**TỔNG QUAN VỀ MAVEN VÀ CÁC BƯỚC ĐỂ TẠO MỘT MAVEN PROJECT**

**Tổng quan về maven.**

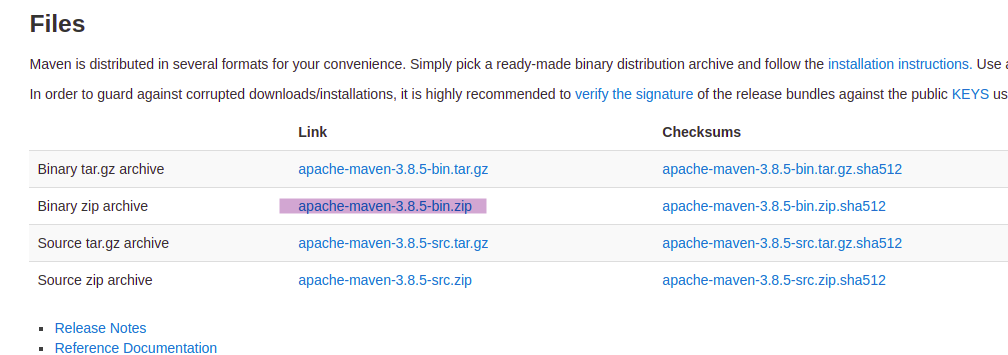
Maven là công cụ quản lý và thiết lập tự động 1 dự án phần mềm. Nó cho phép chúng ta có thể quản lý các module, thành phần phụ thuộc (các thư viện tiện ích: POI - đọc ghi excel, common-lang3- xử lý chuỗi, TestNG, Junit, ...) của dự án theo version.

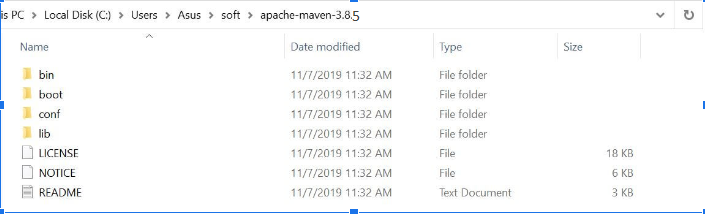
Maven hỗ trợ việc tự động hóa các quá trình tạo dự án ban đầu, thực hiện biên dịch, kiểm thử, đóng gói và triển khai sản phẩm.

