

## DevSecOps with Dr. Chuck Easttom

### Labs

#### Contents

General Instructions .....	4
General Labs (regardless of code repository) .....	5
Lab 1 – Jenkins Initial Setup (Recipe Lab).....	5
Step 1: Navigate to URL.....	5
Step 2: Download the Windows MSI.....	6
Step 3:Install Jenkins .....	7
Step 4:Navigate to http://localhost:8080/.....	7
Step 5: Install Plugins .....	8
Lab 2 – Create a Pipeline in Jenkins(Recipe Lab).....	10
Step 1 Create a New Item .....	10
Step 2: Type in Script.....	11
Step 3: Build Now.....	12
Step 4: View Stages .....	12
Lab 3 - Get WebGoat .....	13
Step 1: Navigate to URL.....	13

Lab 4 - Eclipse Initial Setup (Recipe Lab).....	16
<b>Step 1:</b> Navigate to .....	16
Step 2:Download Eclipse.....	16
Step 3: Unzip and Install.....	17
Lab 5 –Visual Studio Community Initial Setup (Recipe Lab).....	19
Step 1: Navigate to URL and download .....	19
Step 2:Install .....	20
Step 3:Launch .....	21
Lab 6 VS Code Initial Setup .....	22
Lab7– Jira Initial Setup (Recipe Lab).....	22
Step 1: Navigate to URL.....	22
Lab 8: Synch Jenkins with Jira (Recipe Lab) .....	22
Step 1: Navigate to URL and login.....	22
Step 2: Go to New Apps.....	22
Step 3: Navigate to Jenkins.....	24
Step 4: Install Jira Integration.....	25
Step 5: View Jira .....	25
Lab 9 Building an App in Jenkins (experiment lab) .....	28
Lab 10: Integrate Visual Studio Code with Jira (Recipe Lab) .....	29
Step 1: Get Jira in VS Code .....	29
Lab 11: Test WebGoat in VS Code .....	30
Step 1: Open Web Goat.....	30
Step 2: Run Basic Tests.....	31
Lab 12: OWASP ZAP .....	32
Lab 13: Testing in Visual Studio.....	32
Bit Bucket Labs.....	32
Lab 14 – BitBucket Initial Setup (Recipe Lab) .....	32
<b>Step 1:</b> Navigate to URL .....	33
<b>Step 2:</b> Get the Free version .....	33
Step 3:Create an account.....	33
Step 4: Create a workspace.....	34

Step 5:Create a repository .....	35
Lab 15: Synch Jenkins and BitBucket (Recipe Lab) .....	37
Step 1: Go to Plugins .....	37
Step 2: Visit Jenkins - System.....	37
Lab 16 – Setup Project in Bit Bucket and integrate with Eclipse (Recipe Lab).....	38
<b>Step 1:</b> Navigate to URL .....	38
Step 2: Create a Repository.....	38
Step 3: View Project.....	40
Step 4: Open Eclipse.....	41
Step 5: Share Project.....	43
Step 6: Configure Git repository.....	44
Step 7: Commit .....	45
Step 8: Push.....	48
Step 9: View code in BitBucket .....	50
Step 10: Return to Eclipse.....	54
Lab 17: Integrate Bitbucket with VS Code.....	60
Step 1: Get Bitbucket in VS Code .....	60
Step 2: Connect to Bitbucket .....	61
GitHub Labs.....	62
Lab 18 – GitHub Initial Setup (Recipe Lab).....	62
Step 1:Navigate to URL.....	62
Lab 19: Integrate Visual Studio Code with Github (Recipe Lab).....	63
Step 1 Setup Project.....	63
Step 2: Install the GitHub extensions.....	63
Step 3 Connect to Github.....	65
Lab 20: Viewing Activities in GitHub .....	69
Step 1: Login into a GitHub repository.....	69
Step 2: Check Status .....	69
Step 3: Setup Security.....	70
Lab 21: GitHub Collaboration .....	73
Step 1: Invite Collaborator.....	73

