

Atelier Framework Spring-01

Maven-Partie-02

Objectifs

- **Construire un projet Maven**
 - Configurer un artéfact via des commandes Maven
 - Créer un exemple de projet Maven JAVA
 - Compiler un programme java avec Maven
 - Lancer l'exécution d'un projet Maven

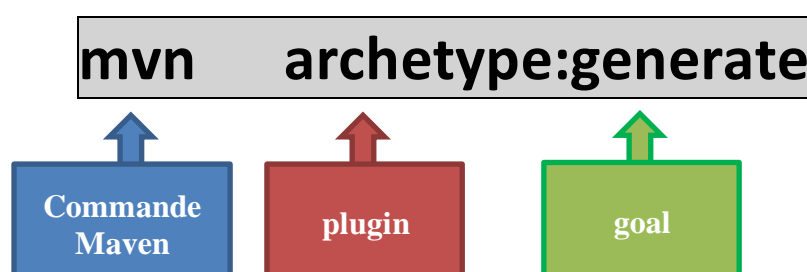
A. Créer un exemple standard d'un projet JAVA avec Maven

- Nous utilisons «**Maven**» pour générer la structure d'un projet JAVA d'une manière interactive (préparation d'une arborescence de dossiers conforme à un projet JAVA standard)
- «**Maven**» offre une commande pour construire un projet et spécifier son type (**simple**, **web**, **Ejb**, **spring**,...) en vous demandant d'indiquer un numéro (relatif au modèle voulu). Il est possible de récupérer cette liste de numéros dans un fichier texte.

Pour se faire, suivre les étapes suivantes :

1. Créer le dossier «**D:\Dev\Spring\workspace**» qui englobe, désormais, tous les projets de ces ateliers.
2. Se positionner avec l'invite de commande dans le dossier :
« **D:\Dev\Spring\workspace**»
3. Puis commencer par la création d'un projet JAVA avec l'outil Maven :

a) Taper la commande suivante :

