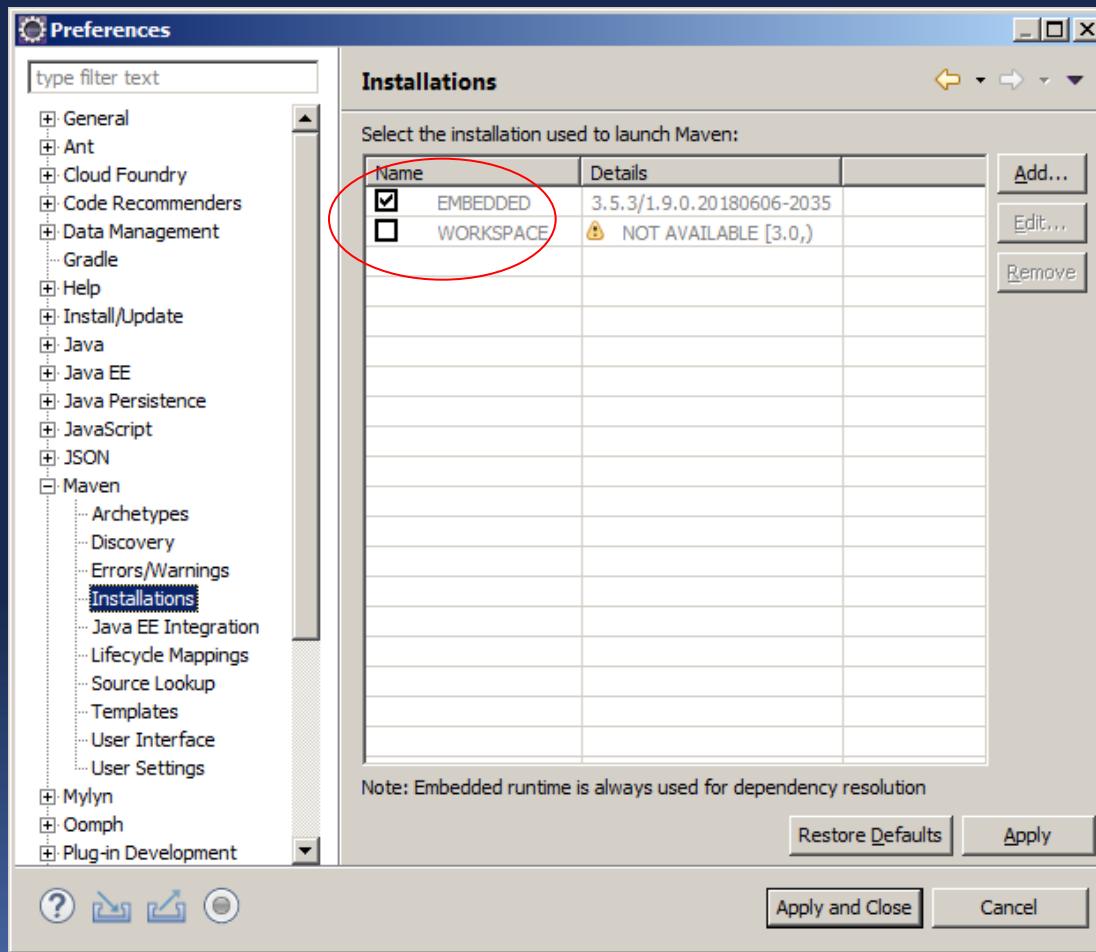
Integração do Maven com Eclipse

- ✓ Com o plugin **M2E** instalado e o Eclipse aberto, acesse o menu **Window > Preferences**;
- ✓ Vá para a opção **Maven > Installations**.



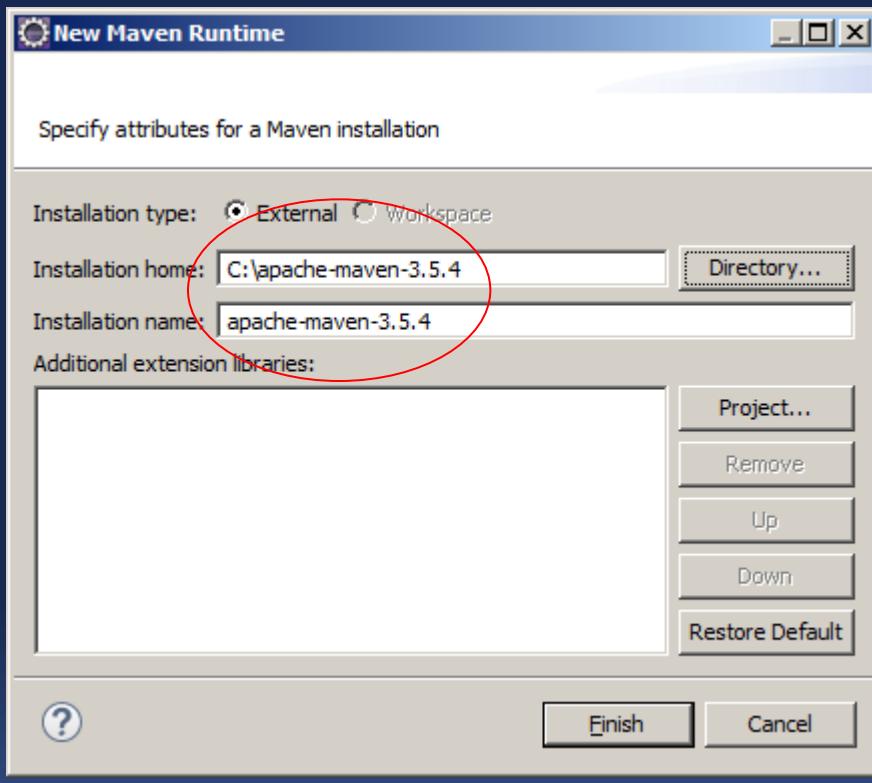
Integração do Maven com Eclipse

- ✓ Observe que já existe uma instalação “**embarcada**”, mas com uma versão desatualizada;
- ✓ Vamos adicionar o nosso **Maven** recentemente instalado;
- ✓ Clique em **Add...**



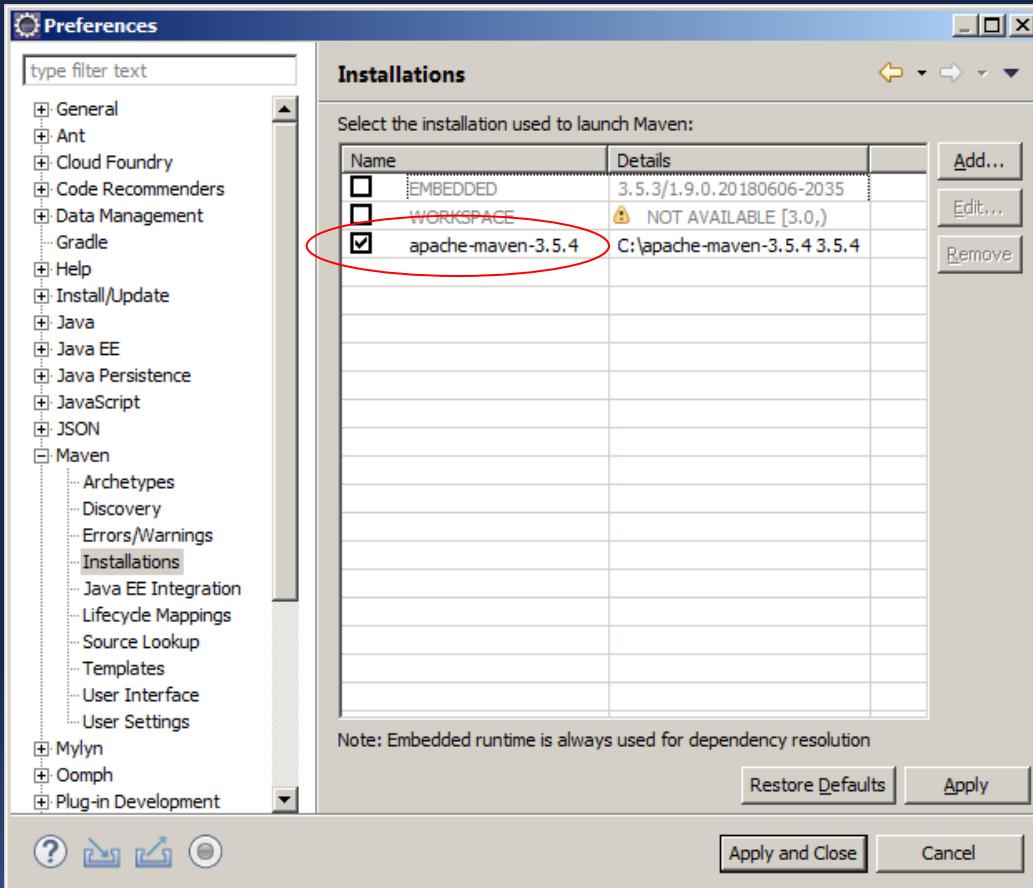
Integração do Maven com Eclipse

- ✓ Selecione a pasta com a nossa instalação e tecle **Finish**.



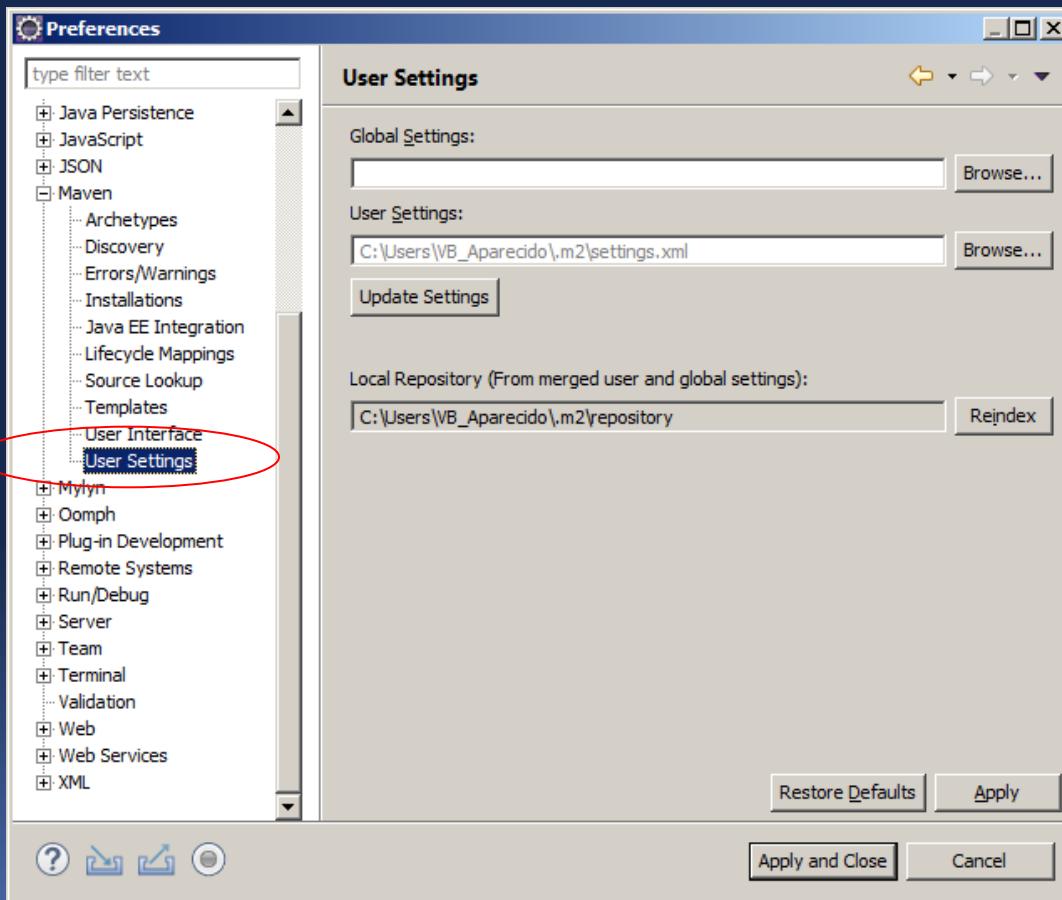
Integração do Maven com Eclipse

- ✓ Marque a nossa instalação e tecle **Apply and Close**.



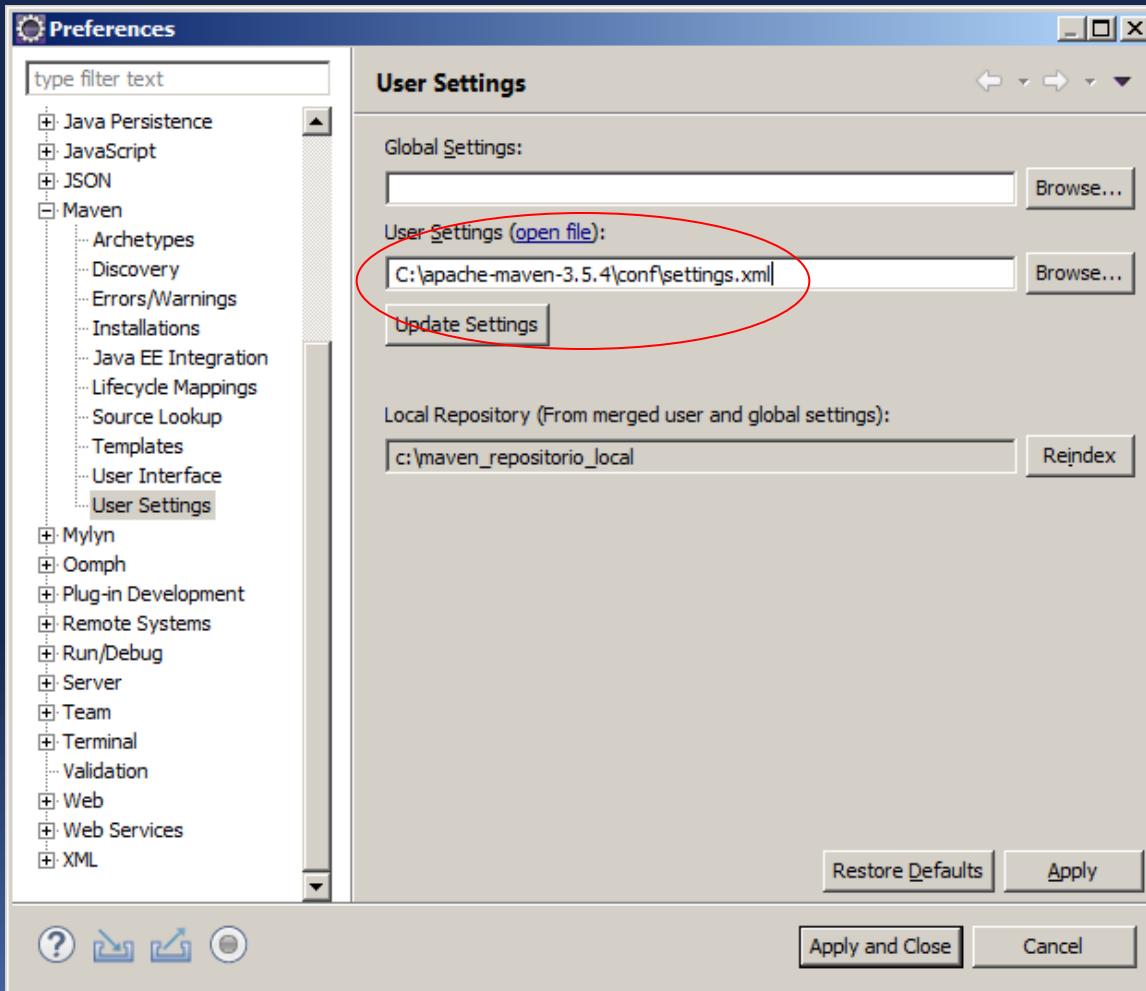
Integração do Maven com Eclipse

- ✓ Agora vamos atualizar o arquivo de configuração do **Maven**;
- ✓ Vá ao menu **User Settings** e defina o arquivo de configuração existente no diretório onde o **Maven** foi instalado.