Step 2: Accept the invitation .....	75
GitLab Labs .....	76
Lab 22 GitLab New project .....	76
Step 1: Start New GitLab project.....	76
Step 2: Use Sample Template .....	77
Step 3: Review results.....	78
Lab 23: Create a Pipeline in GitLab .....	79
Step 1: Setup YAML file.....	79
Step 2: Setup Pipeline in YAML.....	80
Step 3: See your pipeline running.....	81
Lab 24: GitLab Secure Scanning .....	82
Lab 25 Experiment With GitLab.....	82
Step 1 Setup Project in GitLab.....	82
Step 2: Setup CI/CD.....	83
Step 3: Security Configuration.....	84
Online Labs (No VM needed).....	85
Lab 26: HackMe Pipeline Lab .....	85
Lab 27: Hackme DevSecOps Setup (Recipe Lab).....	86
Lab 28: Hackme DevSecOps Setup (Recipe Lab).....	87
Lab 29: Git(Recipe Lab) .....	87
Lab 30: Docker (Recipe Lab) .....	89
Lab 31: Kubernetes (Recipe Lab).....	90
Lab 32: Another Kubernetes Lab (recipe/experiment) .....	91
Lab 33: HackMe Kubernetes .....	93
Lab 34: HackMe Intro to IaC.....	93
Lab 35:Terraform Online Lab (recipe/experiment) .....	94
Lab 36: Online OWAS ZAP .....	95

## **General Instructions**

Most of the labs are what I call ‘recipe labs’. This means you simply follow the specific steps in order and it will work as shown. There is nothing you need to figure out. Then there are some labs that have most of it spelled out for you, with something you have to figure out. These are meant to help you reach the next level in knowledge.

**NOTE:** SHUT DOWN everything you can. This will maximize your performance. If it is not related to this class, turn it off.

## **General Labs (regardless of code repository)**

### **Lab 1 – Jenkins Initial Setup (Recipe Lab)**

Jenkins should already be installed on the VM. There should be a password file on that VM. It will be a text file on the desktop. However, if you do need to download Jenkins, the steps are given here

Estimated time 20 minutes

*Step 1: Navigate to URL*

<https://www.jenkins.io/>

The click download and select



Generic Java package (.war)

SHA-256: 90ccf556133c36fdf7653ad710f00d248bf2895f9fbc26ccee0e2d3ba681b01f



Docker



Kubernetes



Ubuntu/Debian



Red Hat Enterprise Linux and derivatives



Fedora



Windows



openSUSE



FreeBSD

Third party

*Step 2: Download the Windows MSI*

For a Windows machine you can use .war or Windows, I used the Windows.msi

## Welcome to the Jenkins 2.492.3 Setup Wizard

The Setup Wizard will install Jenkins 2.492.3 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.



Back

Next

Cancel

*Step 3: Install Jenkins*

Use default settings

*Step 4: Navigate to <http://localhost:8080/>*

Then navigate to

<http://localhost:8080/> you should see

# Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log ([not sure where to find it?](#)) and this file on the server:

```
C:\ProgramData\Jenkins\.jenkins\secrets\initialAdminPassword
```

Please copy the password from either location and paste it below.

**Administrator password**

After you enter that password you should see the following screen:

Getting Started

×

## Customize Jenkins

Plugins extend Jenkins with additional features to support many different needs.

### Install suggested plugins

Install plugins the Jenkins community finds most useful.

### Select plugins to install

Select and install plugins most suitable for your needs.

#### *Step 5: Install Plugins*

Select 'Select plugins to install' leave all the default but add Dashboard view, MSBuild, Junit, bitbucket, and CVS (git should already be checked)

At the end of installing those plugins you will be asked to create admin username and password

# Create First Admin User

Username

Password

Confirm password

Full name

---

Jenkins 2.492.3

[Skip and continue as admin](#)

[Save and Continue](#)

For this lab make it Adminperson and LetMeIn

Now you should see the following

- + New Item
  - 📅 Build History
  - ⚙️ Manage Jenkins
  - 📄 My Views
- Build Queue ▼

No builds in the queue.
- Build Executor Status 0/2 ▼

[Add description](#)

## Welcome to Jenkins!

This page is where your Jenkins jobs will be displayed. To get started, you can set up distributed builds or start building a software project.

### Start building your software project

[Create a job](#) +

### Set up a distributed build

- [Set up an agent](#) 🖥️
- [Configure a cloud](#) ☁️
- [Learn more about distributed builds](#) ?

localhost:8080/newJob

You are ready to start using Jenkins!!

## Lab 2 – Create a Pipeline in Jenkins(Recipe Lab)

### *Step 1 Create a New Item*


Create a new item




# Jenkins

Dashboard >

+ New Item

 Build History

 Manage Jenkins

 My Views

Enter an item name

firstpipinetest

Select an item type



Freestyle project

Classic, general-purpose job type that checks out from up to one SCM, executes build steps serially, followed by post-build steps like archiving artifacts and sending email notifications.



