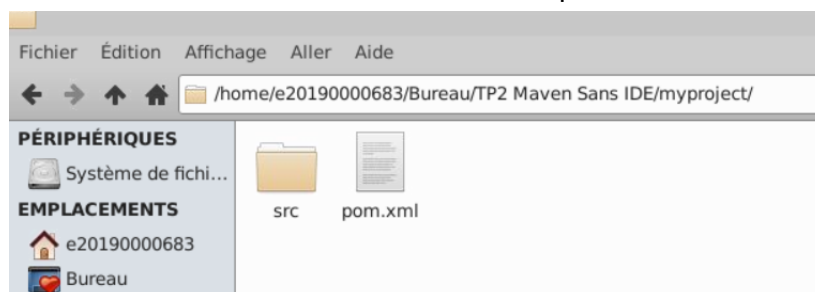


Pour créer notre projet dans le terminal, nous utilisons la commande suivante :

```
mvn archetype:generate -DgroupId=com.yourcompany -DartifactId=myproject
-DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
```

```
[INFO] -----< org.apache.maven:standalone-pom >-----
[INFO] Building Maven Stub Project (No POM) 1
[INFO] -----[ pom ]-----
[INFO] >>> maven-archetype-plugin:3.2.0:generate (default-cli) > generate-source
s @ standalone-pom >>>
[INFO] <<< maven-archetype-plugin:3.2.0:generate (default-cli) < generate-source
s @ standalone-pom <<<
[INFO]
[INFO] --- maven-archetype-plugin:3.2.0:generate (default-cli) @ standalone-pom
---
[INFO] Generating project in Batch mode
[INFO] -----
[INFO] Using following parameters for creating project from Old (1.x) Archetype:
maven-archetype-quickstart:1.0
[INFO] -----
[INFO] Parameter: basedir, Value: /home/e20190000683/Bureau/TP2 Maven Sans IDE
[INFO] Parameter: package, Value: com.yourcompany
[INFO] Parameter: groupId, Value: com.yourcompany
[INFO] Parameter: artifactId, Value: myproject
[INFO] Parameter: packageName, Value: com.yourcompany
[INFO] Parameter: version, Value: 1.0-SNAPSHOT
[INFO] project created from Old (1.x) Archetype in dir: /home/e20190000683/Bureau/TP2 Maven Sans IDE/myproject
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 6.729 s
[INFO] Finished at: 2021-12-06T11:14:35+01:00
[INFO] -----
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1:~/Bureau/TP2 Maven Sans IDE$
```

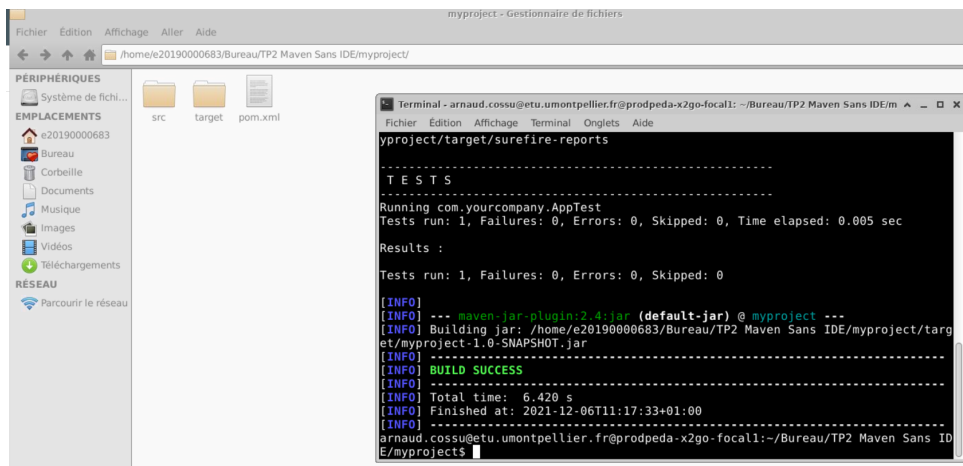
Nous obtenons la structure suivante ainsi qu'un POM :



