Juan Sebastian Velandia

Juan Felipe Ochoa

Lab02

1. Maven contribuye a:

* Administrar dependencias de forma automatizada.
* Estandarizar la organización de los proyectos.
* Automatizar los procedimientos de compilación, evaluaciones y embalaje.
* Integrar constante con herramientas como Jenkins y otras similares.
* **validate** → Verifica si el proyecto es correcto y si tiene todas las dependencias necesarias.
* **compile** → Compila el código fuente del proyecto.
* **test** → Ejecuta pruebas unitarias.
* **package** → Genera el artefacto empaquetado (por ejemplo, un .jar o .war).
* **verify** → Verifica que el paquete cumple con los criterios de calidad.
* **install** → Instala el artefacto en el repositorio local.
* **deploy** → Sube el artefacto a un repositorio remoto para compartirlo.

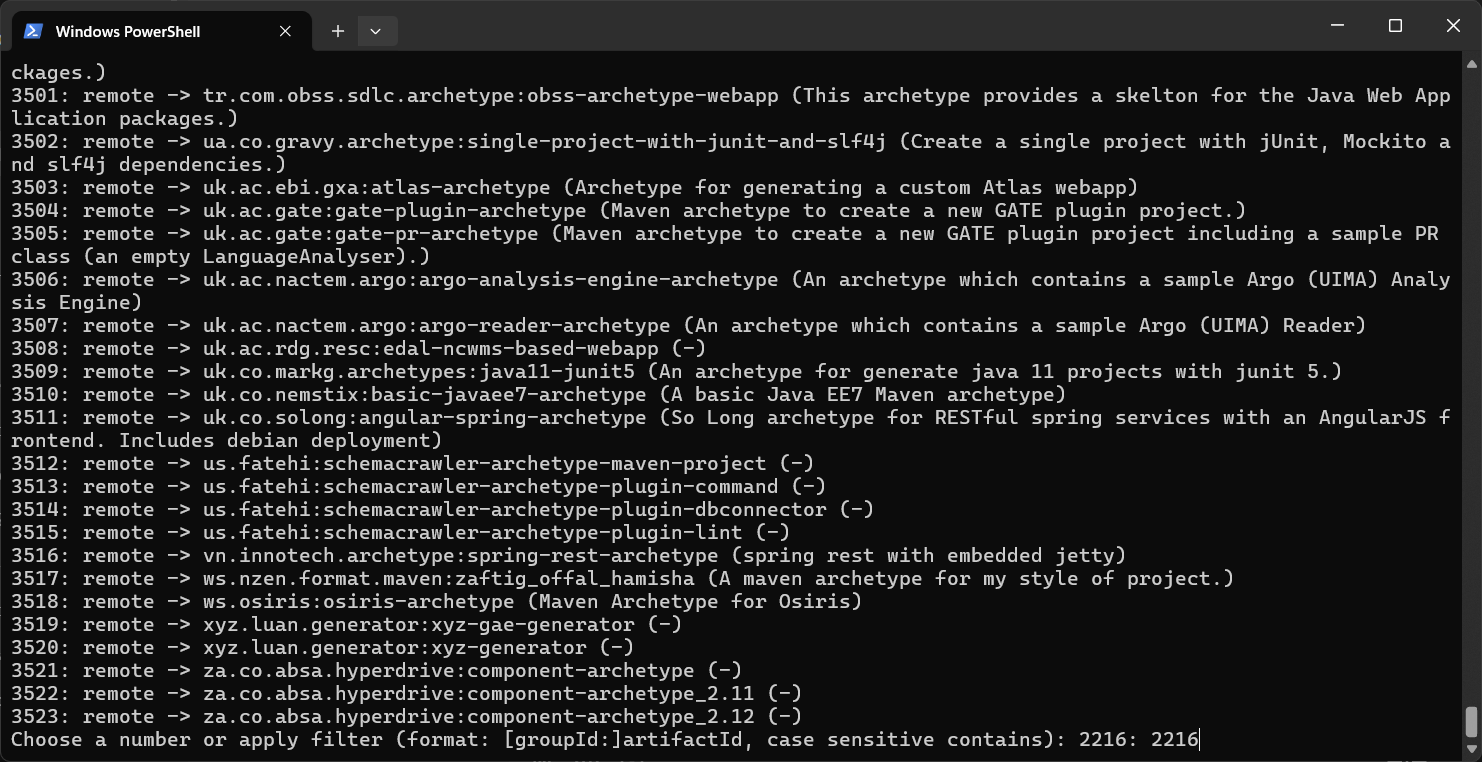
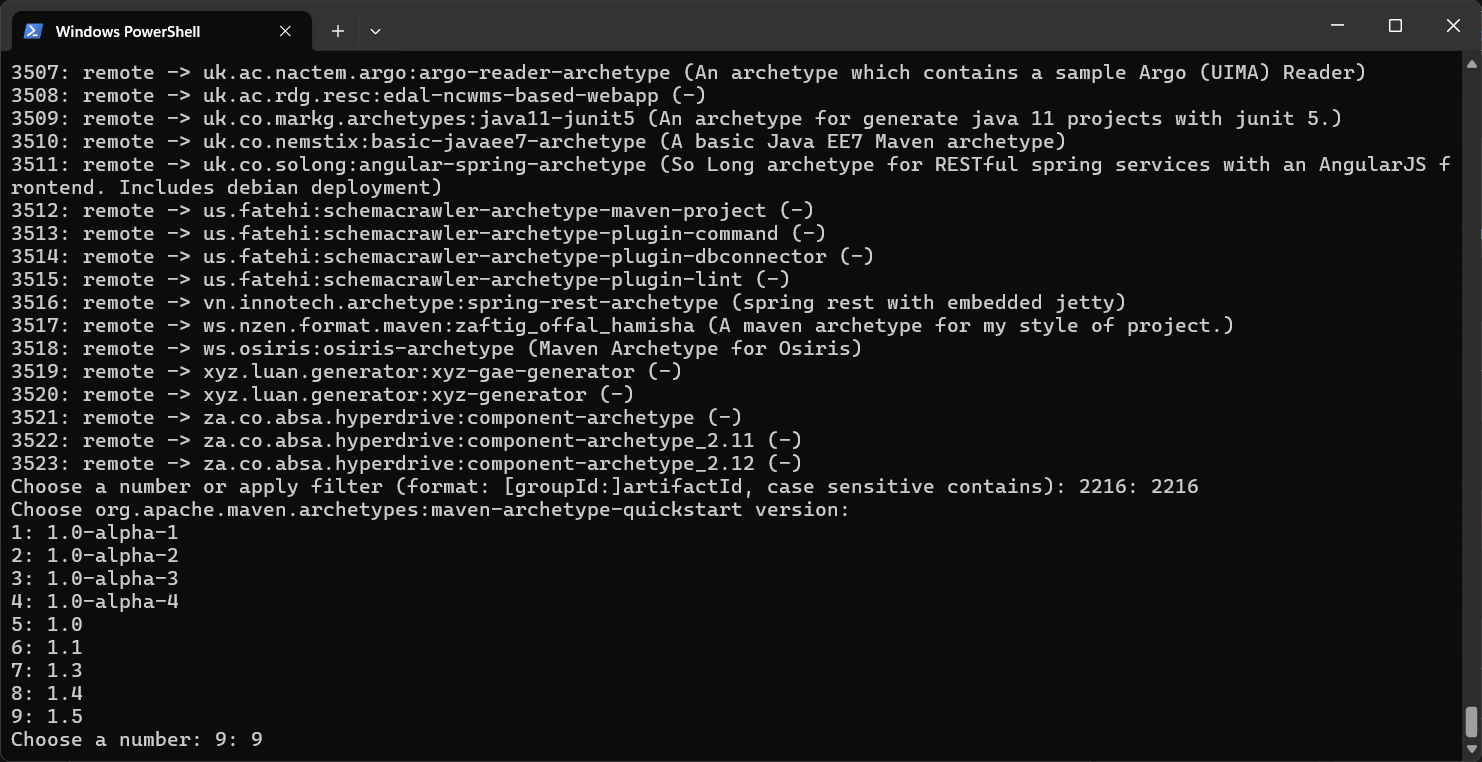
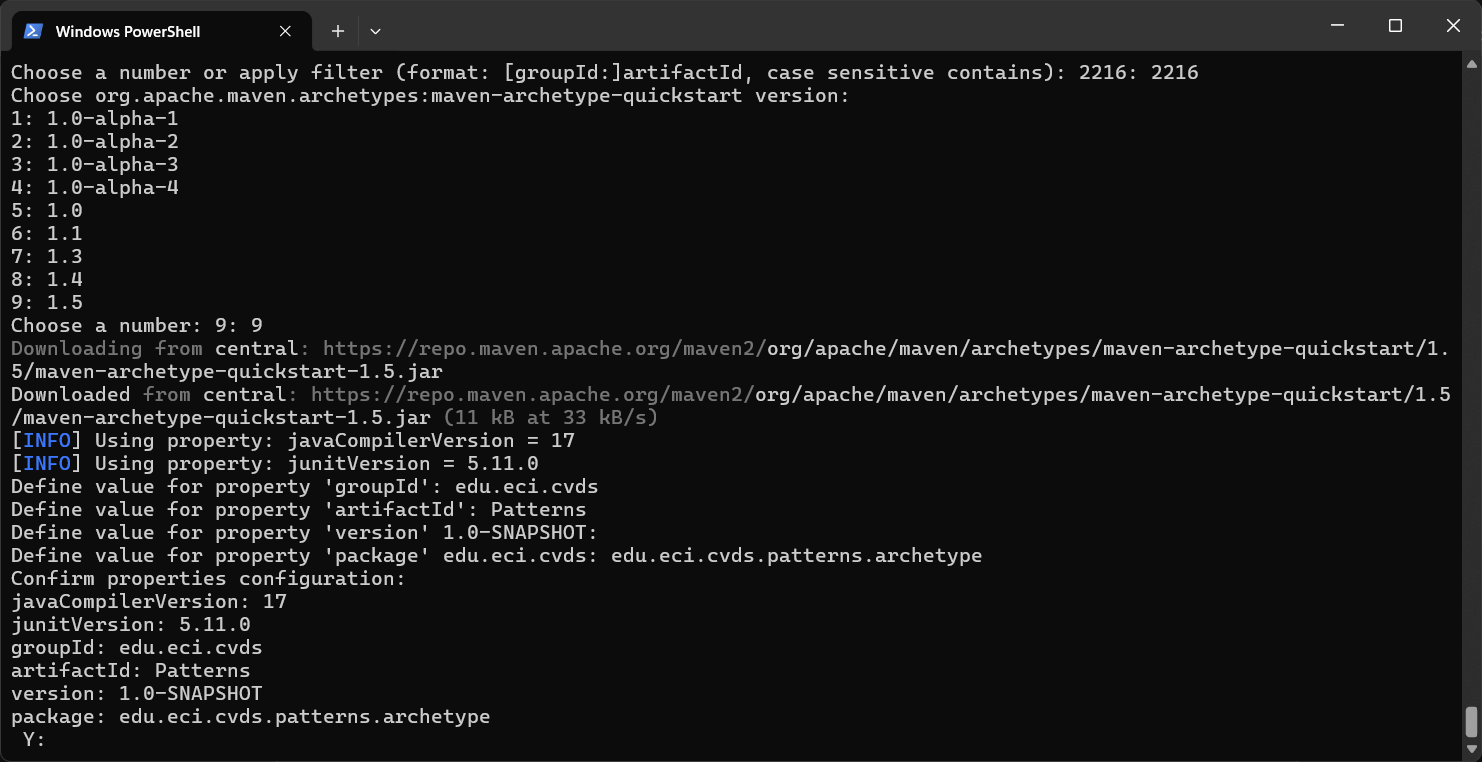
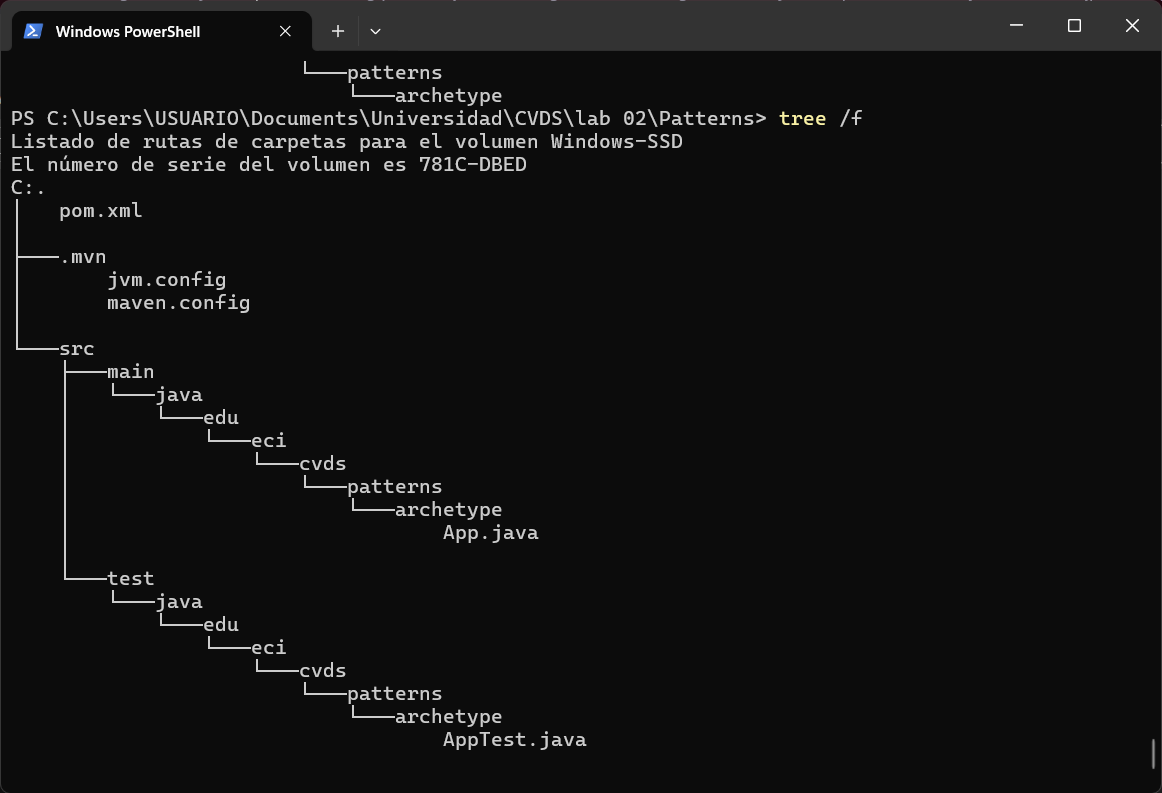
Cada fase se ejecuta en orden y depende de la anterior.

