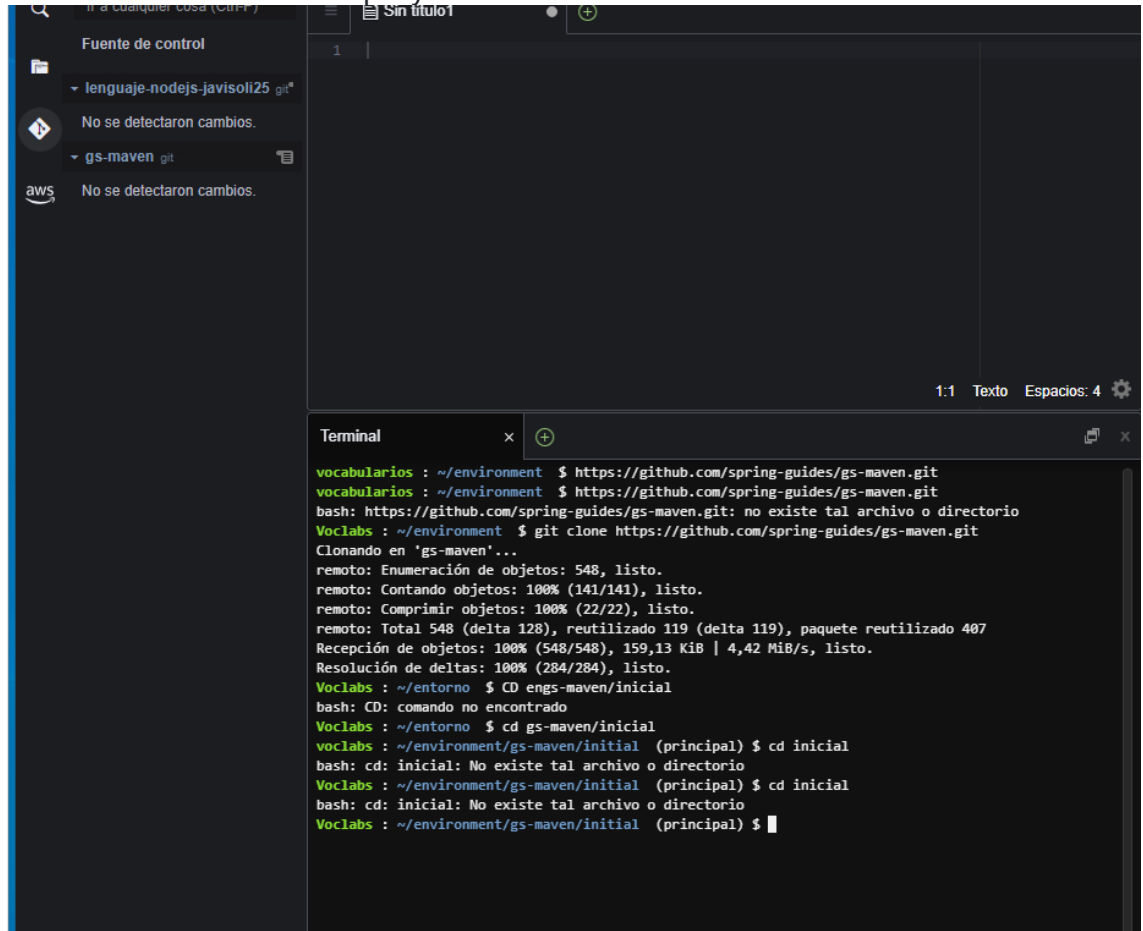


# TAREA JAVA

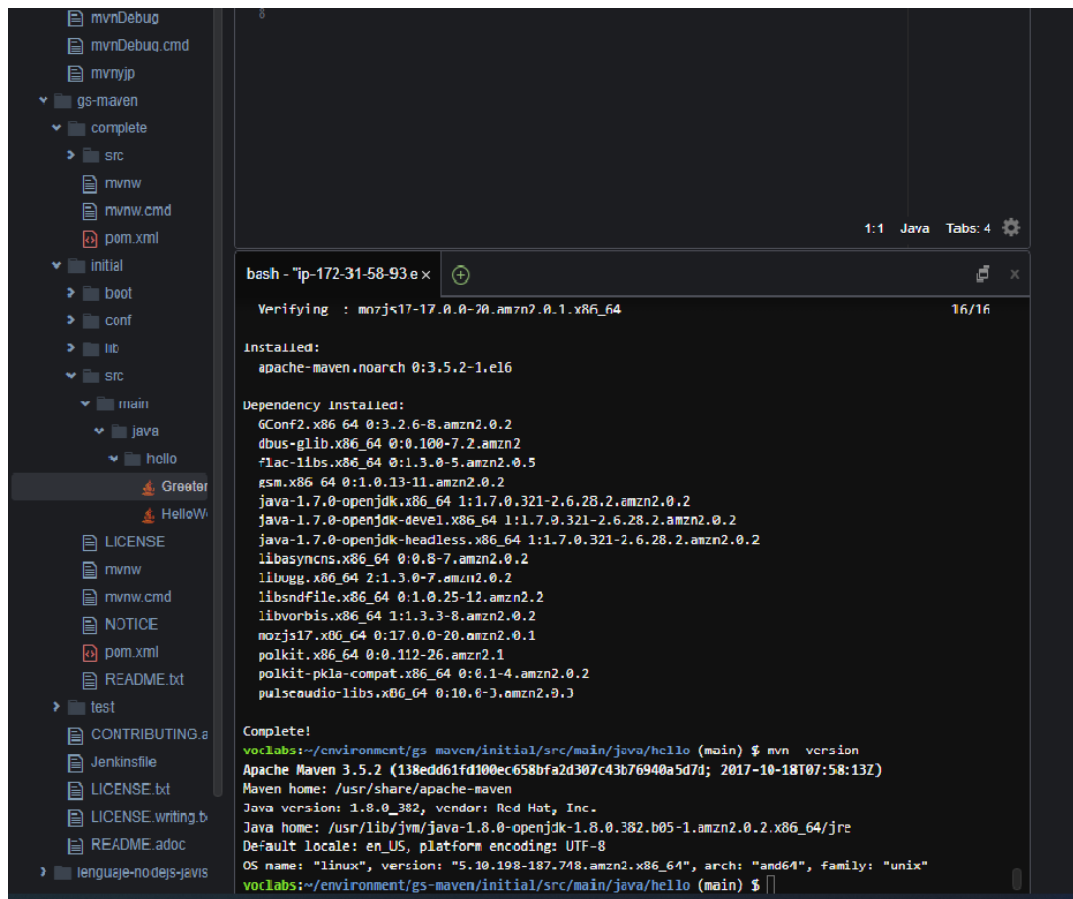
- Clona este proyecto en el entorno de Cloud9.



The screenshot shows the Cloud9 IDE interface. On the left, the 'Fuente de control' (Source Control) panel is visible, showing a list of files: 'lenguaje-nodejs-javisoli25' and 'gs-maven', both with the status 'No se detectaron cambios.' (No changes detected). The main editor area is empty. At the bottom, a terminal window is open, displaying the following commands and output:

```
Voclabios : ~/environment $ https://github.com/spring-guides/gs-maven.git
Voclabios : ~/environment $ https://github.com/spring-guides/gs-maven.git
bash: https://github.com/spring-guides/gs-maven.git: no existe tal archivo o directorio
Voclabios : ~/environment $ git clone https://github.com/spring-guides/gs-maven.git
Clonando en 'gs-maven'...
remoto: Enumeración de objetos: 548, listo.
remoto: Contando objetos: 100% (141/141), listo.
remoto: Comprimir objetos: 100% (22/22), listo.
remoto: Total 548 (delta 128), reutilizado 119 (delta 119), paquete reutilizado 407
Recepción de objetos: 100% (548/548), 159,13 KiB | 4,42 MiB/s, listo.
Resolución de deltas: 100% (284/284), listo.
Voclabios : ~/entorno $ cd engs-maven/inicial
bash: cd: comando no encontrado
Voclabios : ~/entorno $ cd gs-maven/inicial
Voclabios : ~/environment/gs-maven/inicial (principal) $ cd inicial
bash: cd: inicial: No existe tal archivo o directorio
Voclabios : ~/environment/gs-maven/inicial (principal) $ cd inicial
bash: cd: inicial: No existe tal archivo o directorio
Voclabios : ~/environment/gs-maven/inicial (principal) $
```

- Clona también el proyecto indicado en el enlace anterior.
- 
- Inspecciona el archivo `.gitignore` del proyecto de ejemplo. Cópialo en este repositorio.
- Siga los pasos indicados en el tutorial. Debes crear el código en **este repositorio**.
  - Haz las siguientes capturas de pantalla:
    - Resultado de ejecutar el comando `mvn -v` una vez instalado Maven.



