CS489: Applied Software Development

Lesson 1b:

SOFTWARE BUILD AUTOMATION (Apache Maven, Gradle) and CI/CD PIPELINES

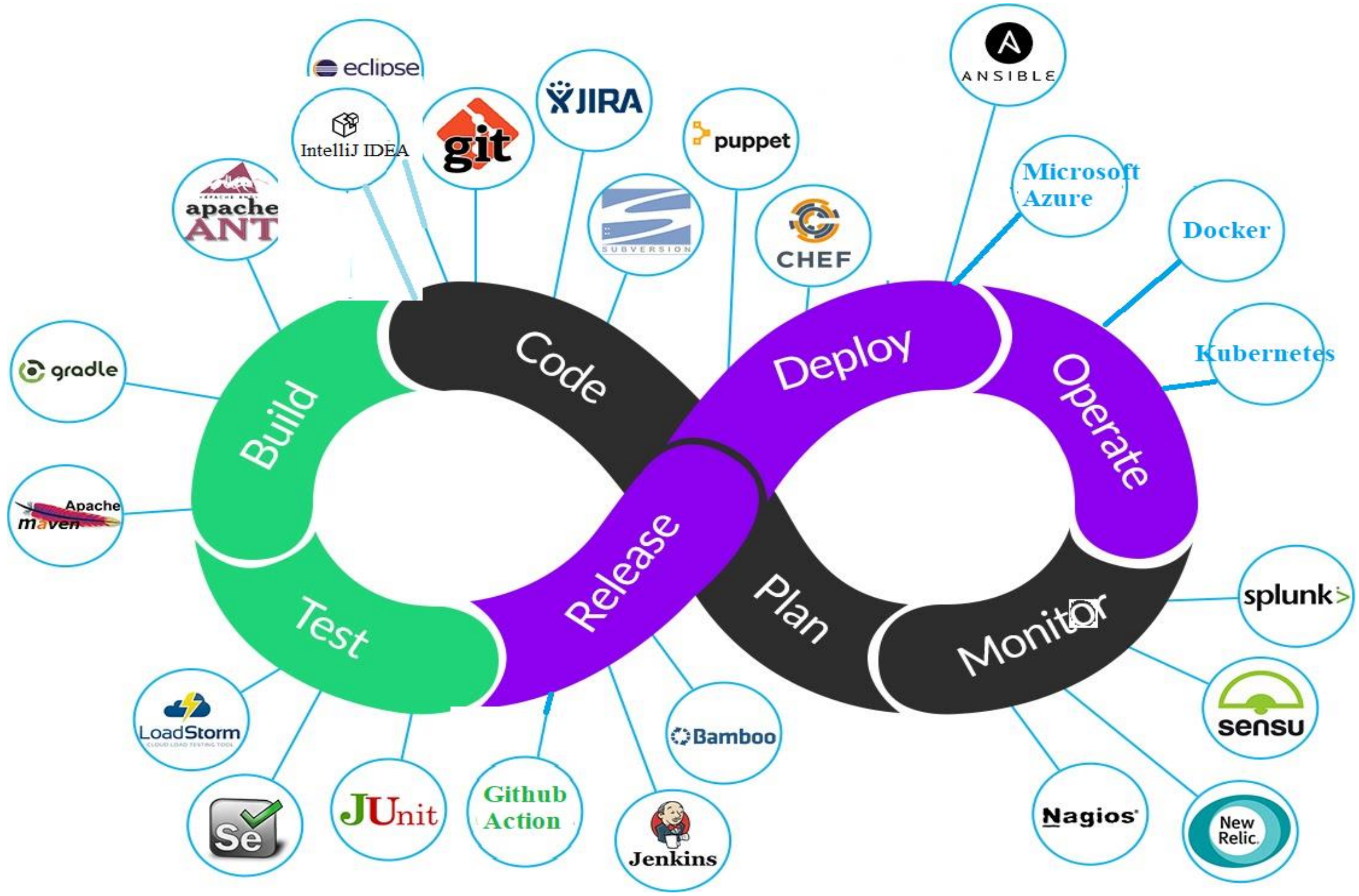
Wholeness

- Automation is the use of technology to perform tasks with reduced or no human assistance/intervention.
- Software Build constitutes the process of compiling source code, executing tests and creating a deployable unit or package/artifact.
- Science of Consciousness: Thought leads to Action. Action leads to Achievement. *Achievement leads to Fulfillment.*

Software Build Automation

- Automation of the Software build process is an essential requirement for a quick and efficient release of software product and subsequent updates.
- For modern business software products the Build -> Test -> Release cycle needs to be fast, frequent and error-free.

DevOps Life-cycle:
Plan -> Code -> Build -> Test -> Release



Build Automation Tools

- Commonly used Software build automation tools:
  - Apache Maven
  - Gradle
  - Github Action
  - Jenkins

Apache Maven

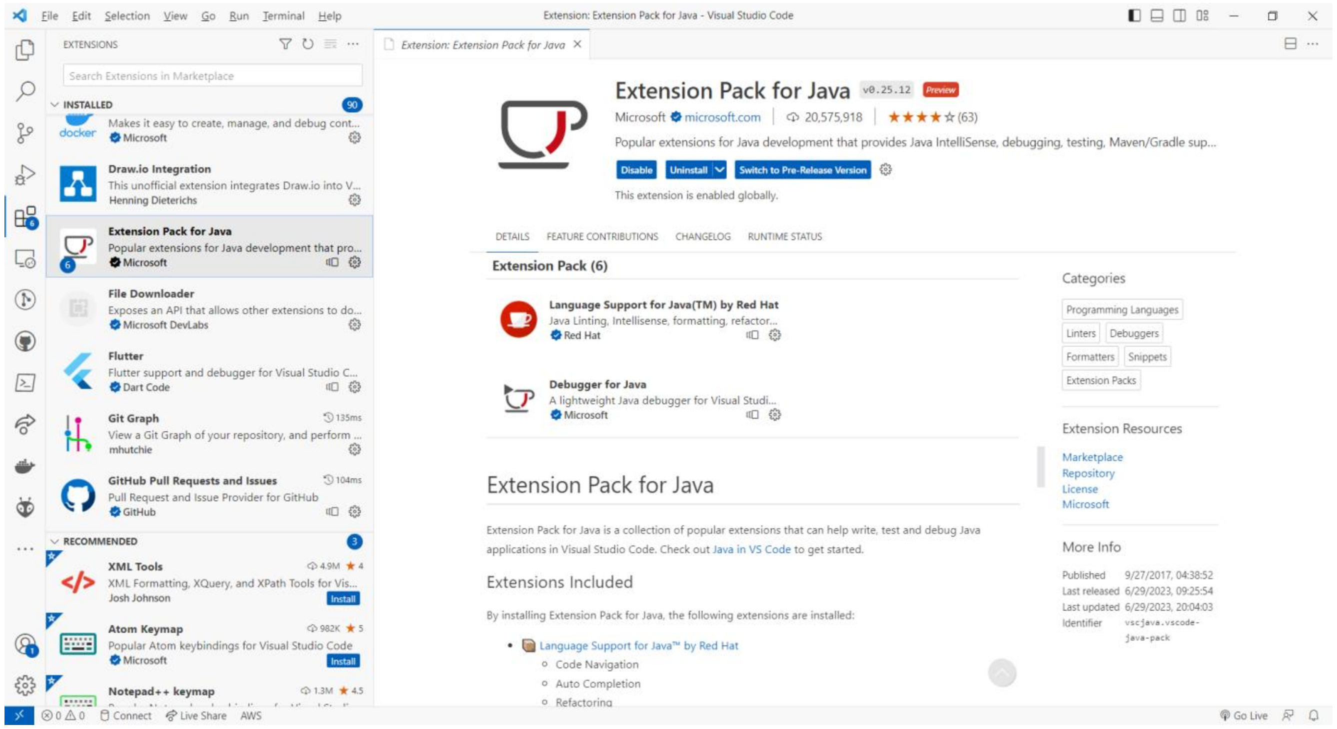
- Apache Maven is an open-source software build automation and dependency management tool, preeminent within the Java application development space.
- Maven uses the concept of a Project Object Model (POM) to manage the software build operation
- Every Maven project uses a pom.xml file which contains build configuration data

Demo: Apache Maven

- Maven requires the JDK installed
- Obtain and setup Apache Maven from the download webpage at: Maven – Download Apache Maven
- (Optional) Add ‘bin’ folder to PATH env var
- (Optional) Add environment variables named:
  - MAVEN_HOME
  - M2_HOME
- Test: Open a cmd/terminal and run – mvn -v

Demo: Apache Maven

- Create a sample Project and execute a Maven build
- Using VS Code with Extension Pack for Java


File Edit Selection View Go Run Terminal Help
Extension: Extension Pack for Java - Visual Studio Code
EXTENSIONS
Search Extensions in Marketplace
INSTALLED
RECOMMENDED
Extension: Extension Pack for Java
Disable
Uninstall
Switch to Pre-Release Version
DETAILS
FEATURE CONTRIBUTIONS
CHANGELOG
RUNTIME STATUS
Install
Go Live
Connect
Live Share
AWS

Makes it easy to create, manage, and debug cont...
Microsoft

Draw.io Integration
This unofficial extension integrates Draw.io into V...
Henning Dieterichs

Extension Pack for Java
Popular extensions for Java development that pro...
Microsoft

File Downloader
Exposes an API that allows other extensions to do...
Microsoft DevLabs

Flutter
Flutter support and debugger for Visual Studio C...
Dart Code

Git Graph 135ms
View a Git Graph of your repository, and perform ...
mhutchie

GitHub Pull Requests and Issues 104ms
Pull Request and Issue Provider for GitHub
GitHub

XML Tools 4.9M ★ 4
XML Formatting, XQuery, and XPath Tools for Vis...
Josh Johnson

Atom Keymap 982K ★ 5
Popular Atom keybindings for Visual Studio Code
Microsoft

Notepad++ keymap 1.3M ★ 4.5

Extension Pack for Java v0.25.12 Preview
Microsoft microsoft.com | 20,575,918 | ★★★★☆ (63)
Popular extensions for Java development that provides Java IntelliSense, debugging, testing, Maven/Gradle sup...
This extension is enabled globally.

Extension Pack (6)

Language Support for Java(TM) by Red Hat
Java Linting, Intellisense, formatting, refactor...
Red Hat

Debugger for Java
A lightweight Java debugger for Visual Studi...
Microsoft

Extension Pack for Java

Extension Pack for Java is a collection of popular extensions that can help write, test and debug Java applications in Visual Studio Code. Check out Java in VS Code to get started.

Extensions Included

By installing Extension Pack for Java, the following extensions are installed:

- Language Support for Java™ by Red Hat
  - Code Navigation
  - Auto Completion
  - Refactoring

Categories
Programming Languages, Linters, Debuggers, Formatters, Snippets, Extension Packs

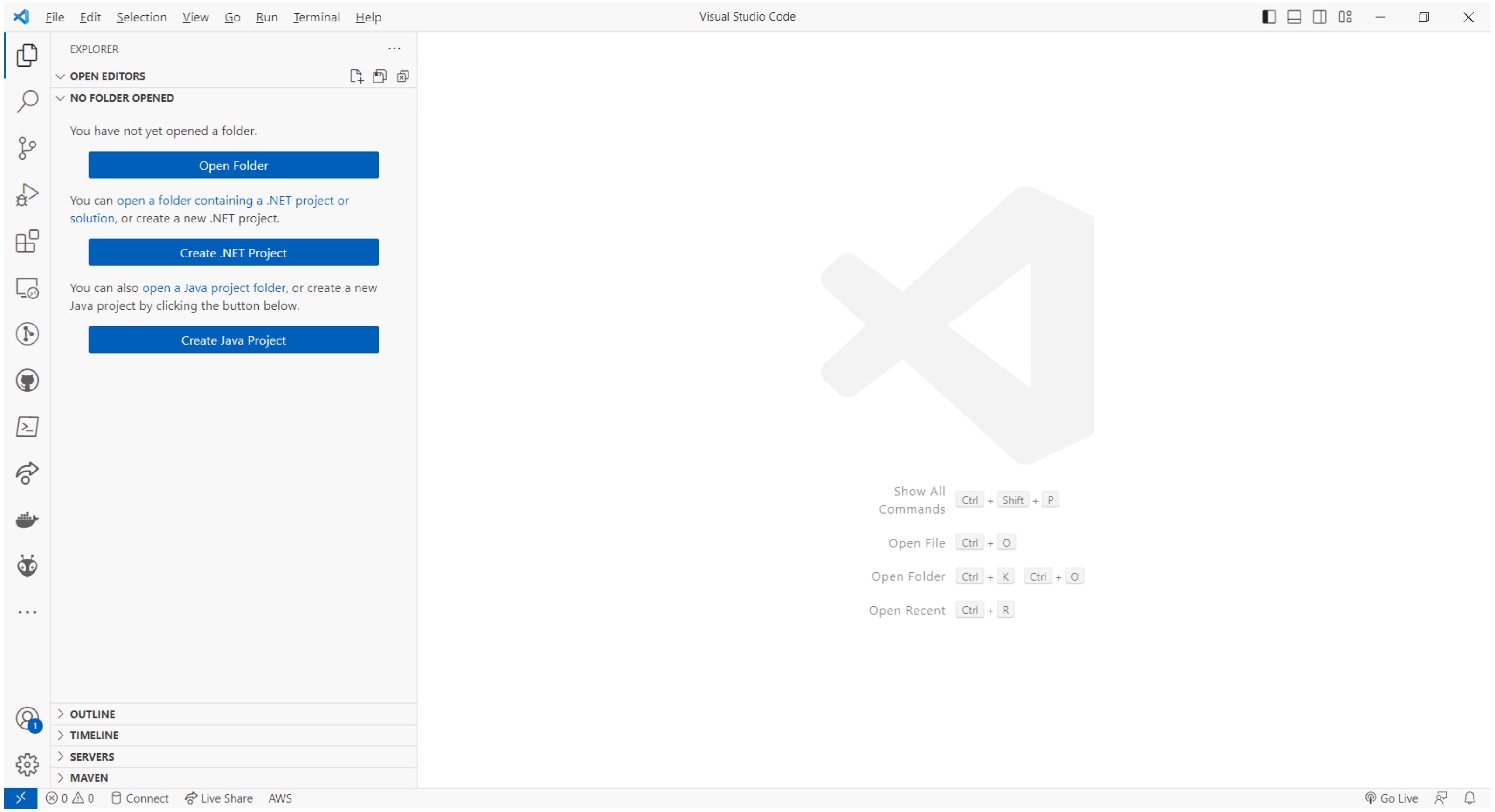
Extension Resources
Marketplace
Repository
License
Microsoft

More Info

Published	9/27/2017, 04:38:52
Last released	6/29/2023, 09:25:54
Last updated	6/29/2023, 20:04:03
Identifier	vscjava.vscode-java-pack

Demo: Apache Maven

- Select the “Create Java Project” button



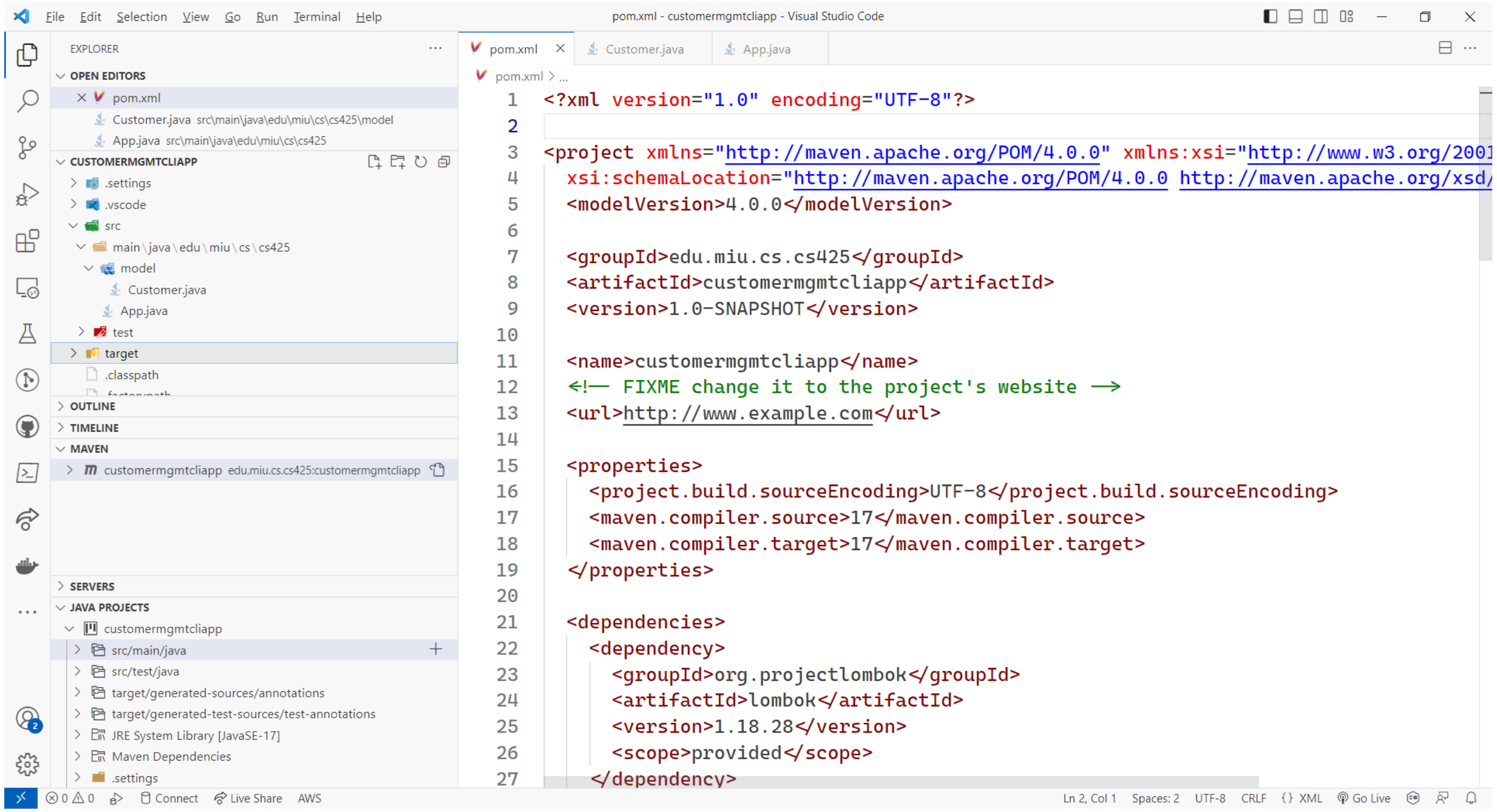
You have not yet opened a folder.