1. Son el conjunto de fases que se deben seguir para pasar un crear un proyecto, hay 3 tipos de ciclos;

* Default: Es el ciclo predeterminado que se encarga del control de la construcción, empaquetamiento y despliegue del proyecto
* Clean: Controla la limpieza del espacio de trabajo, es decir, limpia los archivos generados en compilaciones previas
* Site: Es el encargado de crear la página con la respectiva documentación del proyecto

1. Los plugins en Maven extienden sus funcionalidades y permiten ejecutar tareas específicas dentro del ciclo de vida.
2. Es el principal registro y repositorio de software para componentes, bibliotecas y marcos de Java, así como para lenguajes de Java Virtual Machine (JVM) y sirve para que los desarrolladores no tengan que buscar, descargar y configurar manualmente cada librería, si no solo añadirla al pom.xml

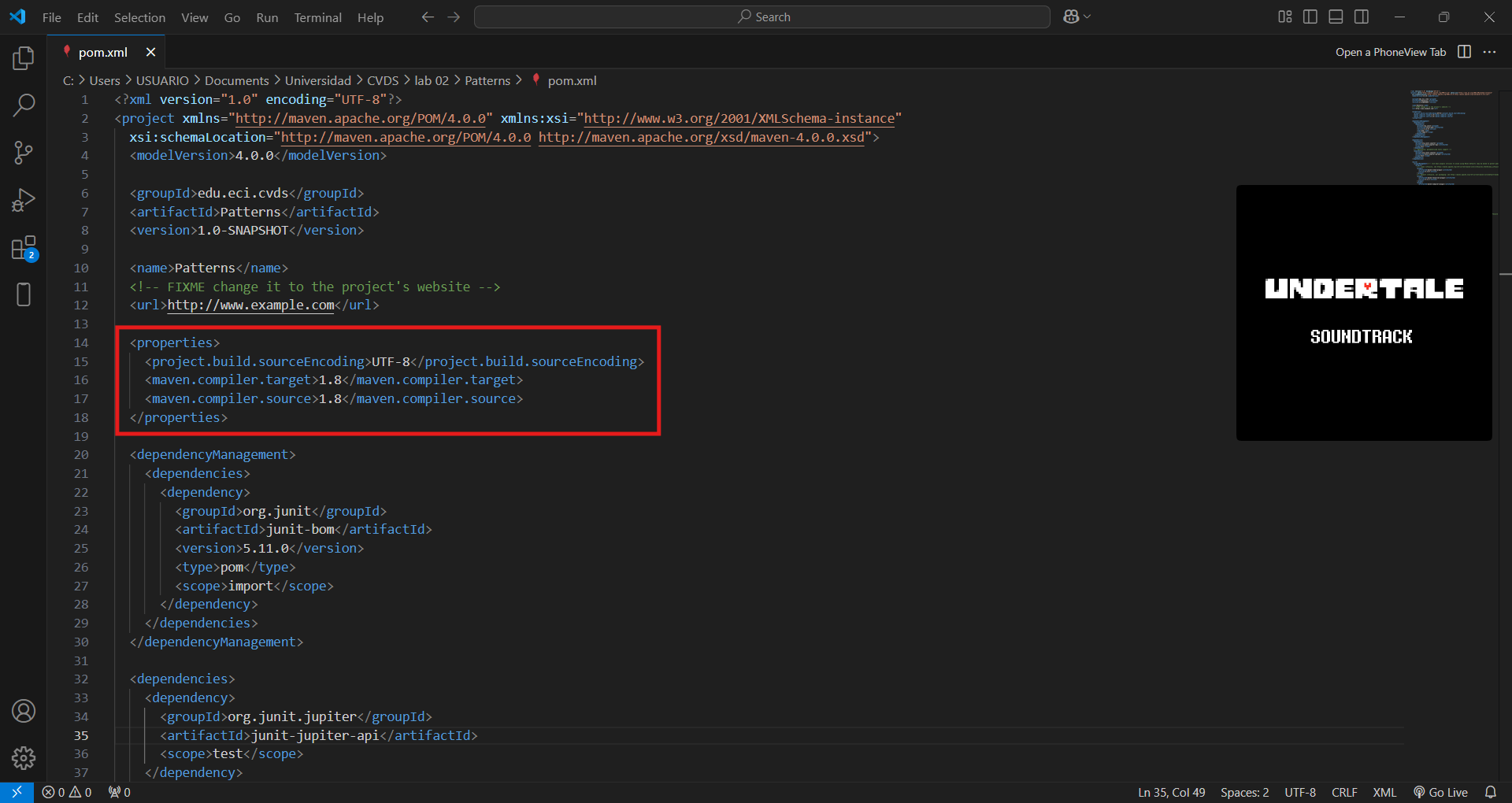
### CREAR UN PROYECTO CON MAVEN

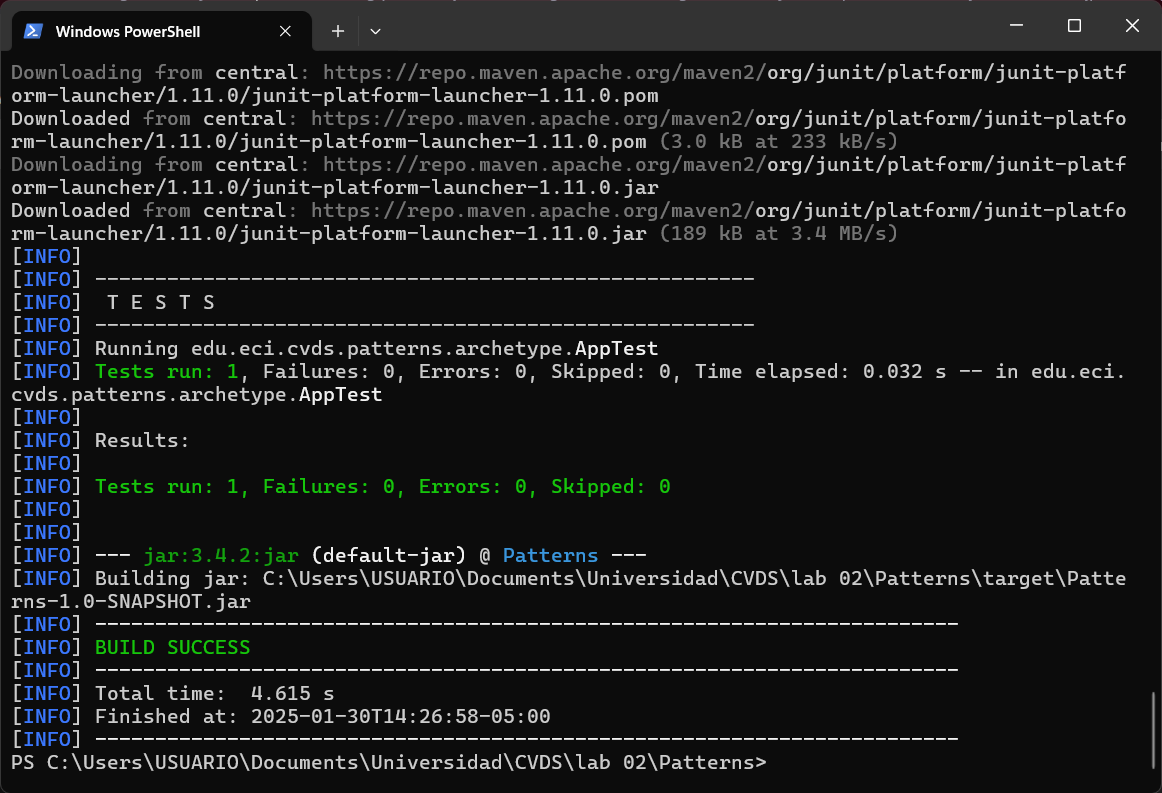
1. Se crea una carpeta donde va a reposar nuestro proyecto, para luego abrir la terminal y abrir la carpeta
2. Luego se ejecuta el comando mvn archetype:generate
3. Despues de ejecutado, nos pedira el arquetipo de proyecto maven, para este caso, usaremos 2216 que es el que generalmente se usa para proyectos java  
   
4. Luego seleccionaremos la version, la cual la sugerida es la ultima:
5. Luego daremos los parámetros solicitados en el laboratorio:
6. Y por ultimo comprobamos el tree que es exactamente igual al del laboratorio:

AJUSTAR ALGUNAS CONFIGURACIONES EN EL PROYECTO

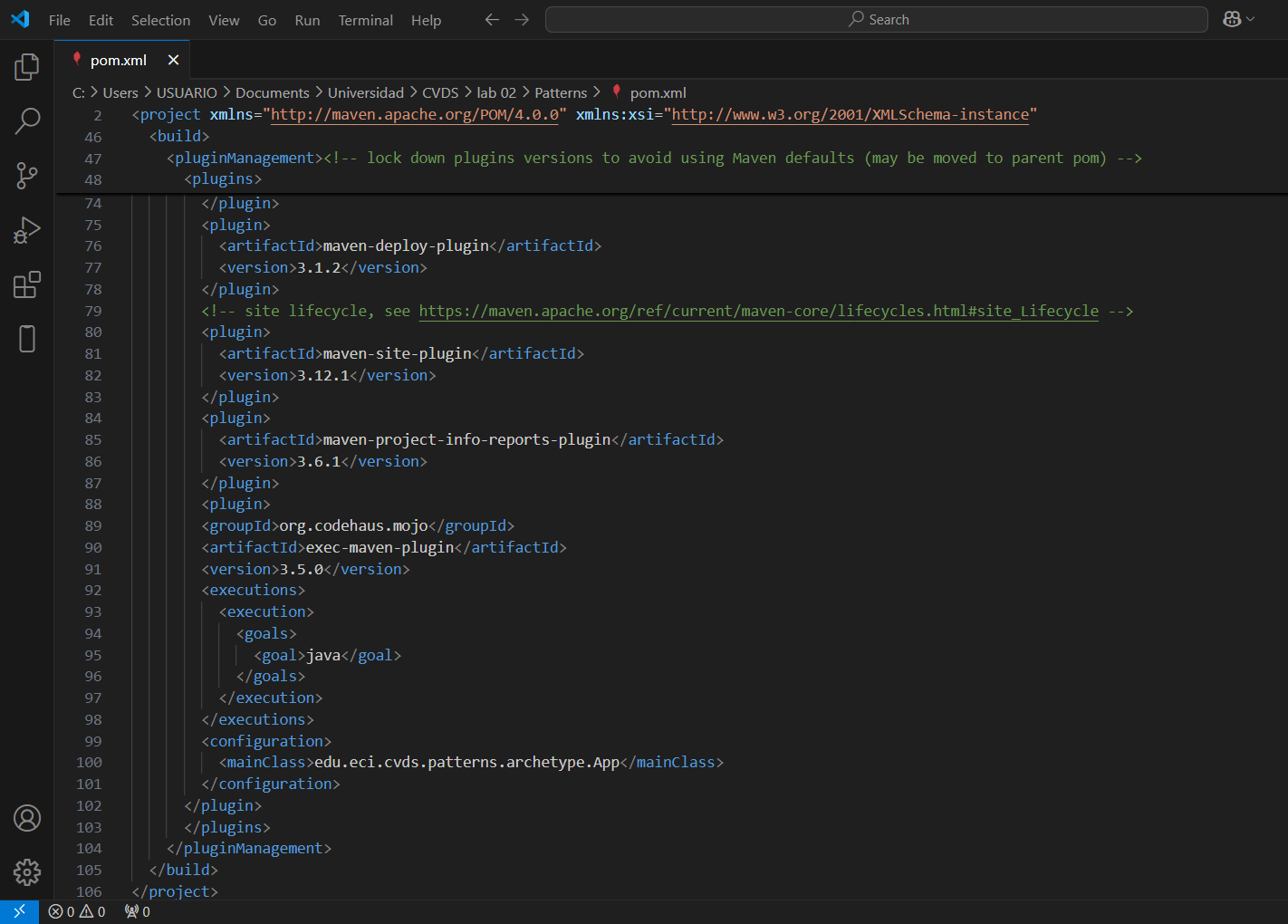
Para devolver la versión del JDK a java 8 tenemos que:

1. Editar el archivo pom.xml de la siguiente manera:



1. Luego ejecutamos el comando mvn package para compilar:

Preguntas:

* El comando package sirve para compilar el código, ejecutar las pruebas unitarias, y empaquetar. Otros parámetros que se pueden enviar a este comando son, por ejemplo, clean, validate, compile, test, install, deploy, site, entre otros.
* Para esto tenemos que modificar el pom.xml añadiendo:
* Luego ejecutamos el comando mvn exec:java