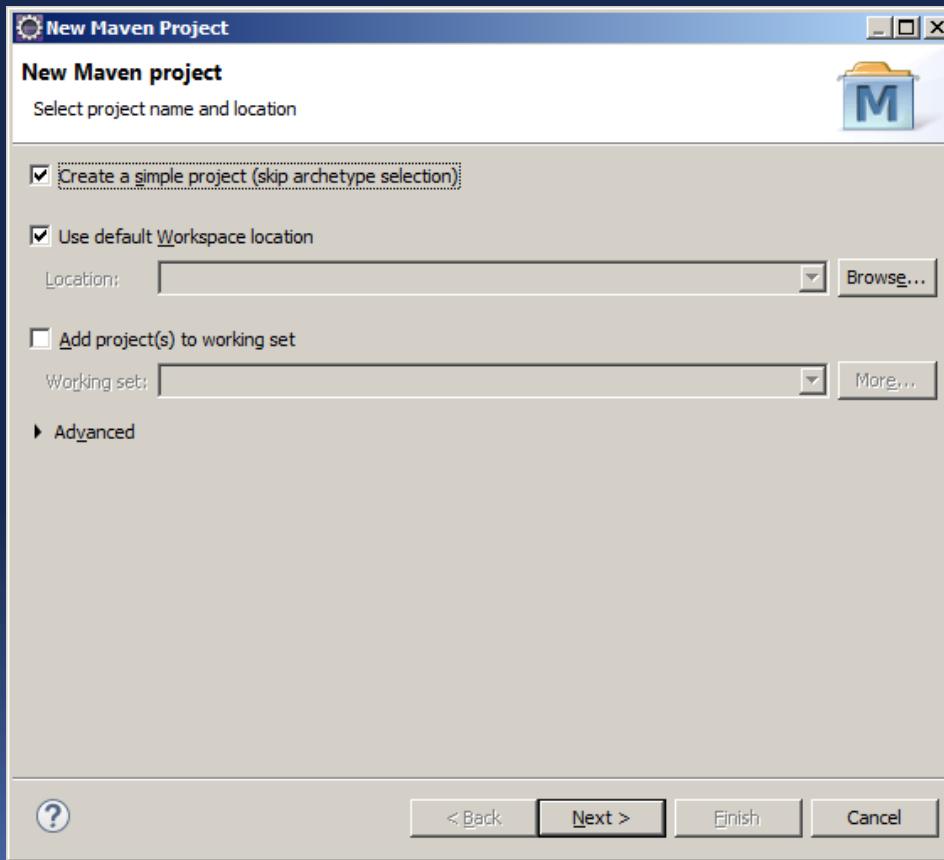
Integração do Maven com Eclipse

- ✓ Após a definição do arquivo de configuração, tecle em **Update Settings**;
- ✓ Em seguida, tecle em **Apply and Close**.



Criação de um Projeto Maven no Eclipse

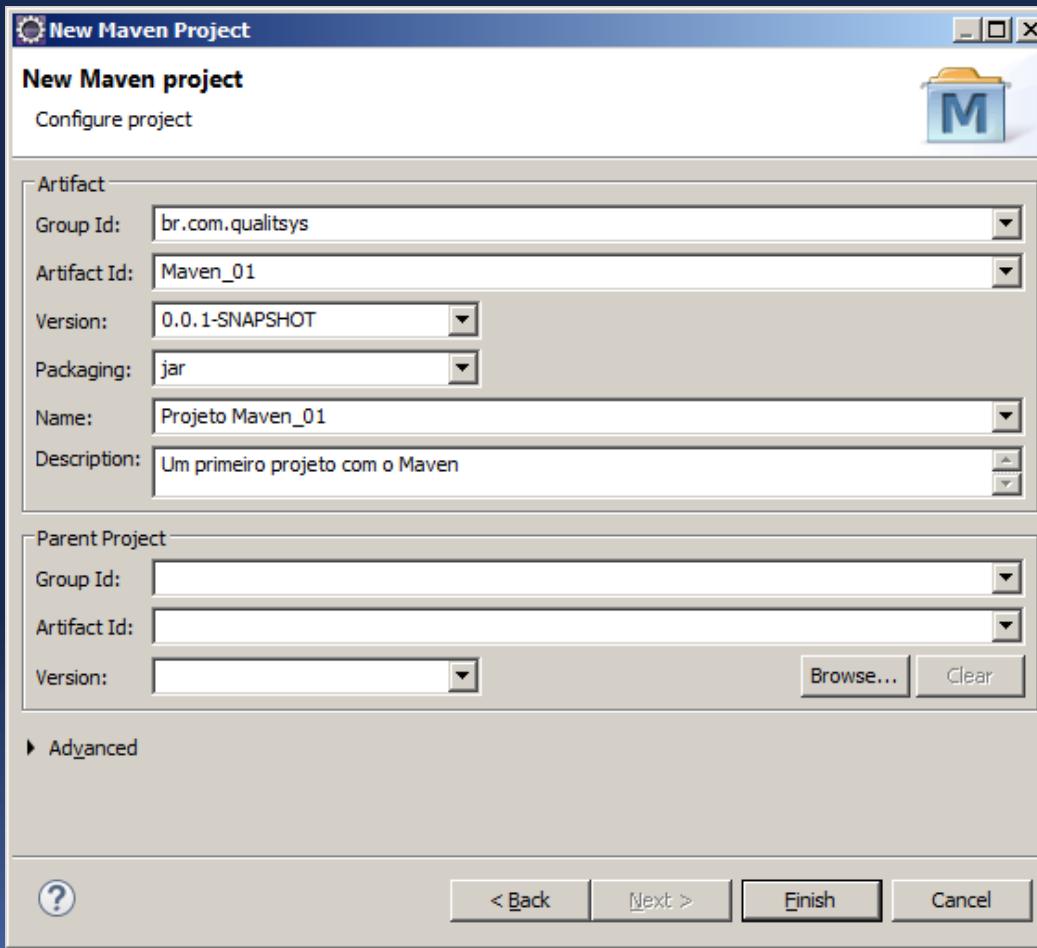
- ✓ Com o Eclipse configurado, pode-se criar um **novo projeto** no Eclipse;
- ✓ Acesse o menu **File > New > Maven Project**.
- ✓ Selecione a opção **Create a simple Project (skip archetypem selection)**.
- ✓ Clique em **Next >**.



Criação de um Projeto Maven no Eclipse



- ✓ **Group Id** representa a organização onde o projeto será desenvolvido;
- ✓ **Artifact Id** corresponde ao projeto que será desenvolvido;
- ✓ **Version** e **Packaging** já deverão estar preenchidos;
- ✓ Clique em **Finish**.



Criação de um Projeto Maven no Eclipse



✓ Projeto Maven criado!

A screenshot of the Eclipse IDE interface. The title bar says "Workspace - Eclipse IDE". The menu bar includes File, Edit, Navigate, Search, Project, Run, Window, Help. The toolbar has various icons for file operations. The "Project Explorer" view on the left shows a Maven project named "Maven_01" with its directory structure: src/main/java, src/main/resources, src/test/java, src/test/resources, JRE System Library [J2SE-1.5], src/main/java, src/main/resources, src/test/java, src/test/resources, target, and pom.xml. The "Outline" view on the right displays the message "An outline is not available." The "Markers" view at the bottom shows 0 errors, 2 warnings, and 0 others. A table lists two items under "Description": "Java Build Path Problems (1 item)" and "Java Problems (1 item)". The status bar at the bottom left says "0 items selected".

An outline is not available.

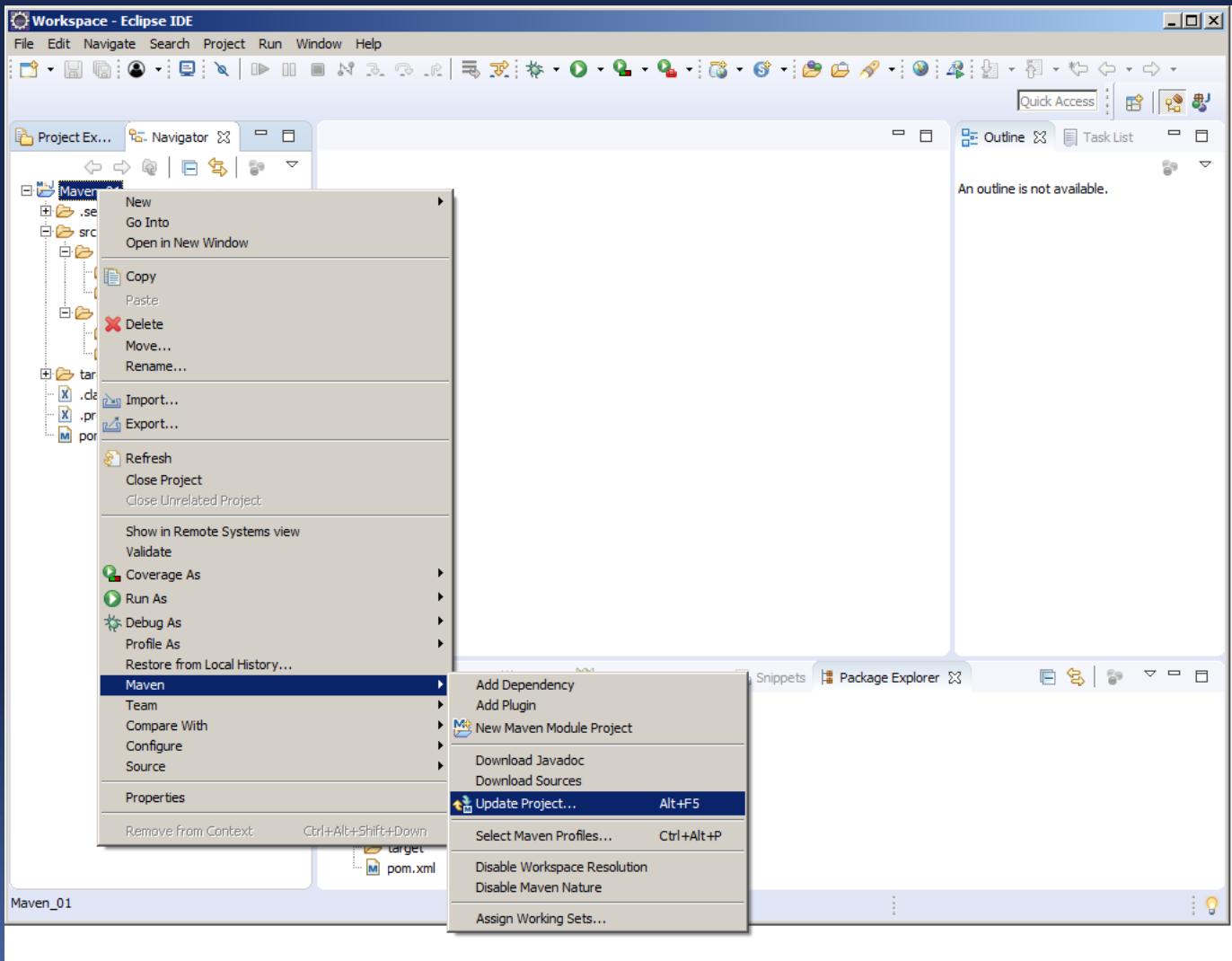
Description	Resource	Path	Location	Type
[+] Java Build Path Problems (1 item)				
[+] Java Problems (1 item)				

0 items selected



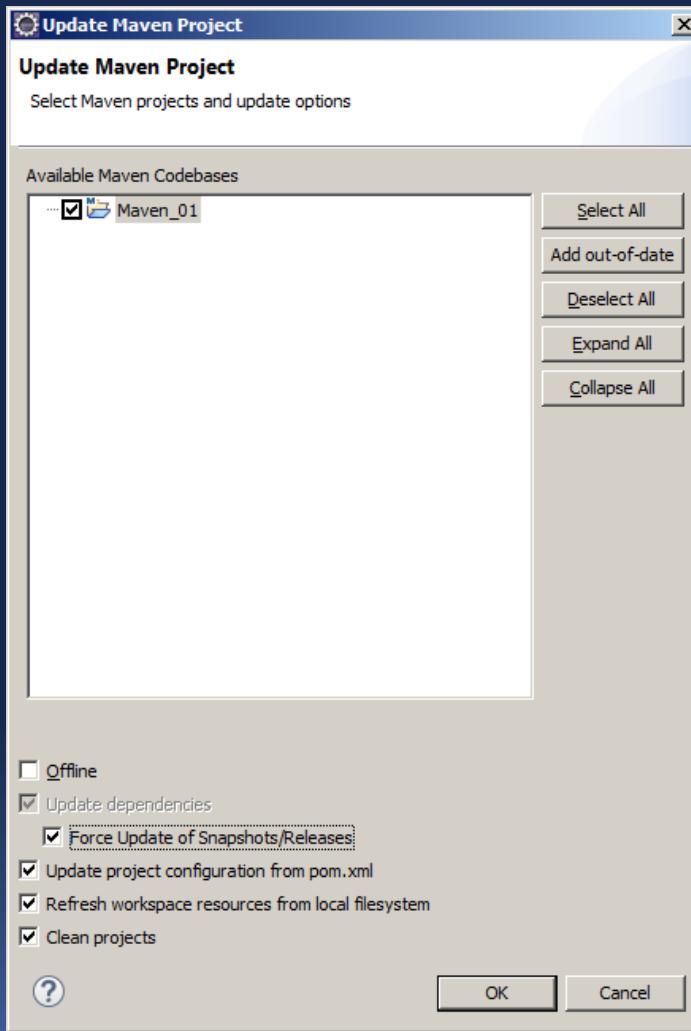
Criação de um Projeto Maven no Eclipse

- ✓ Para configurar o projeto como um projeto Java, clique no botão direito sobre o projeto, acesse o menu **Maven >Update Project ...**



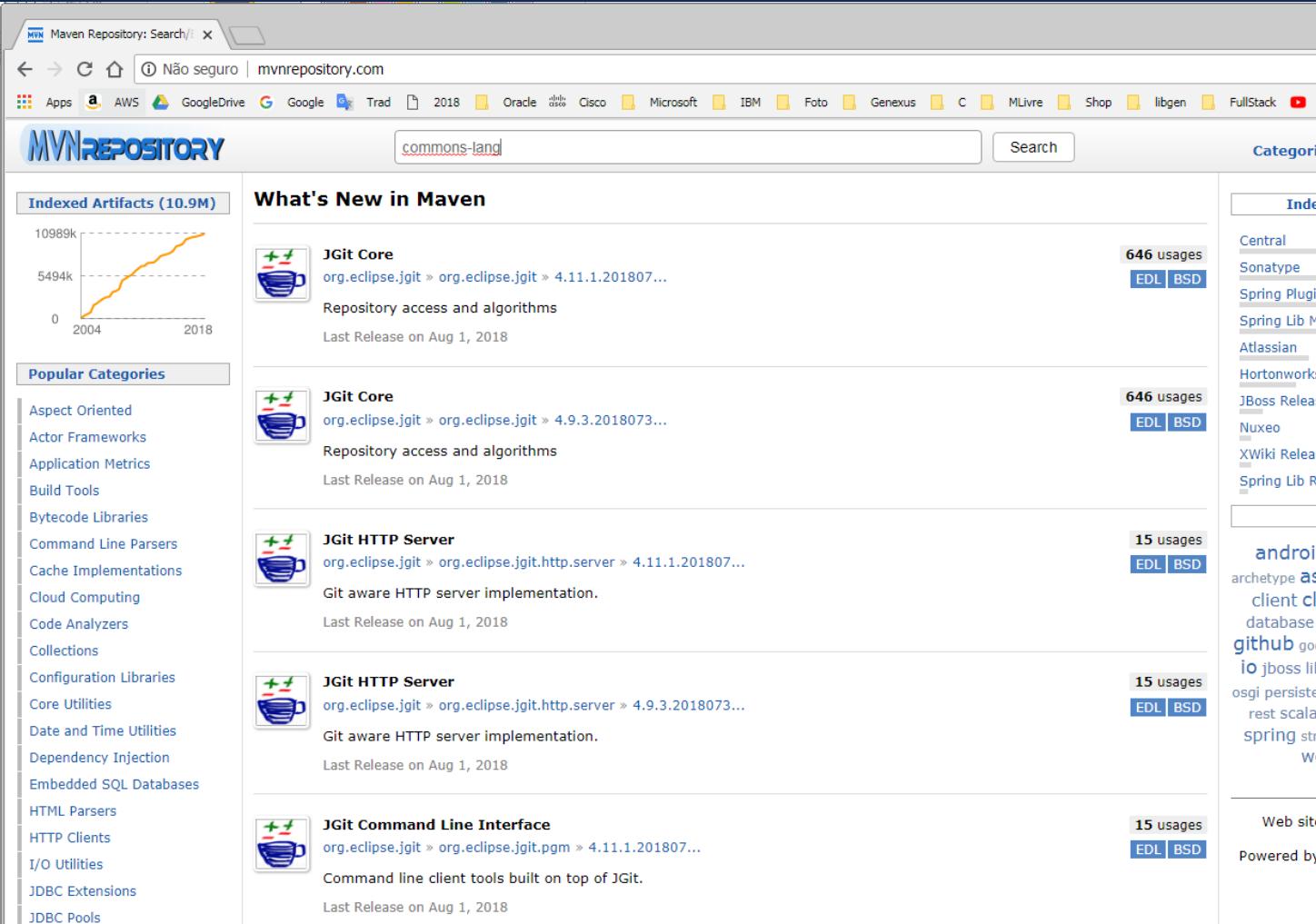
Criação de um Projeto Maven no Eclipse

- ✓ Selecione a opção **Force Update of Snapshots/Releases**;
- ✓ Clique em **OK**.



Adicionando uma dependência

- ✓ Acesse o site <http://mvnrepository.com> que contém uma lista das dependências disponíveis no repositório do Maven;
- ✓ Vamos pesquisar por **Commons-Lang**.