You can open a folder containing a .NET project or solution, or create a new .NET project.

You can also open a Java project folder, or create a new Java project by clicking the button below.

Demo: Apache Maven

- Project code implemented - CustomerMgmtCLIApp

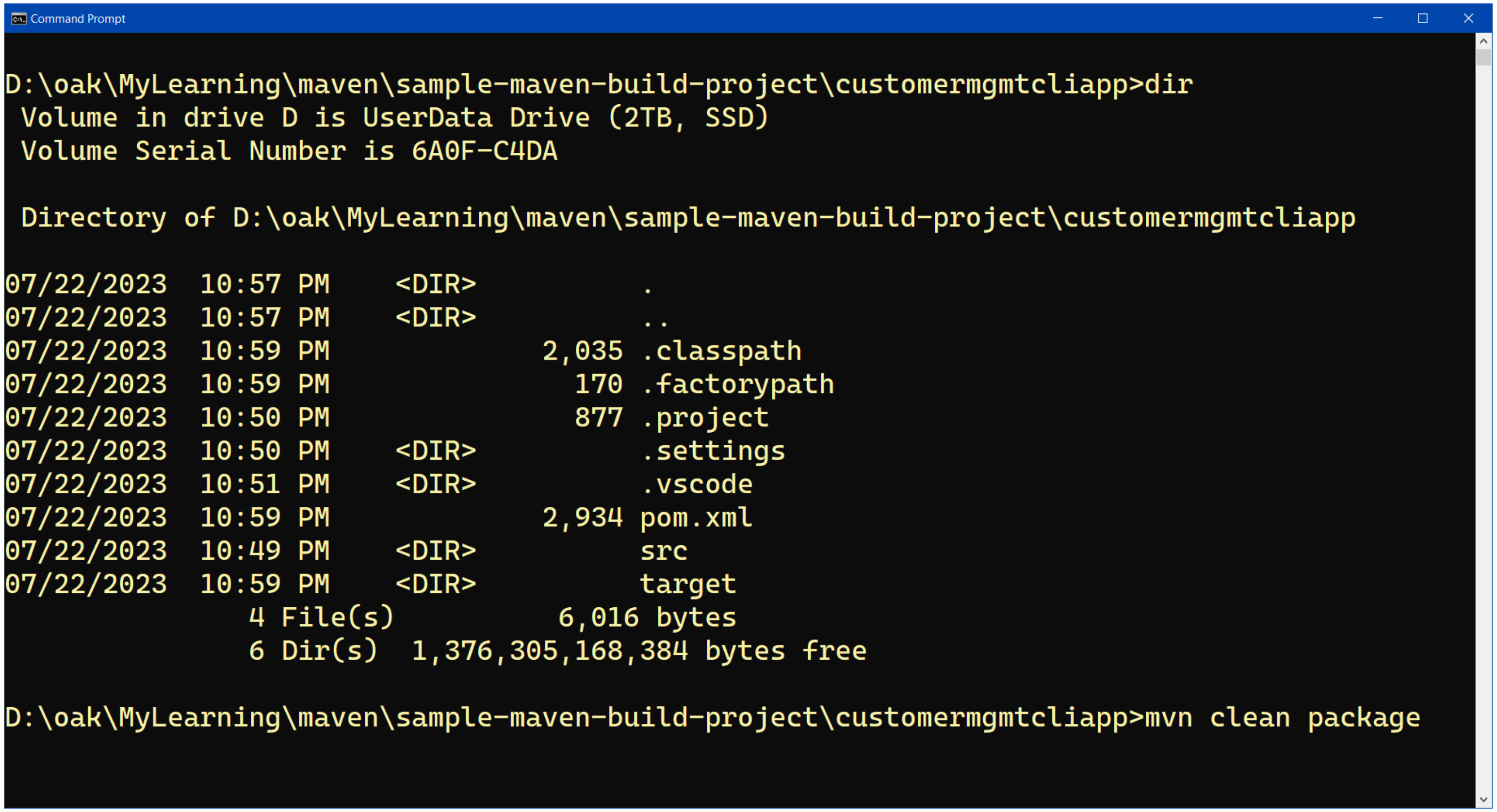


File Edit Selection View Go Run Terminal Help
pom.xml - customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
pom.xml
Customer.java src\main\java\edu\miu\cs\cs425\model
App.java src\main\java\edu\miu\cs\cs425
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
main \ java \ edu \ miu \ cs \ cs425
model
Customer.java
App.java
test
target
.classpath
OUTLINE
TIMELINE
MAVEN
customermgmtcliapp edu.miu.cs.cs425:customermgmtcliapp
SERVERS
JAVA PROJECTS
customermgmtcliapp
src/main/java
src/test/java
target/generated-sources/annotations
target/generated-test-sources/test-annotations
JRE System Library [JavaSE-17]
Maven Dependencies
.settings
pom.xml
Customer.java
App.java
pom.xml > ...
Connect
Live Share
AWS
Ln 2, Col 1
Spaces: 2
UTF-8
CRLF
XML
Go Live

```xml
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/
  <modelVersion>4.0.0</modelVersion>

  <groupId>edu.miu.cs.cs425</groupId>
  <artifactId>customermgmtcliapp</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>customermgmtcliapp</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>17</maven.compiler.source>
    <maven.compiler.target>17</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <version>1.18.28</version>
      <scope>provided</scope>
    </dependency>
```

Demo: Apache Maven

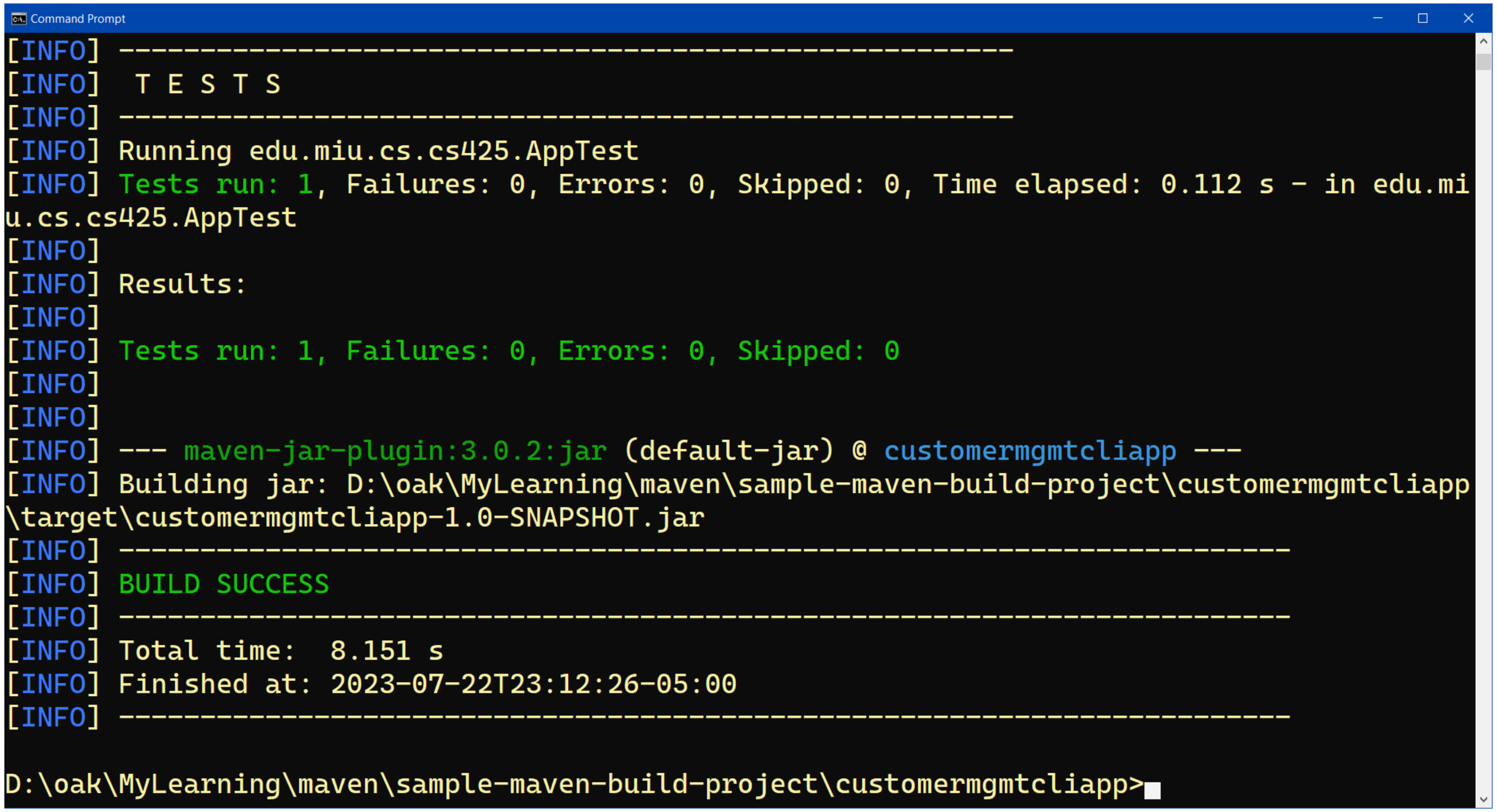
- Execute Maven clean package


Command Prompt

```
D:\oak\MyLearning\maven\sample-maven-build-project\customermgmtcliapp>dir
 Volume in drive D is UserData Drive (2TB, SSD)
 Volume Serial Number is 6A0F-C4DA

 Directory of D:\oak\MyLearning\maven\sample-maven-build-project\customermgmtcliapp

07/22/2023  10:57 PM    <DIR>          .
07/22/2023  10:57 PM    <DIR>          ..
07/22/2023  10:59 PM             2,035 .classpath
07/22/2023  10:59 PM               170 .factorypath
07/22/2023  10:50 PM               877 .project
07/22/2023  10:50 PM    <DIR>          .settings
07/22/2023  10:51 PM    <DIR>          .vscode
07/22/2023  10:59 PM             2,934 pom.xml
07/22/2023  10:49 PM    <DIR>          src
07/22/2023  10:59 PM    <DIR>          target
               4 File(s)          6,016 bytes
               6 Dir(s)  1,376,305,168,384 bytes free

D:\oak\MyLearning\maven\sample-maven-build-project\customermgmtcliapp>mvn clean package
```

Demo: Apache Maven

- Build success



Command Prompt

```
[INFO] -------------------------------------------------------
[INFO]  T E S T S
[INFO] -------------------------------------------------------
[INFO] Running edu.miu.cs.cs425.AppTest
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 0.112 s - in edu.mi
u.cs.cs425.AppTest
[INFO]
[INFO] Results:
[INFO]
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
[INFO]
[INFO]
[INFO] --- maven-jar-plugin:3.0.2:jar (default-jar) @ customermgmtcliapp ---
[INFO] Building jar: D:\oak\MyLearning\maven\sample-maven-build-project\customermgmtcliapp
\target\customermgmtcliapp-1.0-SNAPSHOT.jar
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  8.151 s
[INFO] Finished at: 2023-07-22T23:12:26-05:00
[INFO] ------------------------------------------------------------------------

D:\oak\MyLearning\maven\sample-maven-build-project\customermgmtcliapp>
```

Key features of Apache Maven

- Project generator CLI tool e.g. mvn archetype:generate -DgroupId=ToolsQA -DartifactId=DemoMavenProject -DarchetypeArtifactId=maven-archetype-quickstart -DinteractiveMode=false
- Maven Archetypes – like predefined project templates for creating various types of apps
- Maven Repositories e.g Maven Central repo at https://mvnrepository.com/repos/central
- Maven Phases and Goals: mvn compile, mvn clean, mvn package, mvn test etc.
- Maven Plugins: e.g. maven-compiler-plugin etc

Gradle

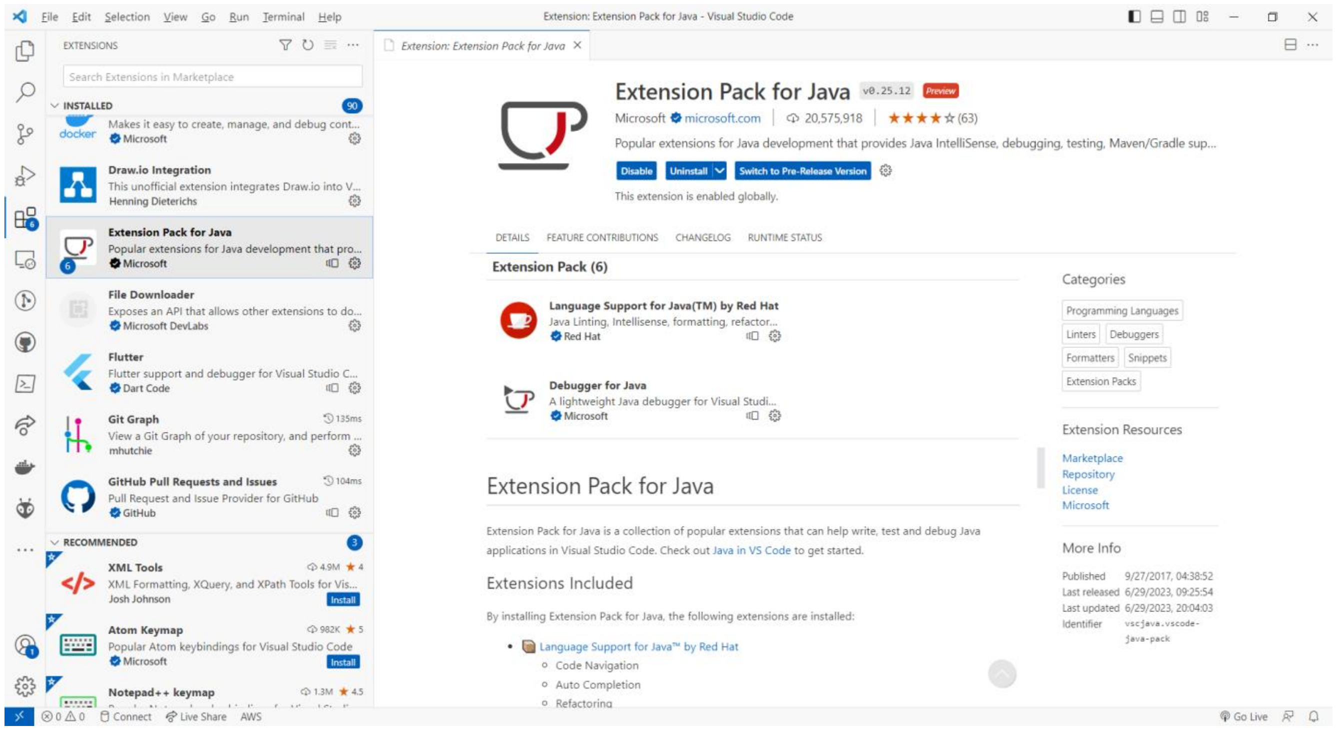
- Gradle is an open-source build automation tool, offering greater flexibility and higher performance than Maven, and popular within the Java application development space.
- Every Gradle project contains a build script, which is a file located in the project’s root directory usually named, *build.gradle*
- Build scripts define tasks, dependencies, plugins and other project configuration data.

Demo: Gradle

- Gradle requires the JDK installed (version 8+)
- Installing a local Gradle:
  - MacOS: You use - brew install gradle
  - For Windows: download zip bundle containing the binary from https://gradle.org/releases/, unzip, set GRADLE_HOME and add GRADLE_HOME\bin to PATH
- (Optional) Add 'bin' folder to PATH env var
- (Optional) Add environment variables named:
  - GRADLE_HOME
- Test: Open a cmd/terminal and run – gradle -v

Demo: Gradle

- Create a sample Project and execute a Gradle build
- Using VS Code with Extension Pack for Java



File Edit Selection View Go Run Terminal Help — Extension: Extension Pack for Java - Visual Studio Code

EXTENSIONS

Search Extensions in Marketplace

INSTALLED 90

docker — Makes it easy to create, manage, and debug cont... — Microsoft

Draw.io Integration — This unofficial extension integrates Draw.io into V... — Henning Dieterichs

Extension Pack for Java — Popular extensions for Java development that pro... — Microsoft

File Downloader — Exposes an API that allows other extensions to do... — Microsoft DevLabs

Flutter — Flutter support and debugger for Visual Studio C... — Dart Code

Git Graph — 135ms — View a Git Graph of your repository, and perform ... — mhutchie

GitHub Pull Requests and Issues — 104ms — Pull Request and Issue Provider for GitHub — GitHub

RECOMMENDED 3

XML Tools — 4.9M ★ 4 — XML Formatting, XQuery, and XPath Tools for Vis... — Josh Johnson — Install

Atom Keymap — 982K ★ 5 — Popular Atom keybindings for Visual Studio Code — Microsoft — Install

Notepad++ keymap — 1.3M ★ 4.5

Extension: Extension Pack for Java

Extension Pack for Java v0.25.12 Preview

Microsoft microsoft.com | 20,575,918 | ★★★★☆ (63)

Popular extensions for Java development that provides Java IntelliSense, debugging, testing, Maven/Gradle sup...