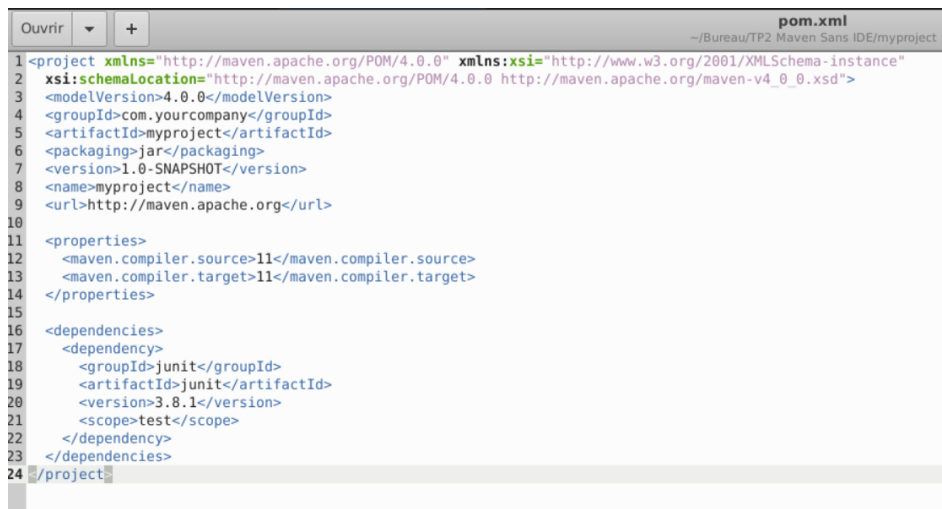
Le premier mvn package est un échec :

```
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1:~/Bureau/TP2 Maven Sans IDE$ mvn package
[INFO] Scanning for projects...
[INFO] -----
[INFO] BUILD FAILURE
[INFO] -----
[INFO] Total time: 0.100 s
[INFO] Finished at: 2021-12-06T11:16:09+01:00
[INFO] -----
[ERROR] The goal you specified requires a project to execute but there is no POM
in this directory (/home/e20190000683/Bureau/TP2 Maven Sans IDE). Please verify
you invoked Maven from the correct directory. -> [Help 1]
[ERROR]
[ERROR] To see the full stack trace of the errors, re-run Maven with the -e swit
ch.
[ERROR] Re-run Maven using the -X switch to enable full debug logging.
[ERROR]
[ERROR] For more information about the errors and possible solutions, please rea
d the following articles:
[ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MissingProject
Exception
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1:~/Bureau/TP2 Maven Sans IDE$
```

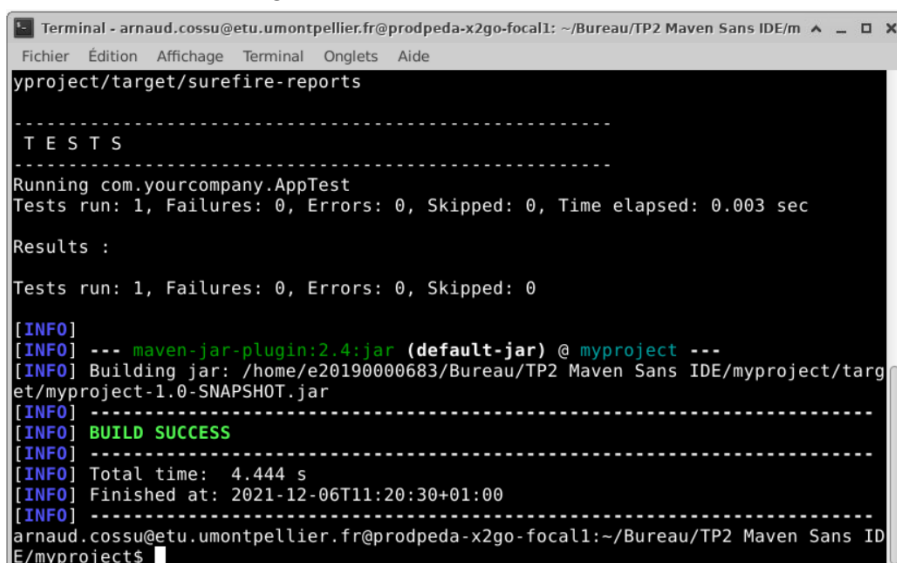
Nous ne sommes pas dans le bon terminal, il faut le lancer dans le projet. En se déplaçant dans le dossier la commande fonctionne parfaitement :



