Lab Experiment 1: Setting Up a Jenkins Job for Maven Build

Objective: Create a Jenkins job that builds a Maven project using Jenkins and triggers the build on changes in the version control repository.

Prerequisites:

- Jenkins server up and running.
- Maven installed on the Jenkins server.
- A Maven project hosted in a version control repository (e.g., Git).

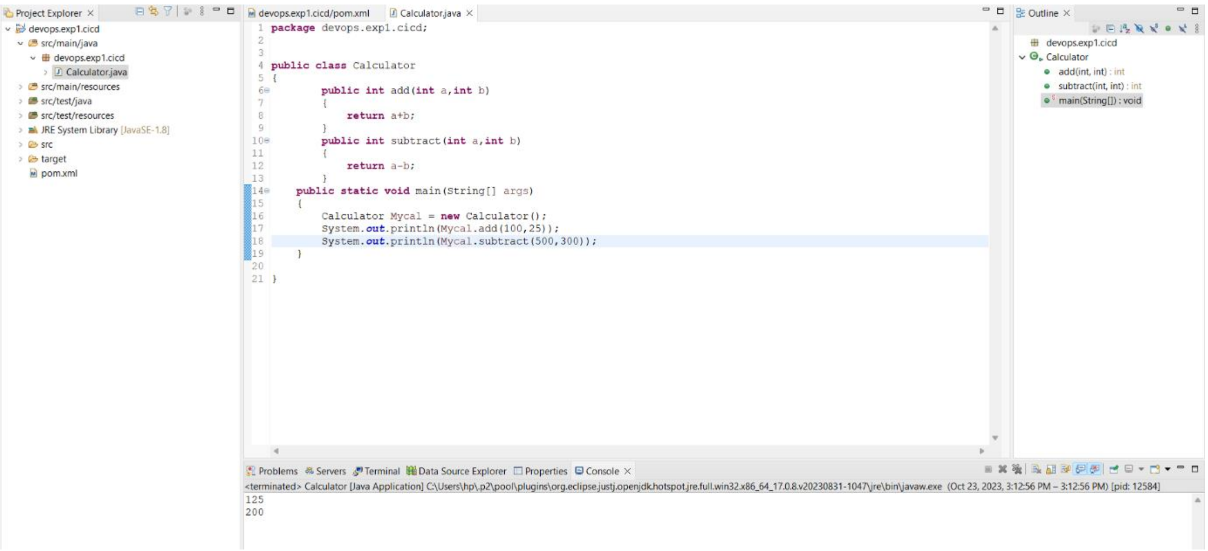
Steps:

Jenkins Configuration:

- Ensure that Jenkins is installed and accessible.
- Install necessary plugins: Maven Integration Plugin.

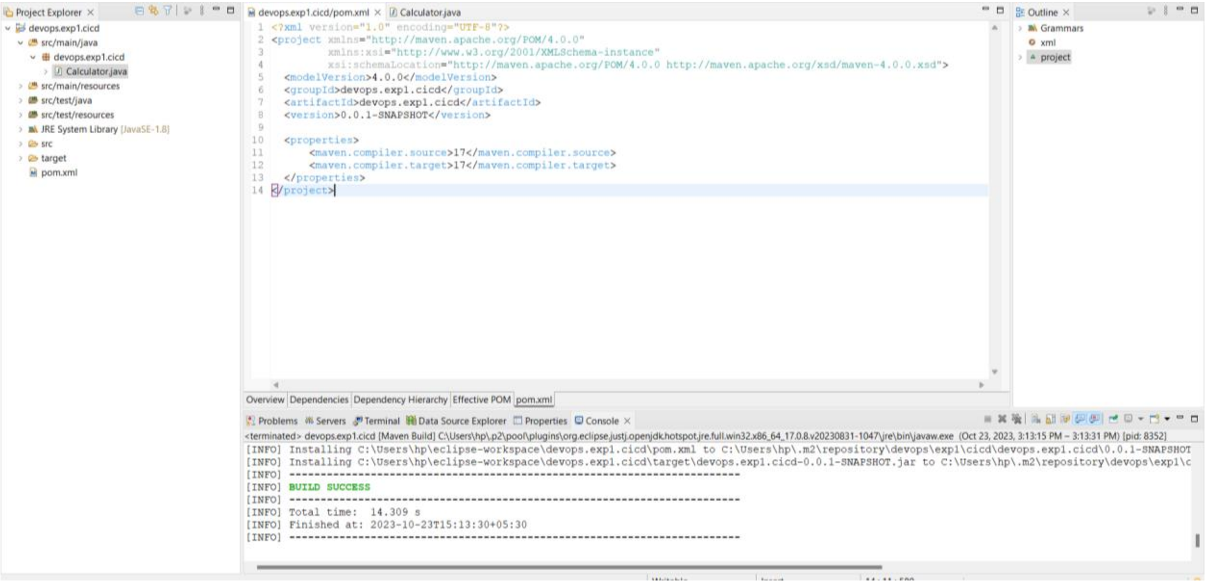
Creating a Jenkins Job:

- Log in to your Jenkins instance.
- Click on "New Item" to create a new Jenkins job.
- Enter a name for the job (e.g., "Maven_Build") and select "Freestyle project."



```java
package devops.exp1.cicd;


public class Calculator
{
        public int add(int a,int b)
        {
            return a+b;
        }
        public int subtract(int a,int b)
        {
            return a-b;
        }
    public static void main(String[] args)
    {
        Calculator Mycal = new Calculator();
        System.out.println(Mycal.add(100,25));
        System.out.println(Mycal.subtract(500,300));
    }

}
```

```
<terminated> Calculator [Java Application] C:\Users\hp\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe  (Oct 23, 2023, 3:12:56 PM – 3:12:56 PM) [pid: 12584]
125
200
```

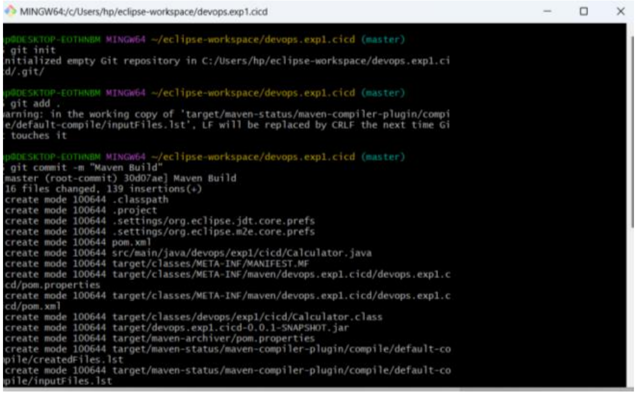


```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">
  <modelVersion>4.0.0</modelVersion>
  <groupId>devops.exp1.cicd</groupId>
  <artifactId>devops.exp1.cicd</artifactId>
  <version>0.0.1-SNAPSHOT</version>

  <properties>
      <maven.compiler.source>17</maven.compiler.source>
      <maven.compiler.target>17</maven.compiler.target>
  </properties>
</project>
```

```
<terminated> devops.exp1.cicd [Maven Build] C:\Users\hp\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.8.v20230831-1047\jre\bin\javaw.exe (Oct 23, 2023, 3:13:15 PM – 3:13:31 PM) [pid: 8352]
[INFO] Installing C:\Users\hp\eclipse-workspace\devops.exp1.cicd\pom.xml to C:\Users\hp\.m2\repository\devops\exp1\cicd\devops.exp1.cicd\0.0.1-SNAPSHOT
[INFO] Installing C:\Users\hp\eclipse-workspace\devops.exp1.cicd\target\devops.exp1.cicd-0.0.1-SNAPSHOT.jar to C:\Users\hp\.m2\repository\devops\exp1\c
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  14.309 s
[INFO] Finished at: 2023-10-23T15:13:30+05:30
[INFO] ------------------------------------------------------------------------
```

