Experiment 3

Maven Build using GitHub Actions

Objective: Set up a GitHub Actions workflow to automatically build a Maven project whenever changes are pushed to a GitHub repository.

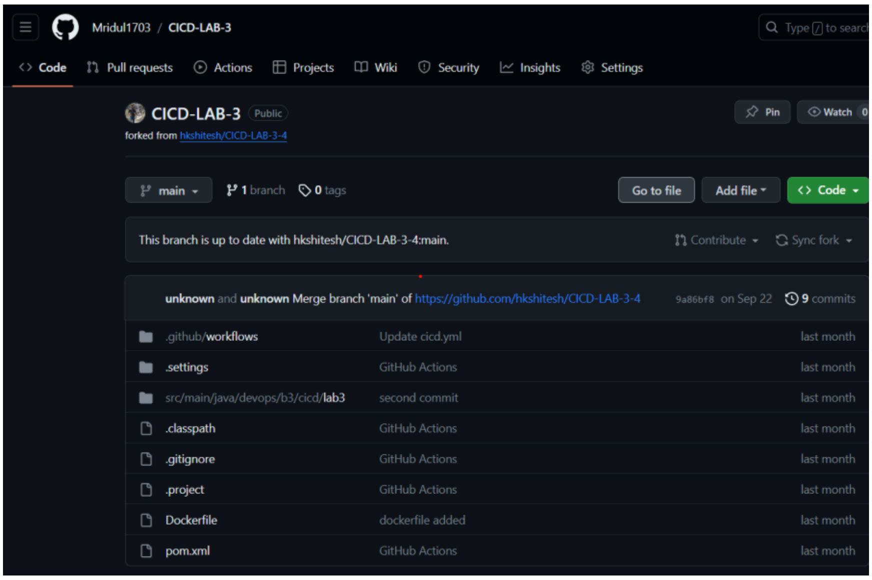
Prerequisites:

- GitHub account
- A Maven-based Java project hosted on GitHub

Exercise Steps:

Step 1: Fork and Clone the Repository

- Fork a sample Maven-based Java project repository on GitHub.
- Clone the forked repository to your local machine.



CICD-LAB-3 Public

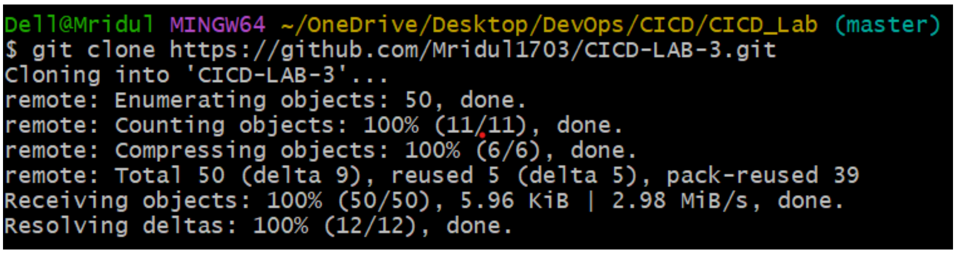
forked from hkshitesh/CICD-LAB-3-4

1 branch 0 tags

This branch is up to date with hkshitesh/CICD-LAB-3-4:main.

unknown and unknown Merge branch 'main' of https://github.com/hkshitesh/CICD-LAB-3-4 9a86bf8 on Sep 22 9 commits