Même si notre mvn package à fonctionné, nous ajoutons les propriétés de maven, entre la ligne URL et Dependencies (et nous sauvegardons) :



Après un mvn package : aucuns problèmes



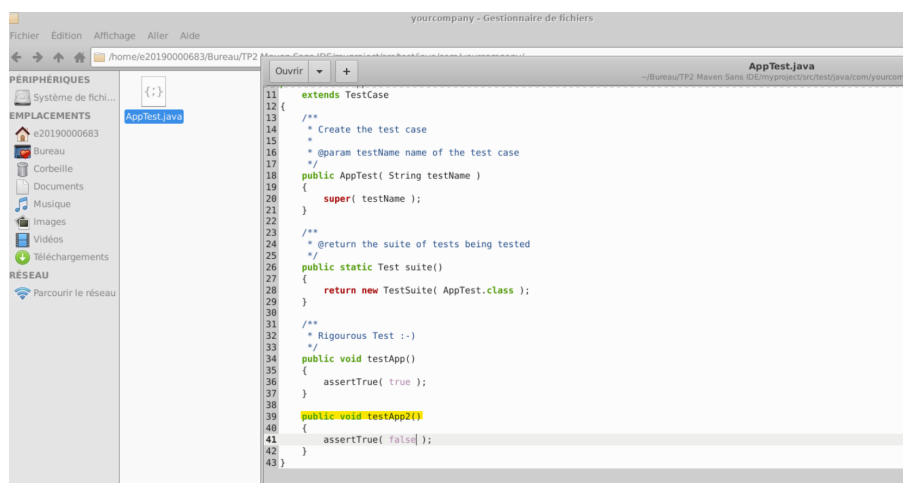
J'effectue ensuite un mvn clean :

```
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1:~/Bureau/TP2 Maven Sans IDE/myproject$ mvn clean
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.yourcompany:myproject >-----
[INFO] Building myproject 1.0-SNAPSHOT
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ myproject ---
[INFO] Deleting /home/e20190000683/Bureau/TP2 Maven Sans IDE/myproject/target
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] -----
[INFO] Total time: 0.534 s
[INFO] Finished at: 2021-12-06T11:21:19+01:00
[INFO] -----
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1:~/Bureau/TP2 Maven Sans IDE/myproject$
```

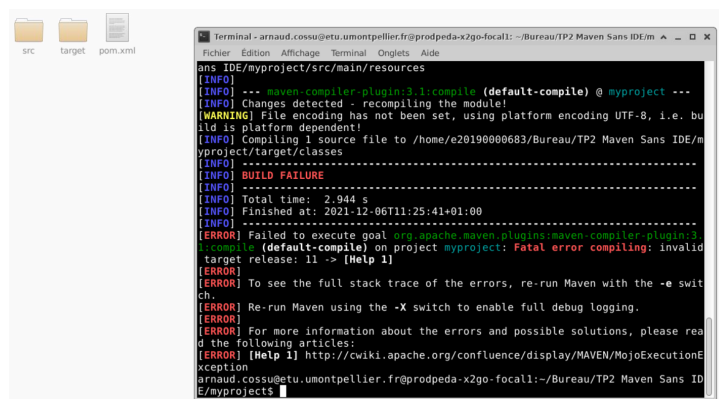
Nous nous rendons donc pour la question suivante dans le dossier qui contient les classes de test, afin de modifier le fichier et ajouter un test qui échoue :

/TP2 Maven Sans IDE/myproject/src/test/java/com/yourcompany/

Nous ajoutons un test nommé testApp2 et qui prend false dans un assertTrue, cela va donc échouer.