Disable | Uninstall | Switch to Pre-Release Version

This extension is enabled globally.

DETAILS | FEATURE CONTRIBUTIONS | CHANGELOG | RUNTIME STATUS

Extension Pack (6)

Language Support for Java(TM) by Red Hat — Java Linting, Intellisense, formatting, refactor... — Red Hat

Debugger for Java — A lightweight Java debugger for Visual Studi... — Microsoft

Extension Pack for Java

Extension Pack for Java is a collection of popular extensions that can help write, test and debug Java applications in Visual Studio Code. Check out Java in VS Code to get started.

Extensions Included

By installing Extension Pack for Java, the following extensions are installed:

- Language Support for Java™ by Red Hat
  - Code Navigation
  - Auto Completion
  - Refactoring

Categories: Programming Languages, Linters, Debuggers, Formatters, Snippets, Extension Packs

Extension Resources: Marketplace, Repository, License, Microsoft

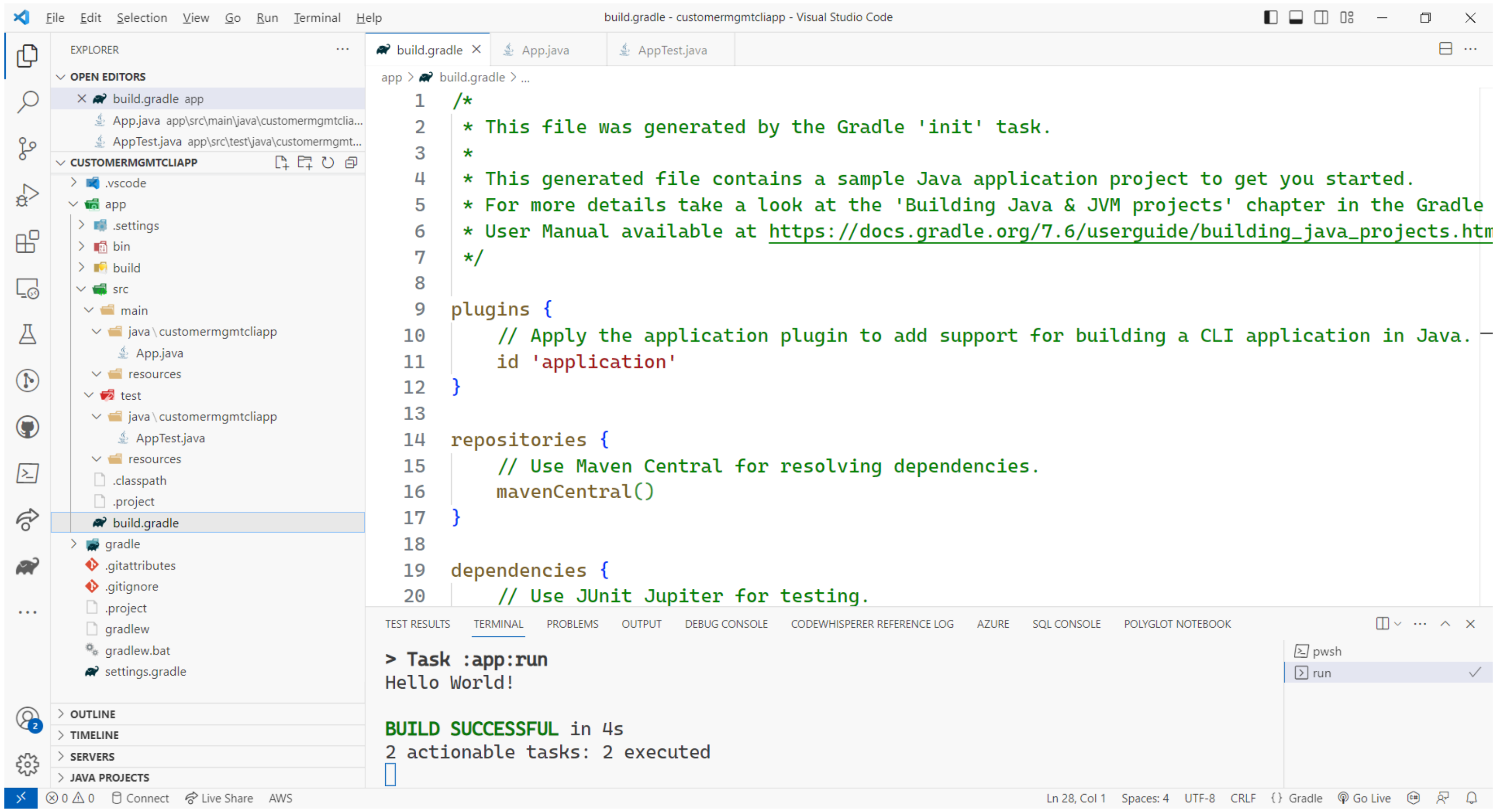
More Info

Published	9/27/2017, 04:38:52
Last released	6/29/2023, 09:25:54
Last updated	6/29/2023, 20:04:03
Identifier	vscjava.vscode-java-pack

⊗ 0 ⚠ 0 Connect Live Share AWS — Go Live

Demo: Gradle

- Project code implemented - CustomerMgmtCLIApp

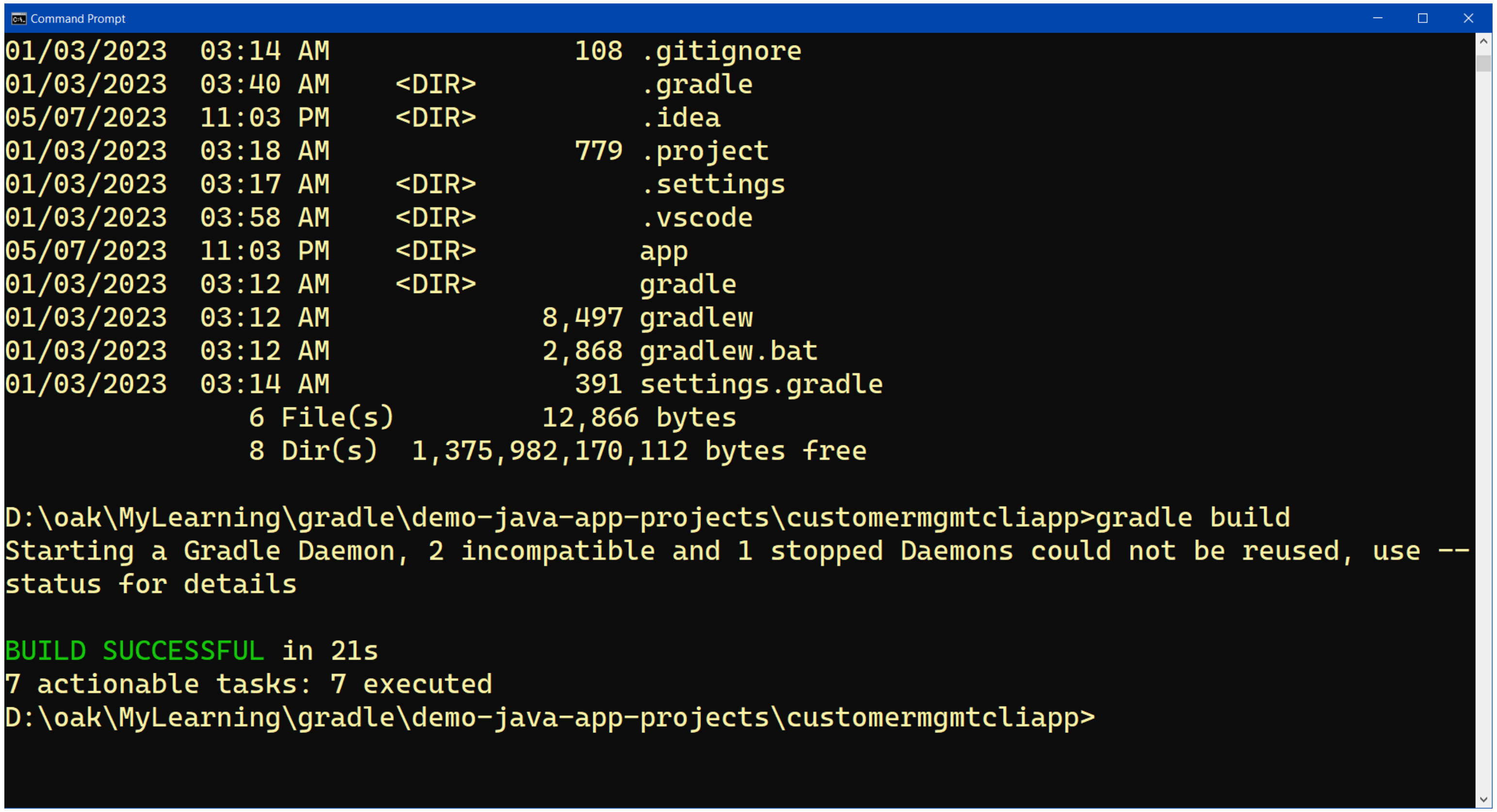


```
/*
 * This file was generated by the Gradle 'init' task.
 *
 * This generated file contains a sample Java application project to get you started.
 * For more details take a look at the 'Building Java & JVM projects' chapter in the Gradle
 * User Manual available at https://docs.gradle.org/7.6/userguide/building_java_projects.htm
 */

plugins {
    // Apply the application plugin to add support for building a CLI application in Java.
    id 'application'
}

repositories {
    // Use Maven Central for resolving dependencies.
    mavenCentral()
}

dependencies {
    // Use JUnit Jupiter for testing.
```

```
> Task :app:run
Hello World!

BUILD SUCCESSFUL in 4s
2 actionable tasks: 2 executed
```

Demo: Apache Maven

- Execute gradle build

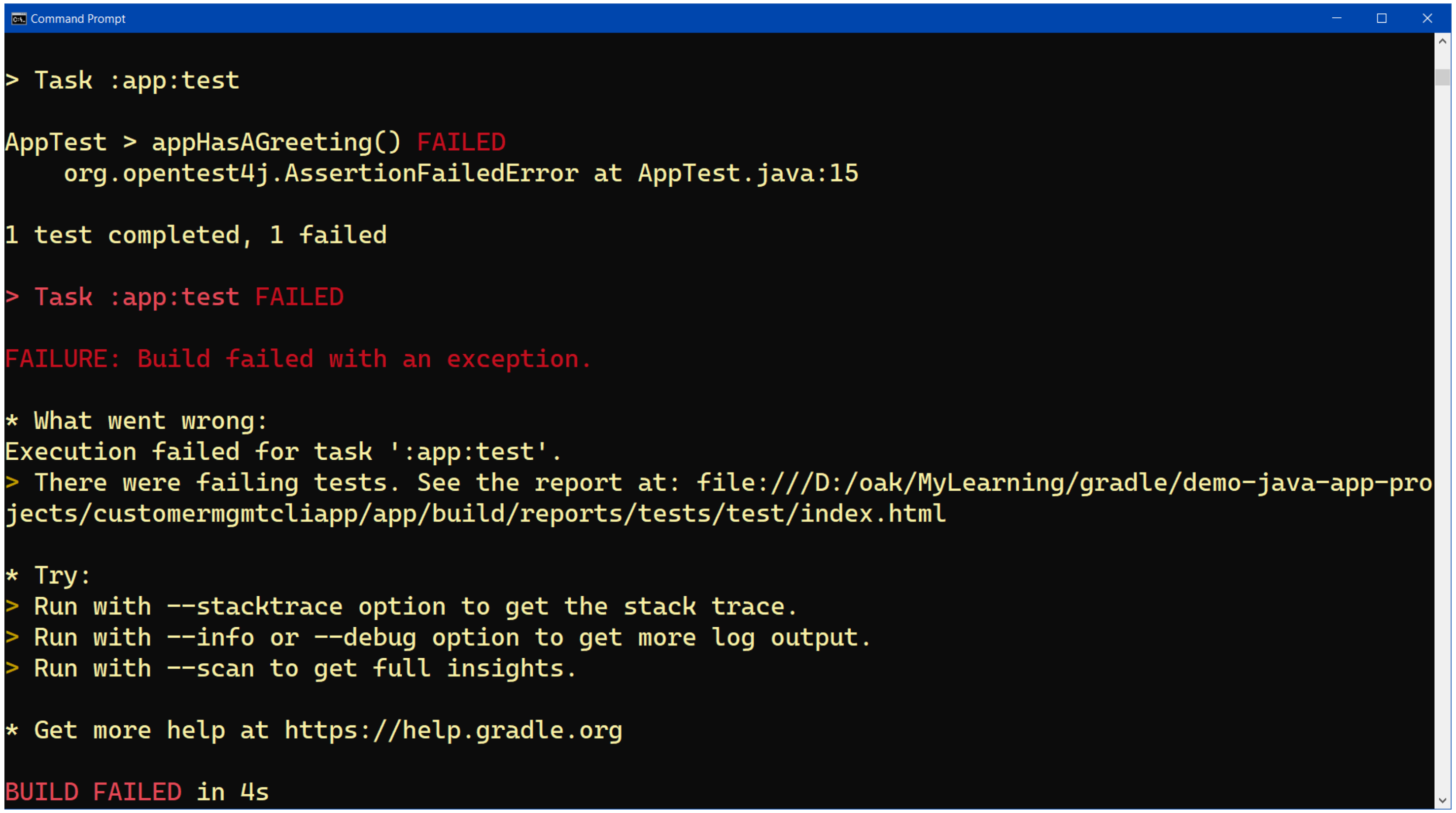


Command Prompt

```
01/03/2023  03:14 AM               108 .gitignore
01/03/2023  03:40 AM    <DIR>          .gradle
05/07/2023  11:03 PM    <DIR>          .idea
01/03/2023  03:18 AM               779 .project
01/03/2023  03:17 AM    <DIR>          .settings
01/03/2023  03:58 AM    <DIR>          .vscode
05/07/2023  11:03 PM    <DIR>          app
01/03/2023  03:12 AM    <DIR>          gradle
01/03/2023  03:12 AM             8,497 gradlew
01/03/2023  03:12 AM             2,868 gradlew.bat
01/03/2023  03:14 AM               391 settings.gradle
               6 File(s)         12,866 bytes
               8 Dir(s)  1,375,982,170,112 bytes free

D:\oak\MyLearning\gradle\demo-java-app-projects\customermgmtcliapp>gradle build
Starting a Gradle Daemon, 2 incompatible and 1 stopped Daemons could not be reused, use --
status for details

BUILD SUCCESSFUL in 21s
7 actionable tasks: 7 executed
D:\oak\MyLearning\gradle\demo-java-app-projects\customermgmtcliapp>
```

Demo: Apache Maven

- Change the Test code to cause/simulate failing test
- Execute gradle build and see that it fails



```
> Task :app:test

AppTest > appHasAGreeting() FAILED
    org.opentest4j.AssertionFailedError at AppTest.java:15

1 test completed, 1 failed

> Task :app:test FAILED

FAILURE: Build failed with an exception.

* What went wrong:
Execution failed for task ':app:test'.
> There were failing tests. See the report at: file:///D:/oak/MyLearning/gradle/demo-java-app-pro
jects/customermgmtcliapp/app/build/reports/tests/test/index.html

* Try:
> Run with --stacktrace option to get the stack trace.
> Run with --info or --debug option to get more log output.
> Run with --scan to get full insights.

* Get more help at https://help.gradle.org

BUILD FAILED in 4s
```

CI/CD

- CI/CD – Continuous Integration (CI), Continuous Delivery (CD). Note: The ‘D’ in CD can sometimes mean ‘Deployment’, which includes additional steps in the workflow beyond ‘Delivery’ (as will be explained later).
- CI/CD is aimed at automating the build, testing and deployment of software so that new features can ship faster and more reliably.
- Automation is a core principle for achieving DevOps success and CI/CD is a critical part.