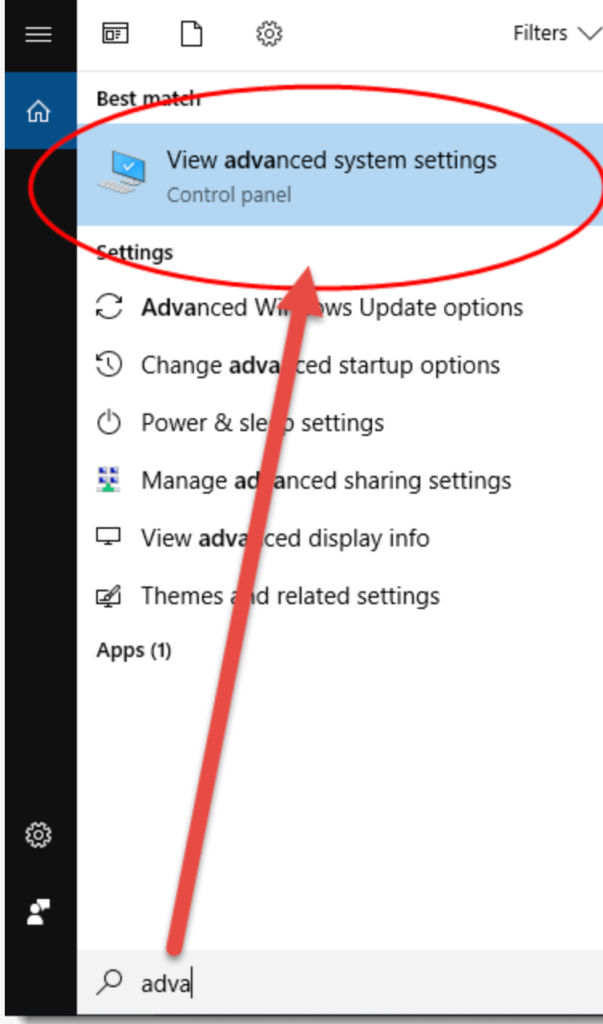
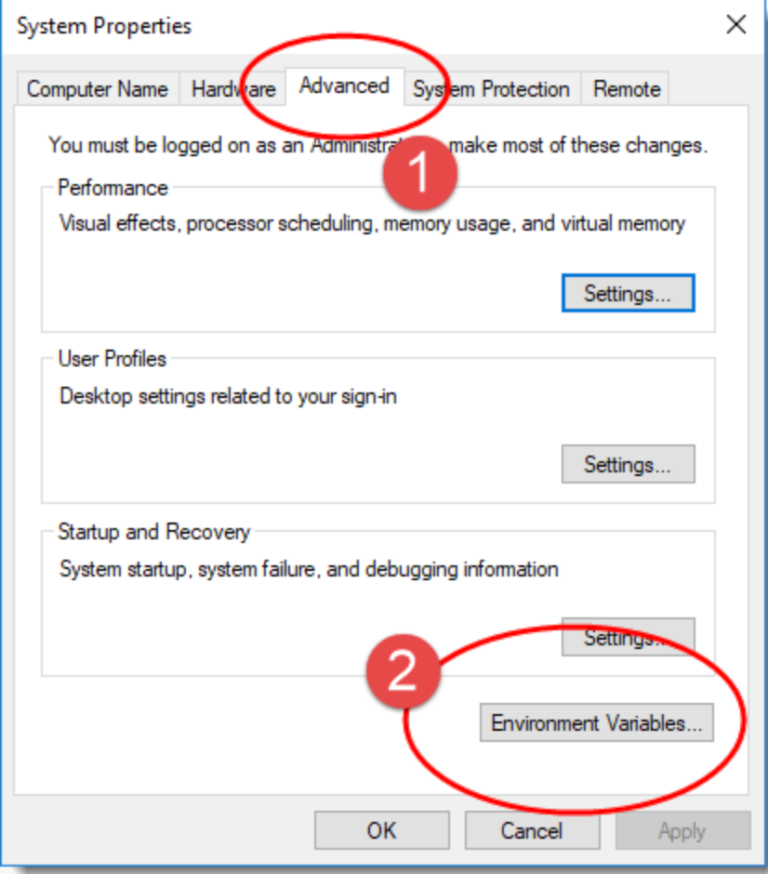
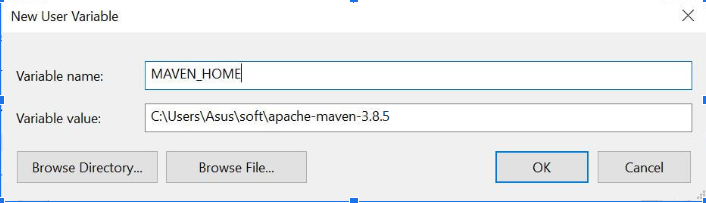
Việc của chúng ta sẽ là khai báo thông tin các module, dependency cần cho dự án. Việc download, build và tích hợp vào dự án Maven sẽ thay chúng ta làm.

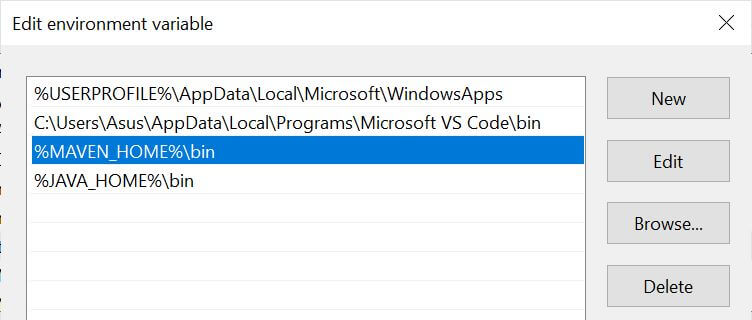
Maven dùng khái niệm Project Object Model (POM) để mô tả việc build project, các thành phần phụ thuộc và các module. Nó định nghĩa trước trình biên dịch, đóng gói và thứ tự hoạt động để mọi việc diễn ra tốt nhất.

