

Supplementary Exercise 4 Report

Name: Shizhen Yang

Pitt ID: shy77

Part 1: CI/CD Pipelines

1. Add Hello World Workflow

Yes

Created and executed a manual workflow (`manual.yml`) successfully via GitHub Actions.

2. Add Maven CI Workflow

Yes

Added `maven-ci.yml` with `maven_test` and `update_dependence_graph` jobs. Configured Maven setup, JDK 11, code checkout, and Jacoco coverage.

3. Debug Maven Test Job

Yes

Initial test failures were caused by placeholder code in `src/`. Replaced `src/` with the working code from Exercise 2. After the update, the `maven_test` job passed with sufficient test coverage.

4. Debug update_dependence_graph Job

No

The job was correctly configured with cache sharing and job dependencies. However, GitHub returned a message: "The Dependency Graph is disabled for this repository." This setting requires organization-level access and cannot be enabled in a student-owned GitHub Classroom repository.

5. Enable Dependabot

Yes

Manually added `.github/dependabot.yml` to schedule weekly checks for Maven dependencies.

6. Add Maven Publish Workflow

Yes

Added `maven-publish.yml` and modified `pom.xml` with distribution management and updated artifact ID.

7. Deploy Maven Package to GitHub Packages

No

Deployment failed with `422 Unprocessable Entity` due to group ID/repo path mismatch. All configuration steps were verified, but the issue persisted. This is a known limitation when publishing packages to GitHub Packages from organization-owned repos.

Reflections (End of Part 1)

I successfully completed the majority of tasks, including setting up workflows, running tests with Maven, generating a build cache, and scheduling Dependabot updates. I learned how GitHub Actions workflows are composed of jobs and steps and how job dependencies, caching, and permission scopes affect pipeline execution.

Two tasks could not be completed due to GitHub restrictions:

1. The `update_dependence_graph` job failed because the dependency graph was disabled by default on organization-owned repositories, and I had no permission to enable it.
2. The `maven-publish` job failed with a 422 error. Despite updating the group ID and verifying the URL, GitHub Packages rejected the deployment, which appears to be due to a metadata mismatch or GitHub's handling of packages in organization repositories.