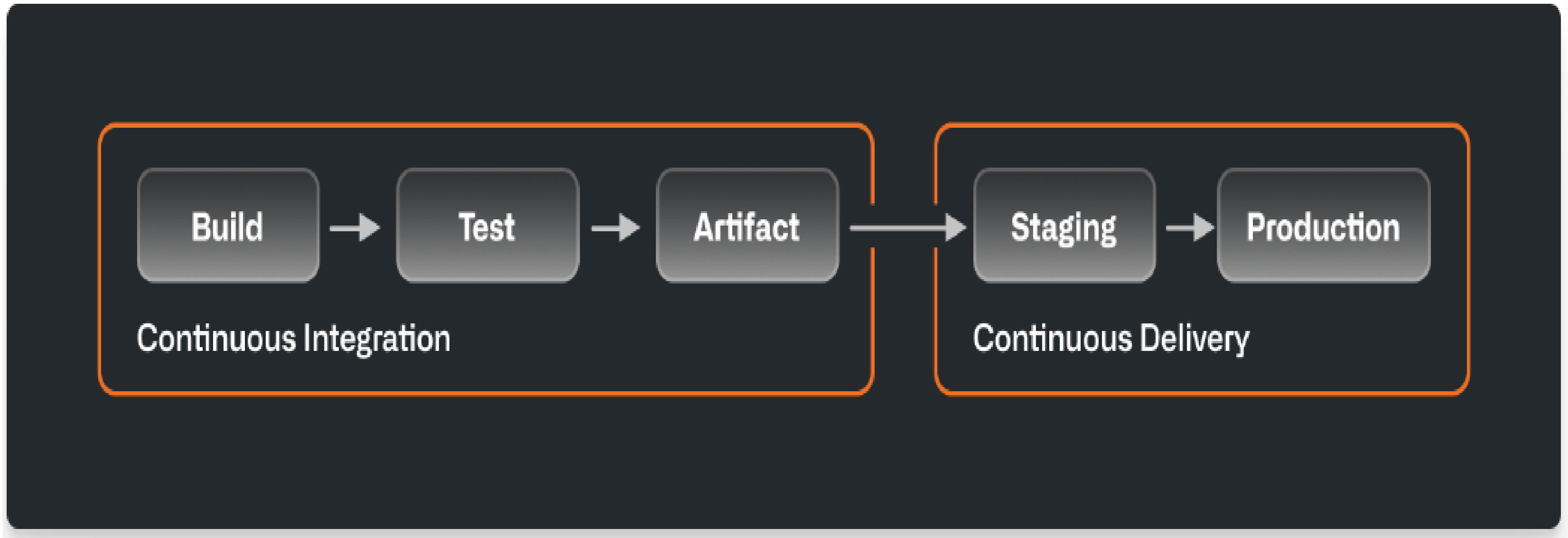
CI/CD

- A **CI/CD Pipeline** is a series of automated workflows setup as part of the DevOps methodology and best practice to enable the elimination or reduction of manual tasks.

- Continuous Integration (CI) automatically **builds, tests and integrates** (packages) code through use of a shared repository;

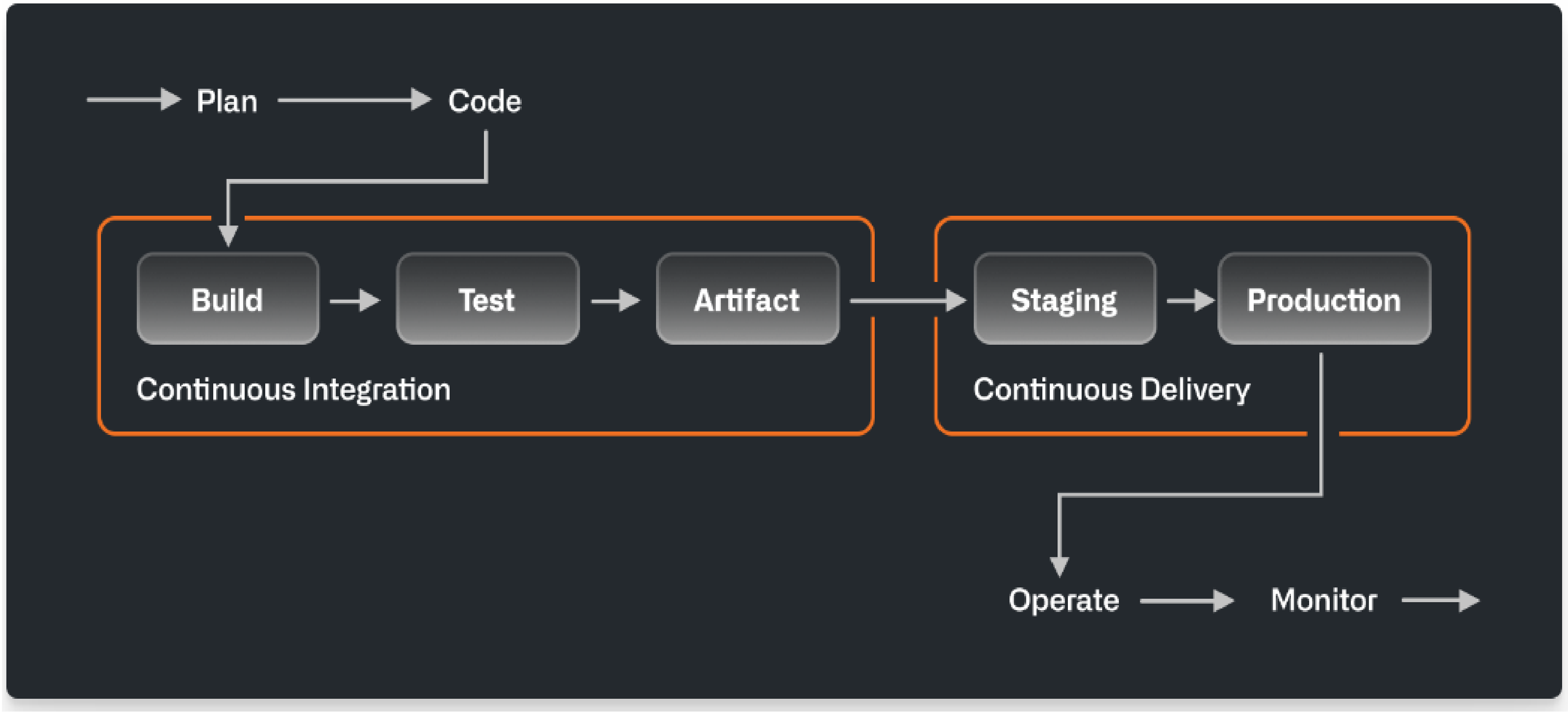
CI/CD

- Continuous Delivery (CD) automatically **delivers** the packaged software artifact to a production-ready (staging) environment for approval.



CI/CD

- Continuous Deployment (CD) automatically deploys the software to end-users or customers directly.



CI/CD

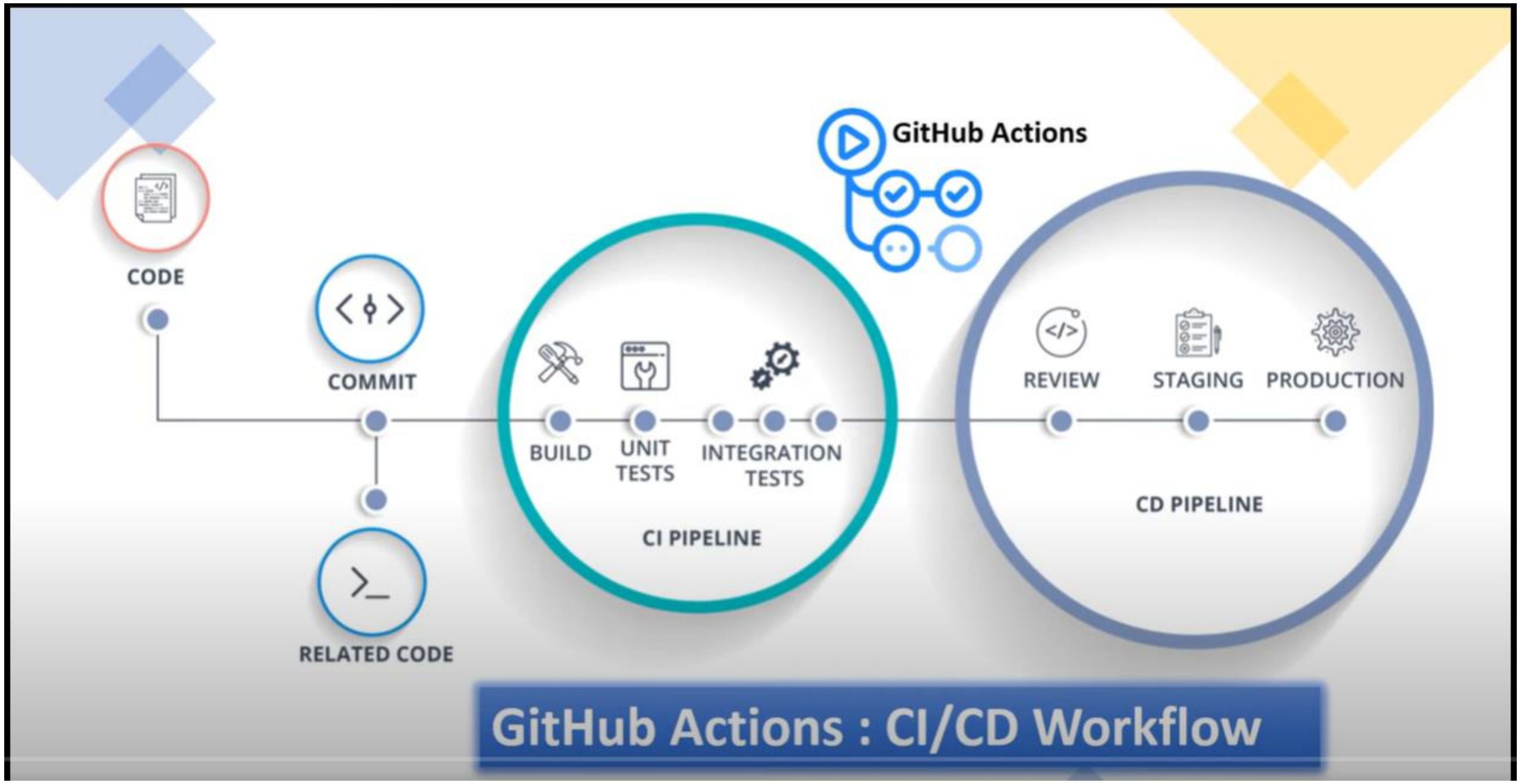
- Most modern software teams make CI/CD an integral part of their development workflow by using a combination of automated process, steps and tools:
  – Version Control system (VCS): CI begins via use of shared repositories, where the team can collaborate on code using tools like Git and Github.
  – Build automation tools: Teams use CI build tools to automatically compile, package code and config files into release candidate consisting of executable, deployable artifacts to be tested for quality, performance and other requirements.

Demo of CI/CD

- To demonstrate the setting-up of a CI/CD pipeline, we will be using **Github Actions**
- (Note: Alternatives – Jenkins, CircleCI, Bamboo, GitLab, Teamcity, Travis etc.)
- Unlike the above mentioned alternative CI/CD tools, Github Action is native to and fully integrated within the Github webapp. Hence a team that uses Github as the code repository will naturally prefer/choose Github Action for CI/CD pipeline.

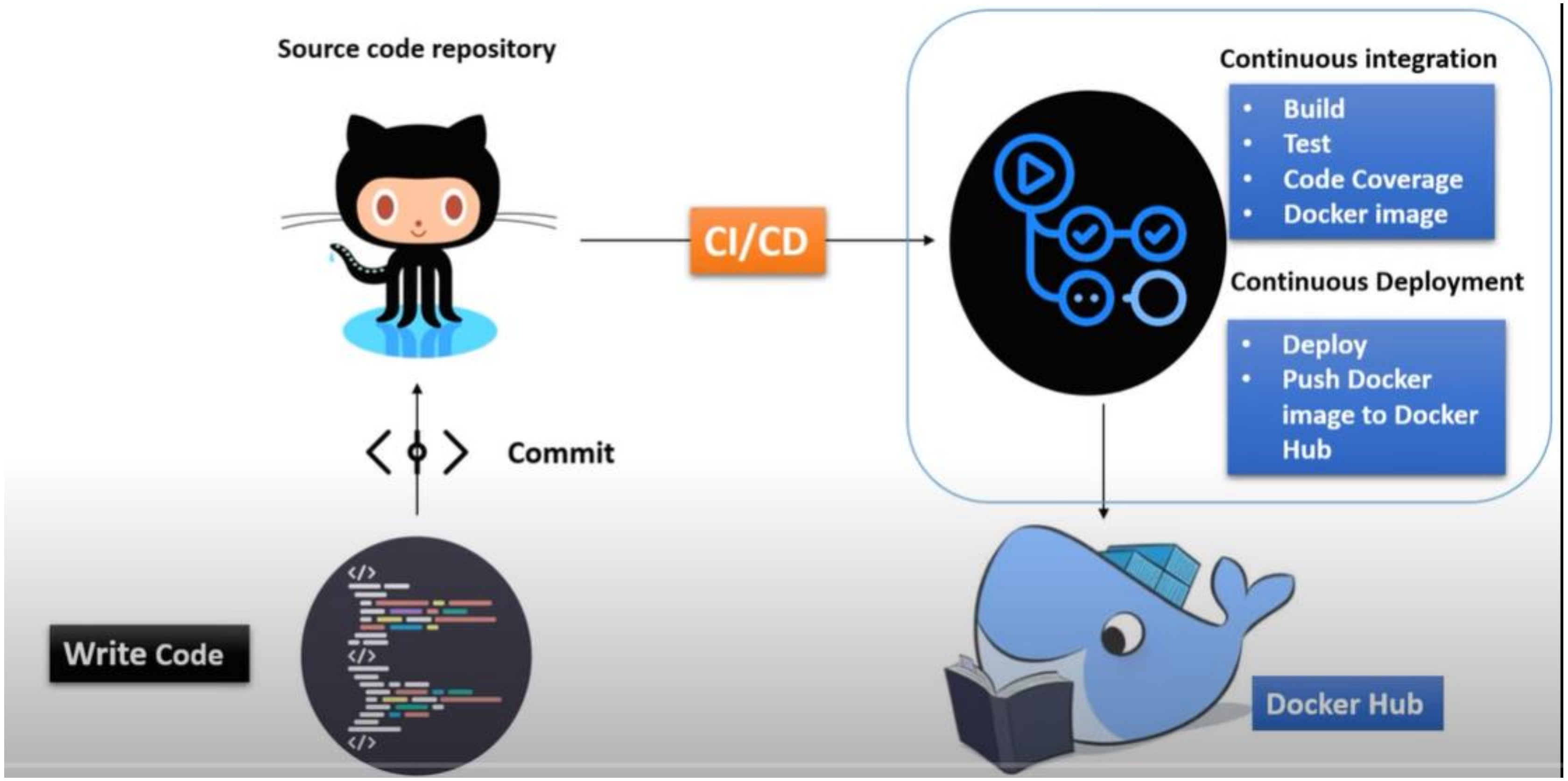
Demo of CI/CD

- Typical steps of activities/tasks with CI and CD



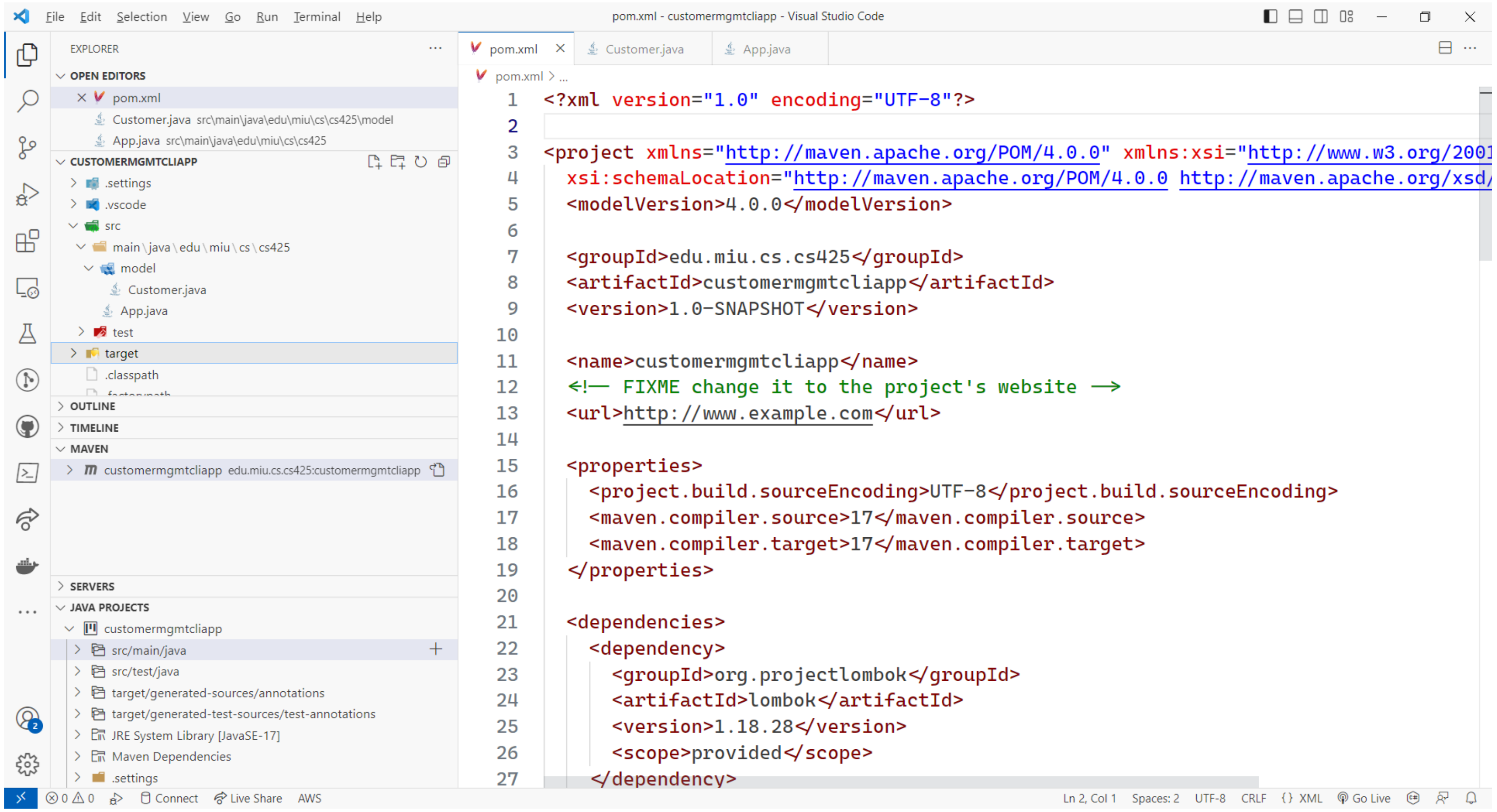
Demo of CI/CD

- Workflow and tools with CI and CD
- For this lesson1b, CI will only Build, Test and Package
- In future demo, Docker hub will be added as deployment target