- Resultado de ejecutar el programa en la consola.

```
java - ip-172-31-58-93.ec2
[INFO]
[INFO] --- maven-surefire-plugin:2.12.4:test (default-test) @ my-app ---
[INFO] Surefire report directory: /home/ec2-user/environment/gs-maven/initial/src/main/java/hello/
my-app/target/surefire-reports

-----
T E S T S
-----
Running com.mycompany.app.AppTest
Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.027 sec

Results :

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

[INFO]
[INFO] --- maven-jar-plugin:2.4:jar (default-jar) @ my-app ---
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time: 3.419 s
[INFO] Finished at: 2023-12-10T16:54:08Z
[INFO] Final Memory: 8M/21M
[INFO]
voclabs:~/environment/gs-maven/initial/src/main/java/hello/my-app (main) $ java -cp target/my-app-
1.0-SNAPSHOT-jar-with-dependencies.jar com.mycompany.app.App my-test-bucket us-east-2
Error: Could not find or load main class com.mycompany.app.App
voclabs:~/environment/gs-maven/initial/src/main/java/hello/my-app (main) $ gradle build
<-----> 0% INITIALIZING [15s]
> Evaluating settings > Generating gradle-api-8.5.jar > Generating 55%
```

```
bash - "ip-172-31-58-93.ec2"
Installing: gradle 8.5
Done installing!

Setting gradle 8.5 as default.
voclabs:~/environment/gs-maven/initial/src/main/java/hello (main) $ gradle -version

Welcome to Gradle 8.5!

Here are the highlights of this release:
- Support for running on Java 21
- Faster first use with Kotlin DSL
- Improved error and warning messages

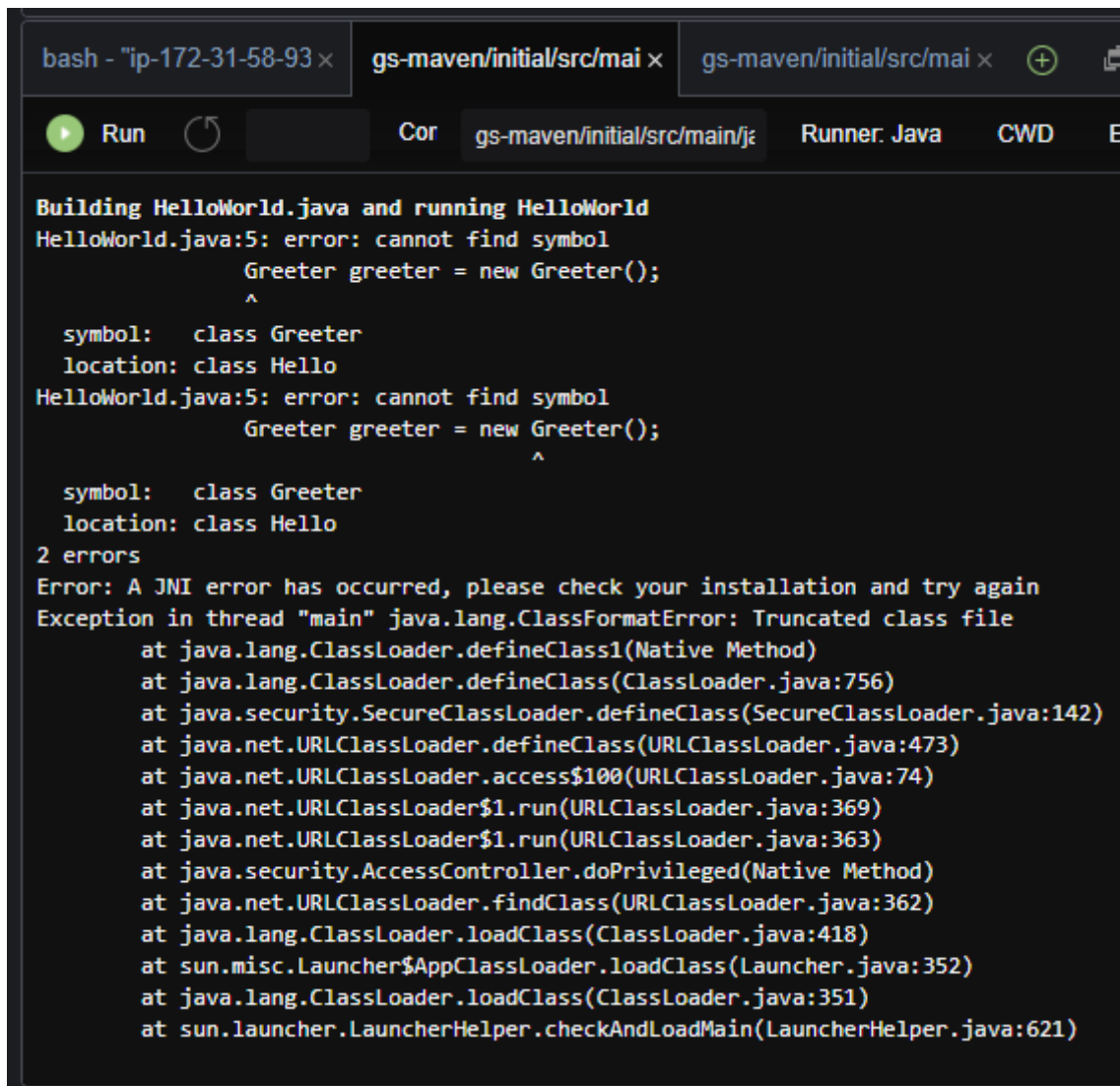
For more details see https://docs.gradle.org/8.5/release-notes.html

-----
Gradle 8.5
-----

Build time: 2023-11-29 14:08:57 UTC
Revision: 28aca86a7180baa17117e0e5ba01d8ea9feca598

Kotlin: 1.9.20
Groovy: 3.0.17
Ant: Apache Ant(TM) version 1.10.13 compiled on January 4 2023
JVM: 1.8.0_382 (Red Hat, Inc. 25.382-b05)
OS: Linux 5.10.198-187.748.amzn2.x86_64 amd64

voclabs:~/environment/gs-maven/initial/src/main/java/hello (main) $
```