The screenshot shows the Maven Repository search results for the term "commons-lang". The search bar at the top contains "commons-lang". The results page features a chart titled "What's New in Maven" showing the growth of indexed artifacts from 2004 to 2018, reaching over 10 million. Below the chart, a list of Maven artifacts is displayed:

- JGit Core**: org.eclipse.jgit > org.eclipse.jgit > 4.11.1.201807... (646 usages, EDL | BSD)
- JGit Core**: org.eclipse.jgit > org.eclipse.jgit > 4.9.3.2018073... (646 usages, EDL | BSD)
- JGit HTTP Server**: org.eclipse.jgit > org.eclipse.jgit.http.server > 4.11.1.201807... (15 usages, EDL | BSD)
- JGit HTTP Server**: org.eclipse.jgit > org.eclipse.jgit.http.server > 4.9.3.2018073... (15 usages, EDL | BSD)
- JGit Command Line Interface**: org.eclipse.jgit > org.eclipse.jgit.pgm > 4.11.1.201807... (15 usages, EDL | BSD)

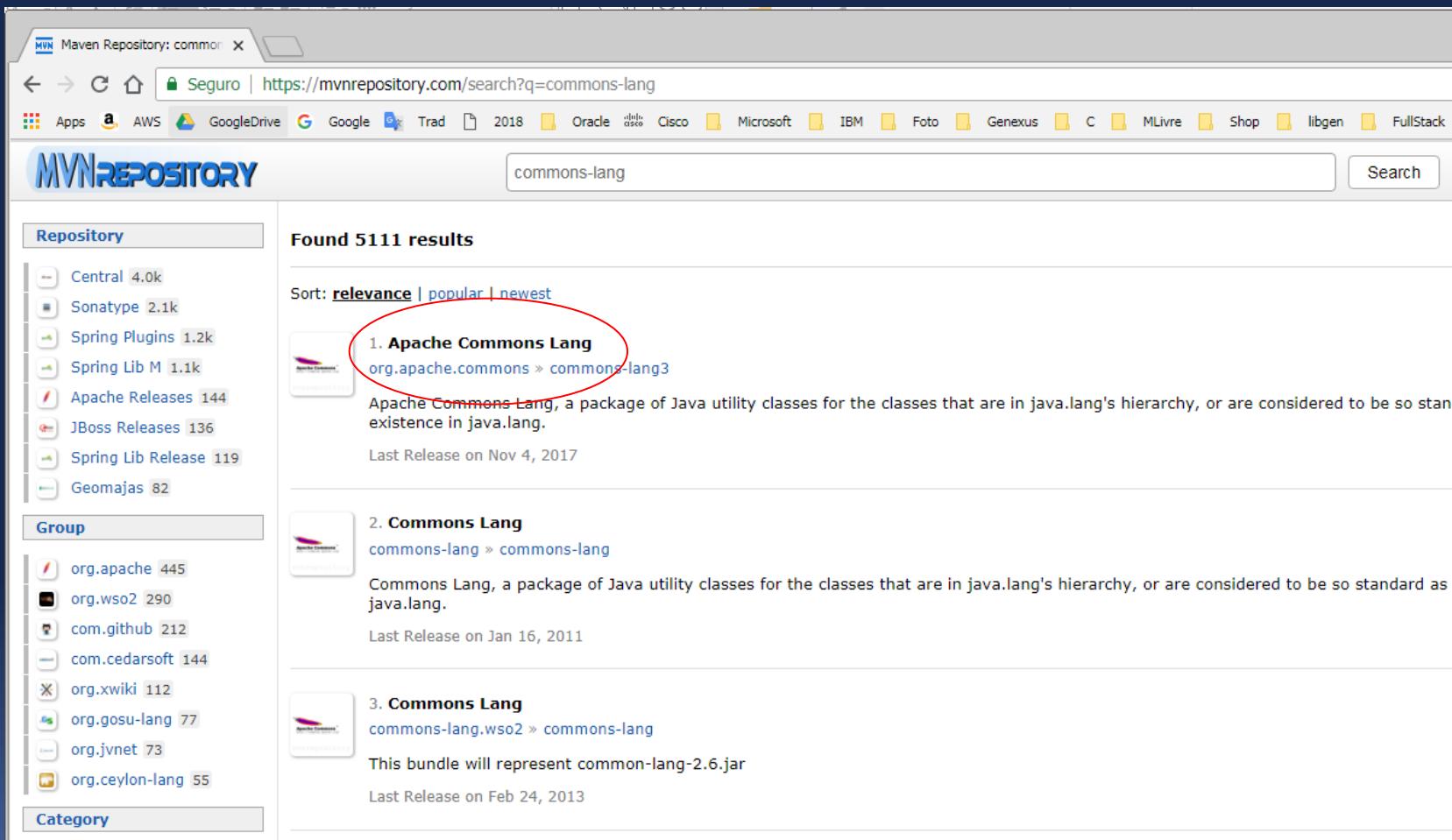
The sidebar on the left lists "Popular Categories" including Aspect Oriented, Actor Frameworks, Application Metrics, Build Tools, Bytecode Libraries, Command Line Parsers, Cache Implementations, Cloud Computing, Code Analyzers, Collections, Configuration Libraries, Core Utilities, Date and Time Utilities, Dependency Injection, Embedded SQL Databases, HTML Parsers, HTTP Clients, I/O Utilities, JDBC Extensions, and JDBC Pools.

The sidebar on the right lists "Categories" such as Central, Sonatype, Spring Plugins, Spring Lib M, Atlassian, Hortonworks, JBoss Releases, Nuxeo, XWiki Releases, Spring Lib Rel, android, archetype, ass, client, clo, database, e, github, io, jboss, libra, osgi, persisten, rest, scala, s, spring, stream, web, and Web site.

At the bottom, it says "Powered by: mavenrepository.com".

Adicionando uma dependência

- ✓ Selecione **Apache Commons Lang**, conforme indicação abaixo.



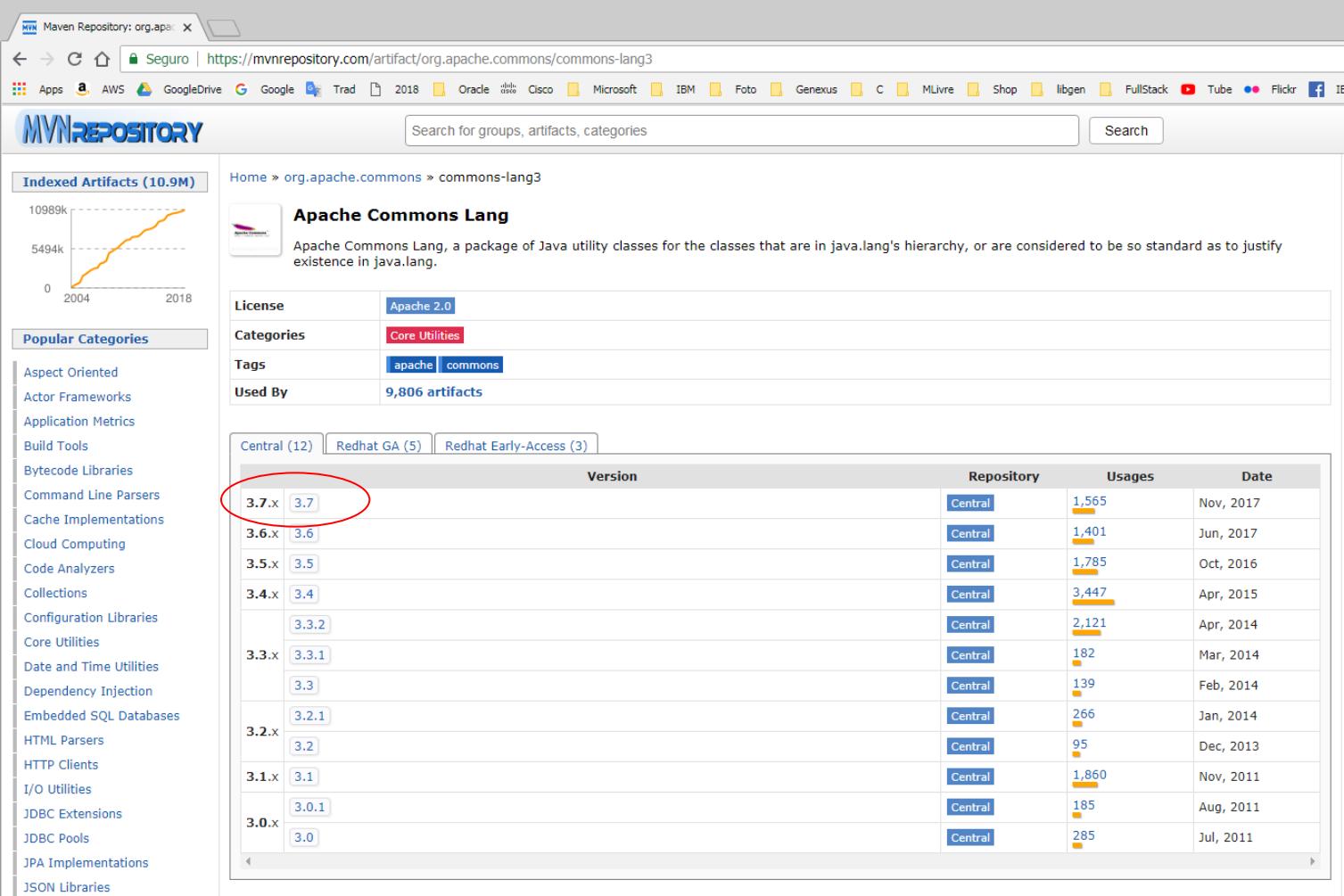
The screenshot shows the MVNRepository search results for the query "commons-lang". The search bar at the top contains "commons-lang". On the left, there are three filter panels: "Repository" (Central 4.0k, Sonatype 2.1k, Spring Plugins 1.2k, Spring Lib M 1.1k, Apache Releases 144, JBoss Releases 136, Spring Lib Release 119, Geomajas 82), "Group" (org.apache 445, org.wso2 290, com.github 212, com.cedarsoft 144, org.xwiki 112, org.gosu-lang 77, org.jvnet 73, org.ceylon-lang 55), and "Category". The main area displays "Found 5111 results" and lists three items:

- 1. Apache Commons Lang**
org.apache.commons » commons-lang3
Apache Commons Lang, a package of Java utility classes for the classes that are in java.lang's hierarchy, or are considered to be so standard as to exist in java.lang.
Last Release on Nov 4, 2017
- 2. Commons Lang**
commons-lang » commons-lang
Commons Lang, a package of Java utility classes for the classes that are in java.lang's hierarchy, or are considered to be so standard as to exist in java.lang.
Last Release on Jan 16, 2011
- 3. Commons Lang**
commons-lang.wso2 » commons-lang
This bundle will represent common-lang-2.6.jar
Last Release on Feb 24, 2013

A red oval highlights the first result, "Apache Commons Lang".

Adicionando uma dependência

- ✓ Selecione a mais **nova** versão, conforme ilustrado na imagem abaixo:



Maven Repository: org.apache.commons/commons-lang3

Apache Commons Lang

Apache Commons Lang, a package of Java utility classes for the classes that are in `java.lang`'s hierarchy, or are considered to be so standard as to justify existence in `java.lang`.

License: Apache 2.0

Categories: Core Utilities

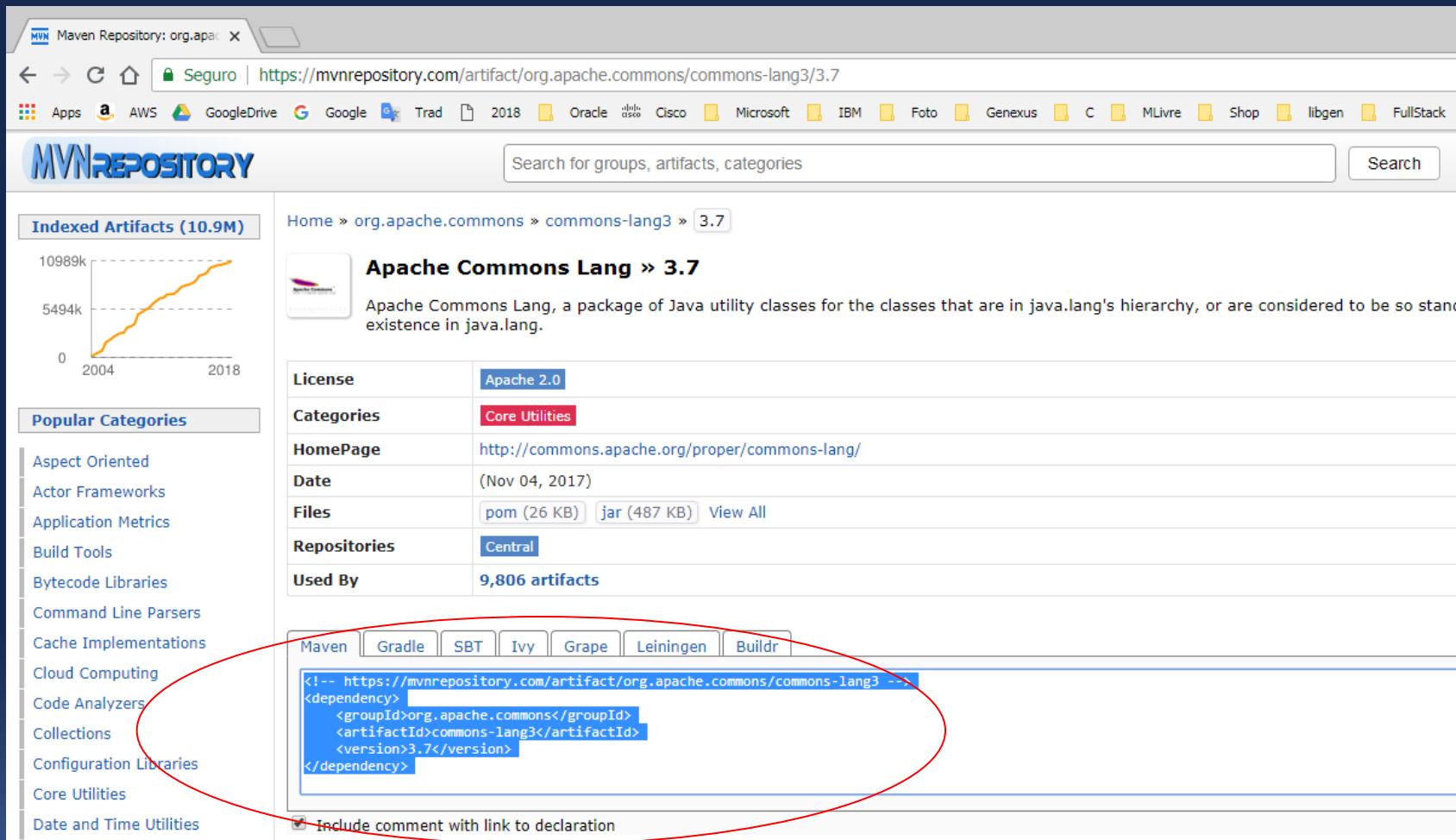
Tags: apache, commons

Used By: 9,806 artifacts

Version	Repository	Usages	Date
3.7.x	Central	1,565	Nov, 2017
3.6.x	Central	1,401	Jun, 2017
3.5.x	Central	1,785	Oct, 2016
3.4.x	Central	3,447	Apr, 2015
3.3.2	Central	2,121	Apr, 2014
3.3.x	Central	182	Mar, 2014
3.3	Central	139	Feb, 2014
3.2.1	Central	266	Jan, 2014
3.2.x	Central	95	Dec, 2013
3.1.x	Central	1,860	Nov, 2011
3.0.1	Central	185	Aug, 2011
3.0.x	Central	285	Jul, 2011

Adicionando uma dependência

- ✓ Copie o **XML** que corresponde à biblioteca **Commons-Lang**



Maven Repository: org.apache.commons/commons-lang3/3.7

Indexed Artifacts (10.9M)

Apache Commons Lang > 3.7

Apache Commons Lang, a package of Java utility classes for the classes that are in java.lang's hierarchy, or are considered to be so standard in existence in java.lang.