Demo: Github Action

- Create a new Maven Java CLI project
- Project code implemented - CustomerMgmtCLIApp

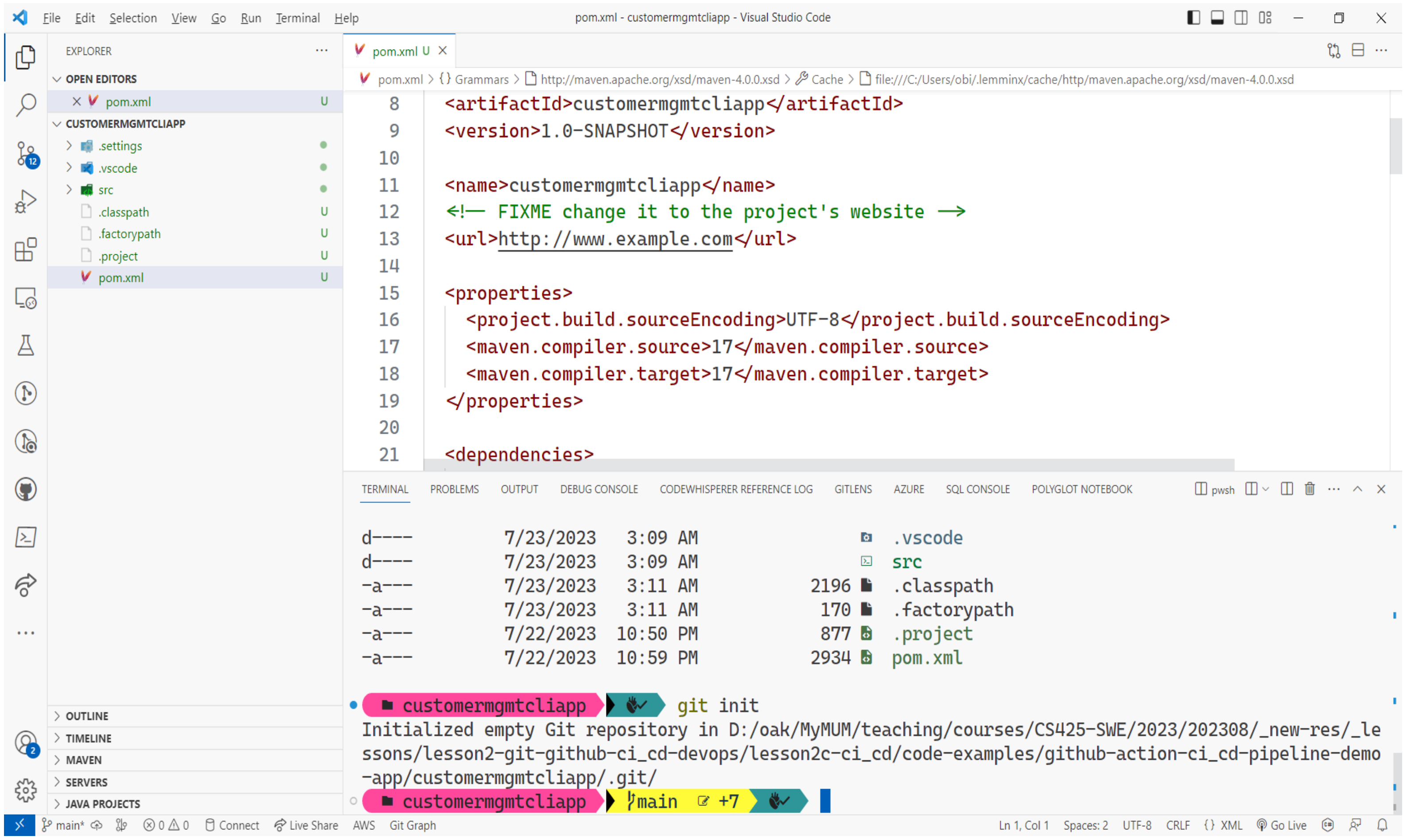


File Edit Selection View Go Run Terminal Help
pom.xml - customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
pom.xml
Customer.java src\main\java\edu\miu\cs\cs425\model
App.java src\main\java\edu\miu\cs\cs425
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
main \ java \ edu \ miu \ cs \ cs425
model
Customer.java
App.java
test
target
.classpath
OUTLINE
TIMELINE
MAVEN
customermgmtcliapp edu.miu.cs.cs425:customermgmtcliapp
SERVERS
JAVA PROJECTS
customermgmtcliapp
src/main/java
src/test/java
target/generated-sources/annotations
target/generated-test-sources/test-annotations
JRE System Library [JavaSE-17]
Maven Dependencies
.settings
pom.xml
Customer.java
App.java
pom.xml > ...
⊗ 0 ⚠ 0
Connect
Live Share
AWS
Ln 2, Col 1
Spaces: 2
UTF-8
CRLF
{} XML
Go Live

```xml
<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/
  <modelVersion>4.0.0</modelVersion>

  <groupId>edu.miu.cs.cs425</groupId>
  <artifactId>customermgmtcliapp</artifactId>
  <version>1.0-SNAPSHOT</version>

  <name>customermgmtcliapp</name>
  <!-- FIXME change it to the project's website -->
  <url>http://www.example.com</url>

  <properties>
    <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    <maven.compiler.source>17</maven.compiler.source>
    <maven.compiler.target>17</maven.compiler.target>
  </properties>

  <dependencies>
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <version>1.18.28</version>
      <scope>provided</scope>
    </dependency>
```

Demo: Github Action

- Initialize a new git local repository – open terminal and execute: git init

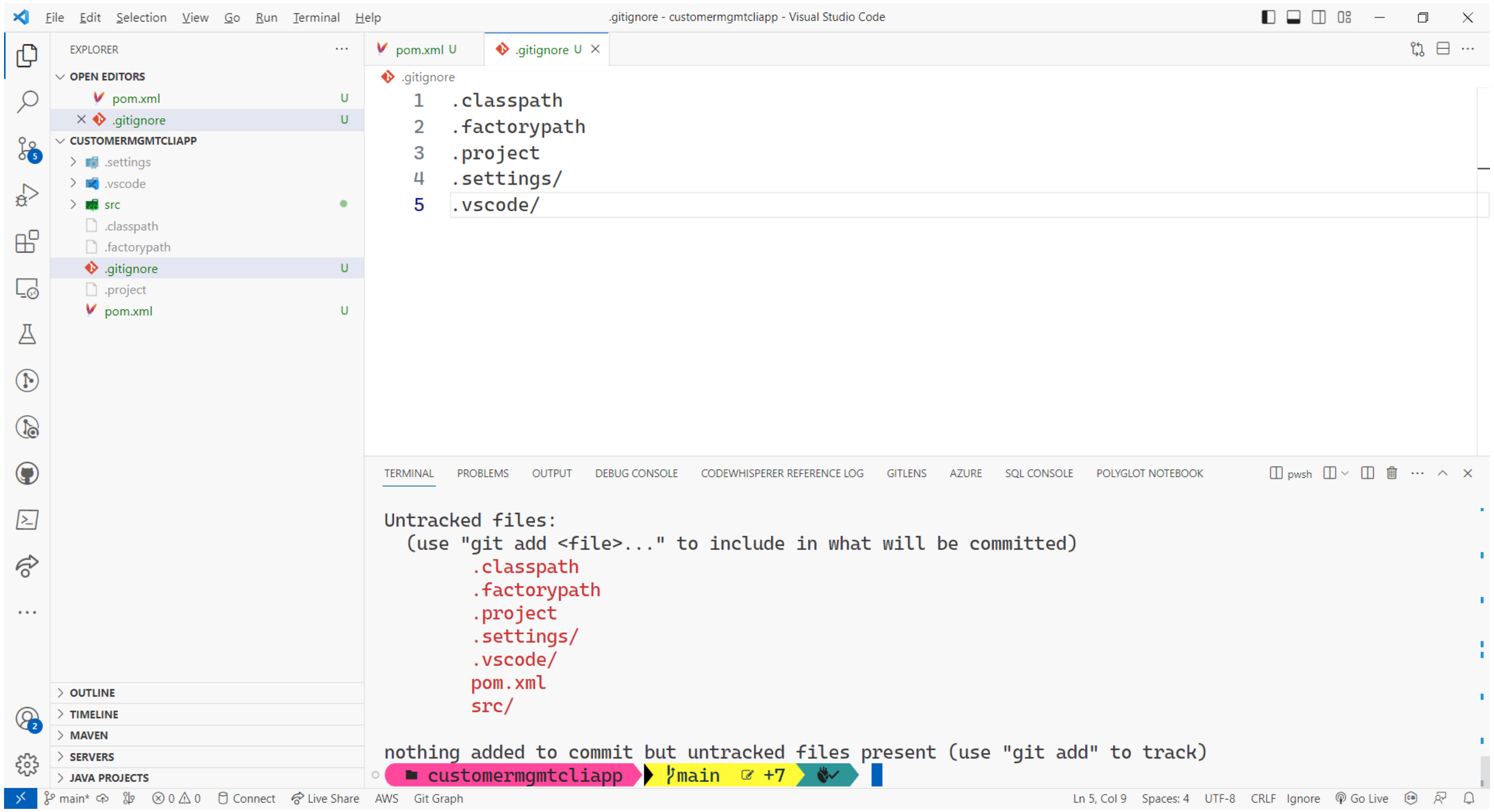

File Edit Selection View Go Run Terminal Help
pom.xml - customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
pom.xml U
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
.classpath U
.factorypath U
.project U
pom.xml U
OUTLINE
TIMELINE
MAVEN
SERVERS
JAVA PROJECTS
pom.xml > {} Grammars > http://maven.apache.org/xsd/maven-4.0.0.xsd > Cache > file:///C:/Users/obi/.lemminx/cache/http/maven.apache.org/xsd/maven-4.0.0.xsd
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE CODEWHISPERER REFERENCE LOG GITLENS AZURE SQL CONSOLE POLYGLOT NOTEBOOK
pwsh
main* 0 0 Connect Live Share AWS Git Graph
Ln 1, Col 1 Spaces: 2 UTF-8 CRLF {} XML Go Live

```xml
<artifactId>customermgmtcliapp</artifactId>
<version>1.0-SNAPSHOT</version>

<name>customermgmtcliapp</name>
<!-- FIXME change it to the project's website -->
<url>http://www.example.com</url>

<properties>
  <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
  <maven.compiler.source>17</maven.compiler.source>
  <maven.compiler.target>17</maven.compiler.target>
</properties>

<dependencies>
```

```
d----          7/23/2023   3:09 AM                .vscode
d----          7/23/2023   3:09 AM                src
-a---          7/23/2023   3:11 AM           2196 .classpath
-a---          7/23/2023   3:11 AM            170 .factorypath
-a---          7/22/2023  10:50 PM            877 .project
-a---          7/22/2023  10:59 PM           2934 pom.xml

customermgmtcliapp  git init
Initialized empty Git repository in D:/oak/MyMUM/teaching/courses/CS425-SWE/2023/202308/_new-res/_lessons/lesson2-git-github-ci_cd-devops/lesson2c-ci_cd/code-examples/github-action-ci_cd-pipeline-demo-app/customermgmtcliapp/.git/
customermgmtcliapp  main  +7
```

Demo: Github Action

- Add .gitignore file



.gitignore - customermgmtcliapp - Visual Studio Code

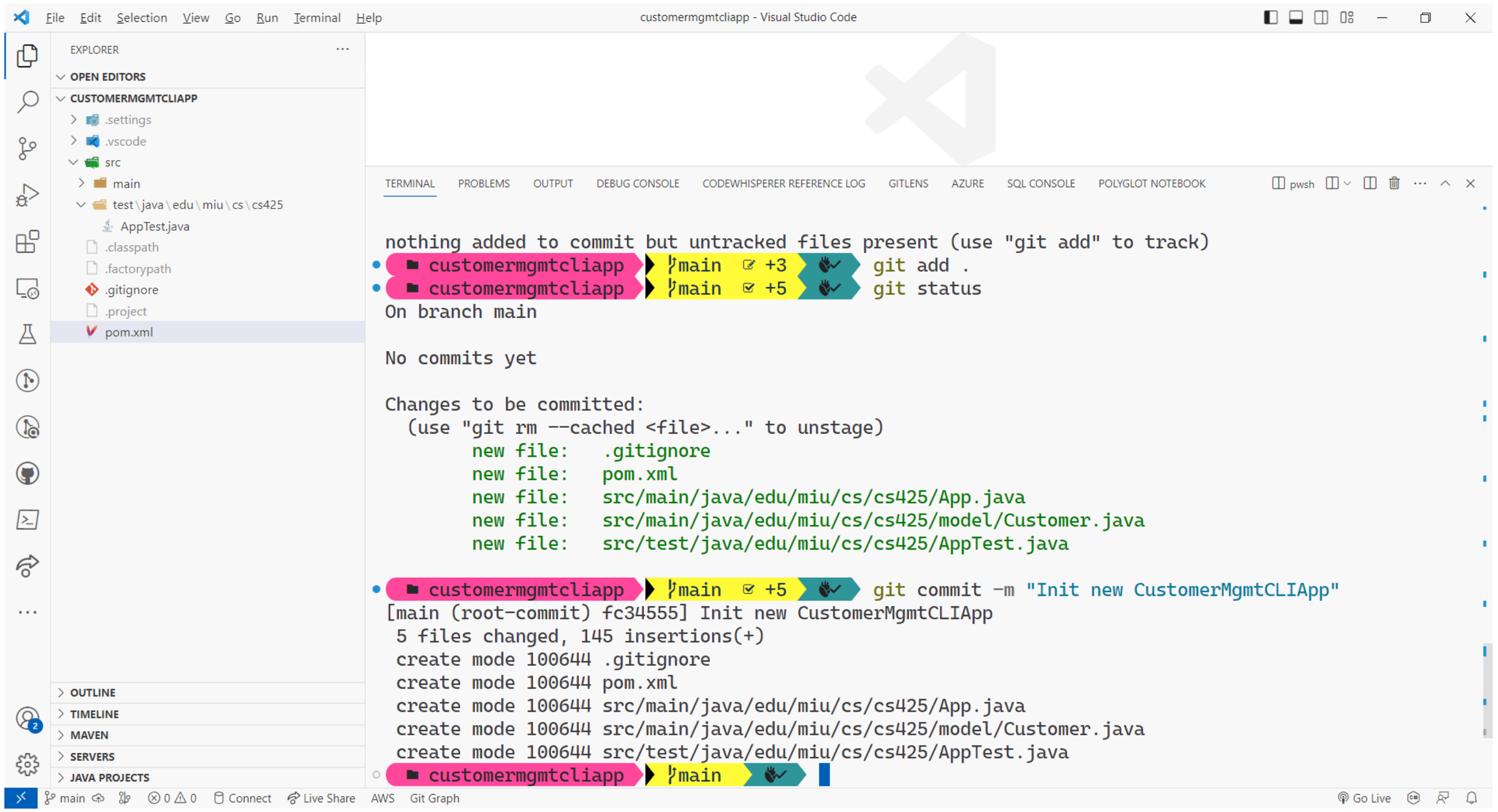
File Edit Selection View Go Run Terminal Help
EXPLORER
OPEN EDITORS
pom.xml U
.gitignore U
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
.classpath
.factorypath
.gitignore U
.project
pom.xml U
OUTLINE
TIMELINE
MAVEN
SERVERS
JAVA PROJECTS
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE CODEWHISPERER REFERENCE LOG GITLENS AZURE SQL CONSOLE POLYGLOT NOTEBOOK
main* Connect Live Share AWS Git Graph Ln 5, Col 9 Spaces: 4 UTF-8 CRLF Ignore Go Live

```
.classpath
.factorypath
.project
.settings/
.vscode/
```

```
Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .classpath
        .factorypath
        .project
        .settings/
        .vscode/
        pom.xml
        src/

nothing added to commit but untracked files present (use "git add" to track)
 customermgmtcliapp  main  +7
```

Demo: Github Action

- Execute: git add .
- Execute git commit –m "Init new CLI app"