Pipeline

Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing complex activities that do not easily fit in free-style job type.



Multi-configuration project

Suitable for projects that need a large number of different configurations, such as testing on multiple environments, platform-specific builds, etc.

OK

*Step 2: Type in Script*









Type in the following script

```
node {  
    stage ('Build')
```

```
{
  echo "Build phase done..."
}
stage ('Test')
{
  echo "Test phase done..."
}
stage ('Deploy')
{
  echo "Deploy phase done..."
}
}
```

Click Save

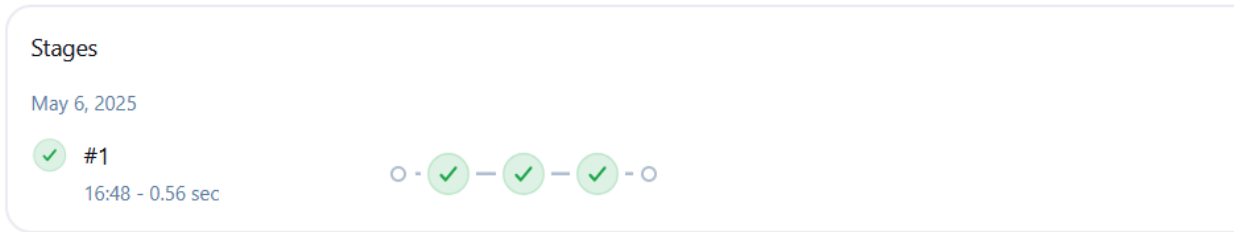
*Step 3: Build Now*

-  Status
-  Changes
-  Build Now
-  Configure
-  Delete Pipeline
-  Stages
-  Rename
-  Pipeline Syntax

*Step 4: View Stages*

Then click on stages and you will see everything went well:

## Stages



If you wish the following link will show you additional things you can do in a basic pipeline  
<https://www.jenkins.io/doc/pipeline/tour/hello-world/>

### Lab 3 - Get WebGoat

Estimated time 10 minutes

This may already be downloaded on the VM. If not, then the instructions on where and how to download it are given in this lab.

*Step 1: Navigate to URL*

<https://owasp.org/www-project-webgoat/>



release **v2025.3**

## Learn the hack - Stop the attack

WebGoat is a deliberately insecure application that allows interested developers just like you to test vulnerabilities commonly found in Java-based applications that use common and popular open source components.

## Description

Web application security is difficult to learn and practice. Not many people have full blown web applications like online book stores or online banks that can be used to scan for vulnerabilities. In addition, security professionals frequently need to test tools against a platform known to be vulnerable to ensure that they perform as

**The OWASP® Foundation** works to improve the security of software through its community-led open source software projects, hundreds of chapters worldwide, tens of thousands of members, and by hosting local and global conferences.

### Downloads

[Standalone jars](#)

[Docker image](#)

### Code Repository

[Source code](#)

Mar 11

github-actions

v2025.3

C5e045a

Compare

v2025.3 Latest

## Version 2025.3

### Bug fixes

- Changed URLs imply other exclusion filters for ZAP ([#2052](#))
- XSS lesson stage 12 (2 issues) ([#1178](#))

**nbaars** chore: new release 2025.3

.github	chore: bump dock
.mvn/wrapper	Update to latest M
config	chore: format all c
docs	chore: add pre-co
src	fix: rewrite questio
.dockerignore	feat: add Docker c
.editorconfig	Refactoring (#1201) 3 years ago
.gitignore	feat: Introduce Playwright for UI testing 4 months ago

Local Codespaces

**Clone**

HTTPS SSH GitHub CLI

Clone using the web URL.

Open with GitHub Desktop

Download ZIP

**Opening WebGoat-2025.3.zip**

You have chosen to open:

**WebGoat-2025.3.zip**

which is: Compressed (zipped) Folder  
from: [codeload.github.com](https://github.com)

**What should Firefox do with this file?**

Open with

Save File

Do this automatically for files like this from now on.

You will be using WebGoat as your sample project for many labs.