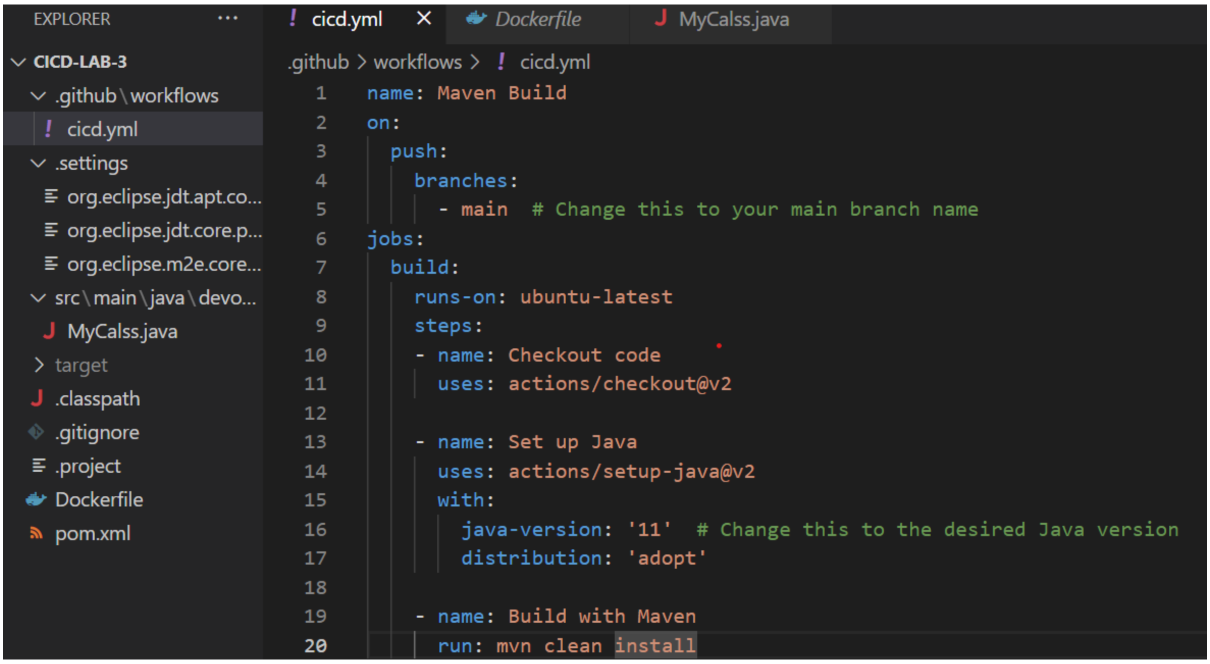
.github/workflows	Update cicd.yml	last month
.settings	GitHub Actions	last month
src/main/java/devops/b3/cicd/lab3	second commit	last month
.classpath	GitHub Actions	last month
.gitignore	GitHub Actions	last month
.project	GitHub Actions	last month
Dockerfile	dockerfile added	last month
pom.xml	GitHub Actions	last month



```
Dell@Mridul MINGW64 ~/OneDrive/Desktop/DevOps/CICD/CICD_Lab (master)
$ git clone https://github.com/Mridul1703/CICD-LAB-3.git
Cloning into 'CICD-LAB-3'...
remote: Enumerating objects: 50, done.
remote: Counting objects: 100% (11/11), done.
remote: Compressing objects: 100% (6/6), done.
remote: Total 50 (delta 9), reused 5 (delta 5), pack-reused 39
Receiving objects: 100% (50/50), 5.96 KiB | 2.98 MiB/s, done.
Resolving deltas: 100% (12/12), done.
```

Step 2: Create a GitHub Actions Workflow

- In your cloned repository, create a directory named .github/workflows if it doesn't exist.
- Inside the .github/workflows directory, create a YAML file (e.g., maven-build.yml) to define your GitHub Actions workflow. You can use any text editor to create the file.
- Edit maven-build.yml and add the following content:

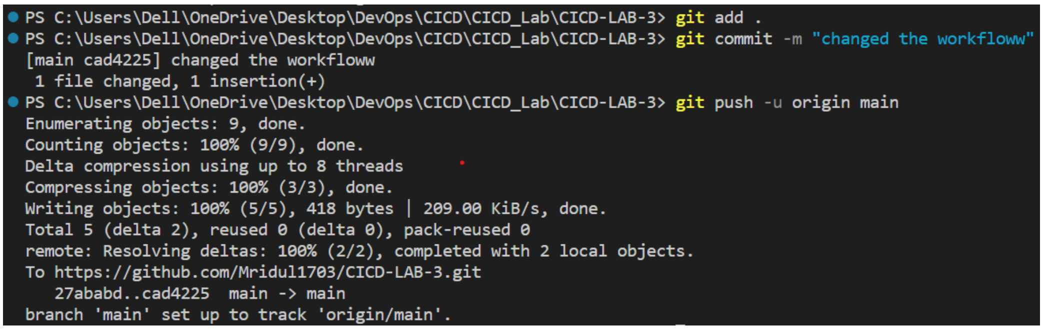


```yaml
name: Maven Build
on:
  push:
    branches:
      - main  # Change this to your main branch name
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
    - name: Checkout code
      uses: actions/checkout@v2

    - name: Set up Java
      uses: actions/setup-java@v2
      with:
        java-version: '11'  # Change this to the desired Java version
        distribution: 'adopt'

    - name: Build with Maven
      run: mvn clean install
```

This workflow will trigger a Maven build whenever changes are pushed to the main branch.