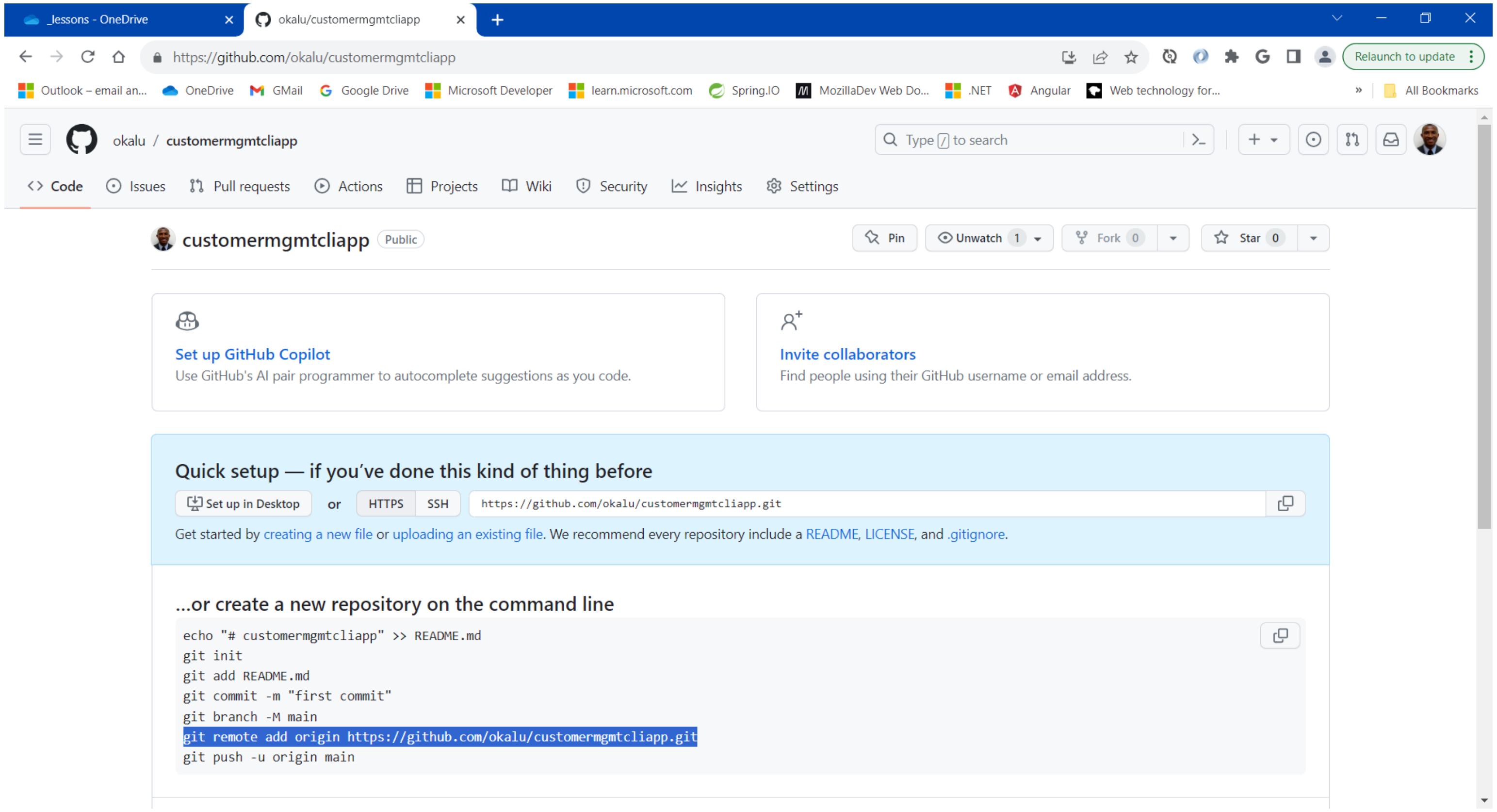


```
nothing added to commit but untracked files present (use "git add" to track)
● ■ customermgmtcliapp ᚠmain ✎ +3  git add .
● ■ customermgmtcliapp ᚠmain ✎ +5  git status
On branch main

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   .gitignore
        new file:   pom.xml
        new file:   src/main/java/edu/miu/cs/cs425/App.java
        new file:   src/main/java/edu/miu/cs/cs425/model/Customer.java
        new file:   src/test/java/edu/miu/cs/cs425/AppTest.java

● ■ customermgmtcliapp ᚠmain ✎ +5  git commit -m "Init new CustomerMgmtCLIApp"
[main (root-commit) fc34555] Init new CustomerMgmtCLIApp
 5 files changed, 145 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 pom.xml
 create mode 100644 src/main/java/edu/miu/cs/cs425/App.java
 create mode 100644 src/main/java/edu/miu/cs/cs425/model/Customer.java
 create mode 100644 src/test/java/edu/miu/cs/cs425/AppTest.java
○ ■ customermgmtcliapp ᚠmain
```

File Edit Selection View Go Run Terminal Help
customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
main
test \ java \ edu \ miu \ cs \ cs425
AppTest.java
.classpath
.factorypath
.gitignore
.project
pom.xml
OUTLINE
TIMELINE
MAVEN
SERVERS
JAVA PROJECTS
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE CODEWHISPERER REFERENCE LOG GITLENS AZURE SQL CONSOLE POLYGLOT NOTEBOOK
pwsh
main ⊗ 0 ⚠ 0 Connect Live Share AWS Git Graph
Go Live

Demo: Github Action

- Create the remote repository on github



https://github.com/okalu/customermgmtcliapp

okalu / customermgmtcliapp

customermgmtcliapp Public

Set up GitHub Copilot
Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Invite collaborators
Find people using their GitHub username or email address.

Quick setup — if you've done this kind of thing before

Set up in Desktop or HTTPS SSH `https://github.com/okalu/customermgmtcliapp.git`

Get started by creating a new file or uploading an existing file. We recommend every repository include a README, LICENSE, and .gitignore.

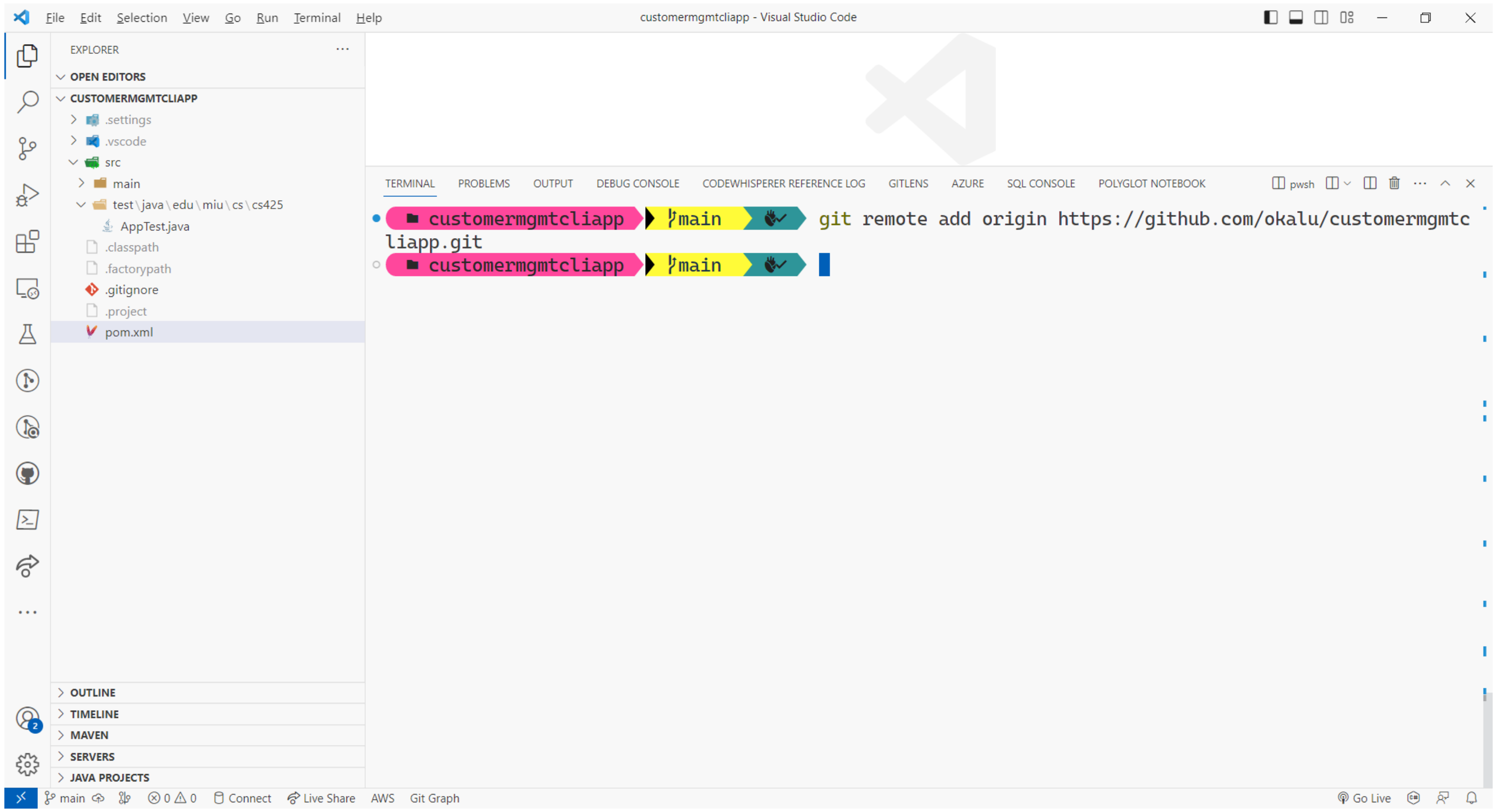
...or create a new repository on the command line

```
echo "# customermgmtcliapp" >> README.md
git init
git add README.md
git commit -m "first commit"
git branch -M main
git remote add origin https://github.com/okalu/customermgmtcliapp.git
git push -u origin main
```

Demo: Github Action

- Add reference named origin pointing to the remote repository by executing:

  git remote add origin <url>



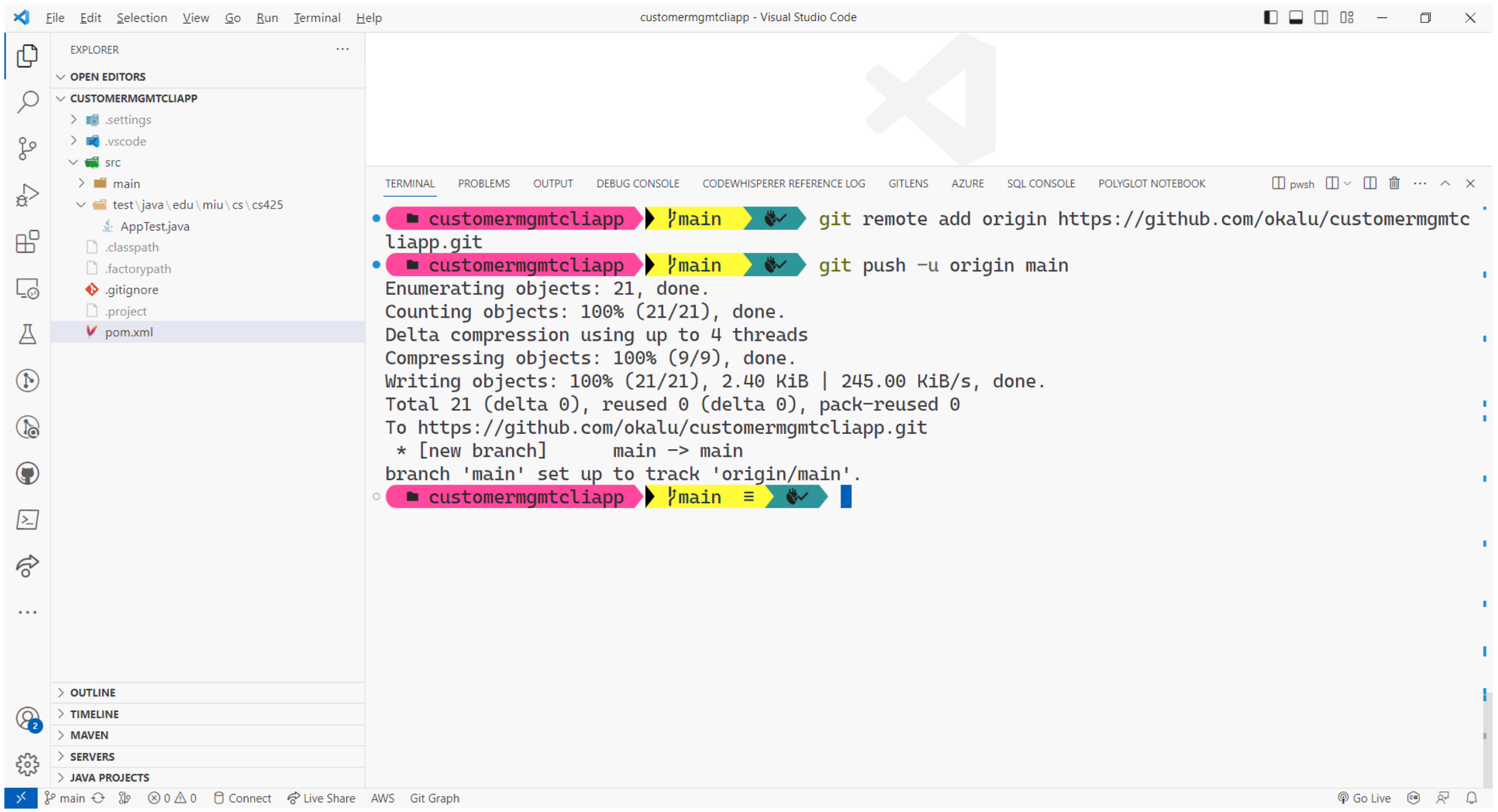
```
customermgmtcliapp  main  git remote add origin https://github.com/okalu/customermgmtcliapp.git
customermgmtcliapp  main
```

File Edit Selection View Go Run Terminal Help
customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
main
test \ java \ edu \ miu \ cs \ cs425
AppTest.java
.classpath
.factorypath
.gitignore
.project
pom.xml
OUTLINE
TIMELINE
MAVEN
SERVERS
JAVA PROJECTS
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE CODEWHISPERER REFERENCE LOG GITLENS AZURE SQL CONSOLE POLYGLOT NOTEBOOK
pwsh
main 0 0 Connect Live Share AWS Git Graph Go Live

Demo: Github Action

- Push the code to the remote repository by executing:

  git push –u origin main

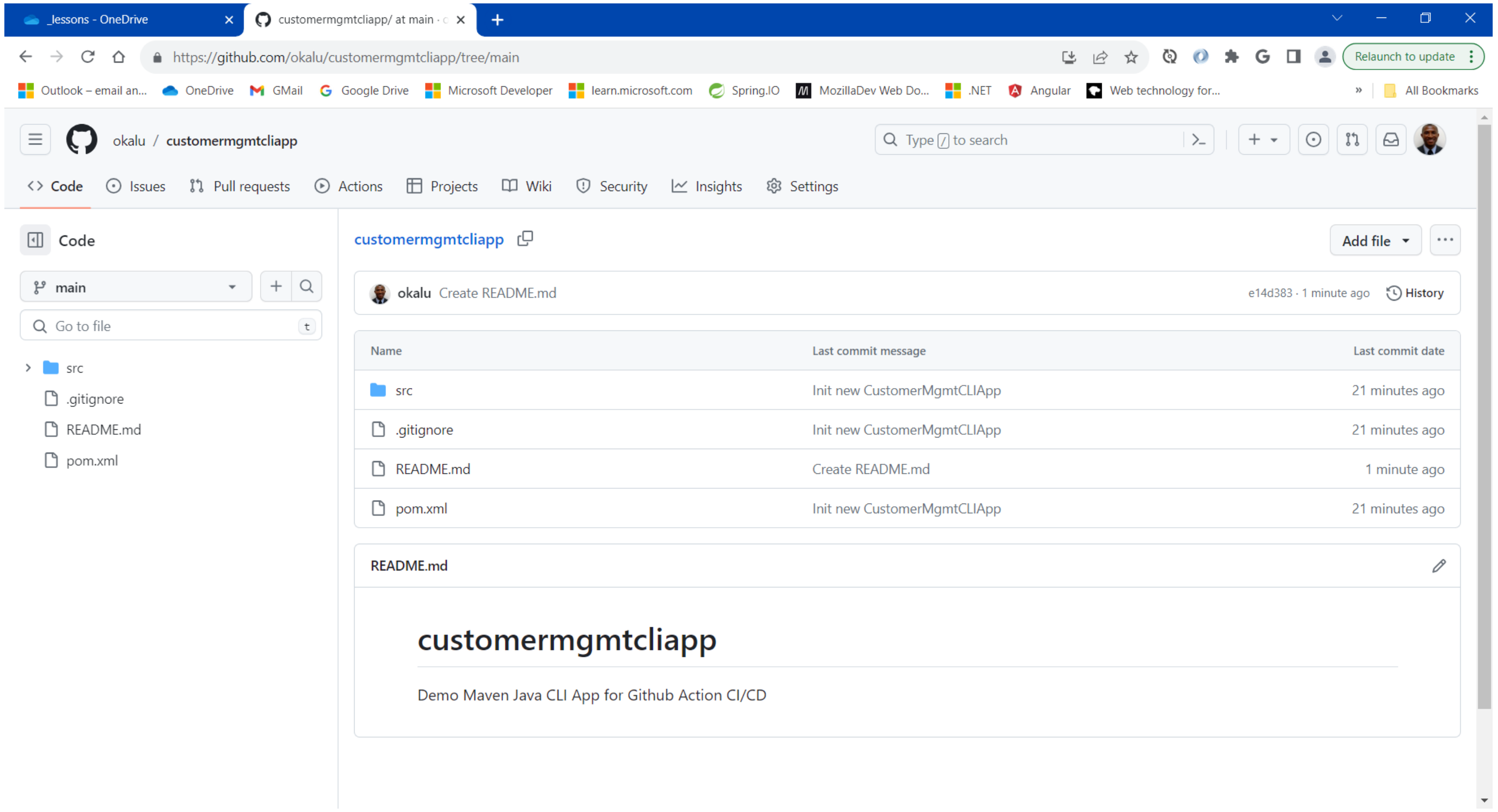


File Edit Selection View Go Run Terminal Help
customermgmtcliapp - Visual Studio Code
EXPLORER
OPEN EDITORS
CUSTOMERMGMTCLIAPP
.settings
.vscode
src
main
test \ java \ edu \ miu \ cs \ cs425
AppTest.java
.classpath
.factorypath
.gitignore
.project
pom.xml
OUTLINE
TIMELINE
MAVEN
SERVERS
JAVA PROJECTS
TERMINAL PROBLEMS OUTPUT DEBUG CONSOLE CODEWHISPERER REFERENCE LOG GITLENS AZURE SQL CONSOLE POLYGLOT NOTEBOOK
main 0 0 Connect Live Share AWS Git Graph Go Live

```
customermgmtcliapp  main  git remote add origin https://github.com/okalu/customermgmtcliapp.git
customermgmtcliapp  main  git push -u origin main
Enumerating objects: 21, done.
Counting objects: 100% (21/21), done.
Delta compression using up to 4 threads
Compressing objects: 100% (9/9), done.
Writing objects: 100% (21/21), 2.40 KiB | 245.00 KiB/s, done.
Total 21 (delta 0), reused 0 (delta 0), pack-reused 0
To https://github.com/okalu/customermgmtcliapp.git
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
customermgmtcliapp  main ≡
```