#### Lab 4 - Eclipse Initial Setup (Recipe Lab)

**Note: This may already be on your VM However, if not, the steps for downloading and installing are given in this lab.**

*Step 1: Navigate to*

Navigate to <https://eclipseide.org/>



*Step 2:Download Eclipse*

Click on the 'Download 2025-03'

Then select Eclipse IDE for Enterprise Java and Web Developers

# Eclipse IDE 2025-03 R Packages

## Eclipse IDE for Enterprise Java and Web Developers

545 MB 211,022 DOWNLOADS



Tools for developers working with Java and Web applications, including a Java IDE, tools for JavaScript, TypeScript, JavaServer Pages and Faces, Yaml, Markdown, Web Services, JPA and Data Tools, Maven and Gradle, Git, and more.



Windows | [x86\\_64](#) | [AArch64](#)  
macOS [x86\\_64](#) | [AArch64](#)  
Linux [x86\\_64](#) | [AArch64](#) | [riscv64](#)

Click [here](#) to raise an issue with the Eclipse Web Tools Platform. Maintainers will move opened issues to the right place.

Click [here](#) to raise an issue with the Eclipse Platform.

Click [here](#) to raise an issue with Maven integration for web projects.

Click [here](#) to raise an issue with Eclipse Wild Web Developer (incubating).

Home ▶ Downloads ▶ Eclipse downloads - Select a mirror

All downloads are provided under the terms and conditions of the [Eclipse Foundation Software User Agreement](#) unless otherwise specified.

 Download

Download from: United States - XMission Internet (https)

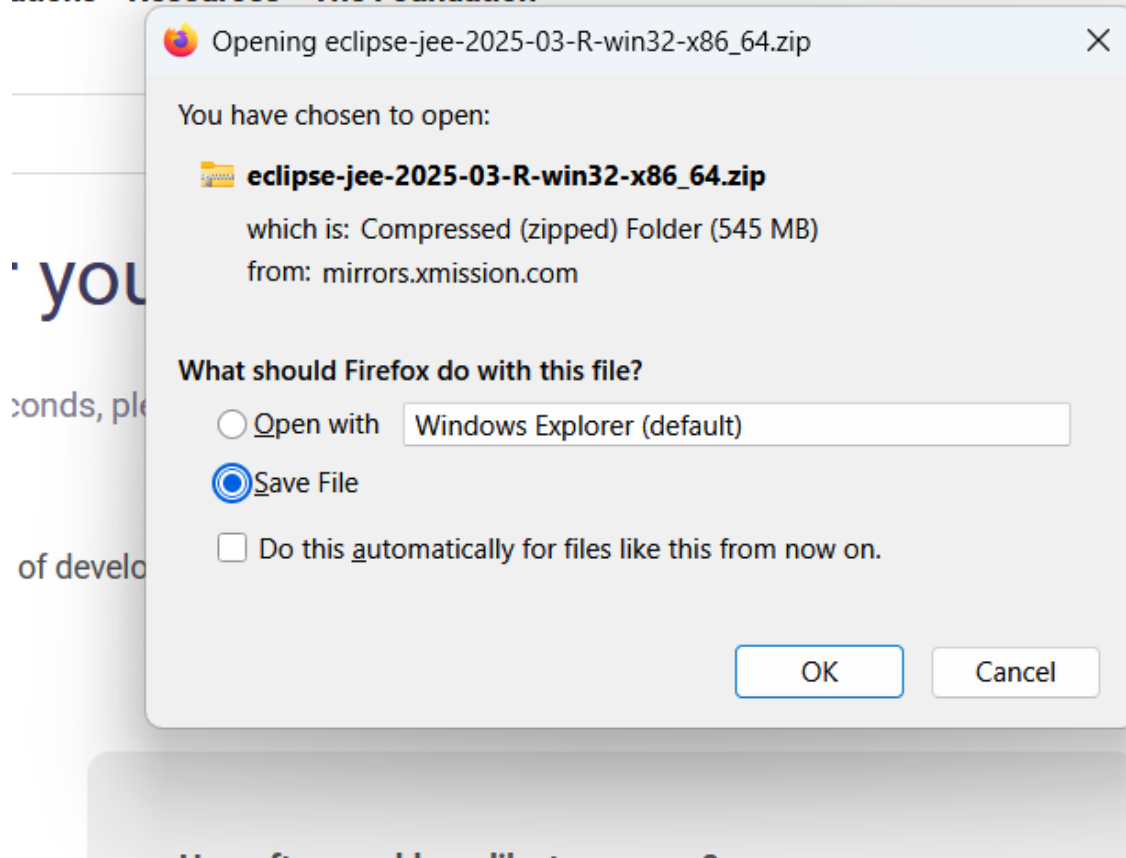
File: [eclipse-jee-2025-03-R-win32-x86\\_64.zip](#) SHA-512