Nous faisons donc mvn clean test et nous voyons bien que nous avons des erreurs :



Nous retournons dans le fichier pour supprimer le test qui échoue :



```
6
7 /**
8  * Unit test for simple App.
9  */
10 public class AppTest
11     extends TestCase
12 {
13     /**
14      * Create the test case
15      * @param testName name of the test case
16      */
17     public AppTest( String testName )
18     {
19         super( testName );
20     }
21
22     /**
23      * @return the suite of tests being tested
24      */
25     public static Test suite()
26     {
27         return new TestSuite( AppTest.class );
28     }
29
30     /**
31      * Rigorous Test :-)
32      */
33     public void testApp()
34     {
35         assertTrue( true );
36     }
37
38 }
```

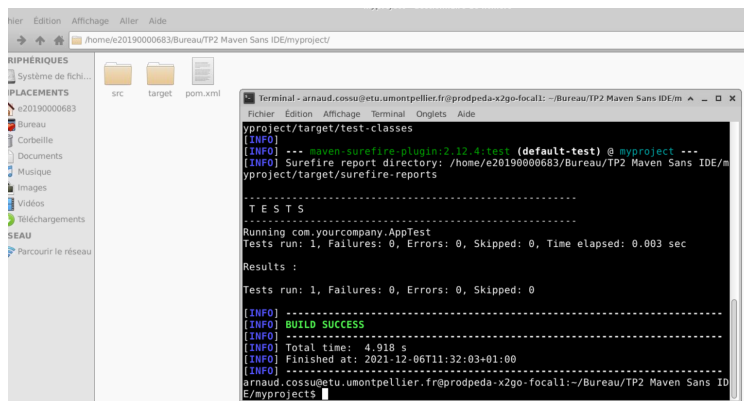
Nous effectuons un nouveau mvn clean test : Cela échoue aussi : Problème la version 11 de java. Nous passons donc en 1.8 dans le pom en changeant les properties :

```
<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/maven-v4_0_0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>com.yourcompany</groupId>
  <artifactId>myproject</artifactId>
  <packaging>jar</packaging>
  <version>1.0-SNAPSHOT</version>
  <name>myproject</name>
  <url>http://maven.apache.org</url>

  <properties>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>junit</groupId>
      <artifactId>junit</artifactId>
      <version>3.8.1</version>
      <scope>test</scope>
    </dependency>
  </dependencies>
</project>
```

Puis nouveau clean test qui cette fois fonctionne :



```
ypjproject/target/test-classes
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ myproject ---
[INFO] Surefire report directory: /home/e2019000683/Bureau/TP2 Maven Sans IDE/m
ypjproject/target/surefire-reports

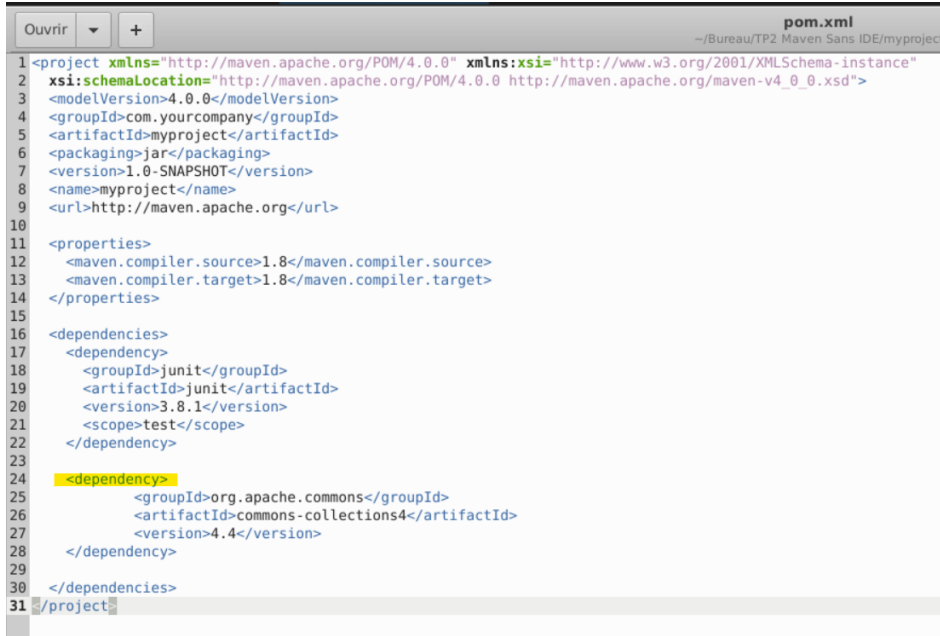
T E S T S
Running com.yourcompany.AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.003 sec

Results :

Tests run: 1, Failures: 0, Errors: 0, Skipped: 0

[INFO] ---
[INFO] BUILD SUCCESS
[INFO] ---
[INFO] Total time: 4.918 s
[INFO] Finished at: 2021-12-06T11:32:03+01:00
[INFO] ---
arnaud.cossu@etu.umontpellier.fr@prodpeda-x2go-focal1: ~/Bureau/TP2 Maven Sans IDE
E/myprojects
```