License: Apache 2.0

Categories: Core Utilities

HomePage: <http://commons.apache.org/proper/commons-lang/>

Date: (Nov 04, 2017)

Files: pom (26 KB) jar (487 KB) View All

Repositories: Central

Used By: 9,806 artifacts

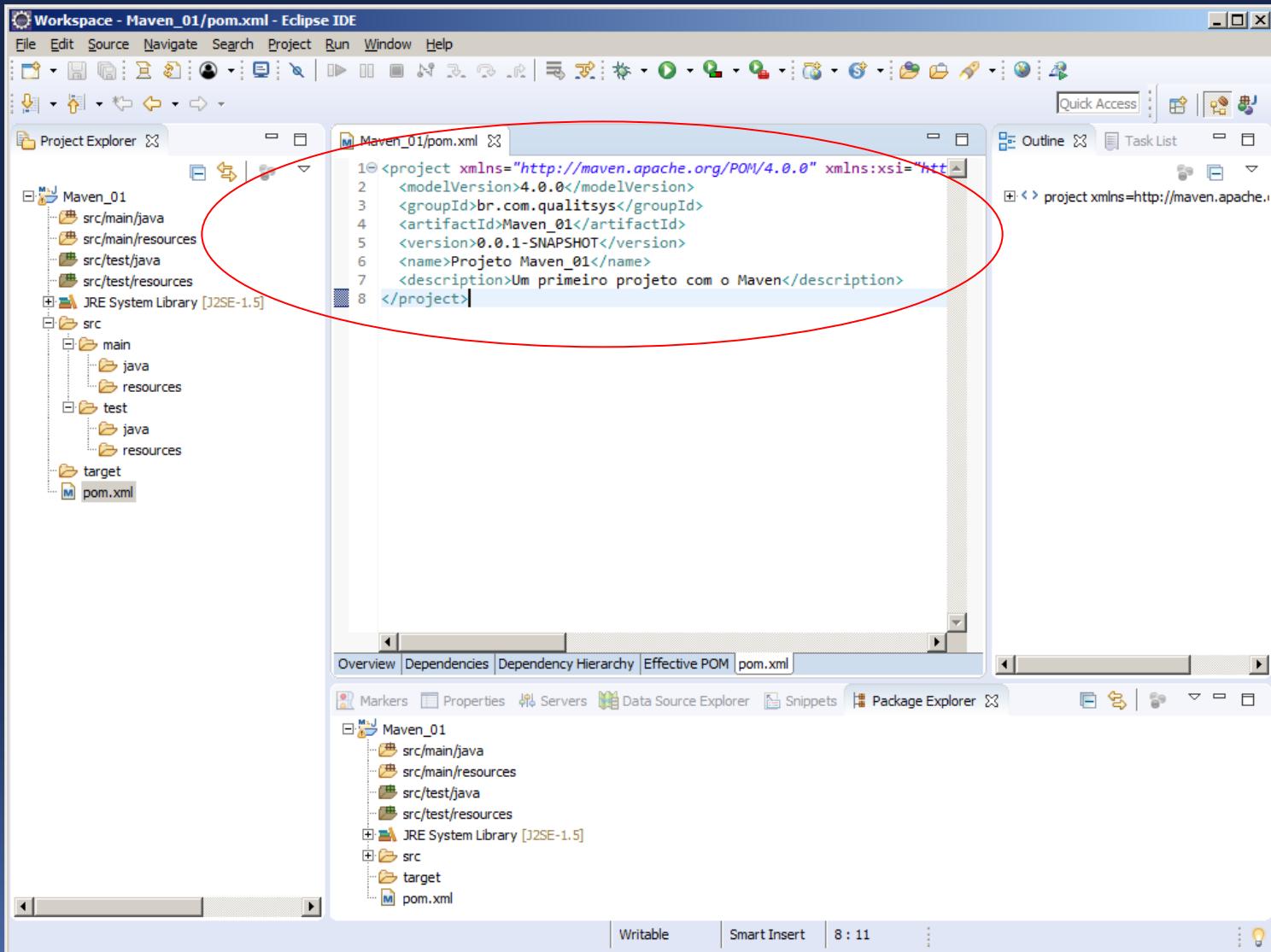
Maven Gradle SBT Ivy Grape Leiningen Buildr

```
<!-- https://mvnrepository.com/artifact/org.apache.commons/commons-lang3 -->
<dependency>
    <groupId>org.apache.commons</groupId>
    <artifactId>commons-lang3</artifactId>
    <version>3.7</version>
</dependency>
```

Include comment with link to declaration

Adicionando uma dependência

- ✓ De volta ao Eclipse, clique duas vezes no arquivo **pom.xml** para editá-lo.
- ✓ Pressione **Ctrl+A** para selecionar todo o conteúdo do arquivo e depois **Ctrl+I** para indentar o arquivo.



Adicionando uma dependência

- ✓ Incluir a dependência no arquivo **pom.xml**

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>br.com.qualitsys</groupId>
    <artifactId>Maven_01</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>Projeto Maven_01</name>
    <description>Um primeiro projeto com o Maven</description>

    <dependencies>

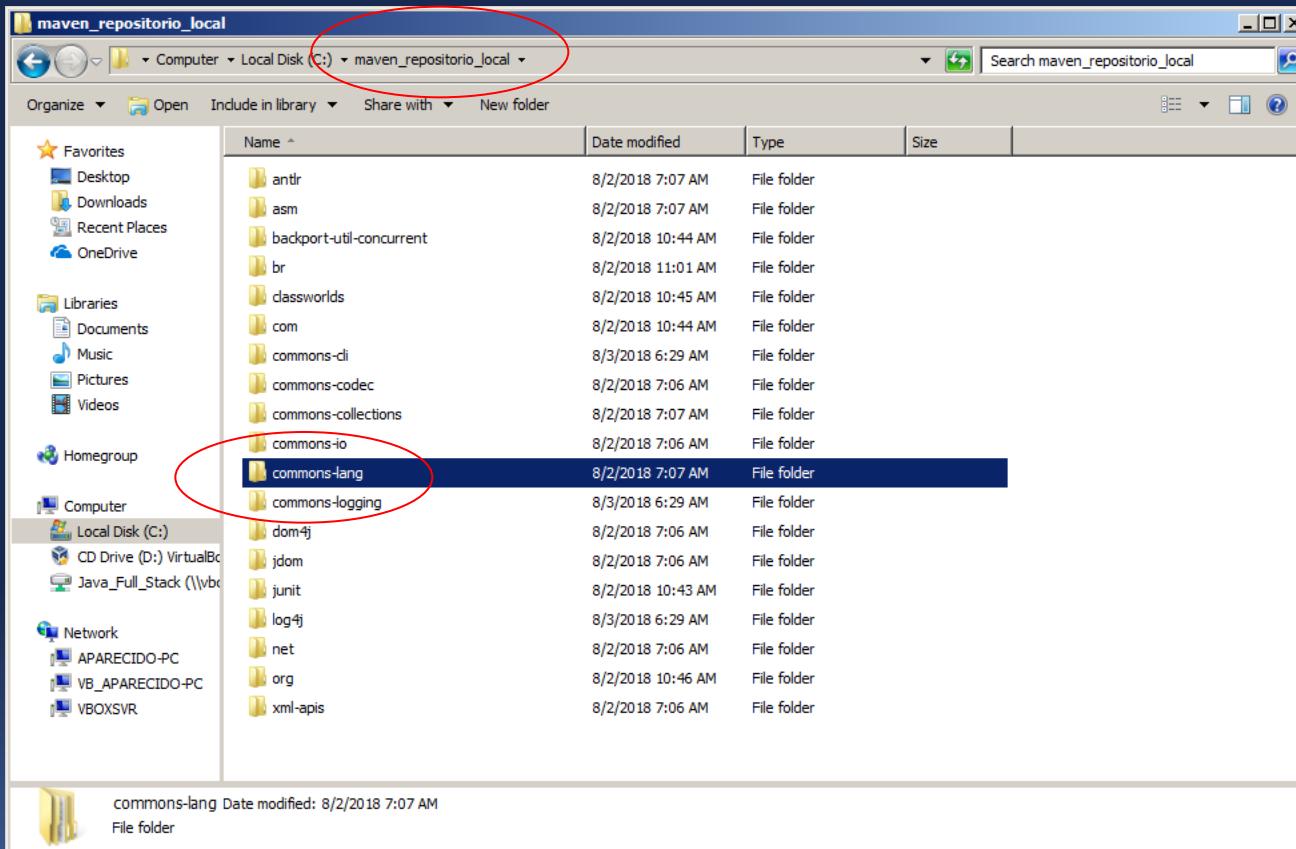
        <dependency>
            <groupId>org.apache.commons</groupId>
            <artifactId>commons-lang3</artifactId>
            <version>3.7</version>
        </dependency>

    </dependencies>

</project>
```

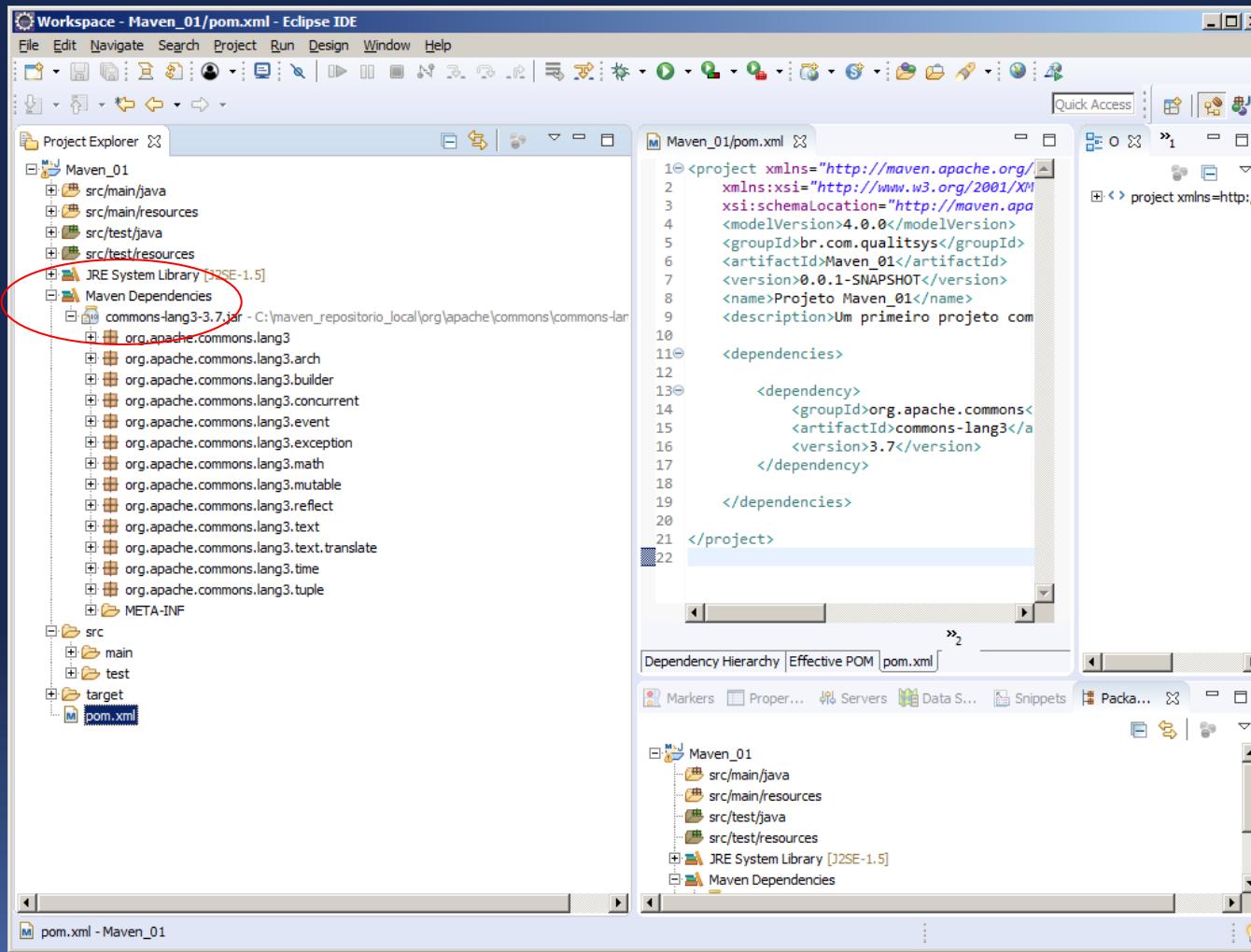
Adicionando uma dependência

- ✓ Salvar o arquivo **pom.xml**;
- ✓ O plugin **M2E** irá identificar a alteração, baixar automaticamente a dependência do repositório central para o seu repositório local e adicioná-la ao classpath do projeto



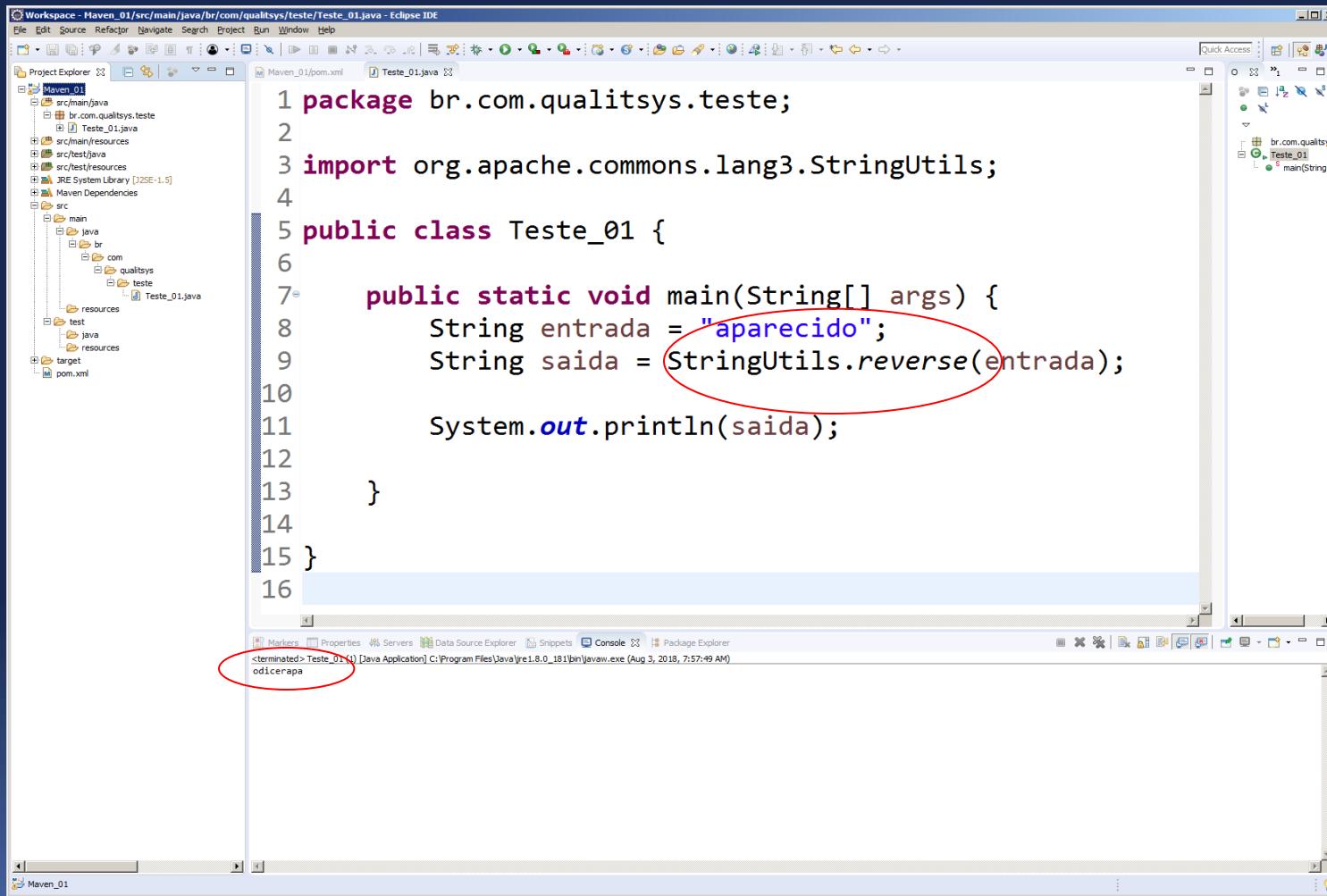
Adicionando uma dependência

- ✓ O plugin **M2E** também irá adicionar a biblioteca baixada ao classpath do projeto



Adicionando uma dependência

- ✓ **Pronto!** Agora podemos utilizar a biblioteca Commons-Lang;
- ✓ No projeto, criaremos o package **br.com.qualitsys.teste**;
- ✓ No package, criaremos a classe **Teste_01** a qual utilizará o método **reverse()** da classe **StringUtils**.



The screenshot shows the Eclipse IDE interface with the following details:

- Project Explorer:** Shows the project structure: Maven_01 > src/main/java > br.com.qualitsys.teste > Teste_01.java.
- Maven_01/pom.xml:** The XML configuration file for the Maven project.
- Teste_01.java:** The Java source code containing the following code:


```

1 package br.com.qualitsys.teste;
2
3 import org.apache.commons.lang3.StringUtils;
4
5 public class Teste_01 {
6
7     public static void main(String[] args) {
8         String entrada = "aparecido";
9         String saida = StringUtils.reverse(entrada);
10
11         System.out.println(saida);
12     }
13
14 }
15
16 
```
- Console Output:** Shows the terminal output of the application running. The output is circled in red and reads:

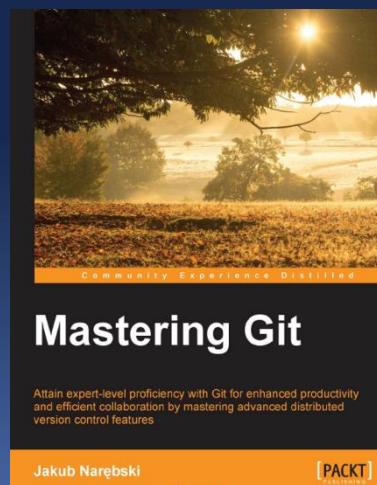
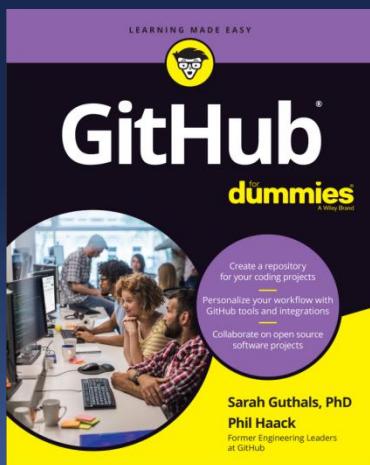
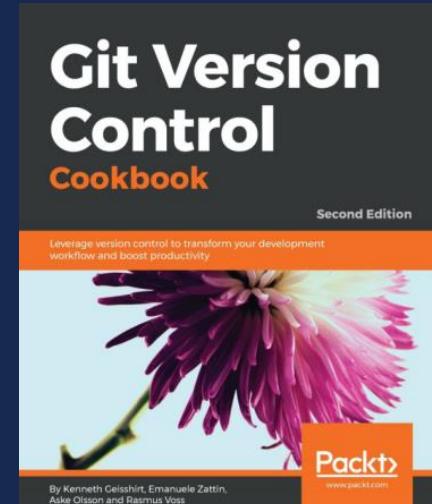
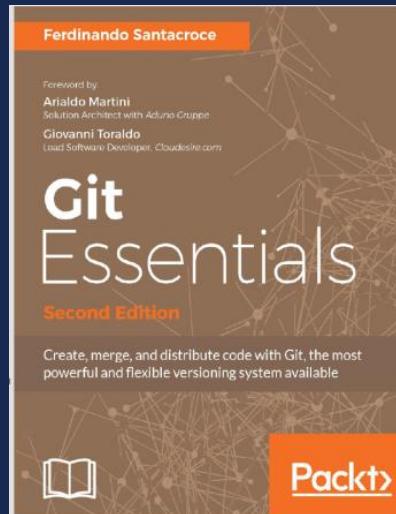
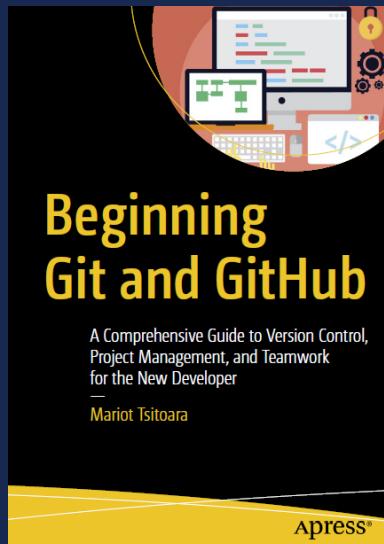
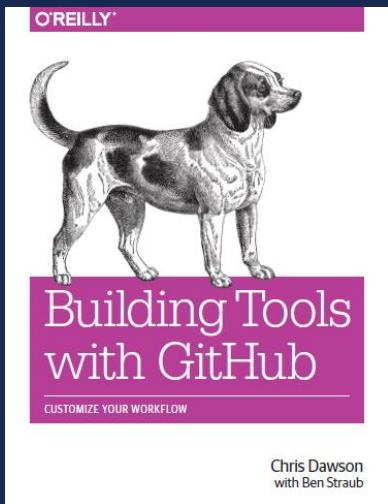

```

<terminated> Teste_01 [Java Application] C:\Program Files\Java\jre1.8.0_181\bin\javaw.exe (Aug 3, 2018, 7:57:49 AM)
odicerapa
      
```

Versionando o Projeto no github



Bibliografia



Github

- Uma das principais ferramentas para suporte ao **desenvolvimento colaborativo** de software;
- **Github** cria um ambiente que permite se armazenar o **código fonte** em um **servidor remoto**;
- Permite **compartilhar** código com outros desenvolvedores;
- Por trás dos bastidores do **github** está o **git**.



Controle de Versões

- **SCM** - Sistema de Controle de Versões (**Source Control Management**)
- Controlam **versões** de um projeto de forma **iterativa**;
- Geralmente uma versão do projeto é criada quando se tem no projeto algo **factível** de ser usado;
- Inicialmente, todo projeto tem uma versão inicial com uma estrutura básica e à medida em que vai sendo desenvolvido novas versões são definidas;
- Caso algo dê errado, **SCM** permite que se **recupere** versões anteriores guardadas no ambiente.



SCM – workflow básico

- Cria-se um **projeto**, tipicamente em um **folder** de seu computador;
- A versão do projeto é informada ao **SCM**;
- Cada vez que o projeto na área de trabalho (**folder local**) estiver em um **estado** aceitável de funcionamento, uma **nova versão** é definida no **SCM**;
- Caso o desenvolvedor necessite **voltar** à uma **versão anterior**, basta informar ao **SCM** para **baixar** a versão solicitada.



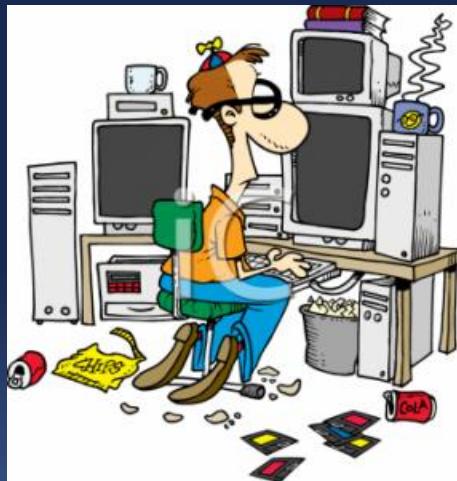
Trabalhando em grupo

- Embora se possa trabalhar sozinho em seu próprio computador, o **SCM** permite que se desenvolva em conjunto com **outros desenvolvedores**;



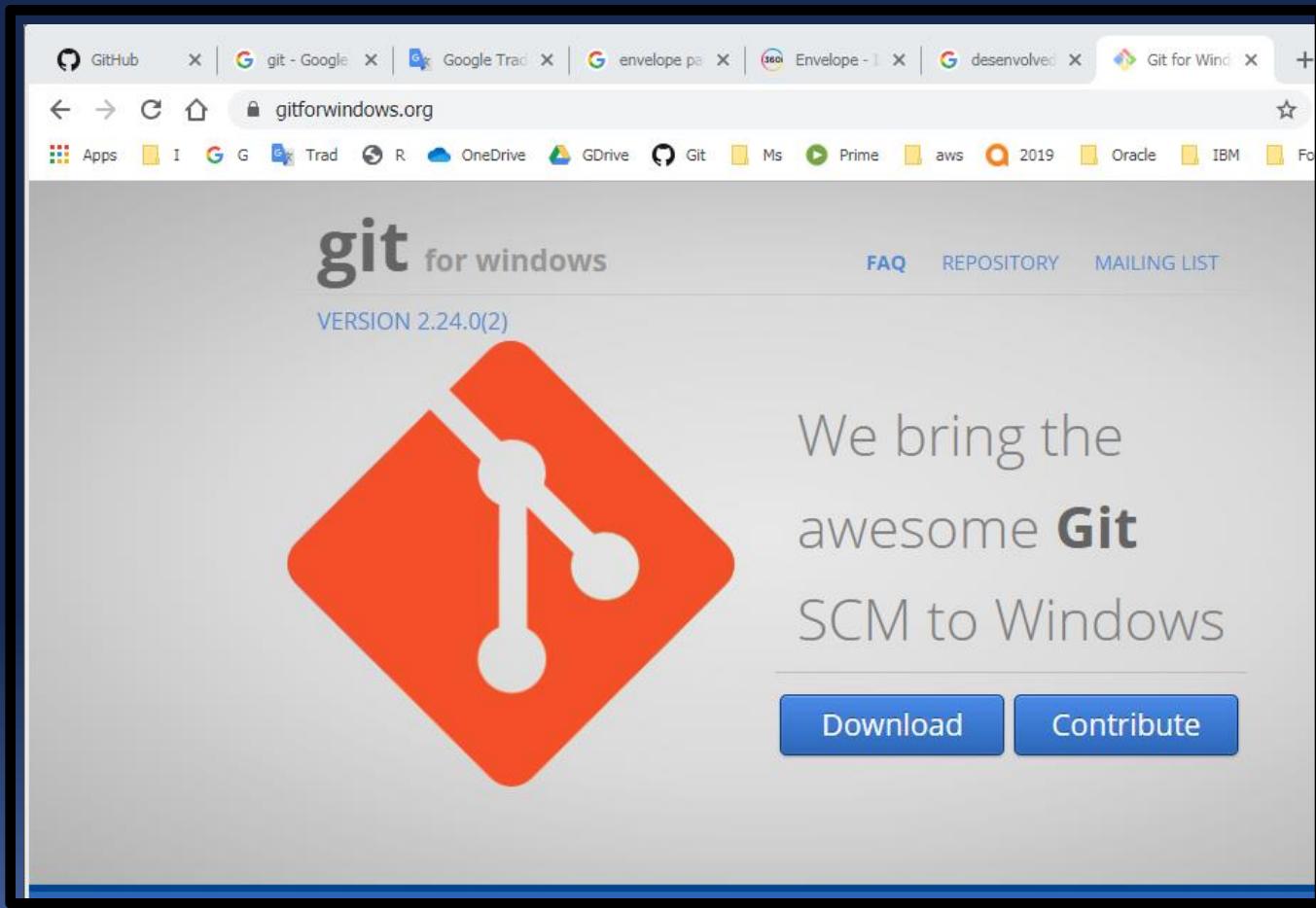
Git - Controle de Versões

- **Github** é baseado em **git**;
- **Git** é um dos muitos sistemas de controle de versões;
- Embora seja usual utilizar **git** com **github**, pode-se trabalhar com **git** de forma isolada, em seu próprio **ambiente local** de desenvolvimento.



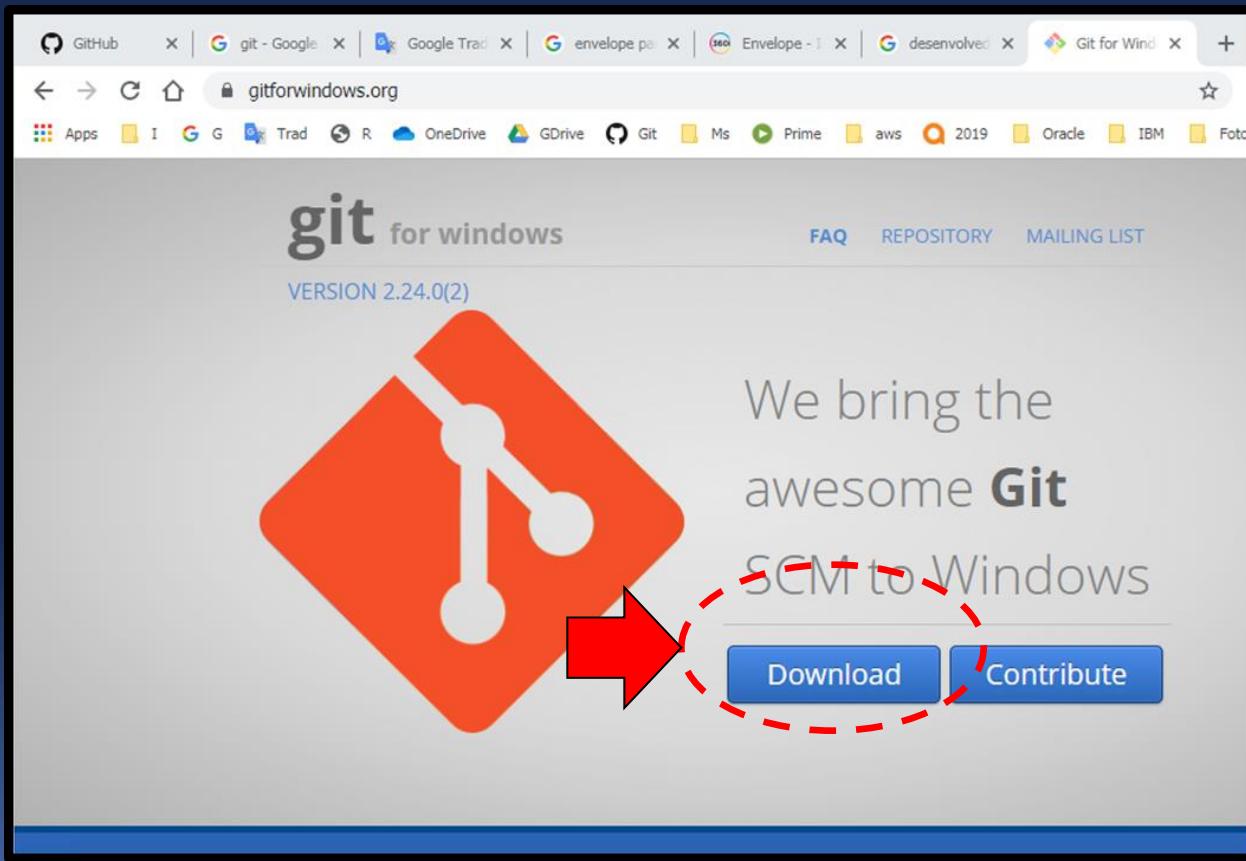
Git no Terminal

- A instalação do git pode ser feita a partir de <https://gitforwindows.org>.

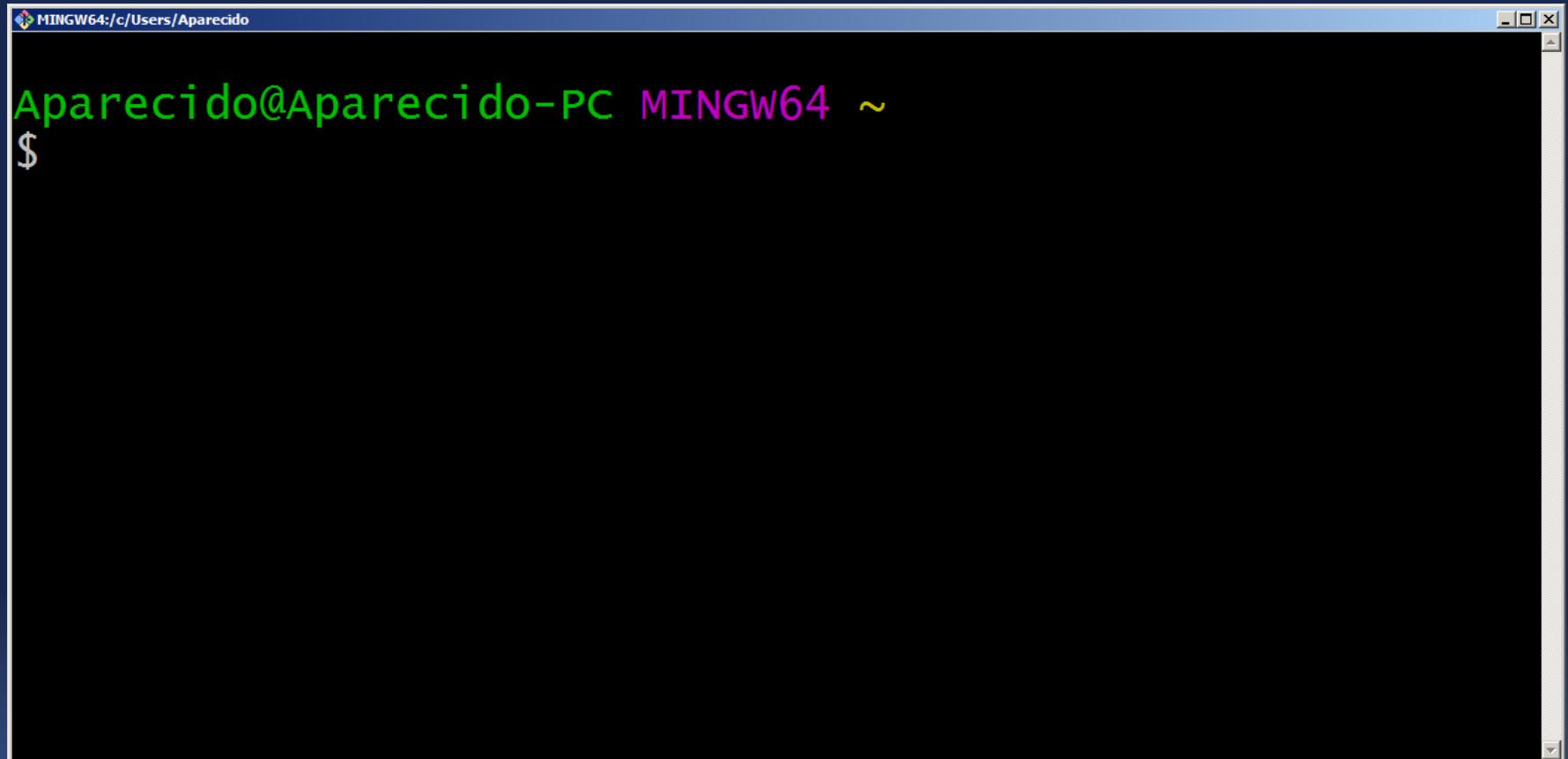


Git no Terminal

- Clicar em **Download** para acessar **Git Bash**, um **emulador** que permite ao desenvolvedor interagir com **Git**, na **console** do terminal.



Git Bash no Terminal

A screenshot of a Windows-style terminal window titled "MINGW64:/c/Users/Aparecido". The title bar also shows the path "MINGW64:/c/Users/Aparecido". The window contains a single character: a dollar sign (\$) in white, indicating the command prompt. The background of the window is black, and the overall interface is reminiscent of a classic Windows application.

Git Bash no Terminal

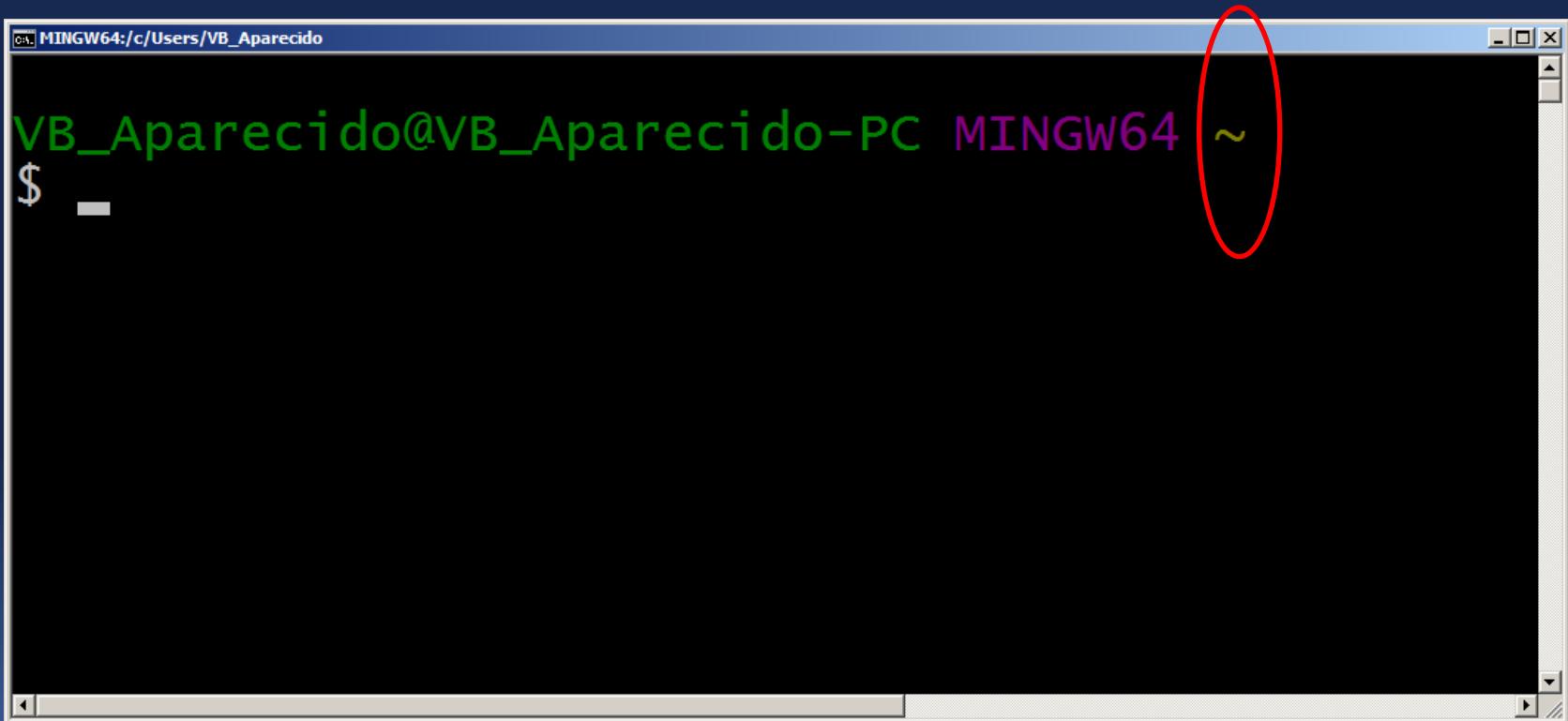
git --version