[>> Select Another Mirror](#)

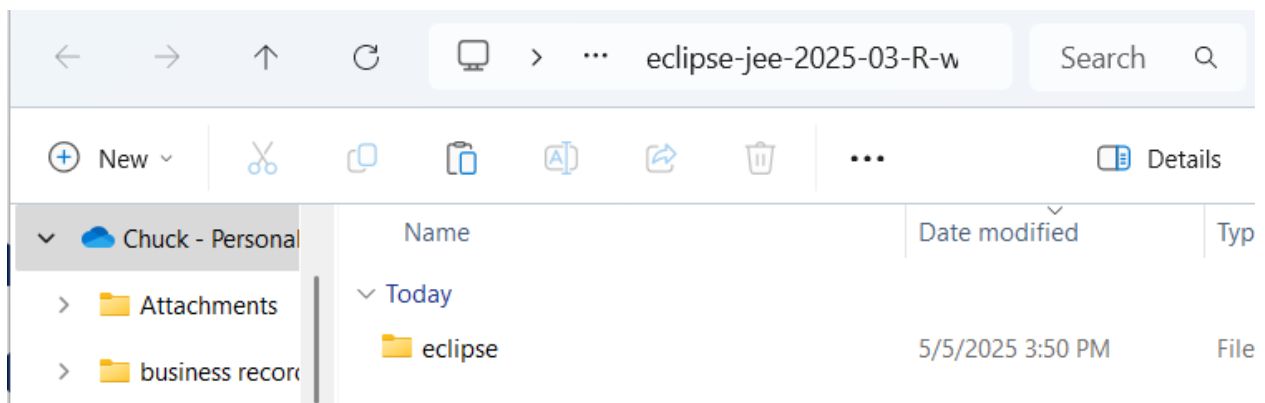
### *Step 3: Unzip and Install*

After it is downloaded unzip and begin installing.