Dans le POM, nous ajoutons ensuite les dépendances vers les apache commons 4.4 :



```
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-v4_0_0.xsd">
3   <modelVersion>4.0.0</modelVersion>
4   <groupId>com.yourcompany</groupId>
5   <artifactId>myproject</artifactId>
6   <packaging>jar</packaging>
7   <version>1.0-SNAPSHOT</version>
8   <name>myproject</name>
9   <url>http://maven.apache.org</url>
10
11   <properties>
12     <maven.compiler.source>1.8</maven.compiler.source>
13     <maven.compiler.target>1.8</maven.compiler.target>
14   </properties>
15
16   <dependencies>
17     <dependency>
18       <groupId>junit</groupId>
19       <artifactId>junit</artifactId>
20       <version>3.8.1</version>
21       <scope>test</scope>
22     </dependency>
23
24     <dependency>
25       <groupId>org.apache.commons</groupId>
26       <artifactId>commons-collections4</artifactId>
27       <version>4.4</version>
28     </dependency>
29
30   </dependencies>
31 </project>
```

Ensuite nous modifions la classe Main afin d'y ajouter les lignes suivantes :

```
import org.apache.commons.collections4.multimap.ArrayListValuedHashMap;
import org.apache.commons.collections4.MultiValuedMap;

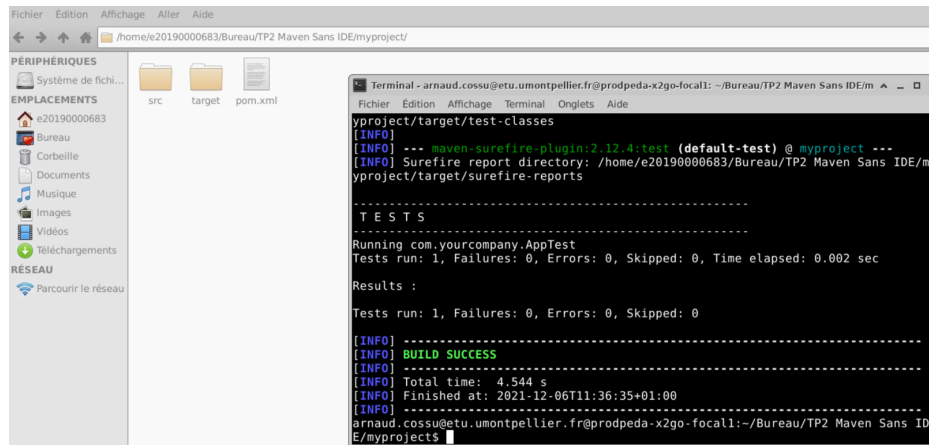
public class App
{
    public static void main(String[] args) {
        MultiValuedMap<String, String> valueMap = new
ArrayListValuedHashMap<String, String>();
        valueMap.put("h1", "value1");
        valueMap.put("h1", "value2");

        System.out.println(valueMap.get("h1"));
    }
}
```



```
1 package com.yourcompany;
2
3 /**
4  * Hello world!
5  *
6  */
7 import org.apache.commons.collections4.multimap.ArrayListValuedHashMap;
8 import org.apache.commons.collections4.MultiValuedMap;
9
10 public class App
11 {
12     public static void main(String[] args) {
13         MultiValuedMap<String, String> valueMap = new ArrayListValuedHashMap<String, String>();
14         valueMap.put("h1", "value1");
15         valueMap.put("h1", "value2");
16
17         System.out.println(valueMap.get("h1"));
18     }
19 }
```

Nous vérifions que nous avons les bons imports et nous effectuons un mvn clean test et tout fonctionne :



Les fichiers téléchargés sont stockés sur la machine.