The screenshot shows an IDE terminal window with the following content:

```
bash - "ip-172-31-58-93" x gs-maven/initial/src/mai x gs-maven/initial/src/mai x +
Run Cor gs-maven/initial/src/main/j Runner: Java CWD E

Building HelloWorld.java and running HelloWorld
HelloWorld.java:5: error: cannot find symbol
    Greeter greeter = new Greeter();
    ^
  symbol:   class Greeter
  location: class Hello
HelloWorld.java:5: error: cannot find symbol
    Greeter greeter = new Greeter();
    ^
  symbol:   class Greeter
  location: class Hello
2 errors
Error: A JNI error has occurred, please check your installation and try again
Exception in thread "main" java.lang.ClassFormatError: Truncated class file
    at java.lang.ClassLoader.defineClass1(Native Method)
    at java.lang.ClassLoader.defineClass(ClassLoader.java:756)
    at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
    at java.net.URLClassLoader.defineClass(URLClassLoader.java:473)
    at java.net.URLClassLoader.access$100(URLClassLoader.java:74)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:369)
    at java.net.URLClassLoader$1.run(URLClassLoader.java:363)
    at java.security.AccessController.doPrivileged(Native Method)
    at java.net.URLClassLoader.findClass(URLClassLoader.java:362)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:418)
    at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:352)
    at java.lang.ClassLoader.loadClass(ClassLoader.java:351)
    at sun.launcher.LauncherHelper.checkAndLoadMain(LauncherHelper.java:621)
```

## instrucciones de entrega

Sube el proyecto Java al repositorio. Incluye además las **capturas** pedidas. Subelo al repositorio. Crea un **tema** y nómbrame mediante @pedroprieto.