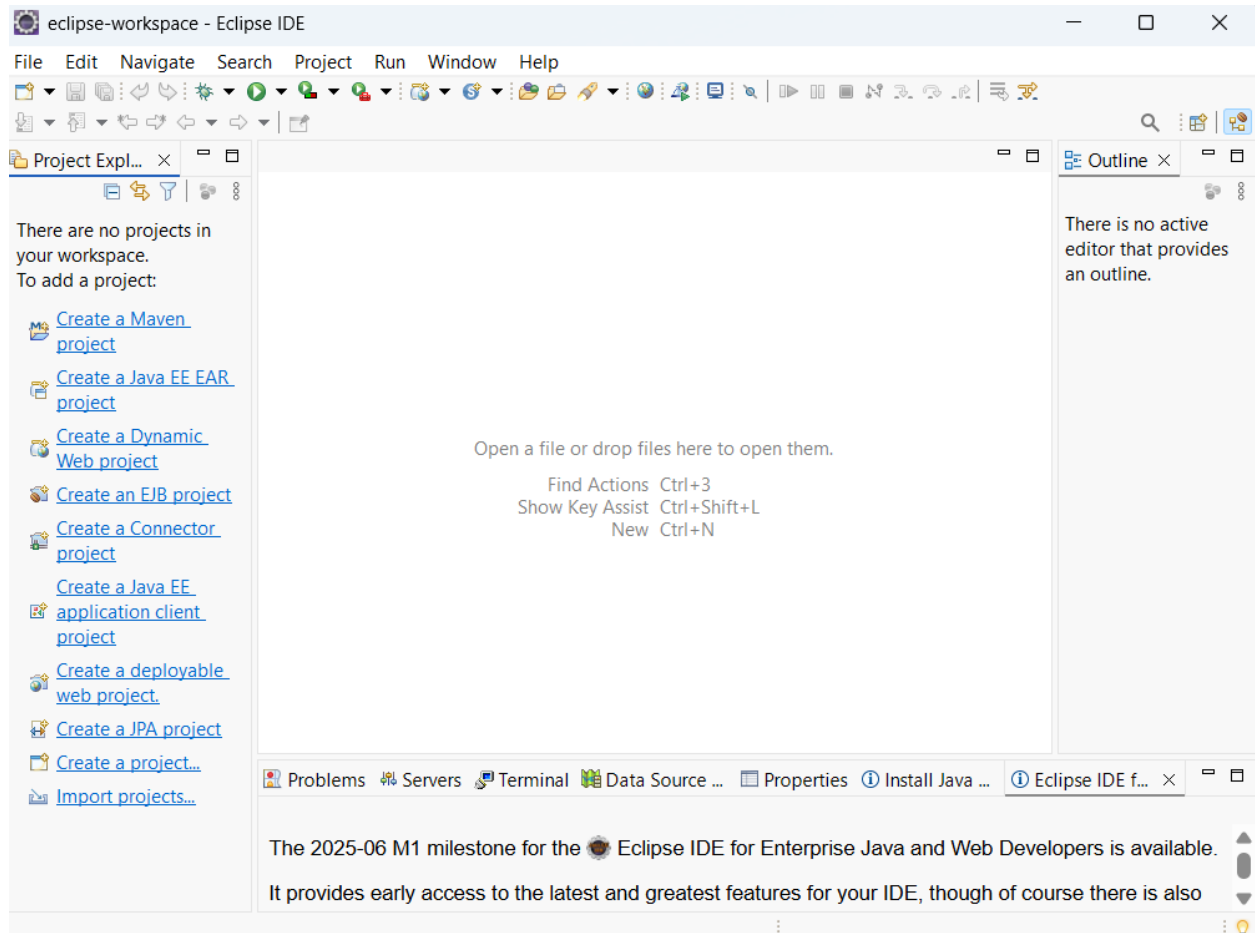
ations Resources The Foundation



When you unzip you will see this:



No open the eclipse folder and double click eclipse.exe



## Lab 5 –Visual Studio Community Initial Setup (Recipe Lab)

**Note: This may already be on your VM, however, if not the steps for downloading and installing are given.**

*Step 1: Navigate to URL and download*

Navigate to <https://visualstudio.microsoft.com/vs/community/>

And click on download

# Visual Studio Community

A fully-featured, extensible, free IDE for creating modern applications for Android, iOS, Windows, as well as web applications and cloud services.

Download

*Step 2: Install*

Begin installing

×

## Visual Studio Installer

Before you get started, we need to set up a few things so that you can configure your installation.

To learn more about privacy, see the [Microsoft Privacy Statement](#).

By continuing, you agree to the [Microsoft Software License Terms](#).

Continue

It should not take long, then go to step 3.

### Step 3: Launch

Launch to make sure it installed

Visual Studio Installer

Installed Available

All installations are up to date.

Visual Studio Community 2022  
17.13.6  
Powerful IDE, free for students, open-source contributors, and individuals  
[Release notes](#)

Modify  
Launch  
More ▾

If you see the next screen you are good to go and can exit for now.

Visual Studio 2022

Open recent

Get started

Clone a repository  
Get code from an online repository like GitHub or Azure DevOps

Open a project or solution  
Open a local Visual Studio project or .sln file

Open a local folder  
Navigate and edit code within any folder

Create a new project  
Choose a project template with code scaffolding to get started

Continue without code →

Older

Project Name	Path	Time
QUERYAPP.sln	C:\court cases\USAA class action\QUERYAPP	11/6/2024 11:07 AM
newinsights-ascend-backend-cbc5eb983b6b	C:\court cases\Rhode Case\newinsights-ascend-backend-production2-14-2024	7/22/2024 11:20 AM
newinsights-di-backend-21cdbf4559de	C:\court cases\Rhode Case\DIBackend3-13	7/22/2024 11:19 AM
javascriptTutorial	C:\Users\chuck\Downloads\javascriptTutorial	6/13/2024 7:08 AM
97503-275971 - Philip Eykamp - Jun 5, 2024 1212 AM	C:\...\Secure Programming Philip Eykamp - Submissions Jun 13, 2024 658 AM	6/13/2024 7:00 AM
CS6387-assignment2-Aaron Vaneps	C:\Users\chuck\Downloads	6/4/2024 9:44 AM
assignment2	C:\Users\chuck\Downloads	6/4/2024 9:41 AM

## Lab 6 VS Code Initial Setup

## Lab7– Jira Initial Setup (Recipe Lab)

Estimated time: 10 minutes

*Step 1: Navigate to URL*

<https://www.atlassian.com/software/jira>


Sign up for the free account.