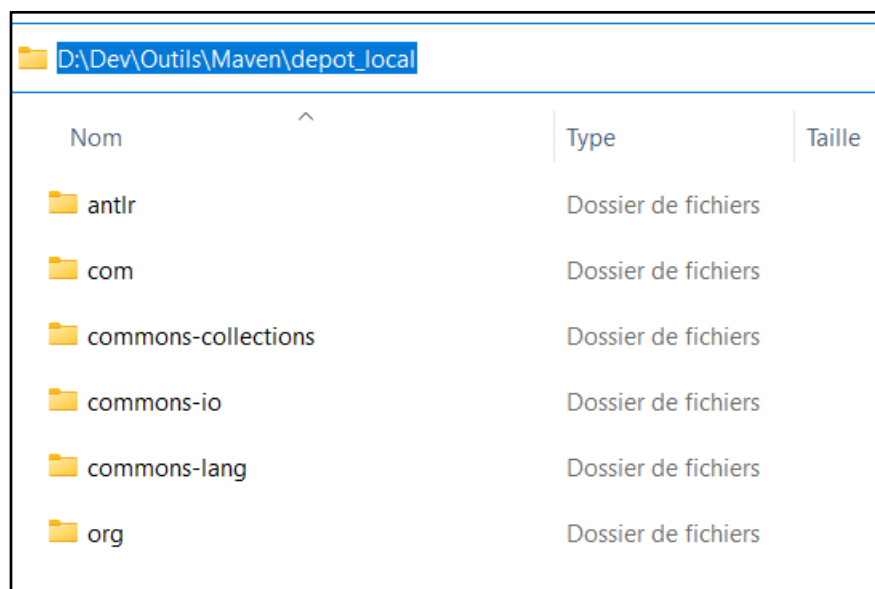


L'exécution de cette commande lance le téléchargement des plugins et des dépendances nécessaires à partir du dépôt central de Maven (ce qui nécessite pour la première fois une connexion Internet) :

```
C:\Windows\System32\cmd.e x + v
Microsoft Windows [version 10.0.26100.4946]
(c) Microsoft Corporation. Tous droits réservés.

D:\Dev\Spring\workspace>mvn archetype:generate
[INFO] Scanning for projects...
Downloading from repo.jenkins-ci.org: https://repo.jenkins-ci.org/public/org/apache/maven/plug
.0/maven-clean-plugin-3.2.0.pom
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-
lean-plugin-3.2.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-c
ean-plugin-3.2.0.pom (5.3 kB at 19 kB/s)
Downloading from repo.jenkins-ci.org: https://repo.jenkins-ci.org/public/org/apache/maven/plug
-plugins-35.pom
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-
5.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-p
.pom (9.9 kB at 146 kB/s)
Downloading from repo.jenkins-ci.org: https://repo.jenkins-ci.org/public/org/apache/maven/mave
5.pom
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/3
```

b) Remarquer que le téléchargement est réalisé dans le dépôt local personnalisé : «D:\Dev\Outils\Maven\depot_local».



c) L'exécution de la commande `mvn archetype:generate` permet de générer la structure d'un projet JAVA selon le modèle (**archetype**) spécifié par l'utilisateur. Ainsi le programme invite l'utilisateur de saisir le nombre relatif au modèle voulu :

```

class (an empty LanguageAnalyser..)
3559: remote -> uk.ac.nactem.argo:argo-analysis-engine-archetype (An archetype which contains a sample Argo (UIMA) Analysis Engine)
3560: remote -> uk.ac.nactem.argo:argo-reader-archetype (An archetype which contains a sample Argo (UIMA) Reader)
3561: remote -> uk.ac.rdg.resc.edal-ncwms-based-webapp (-)
3562: remote -> uk.co.markg.archetypes:java11-junit5 (An archetype for generate java 11 projects with junit 5.)
3563: remote -> uk.co.nemstix:basic-javaee7-archetype (A basic Java EE7 Maven archetype)
3564: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RESTful spring services with an AngularJS frontend. Includes debian deployment)
3565: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)
3566: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)
3567: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)
3568: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)
3569: remote -> vn.innotech.archetype:spring-rest-archetype (spring rest with embedded jetty)
3570: remote -> ws.nzen.format.maven:zaftig_offal_hamisha (A maven archetype for my style of project.)
3571: remote -> ws.osiris.osiris-archetype (Maven Archetype for Osiris)
3572: remote -> xyz.elemental.expath:elemental-expath-package-archetype (Maven Archetype for Elemental EXPath Packages)
3573: remote -> xyz.luan.generator:xyz-gae-generator (-)
3574: remote -> xyz.luan.generator:xyz-generator (-)
3575: remote -> za.co.absa.hyperdrive:component-archetype (-)
3576: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)
3577: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2265:

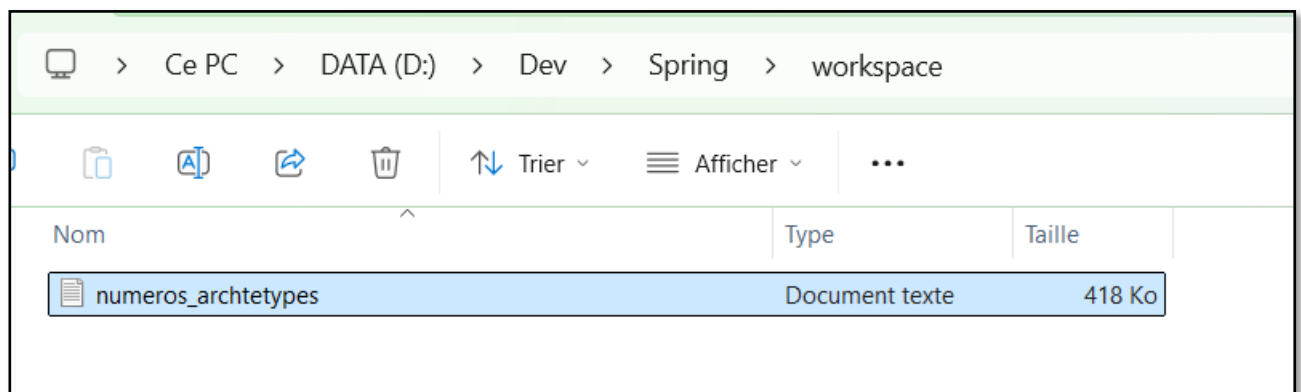
```

d) L'utilisateur peut récupérer le numéro du modèle (**archetype**) désiré à partir de la liste des numéros affichée en mode console ou bien les récupérer dans un fichier comme suit :