**I. Cài đặt maven**

1. Các bạn lên trang chủ của Maven để tải file môi trường Maven về máy: <https://maven.apache.org/download.cgi>  
   
2. Thực hiện giải nén file vừa tải về.  
   Sau đó Giải nén vào thư mục nào bạn thích để cài đặt Maven.  
   Ví dụ: C:\Users\Asus\soft\apache-maven-3.8.5



1. Cài đặt biến môi trường.  
    #1.TÌm kiếm: **View advanced system settings**  
      
     
   #2.Chọn tab **Advanced** và chọn **Environment Variables…**  
     
     
     
   #3. **System variables** -> **New  
     
     
   =>** Trỏ biến môi trường MAVEN\_HOME đến thư mục chúng ta giải nén maven trước đó -> OK

#4. Bước cuối cùng, trong **system variables**, tìm đến **Path**, chọn **Edit**. Trong cửa sổ environment variable , chọn New và thêm **%MAVEN\_HOME%\bin** vào .  
Chọn OK để lưu kết quả.  
  
  
  
#5. Restart máy, mở cửa sổ command line (CMD) - có thể search từ khóa CMD trong ô tìm kiếm.  
Kiểm tra version của maven bằng câu lệnh: **mvn --version**  
  
 C:\Users\Asus>mvn --version

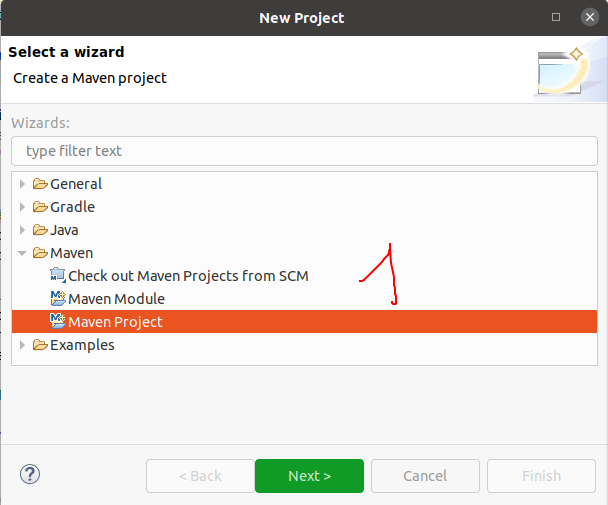
Apache Maven 3.8.5 (05c21c65bdfed0f71a2f2ada8b84da59348c4c5d)

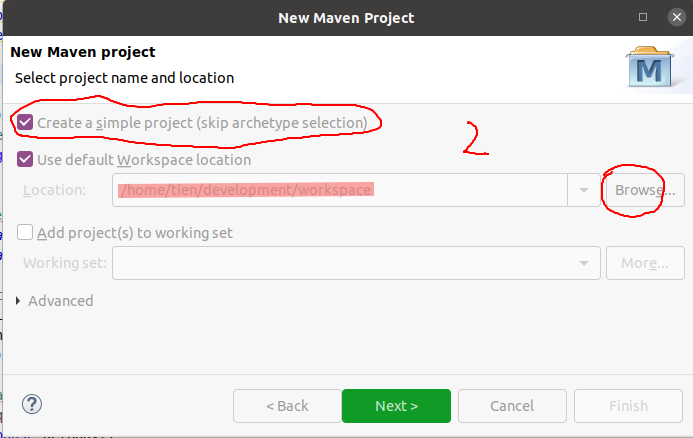
Maven home: C:\Users\Asus\soft\apache-maven-3.8.5\bin\..