Demo: Github Action

- Code successfully pushed to the remote repository on github. And a README.md file was added/committed directly by me



https://github.com/okalu/customermgmtcliapp/tree/main

okalu / customermgmtcliapp

Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

customermgmtcliapp

okalu Create README.md — e14d383 · 1 minute ago — History

Name	Last commit message	Last commit date
src	Init new CustomerMgmtCLIApp	21 minutes ago
.gitignore	Init new CustomerMgmtCLIApp	21 minutes ago
README.md	Create README.md	1 minute ago
pom.xml	Init new CustomerMgmtCLIApp	21 minutes ago

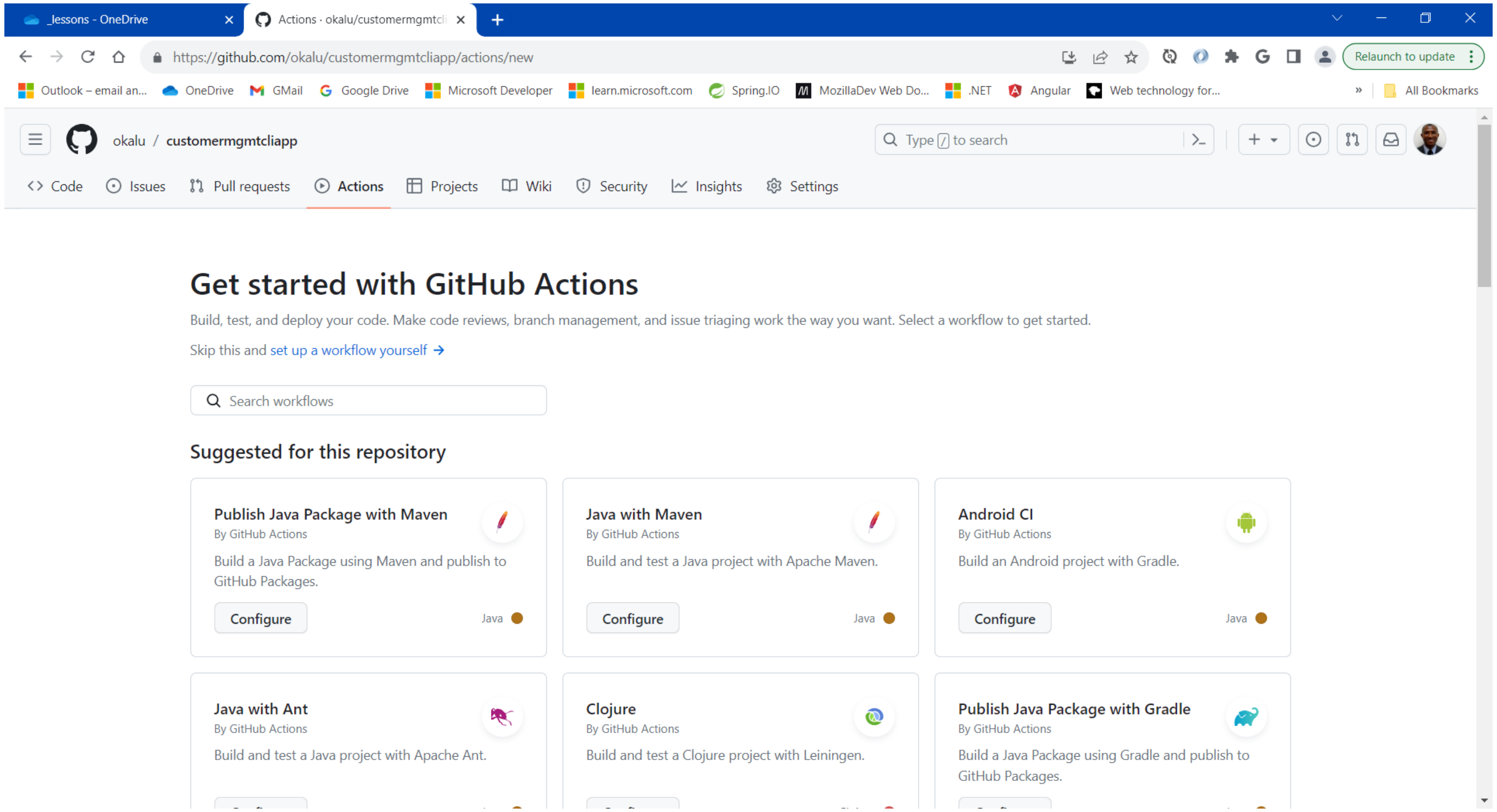
README.md

customermgmtcliapp

Demo Maven Java CLI App for Github Action CI/CD

Demo: Github Action

- Select/open the repository's 'Actions' tab on github



https://github.com/okalu/customermgmtcliapp/actions/new

okalu / customermgmtcliapp

Code | Issues | Pull requests | Actions | Projects | Wiki | Security | Insights | Settings

Get started with GitHub Actions

Build, test, and deploy your code. Make code reviews, branch management, and issue triaging work the way you want. Select a workflow to get started.

Skip this and set up a workflow yourself →

Search workflows

Suggested for this repository

Publish Java Package with Maven
By GitHub Actions
Build a Java Package using Maven and publish to GitHub Packages.
Configure — Java

Java with Maven
By GitHub Actions
Build and test a Java project with Apache Maven.
Configure — Java

Android CI
By GitHub Actions
Build an Android project with Gradle.
Configure — Java

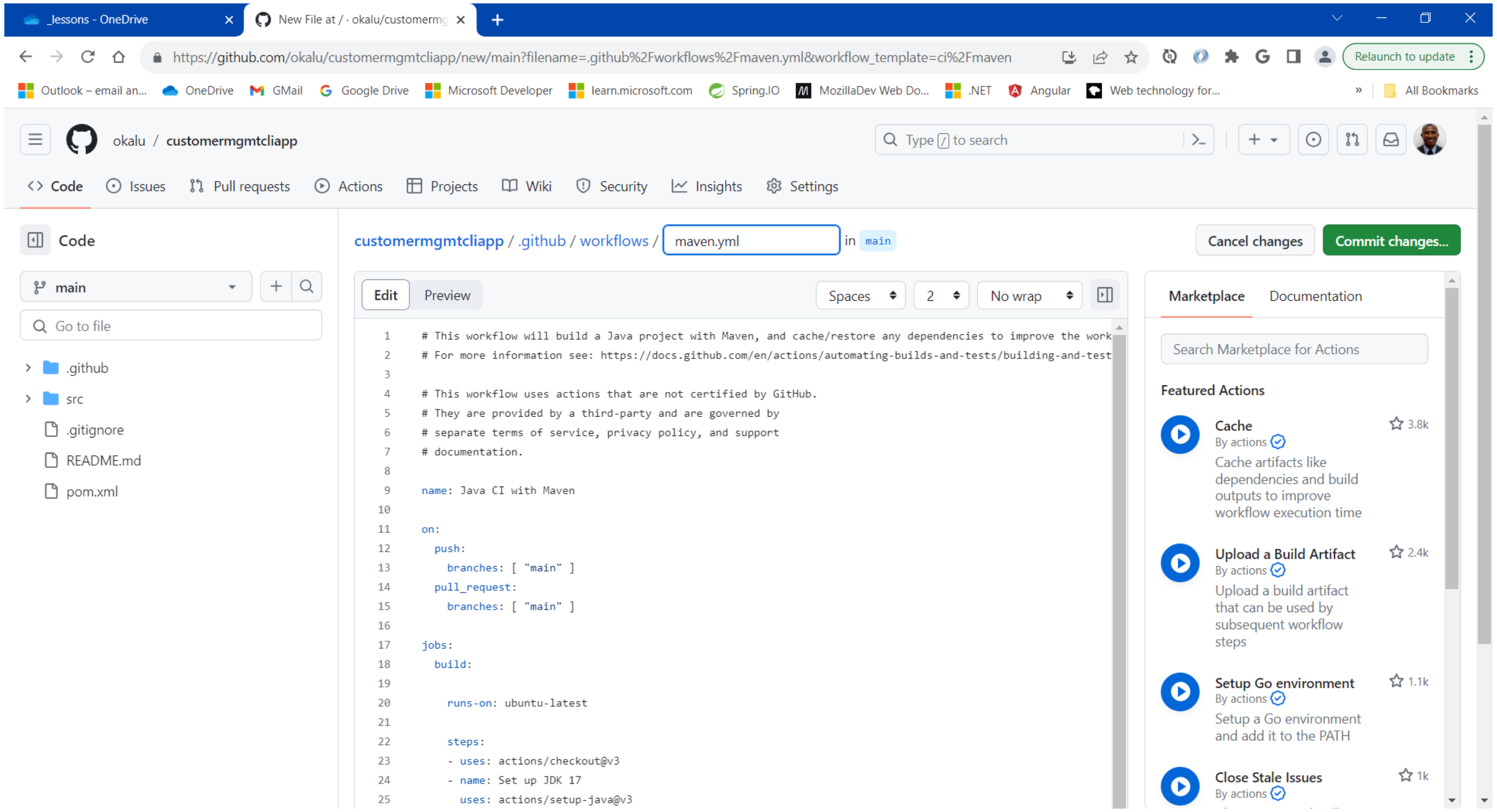
Java with Ant
By GitHub Actions
Build and test a Java project with Apache Ant.

Clojure
By GitHub Actions
Build and test a Clojure project with Leiningen.

Publish Java Package with Gradle
By GitHub Actions
Build a Java Package using Gradle and publish to GitHub Packages.

Demo: Github Action

- Select the preconfigured workflow template named: “Java with Maven”, which builds and test a Java project with Apache Maven. Then, click “Configure”



https://github.com/okalu/customermgmtcliapp/new/main?filename=.github%2Fworkflows%2Fmaven.yml&workflow_template=ci%2Fmaven

okalu / customermgmtcliapp

customermgmtcliapp / .github / workflows / maven.yml in main

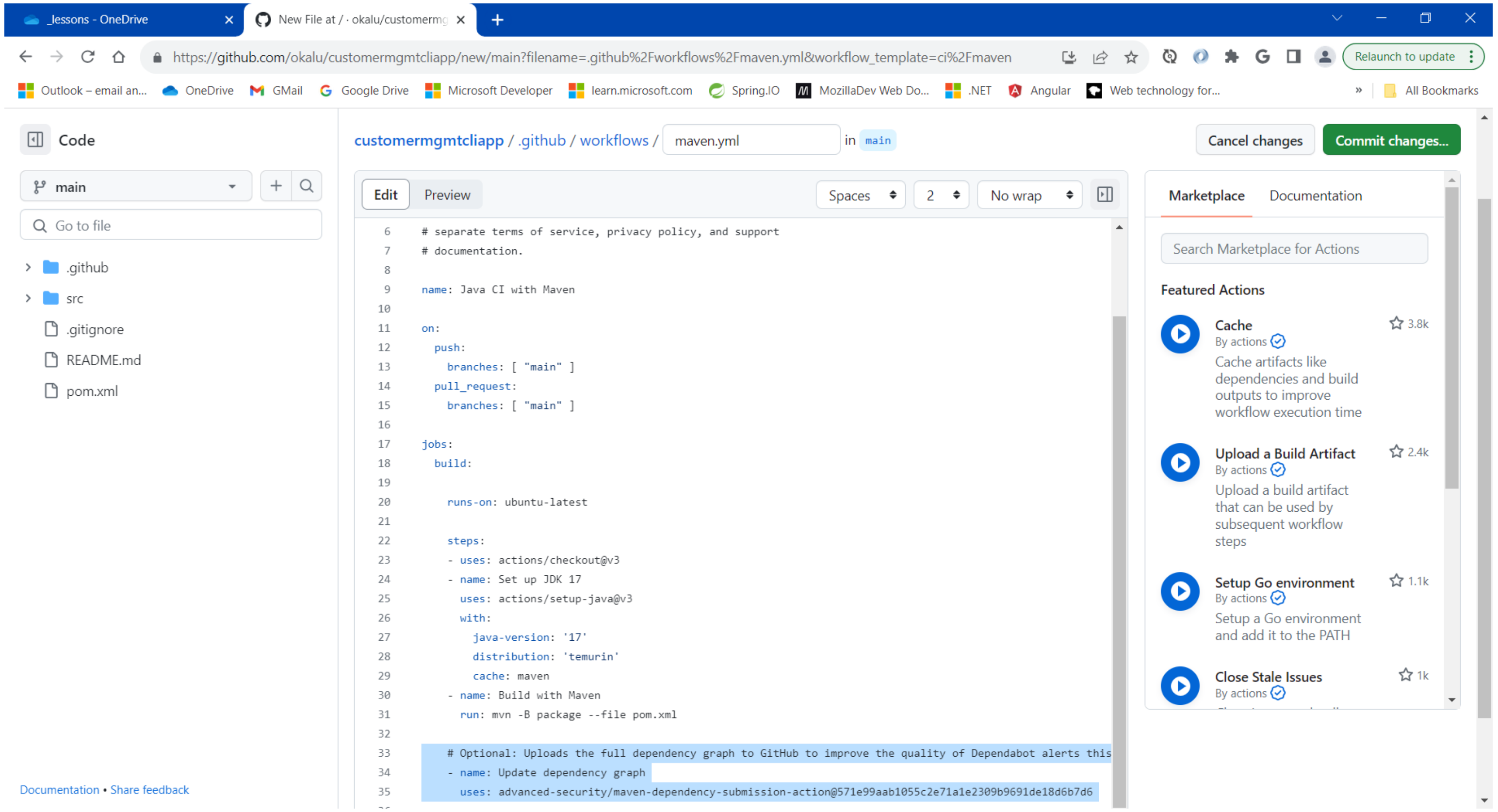
```
# This workflow will build a Java project with Maven, and cache/restore any dependencies to improve the work
# For more information see: https://docs.github.com/en/actions/automating-builds-and-tests/building-and-test

# This workflow uses actions that are not certified by GitHub.
# They are provided by a third-party and are governed by
# separate terms of service, privacy policy, and support
# documentation.

name: Java CI with Maven

on:
  push:
    branches: [ "main" ]
  pull_request:
    branches: [ "main" ]

jobs:
  build:

    runs-on: ubuntu-latest

    steps:
    - uses: actions/checkout@v3
    - name: Set up JDK 17
      uses: actions/setup-java@v3
```

Featured Actions

- Cache — By actions — Cache artifacts like dependencies and build outputs to improve workflow execution time — 3.8k
- Upload a Build Artifact — By actions — Upload a build artifact that can be used by subsequent workflow steps — 2.4k
- Setup Go environment — By actions — Setup a Go environment and add it to the PATH — 1.1k
- Close Stale Issues — By actions — 1k

Demo: Github Action

- Edit the workflow's configuration, as shown below
  - Delete Lines 33, 34 and 35
- Click on the "Commit changes…" button
- Next, execute git pull origin main to download the new github actions workflow config yaml file