- Arrêter le processus courant à travers un « **Ctrl + c** ».
- Relancer la commande en redirigeant la sortie dans un fichier «**numeros_archtetypes.txt**».

```
mvn archetype:generate > numeros_archtetypes.txt
```

- Attendre quelques secondes pour générer le fichier «**numeros_archtetypes.txt**» dans le même dossier « **workspace** » :



- Enfin, arrêter le processus à travers un « **Ctrl + c** ».

- Chercher dans le fichier «**numeros_archetypes.txt**» le numéro qui correspond au modèle d'un simple projet JAVA :

org.apache.maven.archetypes:maven-archetype-quickstart

```

≡ numeros_archetypes.txt X
D: > Dev > Spring > workspace > ≡ numeros_archetypes.txt

2503 2261: remote -> org.apache.maven.archetypes:maven-archetype-plugin (An archetype w
2504 2262: remote -> org.apache.maven.archetypes:maven-archetype-plugin-site (An archety
2505 | existing Maven plugin project.)
2506 2263: remote -> org.apache.maven.archetypes:maven-archetype-portlet (An archetype wh
2507 2264: remote -> org.apache.maven.archetypes:maven-archetype-profiles (-)
2508 2265: remote -> org.apache.maven.archetypes:maven-archetype-quickstart (An archetype
2509 2266: remote -> org.apache.maven.archetypes:maven-archetype-simple (An archetype whi
2510 2267: remote -> org.apache.maven.archetypes:maven-archetype-site (An archetype which
2511 | APT, XDoc, and FML and demonstrates how to i18n your site. This archetype can be
2512 | upon an existing Maven project.)
2513 2268: remote -> org.apache.maven.archetypes:maven-archetype-site-simple (An archety

```

- A l'état actuel de cette version Maven, il s'agit du numéro :
2265. (vérifier le numéro qui convient à votre version)

4. Taper une autre fois la commande :

mvn archetype:generate

5. Un numéro par défaut sera proposé (2265). C'est le même numéro qui correspond au modèle « **maven-archetype-quickstart** » :

```

C:\Windows\System32\cmd.e x + v
class (an empty LanguageAnalyser).)
3559: remote -> uk.ac.nactem.argo:argo-analysis-engine-archetype (An archetype which contains a sample Argo (UIMA) Analy
sis Engine)
3560: remote -> uk.ac.nactem.argo:argo-reader-archetype (An archetype which contains a sample Argo (UIMA) Reader)
3561: remote -> uk.ac.rdg.resc.edal-ncwms-based-webapp (-)
3562: remote -> uk.co.markg.archetypes:javall-junit5 (An archetype for generate java 11 projects with junit 5.)
3563: remote -> uk.co.nemstix:basic-javaee7-archetype (A basic Java EE7 Maven archetype)
3564: remote -> uk.co.solong:angular-spring-archetype (So Long archetype for RESTful spring services with an AngularJS f
rontend. Includes debian deployment)
3565: remote -> us.fatehi:schemacrawler-archetype-maven-project (-)
3566: remote -> us.fatehi:schemacrawler-archetype-plugin-command (-)
3567: remote -> us.fatehi:schemacrawler-archetype-plugin-dbconnector (-)
3568: remote -> us.fatehi:schemacrawler-archetype-plugin-lint (-)
3569: remote -> vn.innotech.archetype:spring-rest-archetype (spring rest with embedded jetty)
3570: remote -> ws.nzen.format.maven:zaftig_offal_hamisha (A maven archetype for my style of project.)
3571: remote -> ws.osiris.osiris-archetype (Maven Archetype for Osiris)
3572: remote -> xyz.elemental.expath:elemental-expath-package-archetype (Maven Archetype for Elemental EXPath Packages)
3573: remote -> xyz.luan.generator:xyz-gae-generator (-)
3574: remote -> xyz.luan.generator:xyz-generator (-)
3575: remote -> za.co.absa.hyperdrive:component-archetype (-)
3576: remote -> za.co.absa.hyperdrive:component-archetype_2.11 (-)
3577: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2265:

```

6. Dans notre cas, choisir cette valeur par défaut (**2265**) en tapant «**ENTREE** ».
7. L'outil «**Maven**» propose plusieurs versions, choisir la dernière version (elle est déjà proposée) :

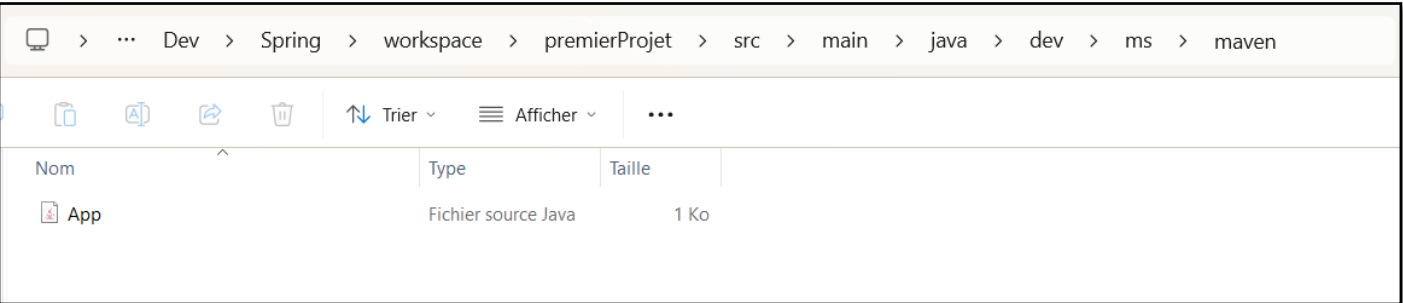
```
3577: remote -> za.co.absa.hyperdrive:component-archetype_2.12 (-)
Choose a number or apply filter (format: [groupId:]artifactId, case sensitive contains): 2265:
Choose org.apache.maven.archetypes:maven-archetype-quickstart version:
1: 1.0-alpha-1
2: 1.0-alpha-2
3: 1.0-alpha-3
4: 1.0-alpha-4
5: 1.0
6: 1.1
7: 1.3
8: 1.4
9: 1.5
Choose a number: 9: |
```

8. Taper « **ENTREE** » et indiquer les valeurs suivantes pour les paramètres suivants :
- **groupId** : **dev.ms.maven**
 - **artifactId** : **premierProjet**
 - **version** : **1.0-SNAPSHOT** (prendre la valeur donnée par défaut, en tapant «**ENTREE** »)
 - **package** : **dev.ms.maven** (même valeur que « **groupId** », prendre la valeur donnée par défaut, en tapant «**ENTREE** »)
9. Finalement, taper « **y** » pour confirmer la génération de la structure du projet.
10. Ainsi, l'outil **Maven** indique le succès de la construction d'un projet « **premierProjet** » avec les caractéristiques indiquées :