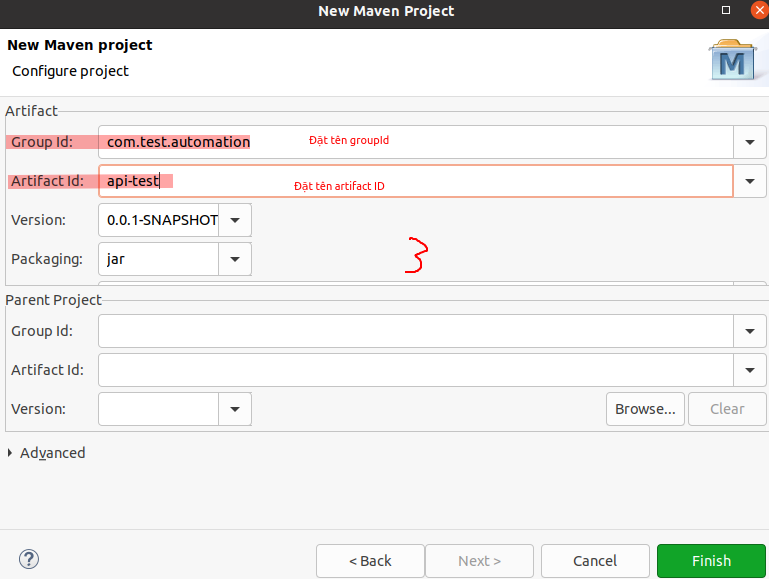
C:\Users\Asus>

**II. Tạo maven project**

Trên Eclipse, các bạn chọn: **File → new → project → Maven Project**



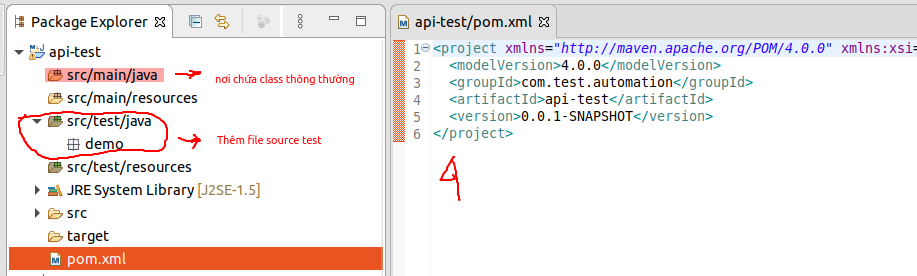




Các bạn có thể đặt tên groupid và artifact Id cho project của các bạn.

=> Finish

#Cấu trúc maven project sẽ như sau



* **src/main/java**  : Folder này, chúng ta có thể tạo ra các package và khai báo các class   
  thông thường như chúng ta vẫn thường làm : Animal.java, Student.java, Main..
* **src/test/java** : Folder này chúng ta sẽ dành cho các package v, class test

testNG, Junit..

* **pom.xml** : file để khai báo/ quản lý các thư viện phụ thuộc.

**II. Thêm mới một thư viện vào project.**

Hiện nay, có rất nhiều thư viện tiện java ích mã nguồn mở được các cá nhân, tổ chức cung cấp miễn phí. Chúng ta có thể tải về và import vào project của chúng ta để sử dụng cho chúng ta. Rõ ràng là việc này sẽ giúp chúng ta nhàn hơn trong việc phát triển dự án thay vì phải ngồi làm tất cả từ đầu.

*Tôi sẽ lấy 1 ví dụ cụ thể:*  
Thư viện Commons Lang 3 do Apache phát hành là một thư viện hỗ trợ rất nhiều class và method tiện ích trong việc xử lý chuỗi String, mảng.  
VD: Lớp StringUtils.java

import org.apache.commons.lang3.StringUtils;

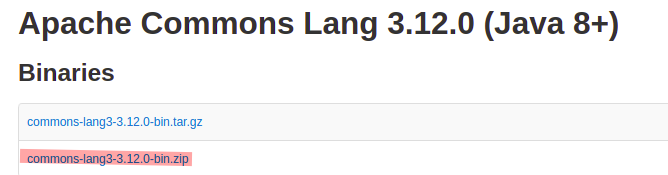
….  
 StringUtils.isNumeric(text): Kiểm tra đoạn chuỗi text có phải toàn là số hay không

StringUtils.isEmpty(text): Kiểm tra đoạn chuỗi text có phải là null hay rỗng ("")  
 Để sử dụng những tiện ích này, việc cần làm của chúng ta là tìm các thư viện này và tải về, sau đó tích hợp vào dự án của chúng ta dưới dạng một thư viện dependency. Các thư viện này được các tổ chức phát hành đóng gói dưới dạng **.jar**

Có 2 cách để tải về và tích hợp thư viện vào dự án.