```
MINGW64:/c/Users/Aparecido
Aparecido@Aparecido-PC MINGW64 ~
$ git --version
git version 2.27.0.windows.1

Aparecido@Aparecido-PC MINGW64 ~
$
```

Trabalhando com git Bash

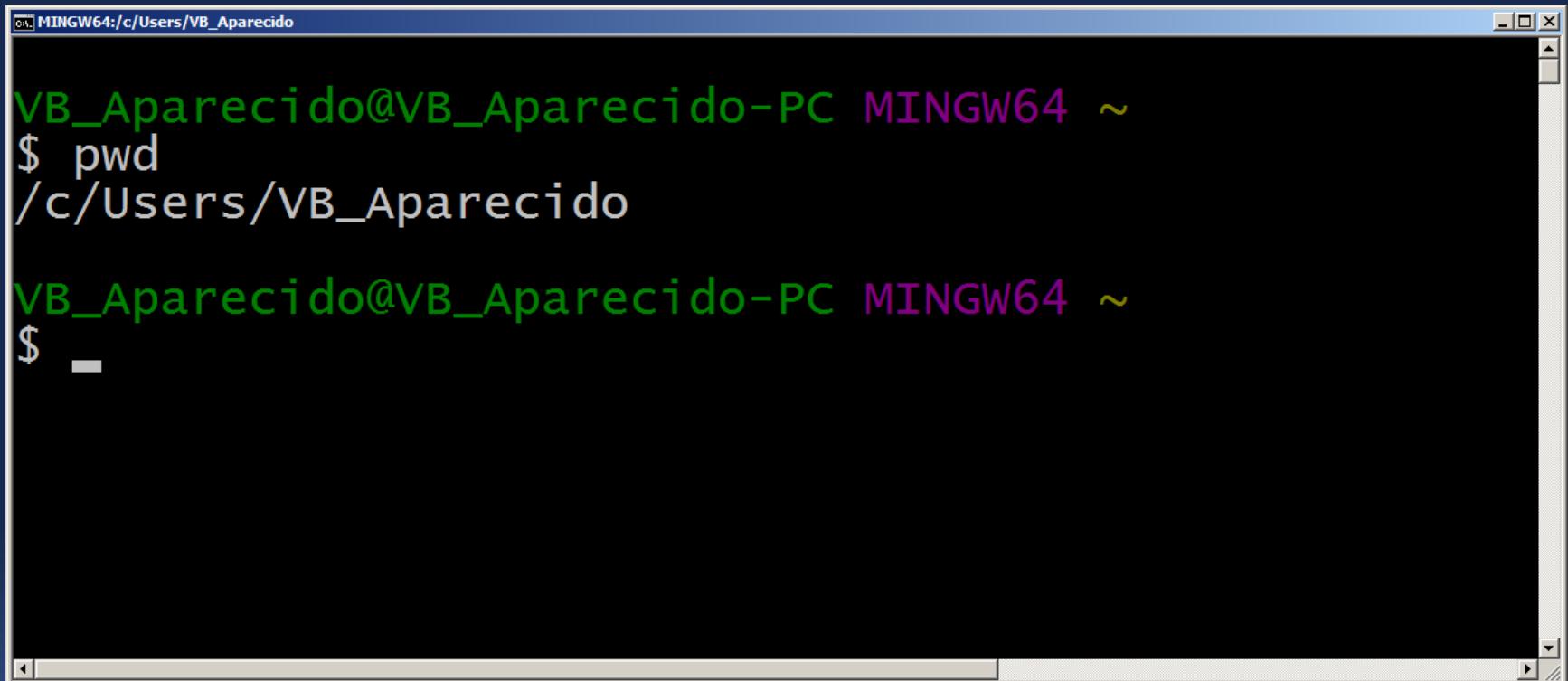
- ✓ O **git Bash** inicia, por padrão, no diretório de seu usuário;
- ✓ Quando o **git Bash** estiver no diretório padrão, ao invés de mostrar todo o caminho, por exemplo: c/Users/Aparecido, ele mostra apenas um ~;



```
MINGW64:c/Users/VB_Aparecido
VB_Aparecido@VB_Aparecido-PC MINGW64
$ -
```

Trabalhando com git Bash

- ✓ O comando **pwd** informa o diretório corrente:

A screenshot of a Windows terminal window titled "MINGW64:/c/Users/VB_Aparecido". The window contains the following text:

```
VB_Aparecido@VB_Aparecido-PC MINGW64 ~
$ pwd
/c/Users/VB_Aparecido

VB_Aparecido@VB_Aparecido-PC MINGW64 ~
$ -
```

The terminal uses color coding for the prompt and the current directory path.

Trabalhando com git Bash

- ✓ O comando **cd** permite ir para algum outro diretório;
comando: **cd E:**



```
MINGW64:/e
Aparecido@Aparecido-PC MINGW64 ~
$ cd e:

Aparecido@Aparecido-PC MINGW64 /e
$ |
```

A green arrow points from the left towards the terminal window. The terminal window shows a command-line interface with a black background and white text. The prompt is "MINGW64:/e". The user enters the command "cd e:" followed by a new line. The prompt then changes to "Aparecido@Aparecido-PC MINGW64 /e" and the cursor is visible at the end of the line.

Trabalhando com git Bash

- ✓ O comando **clear** permite limpar a tela do terminal



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ clear |
```

A screenshot of a terminal window titled "MINGW64:/e/git_repositorio_local". The window shows a command-line interface with the user's name "Aparecido" and computer name "Aparecido-PC" at the top. The current directory is "/e/git_repositorio_local" and the branch is "master". A green arrow points from the left towards the terminal window. Inside the window, the command "\$ clear |" is typed, indicating the user is about to run the "clear" command to clear the screen.

Trabalhando com git bash

- ✓ O comando **dir**, lista as pastas do drive E:



```
Aparecido@Aparecido-PC MINGW64 /e
$ dir
$RECYCLE.BIN
Agenda\ de\ Provas\ Finais\ N1_4AN痈ADS.docx
apache_Maven_Local_Repository
apache-maven-3.6.3
apache-tomcat-9.0.34
Calibre_Library
clojure
Dados_Diversos
Eclipse
EXIMES
FATEC
git_repositorio_local
```

Trabalhando com git bash

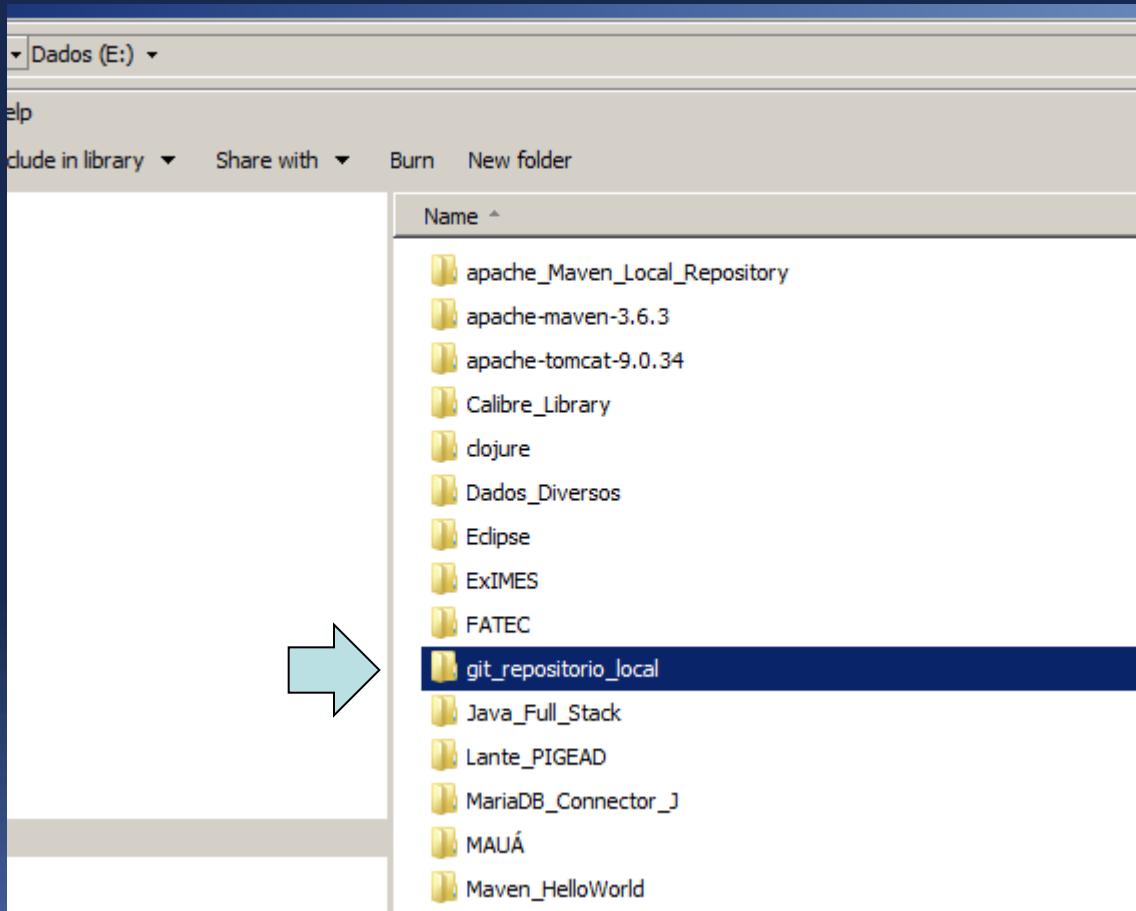
- ✓ O comando **ls**, também lista as pastas do drive E:



```
MINGW64:/e
Aparecido@Aparecido-PC MINGW64 /e
$ ls
'$RECYCLE.BIN'
'Agenda de Provas Finais N1_4AN痈ADS.docx'
'apache_Maven_Local_Repository/'
'apache-maven-3.6.3/'
'apache-tomcat-9.0.34/'
'calibre_Library/'
'clojure/'
'Dados_Diversos/'
'Eclipse/'
'ExIMES/'
'FATEC/'
'git_repositorio_local/'
```

Gitbash no Terminal

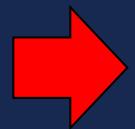
Iremos criar em nosso ambiente um repositório local chamado **E:\git_repositorio_local** e salvaremos nesse diretório os nossos projetos de desenvolvimento.



Gitbash no Terminal

Listando os arquivos do Repositório Local

Comandos: cd E:
 cd git_repositorio_local
 ls



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e
$ cd git_repositorio_local/
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local
$ ls
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local
$
```

O comando **ls** reporta que o repositório local está vazio !

Gitbash no Terminal

Limpando a console e exibindo o diretório corrente

Comandos: **clear**
pwd

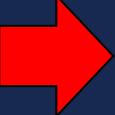
A large red arrow points from the left towards the terminal window, highlighting the output of the commands.

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local
$ pwd
/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local
$ |
```

Gitbash no Terminal

- ✓ Exibindo o status do nosso projeto no Repositório Git Local
- ✓ **git** está informando que o diretório não está sendo monitorado pelo git

Comando: **git status**



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local]
$ git status
fatal: not a git repository (or any of the parent directories): .
git

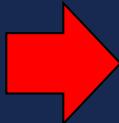
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local]
$
```

A red arrow points from the left towards the terminal window, highlighting the output of the command. The terminal window shows a command-line interface with a black background and white text. It displays the user's name, the computer name, the terminal type, and the current directory. The user then types 'git status' and receives an error message indicating that the current directory is not a git repository. The window ends with a prompt '\$'.

Gitbash no Terminal

Definindo a pasta do projeto como um repositório local git

Comando: **git init**



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local]
$ git init
Initialized empty Git repository in E:/git_repositorio_local/.git/
/
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

A screenshot of a terminal window titled "MINGW64:/e/git_repositorio_local". The window shows a command-line session where the user runs the "git init" command. A red arrow points from the left towards the terminal window. The terminal output indicates that an empty Git repository was initialized in the specified directory.

Gitbash no Terminal

- ✓ Exibindo o status do nosso Repositório Git Local.
- ✓ Git reconhece o diretório, mas informa que não há nada para ser comitado.

Comando: **git status**

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master

No commits yet

nothing to commit (create/copy files and use "git add" to track)

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

Salvando o projeto no repositório git local



- ✓ Salvando projeto Maven_01 no repositório git local

The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** JavaWeb - Maven_01/src/main/java/br/com/qualsys/teste/Teste_01.java - Eclipse IDE
- Menu Bar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help
- Toolbar:** Standard Eclipse toolbar icons.
- Left Sidebar:** Shows the project structure with "Maven_01" selected.
- Central Editor:** Displays the Java code for "Teste_01.java".

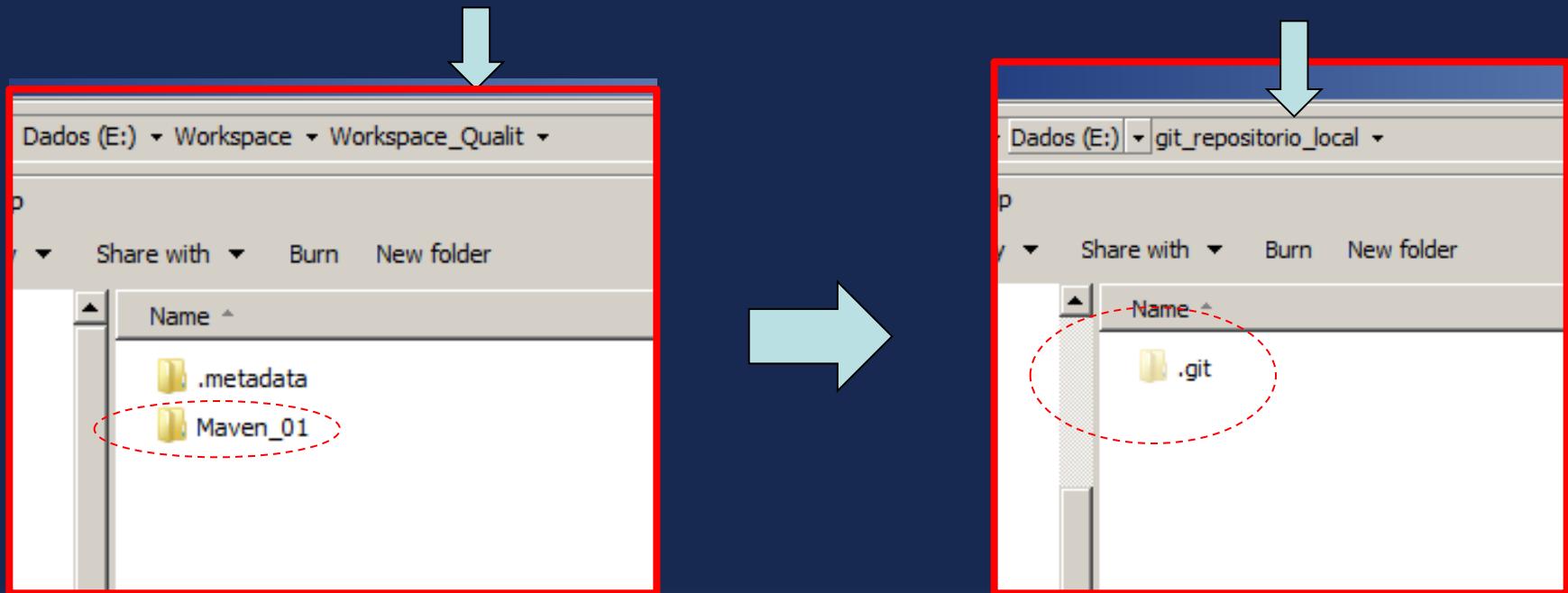
```
1 package br.com.qualitsys.teste;
2
3 import org.apache.commons.lang3.StringUtils;
4
5 public class Teste_01 {
6
7     public static void main(String[] args) {
8         String entrada = new String("aparecer");
9
10        String saida = StringUtils.reverse(entrada);
11    }
12}
```
- Outline View:** Shows the class structure: "br.com.qualitsys.teste" and "Teste_01" with its method "main(String[])".
- Bottom Status Bar:** Shows "Maven_01" and memory usage "188M of 256M".



Salvando o projeto no repositório git local



- ✓ Salvando projeto Maven_01 no repositório git local



Salvando o projeto no repositório git local



- ✓ Listando arquivos no repositório local

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ ls
Maven_01/

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```



Gitbash no Terminal

- ✓ Exibindo o status do nosso projeto no Repositório Git Local
- ✓ git exibe a existência de arquivos não monitorados e solicita que sejam informados (adicionados) os arquivos deverão ser monitorados.

Comando: **git status**



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
    Maven_01/

nothing added to commit but untracked files present (use "git add"
to track)

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Gitbash no Terminal

- ✓ Adicionando **todos** os arquivos da pasta no git para monitorar

Comando: **git add .**

A large red arrow points from the left towards the terminal window, highlighting the command being run.

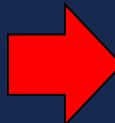
```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git add .