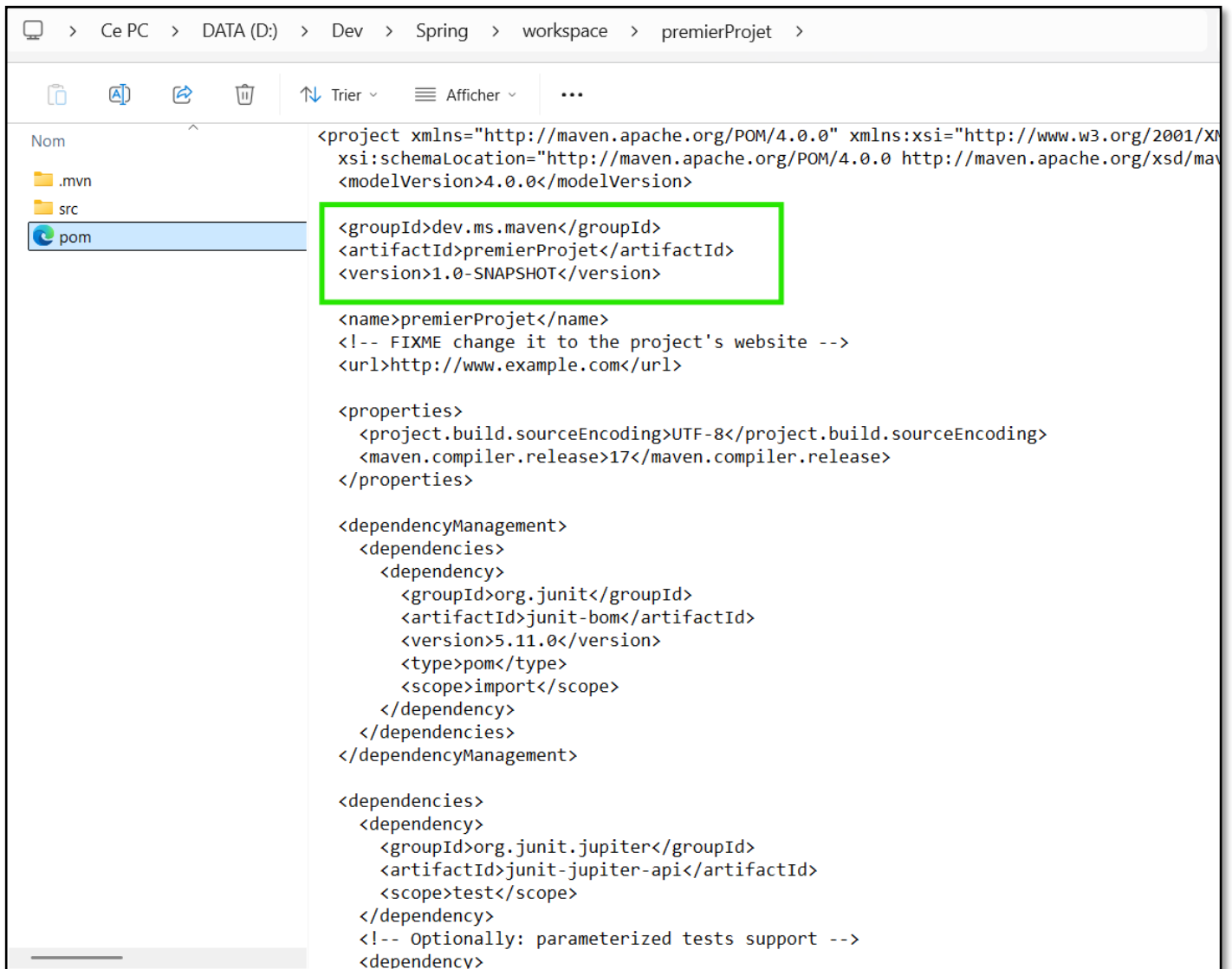
```
C:\Windows\System32\cmd.e  x  +  v

[INFO] Using property: javaCompilerVersion = 17
[INFO] Using property: junitVersion = 5.11.0
Define value for property 'groupId': dev.ms.maven
Define value for property 'artifactId': premierProjet
Define value for property 'version' 1.0-SNAPSHOT:
Define value for property 'package' dev.ms.maven:
Confirm properties configuration:
javaCompilerVersion: 17
junitVersion: 5.11.0
groupId: dev.ms.maven
artifactId: premierProjet
version: 1.0-SNAPSHOT
package: dev.ms.maven
Y: y
[INFO] -----
[INFO] Using following parameters for creating project from Archetype: maven-archetype-quickstart:1.5
[INFO] -----
[INFO] Parameter: groupId, Value: dev.ms.maven
[INFO] Parameter: artifactId, Value: premierProjet
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] Parameter: package, Value: dev.ms.maven
[INFO] Parameter: packageInPathFormat, Value: dev/ms/maven
[INFO] Parameter: junitVersion, Value: 5.11.0
[INFO] Parameter: package, Value: dev.ms.maven
[INFO] Parameter: groupId, Value: dev.ms.maven
[INFO] Parameter: artifactId, Value: premierProjet
[INFO] Parameter: javaCompilerVersion, Value: 17
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[WARNING] Don't override file D:\Dev\Spring\workspace\premierProjet\src\main\java\dev\ms\maven
[WARNING] Don't override file D:\Dev\Spring\workspace\premierProjet\src\test\java\dev\ms\maven
[WARNING] CP Don't override file D:\Dev\Spring\workspace\premierProjet\src\main\java\dev\ms\maven
[INFO] Project created from Archetype in dir: D:\Dev\Spring\workspace\premierProjet
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 04:14 min
[INFO] Finished at: 2025-08-19T17:56:17+01:00
[INFO] -----
```