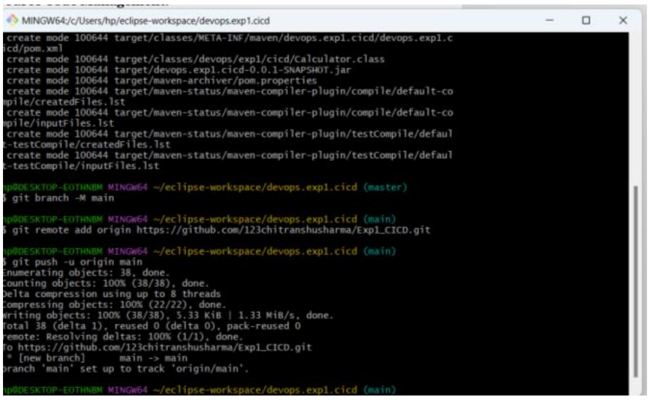
Configuring Source Code Management:

- Under the "Source Code Management" section, choose your version control system (e.g., Git).
- Provide the repository URL and credentials if needed.



MINGW64:/c/Users/hp/eclipse-workspace/devops.exp1.cicd

```
p@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (master)
 git init
nitialized empty Git repository in C:/Users/hp/eclipse-workspace/devops.exp1.ci
d/.git/

p@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (master)
 git add .
arning: in the working copy of 'target/maven-status/maven-compiler-plugin/compi
e/default-compile/inputFiles.lst', LF will be replaced by CRLF the next time Gi
 touches it

p@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (master)
 git commit -m "Maven Build"
master (root-commit) 30d07ae] Maven Build
16 files changed, 139 insertions(+)
create mode 100644 .classpath
create mode 100644 .project
create mode 100644 .settings/org.eclipse.jdt.core.prefs
create mode 100644 .settings/org.eclipse.m2e.core.prefs
create mode 100644 pom.xml
create mode 100644 src/main/java/devops/exp1/cicd/Calculator.java
create mode 100644 target/classes/META-INF/MANIFEST.MF
create mode 100644 target/classes/META-INF/maven/devops.exp1.cicd/devops.exp1.c
cd/pom.properties
create mode 100644 target/classes/META-INF/maven/devops.exp1.cicd/devops.exp1.c
cd/pom.xml
create mode 100644 target/classes/devops/exp1/cicd/Calculator.class
create mode 100644 target/devops.exp1.cicd-0.0.1-SNAPSHOT.jar
create mode 100644 target/maven-archiver/pom.properties
create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-co
pile/createdFiles.lst
create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-co
pile/inputFiles.lst
```