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Gitbash no Terminal

- ✓ Verificando novamente o status do repositório local
- ✓ **git** informa que **há** arquivos para serem comitados

Comando: **git status**



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master

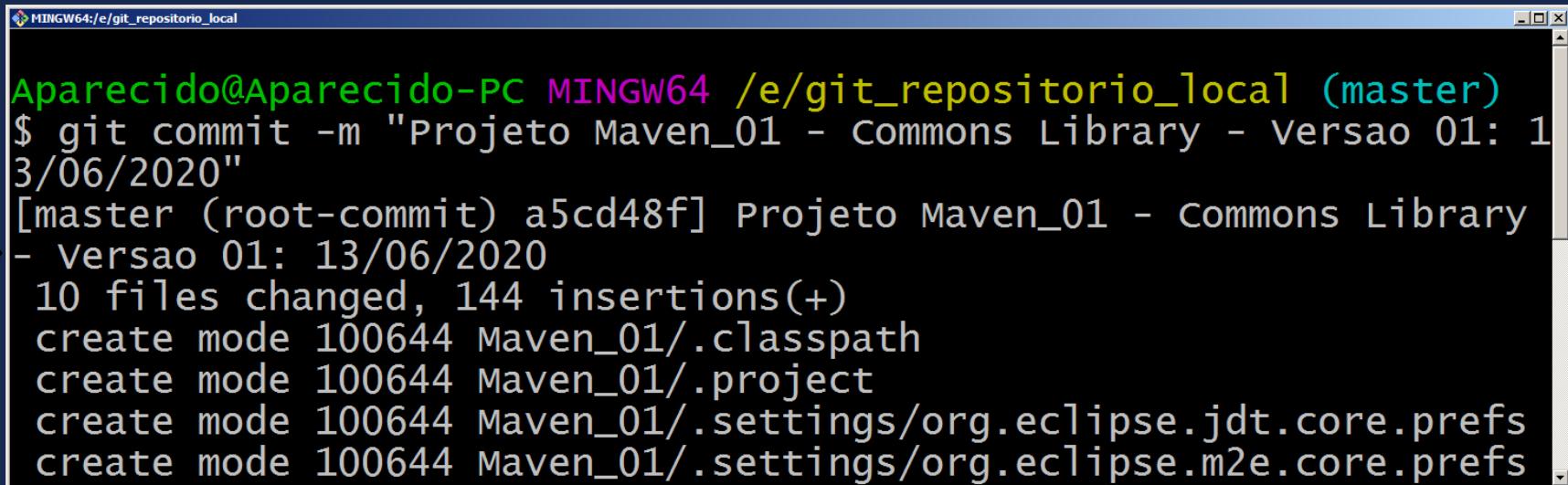
No commits yet

changes to be committed:
(use "git rm --cached <file>..." to unstage)
    new file: Maven_01/.classpath
    new file: Maven_01/.project
    new file: Maven_01/.settings/org.eclipse.jdt.core.prefs
    new file: Maven_01/.settings/org.eclipse.m2e.core.prefs
    new file: Maven_01/pom.xml
    new file: Maven_01/src/main/java/com/qualitsys/teste/T
este_01.java
    new file: Maven_01/target/classes/META-INF/MANIFEST.MF
    new file: Maven_01/target/classes/META-INF/maven/com.q
```

Gitbash no Terminal

- ✓ Incluindo um ponto de controle (emitindo um commit)

Comando: `git commit -m "Projeto Maven_01 – Commons Library - Versão 01: 13/06/20"`



A screenshot of a terminal window titled "MINGW64:/e/git_repositorio_local". The window contains the following text:

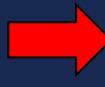
```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git commit -m "Projeto Maven_01 - Commons Library - Versao 01: 13/06/2020"
[master (root-commit) a5cd48f] Projeto Maven_01 - Commons Library - Versao 01: 13/06/2020
 10 files changed, 144 insertions(+)
  create mode 100644 Maven_01/.classpath
  create mode 100644 Maven_01/.project
  create mode 100644 Maven_01/.settings/org.eclipse.jdt.core.prefs
  create mode 100644 Maven_01/.settings/org.eclipse.m2e.core.prefs
```

Two red arrows point to the commit message and the log output respectively.

Gitbash no Terminal

- ✓ Verificando novamente o status do repositório local
- ✓ git informa que não há nada para comitar !

Comando: **git status**



```
MINGW64:/e/git_repositorio_local
$ Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
On branch master
nothing to commit, working tree clean

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Gitbash no Terminal

- ✓ checando as alterações processadas desde o primeiro commit

Comando: **git log**



```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git log
commit 0c7056d992823830dc902e673bfa9484f0da62b6 (HEAD -> master)
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 18:00:24 2020 -0300

    Projeto Maven_01 - Commons Library - versao 01: 13/06/2020

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Gitbash no Terminal

- ✓ Exibindo eventuais alterações feitas no projeto

Comando: **git show**

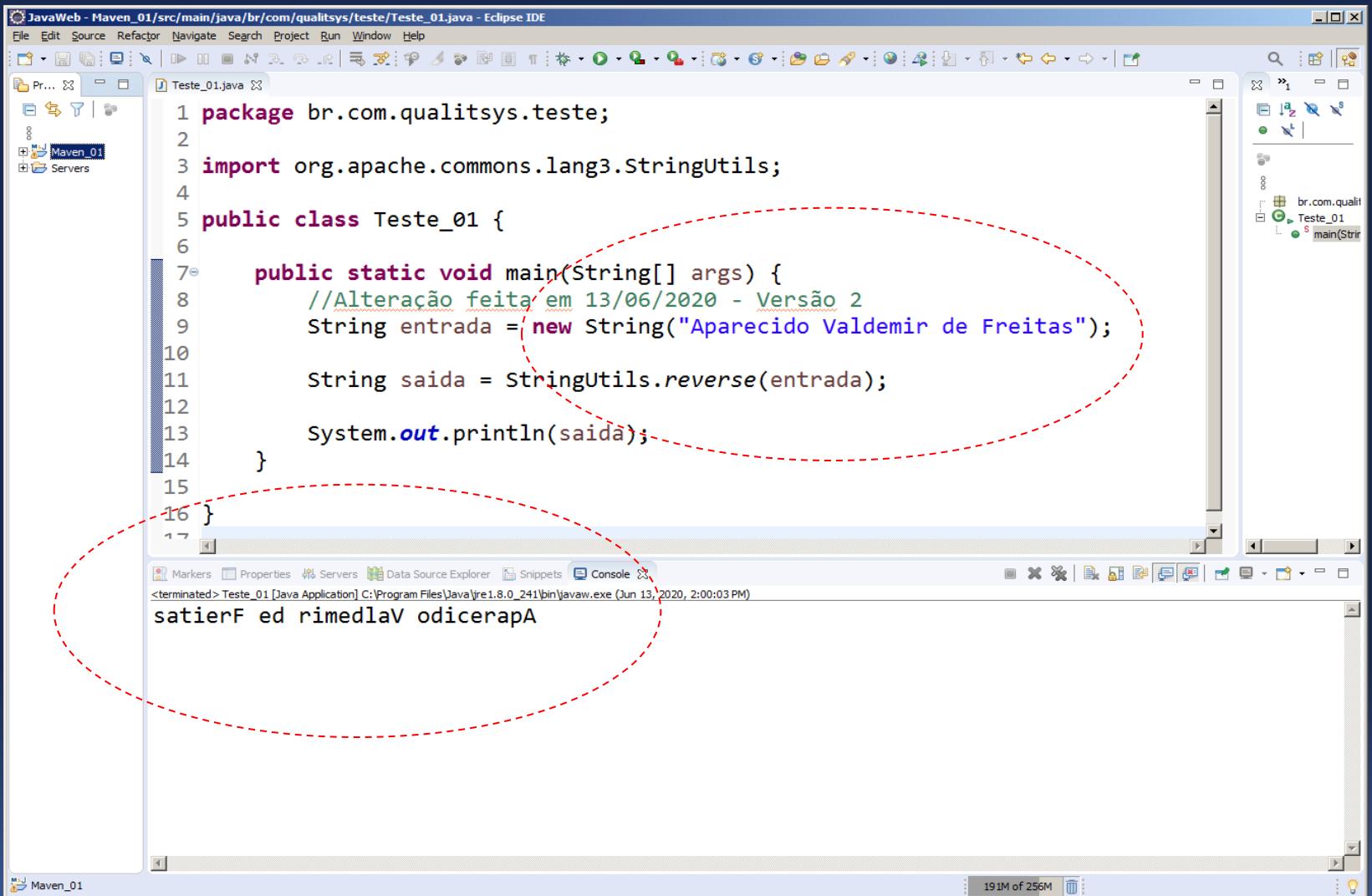


```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git show
commit 0c7056d992823830dc902e673bfa9484f0da62b6 (HEAD -> master)
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 18:00:24 2020 -0300

Projeto Maven_01 - Commons Library - Versao 01: 13/06/2020

diff --git a/Maven_01/.classpath b/Maven_01/.classpath
new file mode 100644
index 000000..90f81ed
--- /dev/null
+++ b/Maven_01/.classpath
@@ -0,0 +1,38 @@
+<?xml version="1.0" encoding="UTF-8"?>
+<classpath>
+    <classpathentry kind="src" output="target/classes" path="sr
```

Vamos agora fazer uma mudança no projeto!



The screenshot shows the Eclipse IDE interface with the following details:

- Title Bar:** JavaWeb - Maven_01/src/main/java/br/com/qualitsys/teste/Teste_01.java - Eclipse IDE
- Toolbar:** File, Edit, Source, Refactor, Navigate, Search, Project, Run, Window, Help.
- Left Sidebar:** Shows the project structure with "Maven_01" selected.
- Central Editor:** Displays the Java code for "Teste_01.java".

```
1 package br.com.qualitsys.teste;
2
3 import org.apache.commons.lang3.StringUtils;
4
5 public class Teste_01 {
6
7     public static void main(String[] args) {
8         //Alteração feita em 13/06/2020 - Versão 2
9         String entrada = new String("Aparecido Valdemir de Freitas");
10
11        String saída = StringUtils.reverse(entrada);
12
13        System.out.println(saída);
14    }
15
16 }
```

A red dashed oval highlights the string literal "Aparecido Valdemir de Freitas" on line 9.
- Right Sidebar:** Shows the package structure: br.com.qualitsys.Teste_01.main(String).
- Bottom Console:** Shows the terminal output:

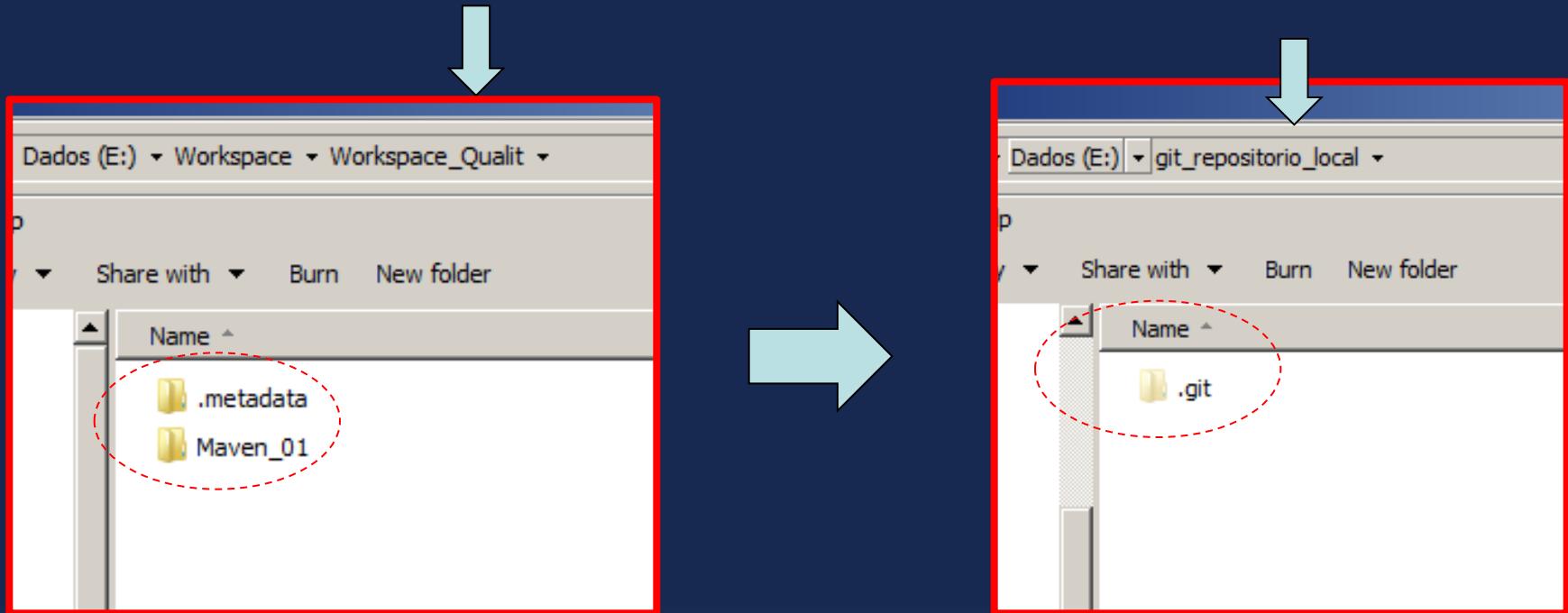
```
<terminated> Teste_01 [Java Application] C:\Program Files\Java\jre1.8.0_241\bin\javaw.exe (Jun 13, 2020, 2:00:03 PM)
satierF ed rimedlaV odicerapA
```

A red dashed oval highlights the reversed string "satierF ed rimedlaV odicerapA" in the output.

Salvando o projeto no repositório git local



- ✓ Salvando projeto Maven_01 no repositório git local



Executando novamente git status !

- ✓ Verificando novamente o status do repositório local
- ✓ **git** informa que houve modificação no projeto

Comando: **git status**

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
    (use "git restore <file>..." to discard changes in working directory)
          modified:   Maven_01/src/main/java/br/com/qualitsys/teste/
Maven_01.java
          modified:   Maven_01/target/classes/br/com/qualitsys/teste/
/Maven_01.class

no changes added to commit (use "git add" and/or "git commit -a")

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```



Gitbash no Terminal

- ✓ Adicionando **mudanças** para serem monitoradas pelo git

Comando: **git add .**

A screenshot of a terminal window titled "MINGW64:/e/git_repositorio_local". The window shows a command-line session where the user has run the command "git add ." and is awaiting further input. The terminal has a dark background with white text and a blue header bar.

```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git add .