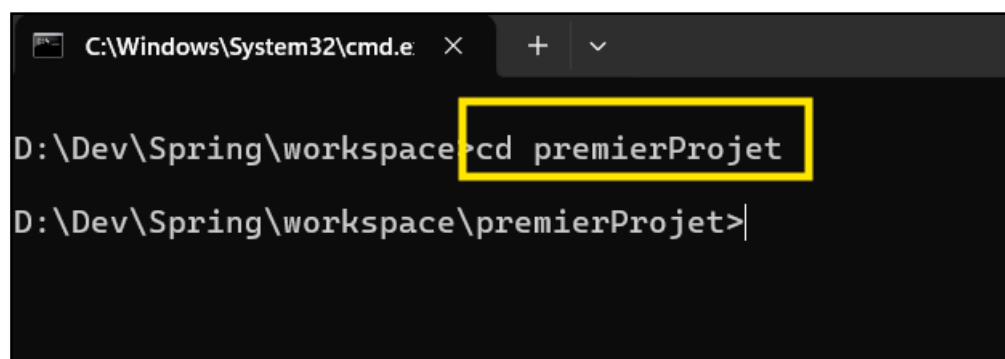
11. Remarquer la construction du projet « **premierProjet** » dans le dossier « **workspace** » comportant un fichier source « **App.java** » :



12. Un projet de type **Maven** est caractérisé par un fichier « **pom.xml** » qui contient la description du projet et les informations nécessaires à sa génération :



13. Revenons maintenant au mode console, accéder au dossier racine du projet « **premierProjet** » :



14. Compiler le projet en utilisant l'outil **Maven** avec le plugin **compile**

mvn compile


```

C:\Windows\System32\cmd.e x + v
D:\Dev\Spring\workspace\premierProjet>mvn compile
[INFO] Scanning for projects...
[INFO] -----< dev.ms.maven:premierProjet >-----
[INFO] Building premierProjet 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] -----[ jar ]-----
Downloading from repo.jenkins-ci.org: https://repo.jenkins-ci.org/public/org/apache/maven/maven-plugin
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-plugin
at 29 kB/s)
Downloading from repo.jenkins-ci.org: https://repo.jenkins-ci.org/public/org/apache/maven/maven-jar-plugin
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-jar-plugin
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-jar-plugin

```

Toujours, lorsque vous utilisez un plugin pour la première fois, Maven télécharge tout d'abord le plugin puis il l'utilise.