1. **Cách truyền thống**

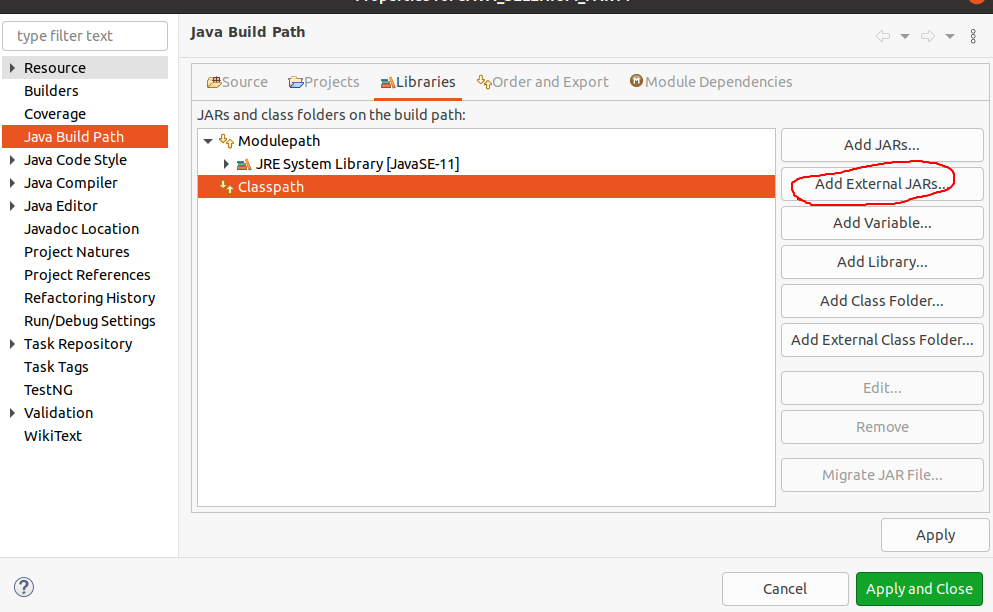
Chúng ta sẽ phải lên trang chủ của tổ chức phát hành, tìm kiếm thư viện và version phù hợp sau đó download file .  
VD: https://commons.apache.org/proper/commons-lang/download\_lang.cgi



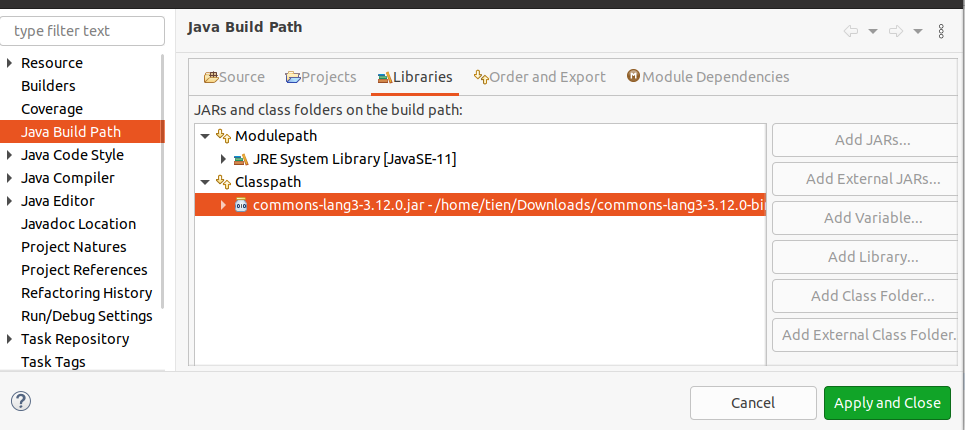
Sau khi giải nén, thì chúng ta chỉ quan tâm đến file: **commons-lang3\*\*.jar**

Bước tiếp theo đó là tích hợp vào project của chúng ta:

* Chuột phải vào project
* Chọn Build Path.
* Chọn Configure Build Path.



* Chọn Libraries và Add External JARs.
* Chọn common-lang3\*\*.jar file ở thự mục vừa giải nén.