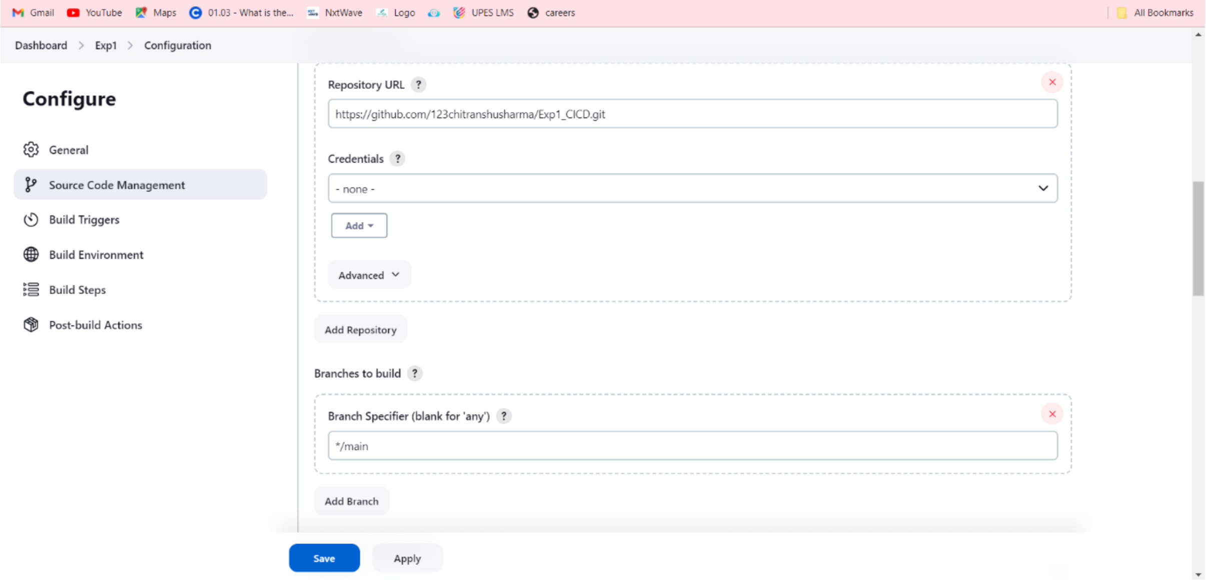


MINGW64:/c/Users/hp/eclipse-workspace/devops.exp1.cicd

```
 create mode 100644 target/classes/META-INF/maven/devops.exp1.cicd/devops.exp1.c
icd/pom.xml
 create mode 100644 target/classes/devops/exp1/cicd/Calculator.class
 create mode 100644 target/devops.exp1.cicd-0.0.1-SNAPSHOT.jar
 create mode 100644 target/maven-archiver/pom.properties
 create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-co
mpile/createdFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/compile/default-co
mpile/inputFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/testCompile/defaul
t-testCompile/createdFiles.lst
 create mode 100644 target/maven-status/maven-compiler-plugin/testCompile/defaul
t-testCompile/inputFiles.lst

np@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (master)
$ git branch -M main

np@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (main)
$ git remote add origin https://github.com/123chitranshusharma/Exp1_CICD.git

np@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (main)
$ git push -u origin main
Enumerating objects: 38, done.
Counting objects: 100% (38/38), done.
Delta compression using up to 8 threads
Compressing objects: 100% (22/22), done.
Writing objects: 100% (38/38), 5.33 KiB | 1.33 MiB/s, done.
Total 38 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), done.
To https://github.com/123chitranshusharma/Exp1_CICD.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.

np@DESKTOP-EOTHNBM MINGW64 ~/eclipse-workspace/devops.exp1.cicd (main)
```

Configuring the Build:

- In the "Build" section, click on "Add build step" and select "Invoke top-level Maven targets."
- In the "Goals" field, enter the Maven goals you want to execute (e.g., "clean install").
- Setting Up Polling for Changes:
- Scroll down to the "Build Triggers" section.
- Choose the option "Poll SCM" and specify the polling schedule (e.g., "* * * * *" for polling every minute).

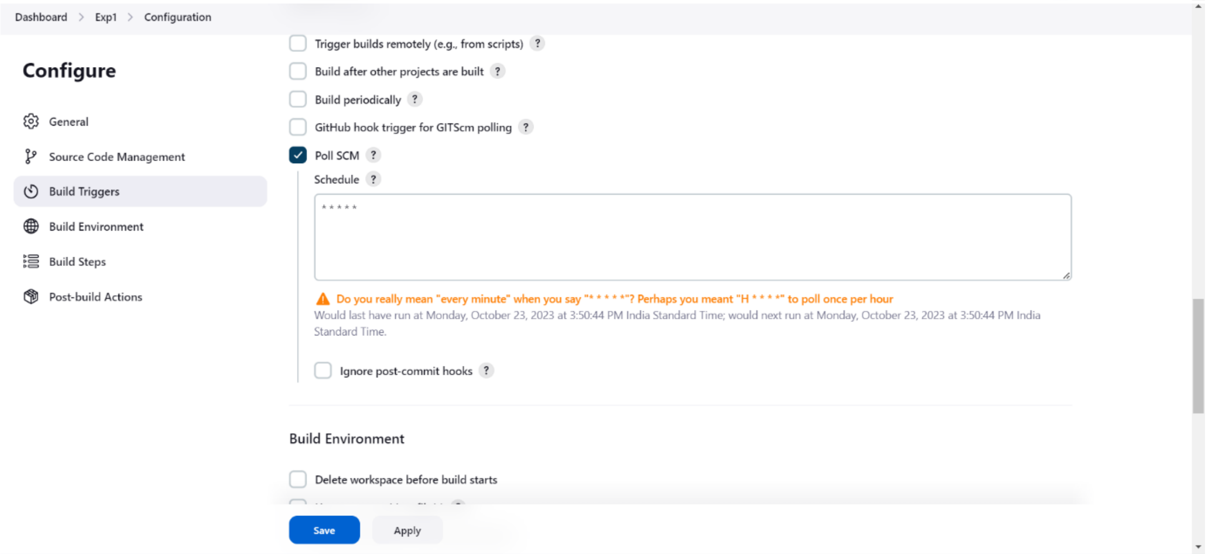


Repository URL ?
https://github.com/123chitranshusharma/Exp1_CICD.git

Credentials ?
- none -

Branches to build ?