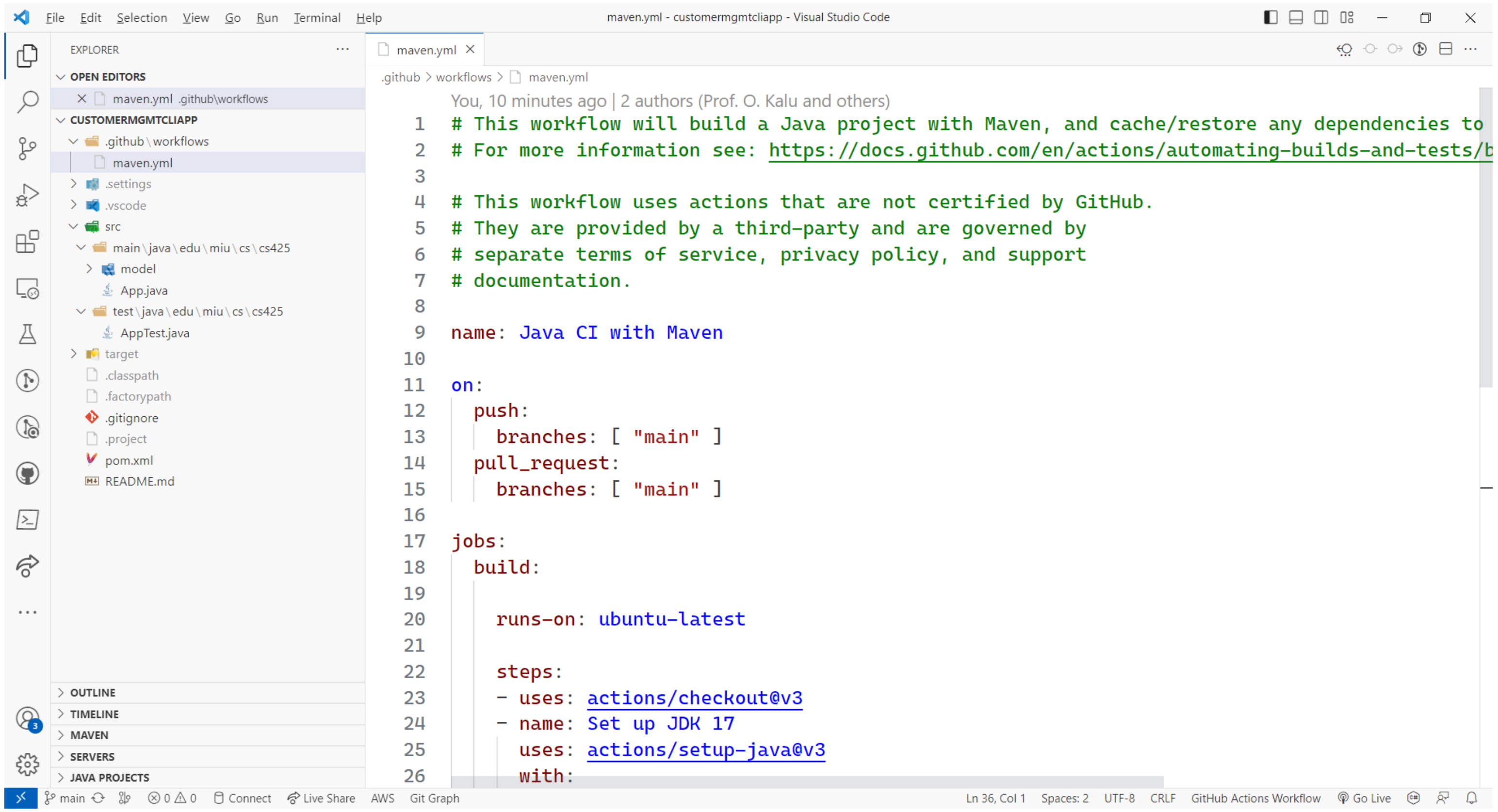


_lessons - OneDrive
New File at / · okalu/customermg
https://github.com/okalu/customermgmtcliapp/new/main?filename=.github%2Fworkflows%2Fmaven.yml&workflow_template=ci%2Fmaven
Relaunch to update
Outlook – email an... OneDrive GMail Google Drive Microsoft Developer learn.microsoft.com Spring.IO MozillaDev Web Do... .NET Angular Web technology for... All Bookmarks
Code
main
Go to file
.github
src
.gitignore
README.md
pom.xml
Documentation • Share feedback
customermgmtcliapp / .github / workflows / maven.yml in main
Cancel changes
Commit changes...
Edit
Preview
Spaces
2
No wrap
Marketplace
Documentation
Search Marketplace for Actions
Featured Actions
Cache
By actions
3.8k
Cache artifacts like dependencies and build outputs to improve workflow execution time
Upload a Build Artifact
By actions
2.4k
Upload a build artifact that can be used by subsequent workflow steps
Setup Go environment
By actions
1.1k
Setup a Go environment and add it to the PATH
Close Stale Issues
By actions
1k

```
6   # separate terms of service, privacy policy, and support
7   # documentation.
8
9   name: Java CI with Maven
10
11  on:
12    push:
13      branches: [ "main" ]
14    pull_request:
15      branches: [ "main" ]
16
17  jobs:
18    build:
19
20      runs-on: ubuntu-latest
21
22      steps:
23      - uses: actions/checkout@v3
24      - name: Set up JDK 17
25        uses: actions/setup-java@v3
26        with:
27          java-version: '17'
28          distribution: 'temurin'
29          cache: maven
30      - name: Build with Maven
31        run: mvn -B package --file pom.xml
32
33      # Optional: Uploads the full dependency graph to GitHub to improve the quality of Dependabot alerts this
34      - name: Update dependency graph
35        uses: advanced-security/maven-dependency-submission-action@571e99aab1055c2e71a1e2309b9691de18d6b7d6
```

Demo: Github Action

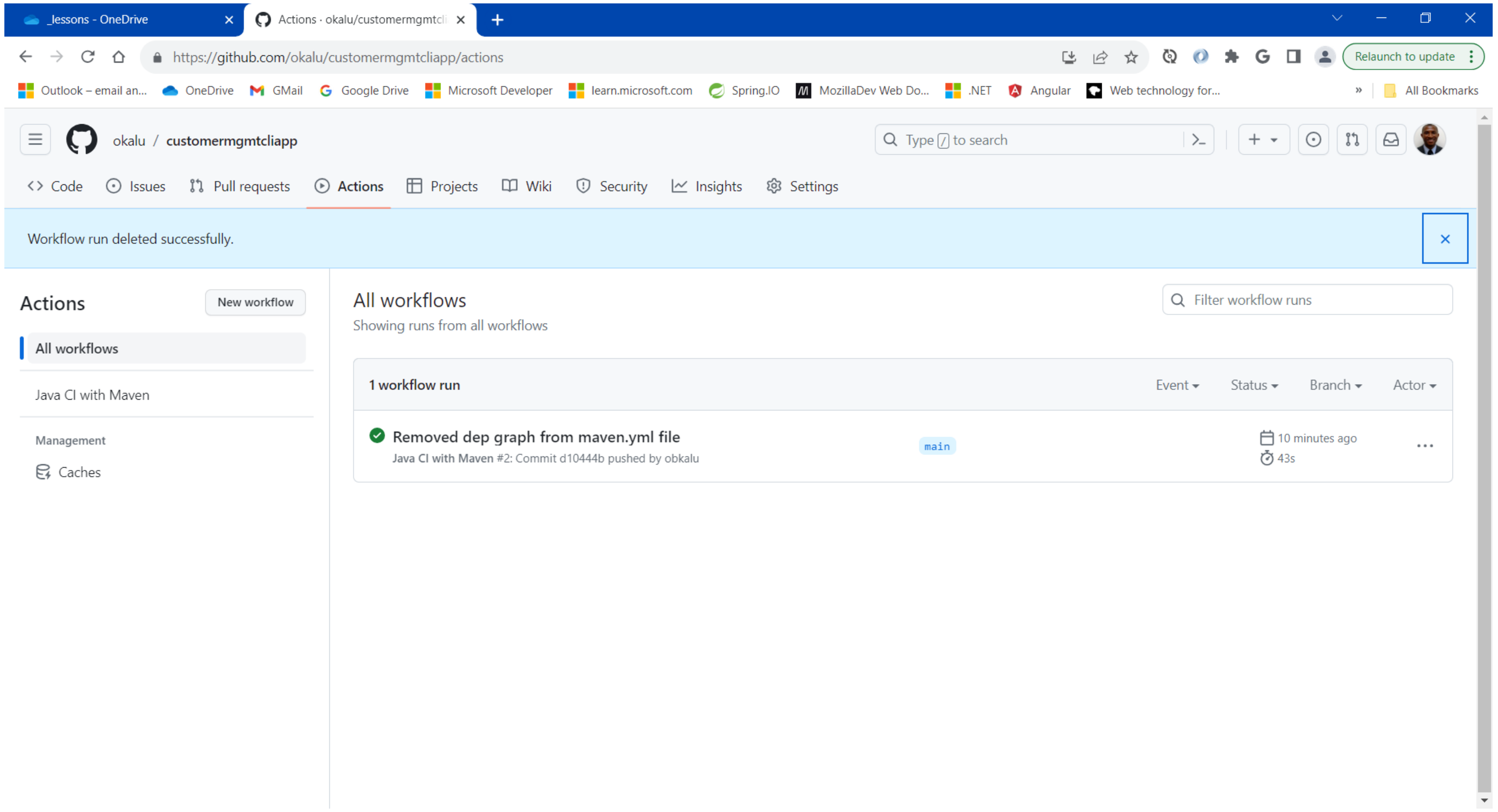
- Next, execute 'git pull origin main' to download the new github actions workflow config yaml file



```yaml
# This workflow will build a Java project with Maven, and cache/restore any dependencies to
# For more information see: https://docs.github.com/en/actions/automating-builds-and-tests/b

# This workflow uses actions that are not certified by GitHub.
# They are provided by a third-party and are governed by
# separate terms of service, privacy policy, and support
# documentation.

name: Java CI with Maven

on:
  push:
    branches: [ "main" ]
  pull_request:
    branches: [ "main" ]

jobs:
  build:

    runs-on: ubuntu-latest

    steps:
    - uses: actions/checkout@v3
    - name: Set up JDK 17
      uses: actions/setup-java@v3
      with:
```

Demo: Github Action

- Check the Github Actions tab and see that the workflow executes successfully, having been triggered by the commit/push made.



Workflow run deleted successfully.

Actions — New workflow

- All workflows
- Java CI with Maven

Management

- Caches

All workflows

Showing runs from all workflows

Filter workflow runs

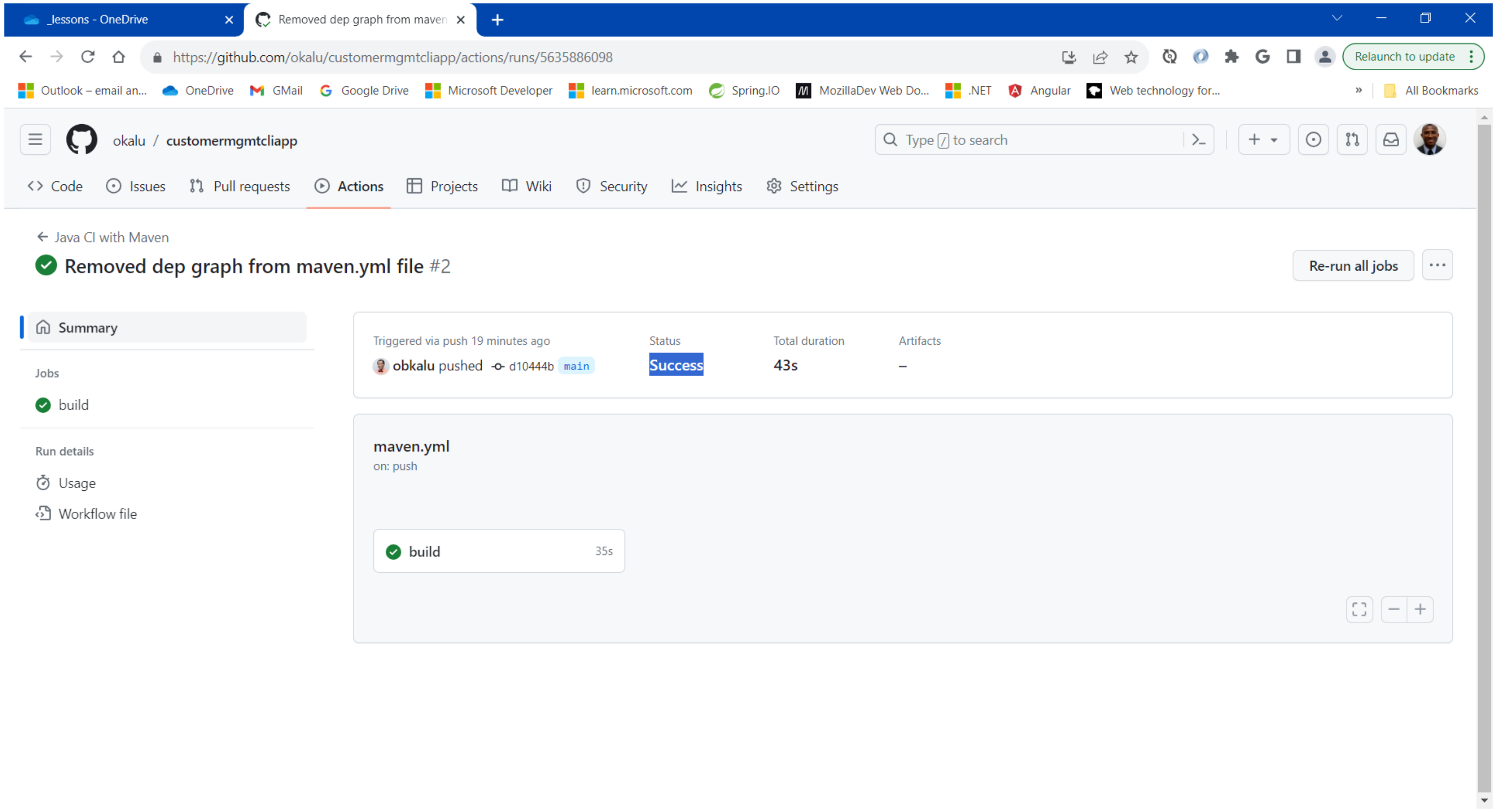
1 workflow run — Event ▾ Status ▾ Branch ▾ Actor ▾

✓ Removed dep graph from maven.yml file — main — 10 minutes ago — 43s

Java CI with Maven #2: Commit d10444b pushed by obkalu

Demo: Github Action

- Check the Github Actions tab and see that the workflow executes successfully, having been triggered by the commit/push made.



https://github.com/okalu/customermgmtcliapp/actions/runs/5635886098

okalu / customermgmtcliapp

Java CI with Maven

Removed dep graph from maven.yml file #2

Triggered via push 19 minutes ago
obkalu pushed d10444b main

Status
Success

Total duration
43s

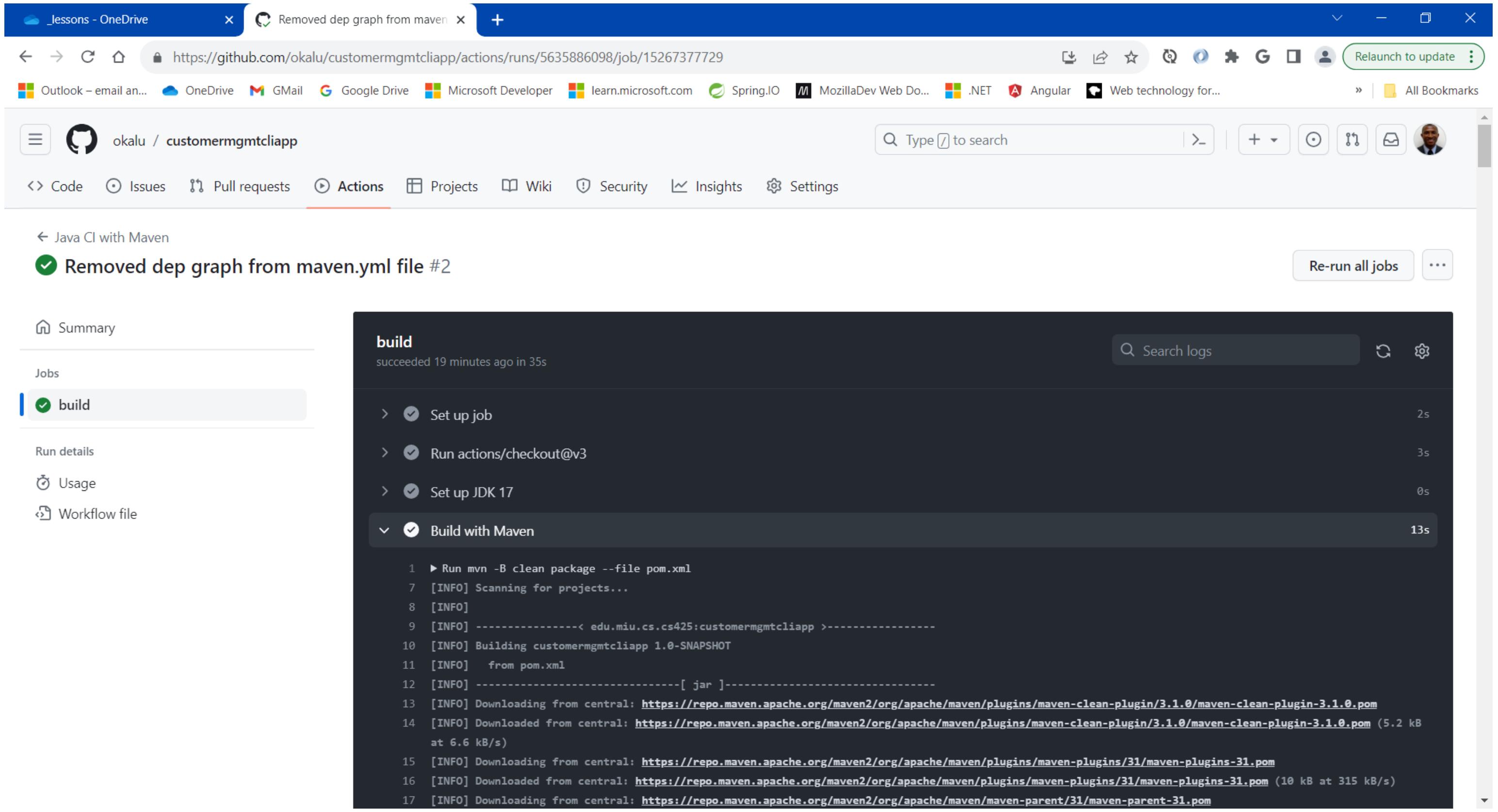
Artifacts
–

maven.yml
on: push

build 35s

Demo: Github Action

- Check the Github Actions tab and see that the workflow executes successfully, having been triggered by the commit/push made.
- View the execution log



Removed dep graph from maven.yml file #2

build
succeeded 19 minutes ago in 35s

- Set up job 2s
- Run actions/checkout@v3 3s
- Set up JDK 17 0s
- Build with Maven 13s

```
▶ Run mvn -B clean package --file pom.xml
[INFO] Scanning for projects...
[INFO]
[INFO] ---------------< edu.miu.cs.cs425:customermgmtcliapp >----------------
[INFO] Building customermgmtcliapp 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] --------------------------------[ jar ]---------------------------------
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/3.1.0/maven-clean-plugin-3.1.0.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-clean-plugin/3.1.0/maven-clean-plugin-3.1.0.pom (5.2 kB
at 6.6 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/31/maven-plugins-31.pom
[INFO] Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-plugins/31/maven-plugins-31.pom (10 kB at 315 kB/s)
[INFO] Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-parent/31/maven-parent-31.pom
```

CS489:
Applied Software Development