15. Le plugin «**compile**» compile tous les fichiers sources JAVA situés sous «**src/main/java**» (Dans ce cas, il s'agit d'un seul fichier «**App.java**»). Le résultat de la compilation est enregistré dans un dossier «**target/classes**» :

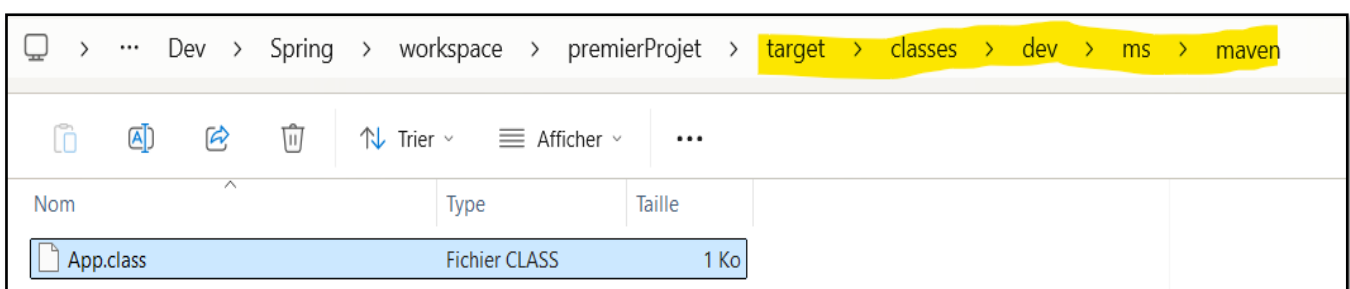
```

Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-xml/3.0.0
Downloaded from central: https://repo.maven.apache.org/maven2/com/thoughtworks/qdox/qdox/2.0.3/qdox-2.0.3.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-utils/4.0.0
[INFO] Recompiling the module because of changed source code.
[INFO] Compiling 1 source file with javac [debug release 17] to target/classes
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 18.057 s

```

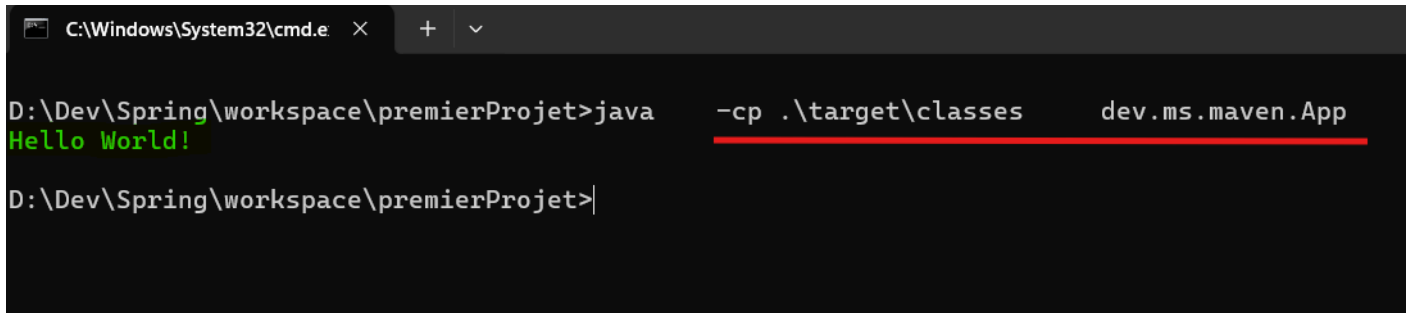
16. Suite à une compilation avec succès, un fichier «**App.class**» est généré sous le dossier :

D:\Dev\Micro_Services\workspace\premierProjet\target\classes\dev\ms\maven



17. Lancer, finalement, l'exécution de la class «**App**» en utilisant la commande :


```
java -cp .\target\classes dev.ms.maven.App
```



A screenshot of a Windows Command Prompt window. The title bar shows 'C:\Windows\System32\cmd.e' with a close button and a dropdown menu. The command prompt shows the following text:

```
D:\Dev\Spring\workspace\premierProjet>java -cp .\target\classes dev.ms.maven.App
Hello World!
D:\Dev\Spring\workspace\premierProjet>
```

The command `java -cp .\target\classes dev.ms.maven.App` is underlined in red. The output `Hello World!` is displayed in green text.