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

Gitbash no Terminal

- ✓ Verificando novamente o status do repositório local
- ✓ git informa que há arquivos para serem comitados

Comando: **git status**



```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master
changes to be committed:
  (use "git restore --staged <file>..." to unstage)
    modified:   Maven_01/src/main/java/br/com/qualitsys/teste/
Maven_01.java
    modified:   Maven_01/target/classes/br/com/qualitsys/teste/
/Maven_01.class

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

Gitbash no Terminal

- ✓ Incluindo um ponto de controle (emitindo um commit)

Comando: `git commit -m "Projeto Maven_01 – Commit Versão 02: 13/06/20..."`



A screenshot of a terminal window titled "MINGW64:/e/git_repositorio_local". The window shows the following command being run:

```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git commit -m "Projeto Maven_01 - Commit Versao 02 : 13/06/2020
"
[master f53c784] Projeto Maven_01 - Commit Versao 02 : 13/06/2020
 2 files changed, 2 insertions(+), 2 deletions(-)

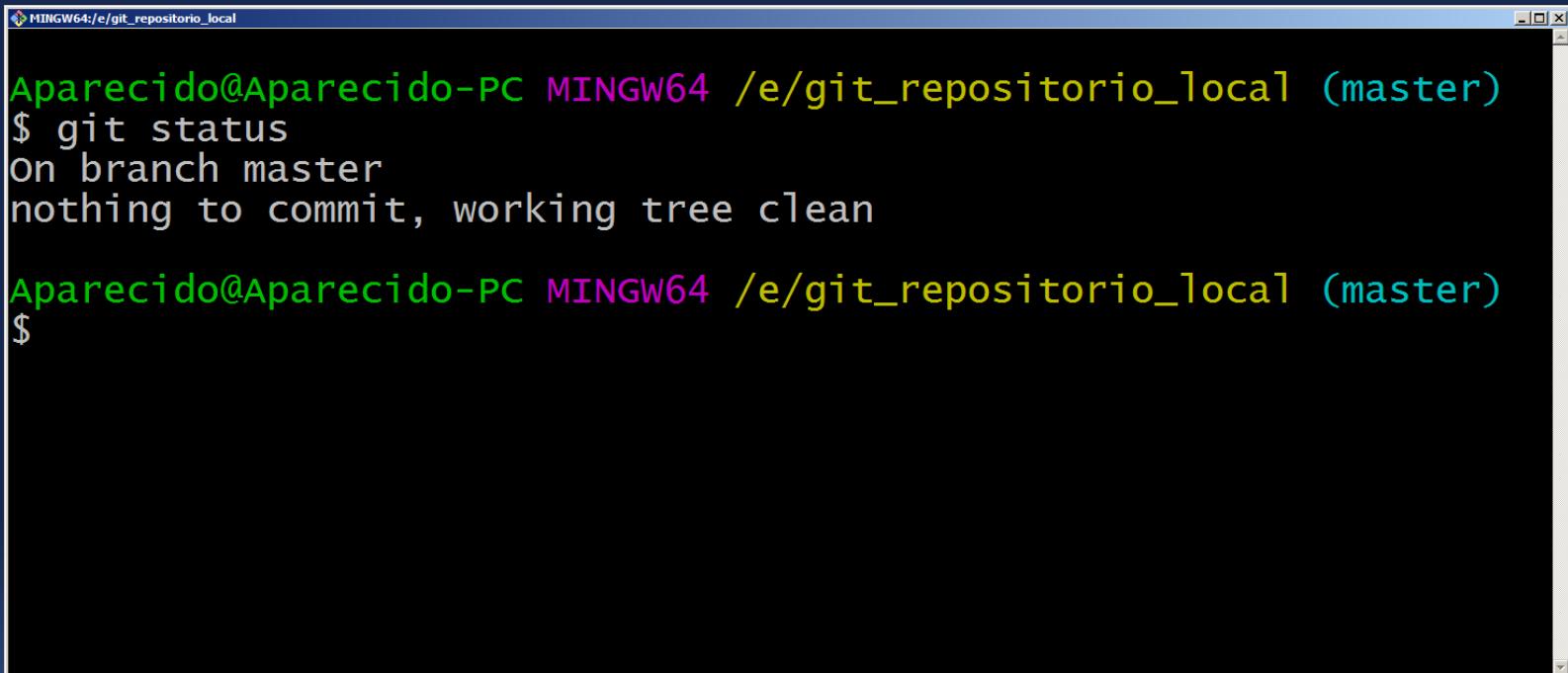
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

Two red arrows point to the commit message and the resulting commit hash and details. The terminal window has a blue header bar and a black body.

Gitbash no Terminal

- ✓ Verificando novamente o status do repositório local
- ✓ git informa que não há nada para comitar !

Comando: **git status**



A terminal window titled "MINGW64:/e/git_repositorio_local" is shown. It contains the following text:

```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git status
on branch master
nothing to commit, working tree clean

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Two red arrows point to the first two lines of the terminal output, specifically highlighting the command "\$ git status" and its response "nothing to commit, working tree clean".

Gitbash no Terminal

- ✓ checando as alterações processadas desde o primeiro commit

Comando: **git log**

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git log
commit f53c784e4a57d08f5bee8fe85676606dfbb3bee4 (HEAD -> master)
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 22:06:41 2020 -0300

    Projeto Maven_01 - Commit Versao 02 : 13/06/2020

commit a5cd48f02366102dc739535a4529e2fc5c7751cc
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 21:58:27 2020 -0300

    Projeto Maven_01 - commons Library - versao 01: 13/06/2020

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Gitbash no Terminal

- ✓ Exibindo alterações feitas no projeto

Comando: **git show**

```
MINGW64:/e/git_repositorio_local
1.java b/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
index 3019b3c..1c64716 100644
--- a/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
+++ b/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
@@ -6,8 +6,8 @@
 public class Maven_01 {

     public static void main(String[] args) {

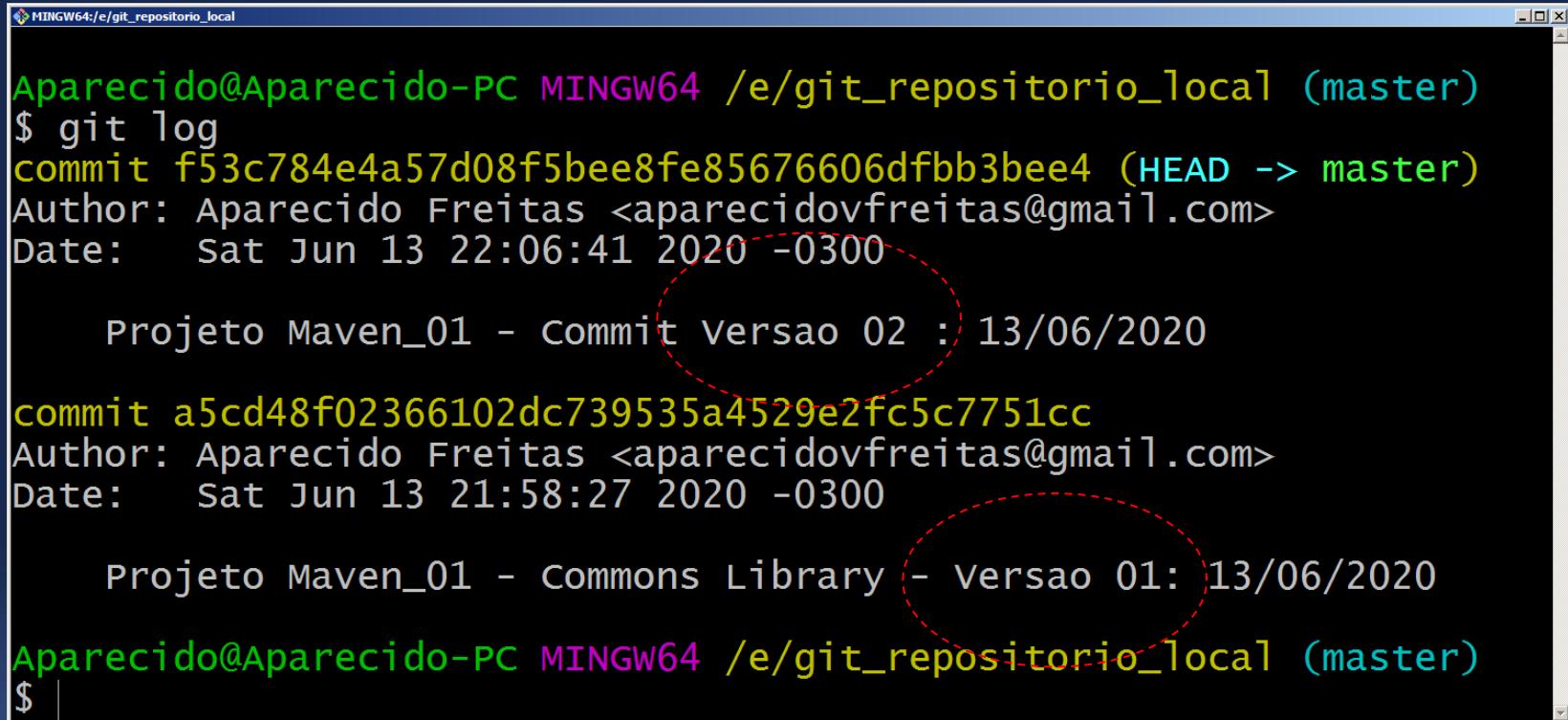
-
-        String entrada = new String("aparecido");
+        //Alteracao feita em 13/06/2020 - versao 2
+        String entrada = new String("Aparecido Valdemir de
Freitas");

        String saida = StringUtils.reverse(entrada);
```



Git log

Hash do Commit => Identificação única do Commit



```
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git log
commit f53c784e4a57d08f5bee8fe85676606dfbb3bee4 (HEAD -> master)
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 22:06:41 2020 -0300

    Projeto Maven_01 - Commit versao 02 : 13/06/2020

commit a5cd48f02366102dc739535a4529e2fc5c7751cc
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 21:58:27 2020 -0300

    Projeto Maven_01 - Commons Library - versao 01: 13/06/2020

Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ |
```

git log -p

```
MINGW64:/e/git_repositorio_local
commit f53c784e4a57d08f5bee8fe85676606dfbb3bee4 (HEAD -> master)
Author: Aparecido Freitas <aparecidovfreitas@gmail.com>
Date:   Sat Jun 13 22:06:41 2020 -0300

    Projeto Maven_01 - Commit Versao 02 : 13/06/2020

diff --git a/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
a b/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
index 3019b3c..1c64716 100644
--- a/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
+++ b/Maven_01/src/main/java/br/com/qualitsys/teste/Maven_01.java
@@ -6,8 +6,8 @@ public class Maven_01 {

    public static void main(String[] args) {

-
-        String entrada = new String("aparecido");
+        //Alteracao feita em 13/06/2020 - versao 2
+        String entrada = new String("Aparecido Valdemir de Frei
tas");

        String saida = StringUtils.reverse(entrada);
    }
}
```

Git log --oneline

```
MINGW64:/e/git_repositorio_local
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$ git log --oneline
f53c784 (HEAD -> master) Projeto Maven_01 - Commit Versao 02 : 13/06/2020
a5cd48f Projeto Maven_01 - Commons Library - Versao 01: 13/06/2020
Aparecido@Aparecido-PC MINGW64 /e/git_repositorio_local (master)
$
```

Branching

- ✓ Função **git** que essencialmente faz cópia do código (cada branch é uma cópia do código), permite que se façam mudanças em uma cópia específica e então faz **merge** de volta o **branch** principal (**main**);

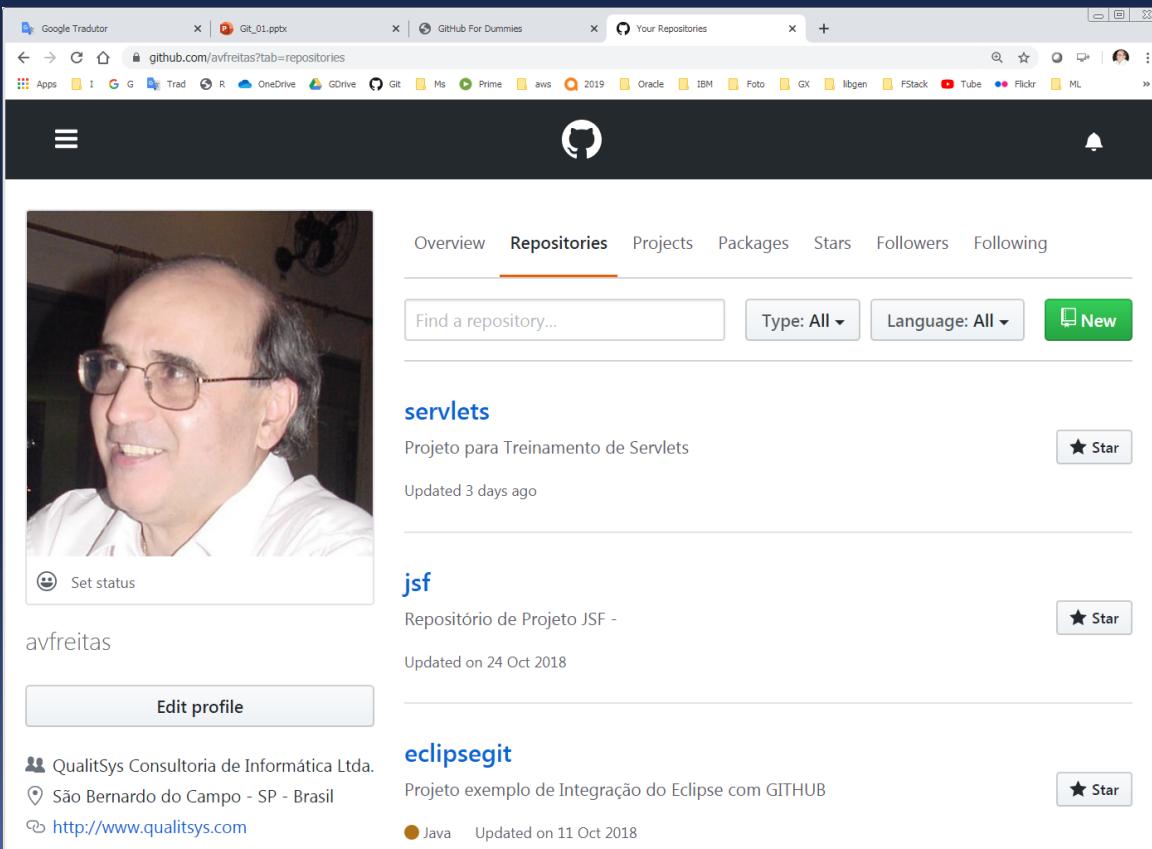


Github - Branching

- ✓ Pode-se definir uma determinada **branch** para uma frente de desenvolvimento;
- ✓ E outra para uma **branch** de produção, que representa o que os usuários irão visualizar no sistema;
- ✓ Pode-se também, por exemplo, criar três diferentes **branches**, cada qual para um **layout** diferente de um **website**. Após decidir qual **layout** será usado, pode-se deletar os outros dois **branches** e fazer **merge** do **branch** favorito no **master**.

github

- ✓ Host para repositórios **git**;
- ✓ Pode ser empregado para backup e local compartilhado para que outros desenvolvedores possam colaborar com seu projeto;



The screenshot shows a web browser window with multiple tabs open. The active tab is 'Your Repositories' on GitHub. The profile picture is a photo of a man with glasses. The 'Repositories' tab is selected, showing three repositories:

- servlets**: Projeto para Treinamento de Servlets. Updated 3 days ago. Star button.
- jsf**: Repositório de Projeto JSF -. Updated on 24 Oct 2018. Star button.
- eclipsegit**: Projeto exemplo de Integração do Eclipse com GITHUB. Updated on 11 Oct 2018. Star button.

Below the repositories, there is profile information for 'avfreitas' and a 'Edit profile' button. At the bottom, there is contact information for QualitSys Consultoria de Informática Ltda. in São Bernardo do Campo - SP - Brasil, with a link to <http://www.qualitsys.com>.

Github – features suportadas

- » **Repository:** Each repository contains all the files and folders related to your project and gives you control of permissions and collaborators' interaction with your code.
- » **Clone:** When you want to make changes to your code, you will often want to create a copy, or clone, of the project on your local computer. The cloned project is still tightly connected with the version on GitHub.com; it's just your local copy.

Github – features suportadas

- » **Fork:** *Forking* a project is when you create your own copy of the entire project. When you fork a project, GitHub.com creates a new repository with your copy of all the files. You can still suggest changes back to the original copy, but you can also take your version and go in a new direction.
- » **Branches:** GitHub.com supports branching and even provides a useful tool — *pull requests* — to compare the diff between branches and merge branches.
- » **Commits:** GitHub.com tracks all the commits that you push to its servers and gives you an easy interface for browsing the code at different branches and different commits.

Github

- ✓ Github.com oferece repositórios **públicos e privados gratuitos e ilimitados**;
- ✓ Contas **privadas gratuitas** são limitadas a **três** colaboradores;
- ✓ **Público** significa que qualquer pessoa pode ver seu código e, portanto, usar seu código;



Signup - github

- ✓ Vá até **github.com**;
- ✓ Escolha o seu plano de acesso;
- ✓ Complete os dados no formulário

Join GitHub

Create your account

There were problems creating your account.

Username *

Username can't be blank

Email

Email can't be blank

Password

Password can't be blank

8 characters OR at least 8 characters including a number and a lowercase letter.

[Learn more](#)

Email preferences

Send me occasional product updates, announcements, and offers.