Step 3: Commit and Push Changes

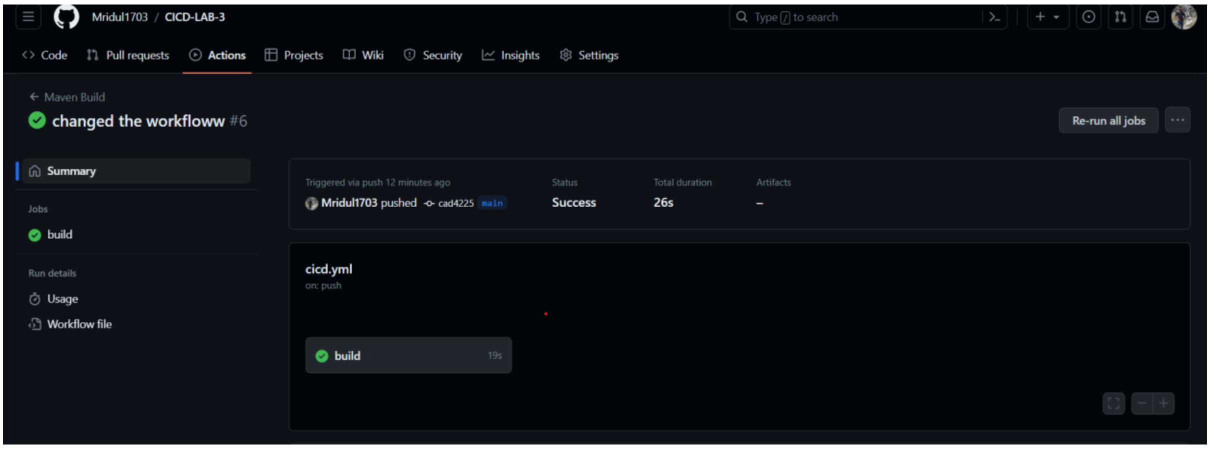
- Save the maven-build.yml file.
- Commit the changes to your local repository:



```
PS C:\Users\Dell\OneDrive\Desktop\DevOps\CICD\CICD_Lab\CICD-LAB-3> git add .
PS C:\Users\Dell\OneDrive\Desktop\DevOps\CICD\CICD_Lab\CICD-LAB-3> git commit -m "changed the workfloww"
[main cad4225] changed the workfloww
 1 file changed, 1 insertion(+)
PS C:\Users\Dell\OneDrive\Desktop\DevOps\CICD\CICD_Lab\CICD-LAB-3> git push -u origin main
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 8 threads
Compressing objects: 100% (3/3), done.
Writing objects: 100% (5/5), 418 bytes | 209.00 KiB/s, done.
Total 5 (delta 2), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (2/2), completed with 2 local objects.
To https://github.com/Mridul1703/CICD-LAB-3.git
   27ababd..cad4225  main -> main
branch 'main' set up to track 'origin/main'.
```

Step 4: Check the Workflow Status

- Go to your GitHub repository on the GitHub website.
- Click on the "Actions" tab to see the workflow running. You should see a workflow named "Maven Build" or the name you specified in the YAML file.
- Monitor the workflow's progress, and once it completes successfully, you should see a green checkmark indicating a successful build.



← Maven Build

✓ changed the workfloww #6

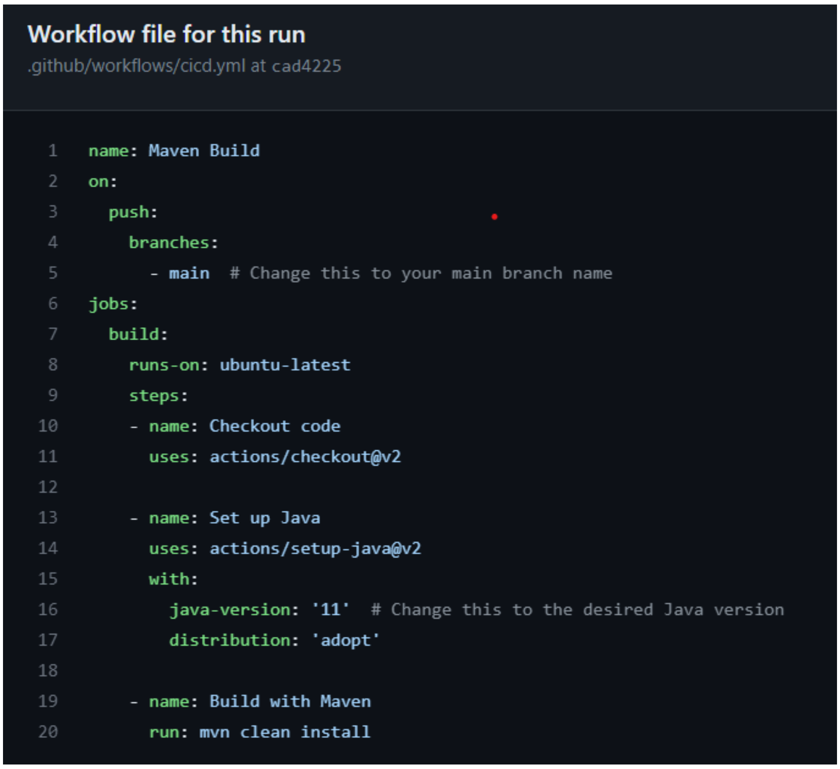
Triggered via push 12 minutes ago	Status	Total duration	Artifacts
Mridul1703 pushed -o- cad4225 main	Success	26s	–

cicd.yml
on: push

✓ build 19s

Step 5: Verify the Build Artifacts

- If the build was successful, navigate to the "Actions" tab on your GitHub repository, and click on the latest workflow run.
- In the workflow details, you can find the "Artifacts" section. Click on the artifact(s) to download and verify the build artifacts.



Workflow file for this run
.github/workflows/cicd.yml at cad4225

```yaml
name: Maven Build
on:
  push:
    branches:
      - main  # Change this to your main branch name
jobs:
  build:
    runs-on: ubuntu-latest
    steps:
    - name: Checkout code
      uses: actions/checkout@v2

    - name: Set up Java
      uses: actions/setup-java@v2
      with:
        java-version: '11'  # Change this to the desired Java version
        distribution: 'adopt'

    - name: Build with Maven
      run: mvn clean install
```