Click and Apply and Ok

Với cách này thì chúng ta phải làm thủ công rất nhiều bước.   
Trường hợp số lượng các gói thư viện tiện ích được thêm vào dự án càng ngày càng phình to sẽ khiến chúng ta gặp khó khăn để quản lý chúng.  
Đặc biệt là, nếu muốn nâng cấp/thay đổi version của một thư viện trong đó thì chúng ta lại phải tải mới về và import từ bước đầu, rât tốn công và mất thời gian

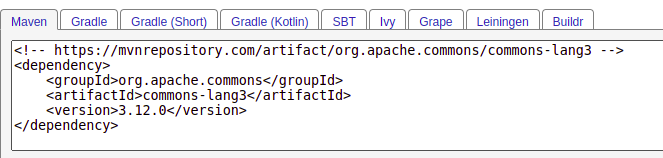
1. **Sử dụng maven, pom.xml file**

Để khắc phục nhược điểm của phương pháp truyền thống, Maven đã ra đời giúp quản lý, nâng cấp các thư viện hiệu quả hơn. Các thư viện được phát triển bởi các cá nhân, tổ chức sau đó được quản lý bới Maven dựa trên thông tin group-id và artifact Id.  
Chúng ta chỉ cần khai báo thông tin gói thư viện (group-id, artifact-id, version, ...) trong file pom.xml, thư viện sẽ được tải và tích hợp vào dự án một cách tự động.  
Nếu chúng ta muốn nâng/hạ version của thư viện thì version mới cũng sẽ được tự động cập   
nhật.

VD: Thông tin về thư viện common-lang 3 trên trang chủ của maven.  
 <https://mvnrepository.com/artifact/org.apache.commons/commons-lang3>

Version 3.12:

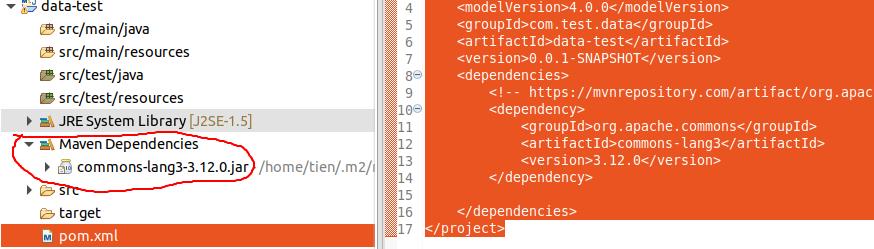
<https://mvnrepository.com/artifact/org.apache.commons/commons-lang3/3.12.0>  
 #Thông tin mô tả:



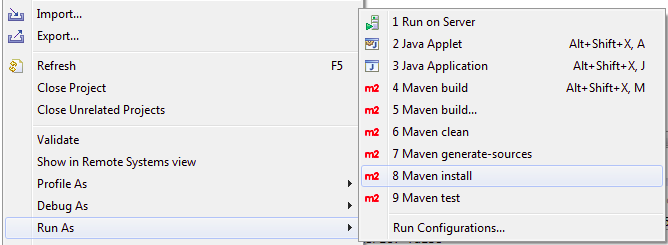
Các bạn chỉ cần copy thông tin về thư viện và đặt vào trong thẻ tab <dependencies> trong   
 pom.xml. (nếu chưa có, các bạn có thể tự tạo thẻ)

| <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 https://maven.apache.org/xsd/maven-4.0.0.xsd"*>  <modelVersion>4.0.0</modelVersion>  <groupId>com.test.data</groupId>  <artifactId>data-test</artifactId>  <version>0.0.1-SNAPSHOT</version>  <dependencies>  <dependency>  <groupId>org.apache.commons</groupId>  <artifactId>commons-lang3</artifactId>  <version>3.12.0</version>  </dependency>  </dependencies>  </project> |
| --- |

Sau đó, sử dụng tổ hợp phím **Ctrl + S**: lưu file, thư viện sẽ được tự động tải về và tích hợp vào project của các bạn.  
Kiểm tra kết quả:



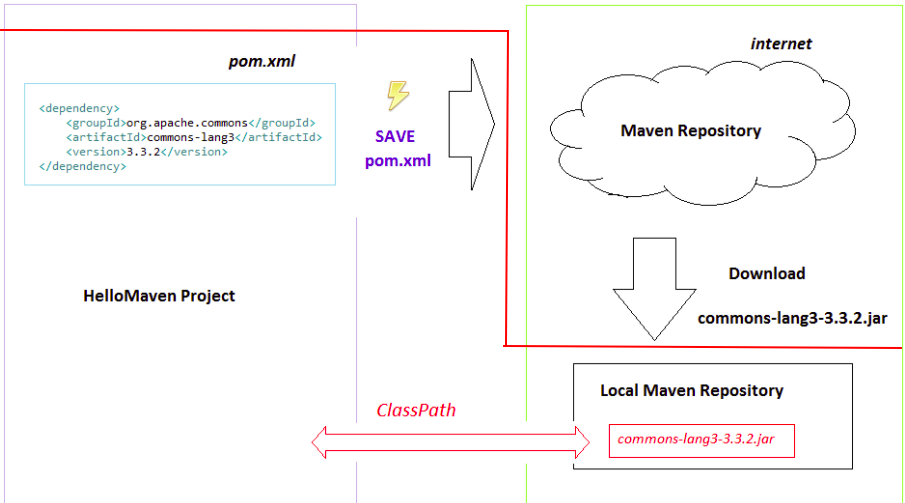
Trong trường hợp lỗi, các bạn có thể dùng một trong 2 cách sau để chủ động tải lại thư viện.  
1. Chuột phải pom.xml → Run as → Maven clean → Maven install



2. 1. Chuột phải pom.xml → Maven →Update project → Chọn Project cần update → OK

**III. Nguyên lý làm việc của Maven**

Câu hỏi đặt ra là, Maven tự động tải gói thư viện từ trên internet về thì sẽ để nó ở đâu trên máy tính chúng ta?  
Để làm rõ điều này, các bạn hãy xem hình ảnh dưới đây nói về cách làm việc của maven.

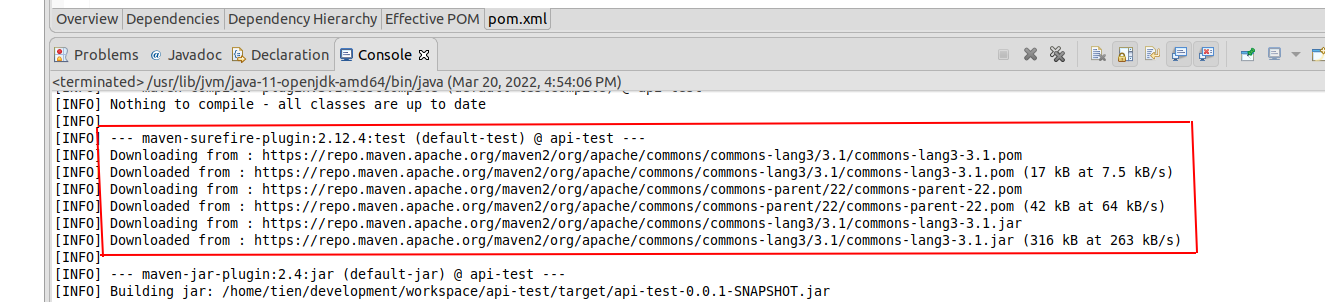
1. Người dùng khai báo thông tin thư viện & version trong pom.xml 

Sau đó Ctrl + S để save file.

1. Maven sẽ kiểm tra xem thư viện này đã có tại Local Repository chưa (đây là nơi lưu trữ các thư viện trên máy tính của bạn). Nếu chưa có nó sẽ thực hiện download về từ Maven Repository (Nơi lưu trữ các thư viện trên Internet) xuống và lưu lại vào Local Repository .
2. Cuối cùng là Maven sẽ tự động tích hợp thư viện vào Project của các bạn.

**Đâu là Maven Repository?**

Khi các bạn khai báo một thư viện cùng 1 version mới trong file pom.xml dự án của các bạn, ngay khi các bạn save file hoặc run as→ Maven install. Các bạn sẽ thấy được quá trình maven download file từ Maven Repository.



**Đâu là Local Maven Repository?**

Trên Eclipse → window → preferences → Maven → User settings