Branch Specifier (blank for 'any') ?
*/main



□ Trigger builds remotely (e.g., from scripts) ?

□ Build after other projects are built ?

□ Build periodically ?

□ GitHub hook trigger for GITScm polling ?

☑ Poll SCM ?

Schedule ?

* * * * *

⚠ Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour

Would last have run at Monday, October 23, 2023 at 3:50:44 PM India Standard Time; would next run at Monday, October 23, 2023 at 3:50:44 PM India Standard Time.

□ Ignore post-commit hooks ?

Build Environment

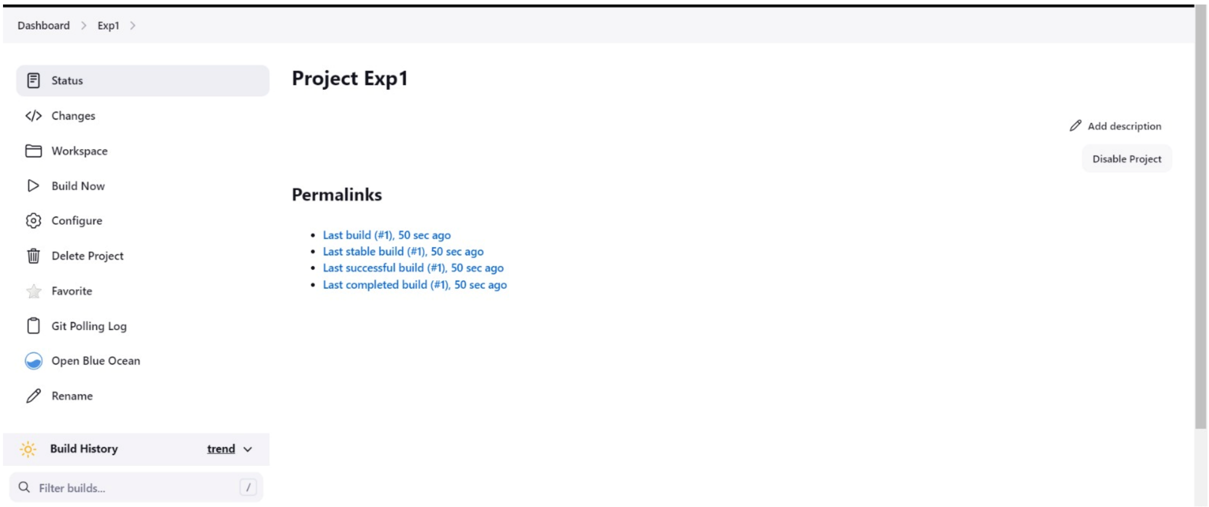
□ Delete workspace before build starts

Save and Run the Job:

- Click on "Save" to save the job configuration.
- Click on "Build Now" to manually trigger the job initially.

Observing the Results:

- Monitor the job's console output to see the Maven build process.
- Check the build status (success/failure) on the Jenkins dashboard.



Project Exp1

Permalinks

- Last build (#1), 50 sec ago
- Last stable build (#1), 50 sec ago
- Last successful build (#1), 50 sec ago
- Last completed build (#1), 50 sec ago

Automated Builds on Repository Changes:

- Make changes to your Maven project and push them to the version control repository.
- Observe that Jenkins automatically triggers the job based on the polling schedule.

This lab experiment will guide you through the process of setting up a basic Jenkins job for building a Maven project. You can further enhance the experiment by exploring advanced configurations such as parameterized builds, post-build actions, integration with artifact repositories, and deploying the built artifacts to various environments.