The [new navigation](#) is now available and will be automatically turned on in the coming weeks. Turn it on for your team now or start with just yourself. [Go to settings](#)

Jira Your work ▾ Projects ▾ Filters ▾ Dashboards ▾ Teams ▾ Plans ▾ Apps ▾ Create Premium trial

Projects Create project Templates

Search Projects 🔍 Filter by product ▾



No projects were found that match your search

Try using specific project names or terms.

## Lab 8: Synch Jenkins with Jira (Recipe Lab)

*Step 1: Navigate to URL and login*

Login to Atlassian [.atlassian.com](https://www.atlassian.com)

*Step 2: Go to New Apps*

On the top menu bar select Apps > Find new apps:

Apps ▾

Create

### Your apps

### Recommended for your team

Ship faster with marketplace apps that integrate your team's tools with Jira.

 Power BI (Analytics)

 eazyBI (Analytics)

 Slack

Explore more apps

Manage your apps

View app requests

Promotions

- Apps
- Atlassian Marketplace
- Find new apps
- Manage apps
- App requests
- Promotions
- OAuth credentials
- Apps
  - Jenkins for Jira

Jira / Marketplace apps

## Explore apps for Jira

Search: jenkins

Suggestions for 'jenkins'

- jenkins for jira
- jenkins integration for jira
- jenkins integration
- jenkins for jira official

by Adaptavist

The must-have app for Jira admins of every technical background: the ultimate toolkit for unlimited automation and customisation

3.7/4 ★★★★★ (805) 38.1k

CLOUD FORTIFIED

by Xtena

Native Test Management. Built for every member of your team to plan, test, track and release great software

3.5/4 ★★★★★ (517) 32k

CLOUD FORTIFIED

It will install in just a few minutes

### Jenkins for Jira

Share page

To receive build and deployment data, your teams must follow the unique setup guide for each connected server.



jirajenkinslab **PENDING**


[Recent events \(0\)](#) [Set up guide](#)


Now we have to go to Jenkins to finish


*Step 3: Navigate to Jenkins*

Login to Jenkins <http://localhost:8080>. Now go to manage Jenkins

+ New Item

 Build History

 Manage Jenkins

 My Views

And choose plugins:

## Manage Jenkins

Q Search settings

Upgrade to Jenkins 2.504.2 is complete, awaiting [restart](#).

Restore the previous version of Jenkins

Downgrade to 2.504.1

### System Configuration



**System**  
Configure global settings and paths.



**Tools**  
Configure tools, their locations and automatic installers.



**24 Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.

### *Step 4: Install Jira Integration*

Search for *jira integration* and check the 'Install' checkbox

Click 'Download now and install after restart'

Check 'Restart Jenkins when installation is complete and no jobs are running' to restart Jenkins.

### *Step 5: View Jira*

Now we are back to Jira. Click the Gear and find apps

[Add payment details](#)



### Personal settings



#### Atlassian account settings

Manage your language, time zone, and other profile information.



#### Personal Jira settings

Manage your email notifications and other Jira settings.

### Jira settings

**Tip:** Try `ctrl k` to search Jira settings



#### System

Manage your general configuration, global permissions, look and feel and more.



#### Products

Manage your Jira products' settings and integrations.



#### Projects

Manage your project settings, categories, and more.



#### Work items

Configure your work types, workflows, screens, custom fields and more.



#### Apps

Add and manage Jira Marketplace apps.

### Atlassian admin

Select Jenkins Integration


# Explore apps for Jira

Q Jenkins integration

Suggestions for 'Jenkins integration'

- Q jenkins integration
- Q jenkins integration for jira

You will actually see multiple apps for Jenkins, choose this one



## Connector for Jenkins and Jira

by ILA eSolution


**Integration** Gets and displays **Jenkins** CICD Pipeline Builds and Statuses on a Jira Issue panel

4/4 ★ ★ ★ ★ (1) ↓ 40

You then select your site

## Select a site

### Select a site

 chuckeasttom (https://chuckeasttom.atlassian.net) ▼

App will be installed on the following product(s):

 **Jira** E-43G-SQY-7UV-2VW

Cancel

Review

The plugin will then be installed:

Jira / Marketplace apps / Connector for Jenkins and Jira Feedback

 **Connector for Jenkins and Jira**  
by ILA eSolution Free

OVERALL RATINGS INSTALLS

4/4 ★★★★★ (1) 📄 40

[Overview](#) [Privacy & Security](#) [Support](#)

---

**Integration Gets and displays Jenkins CICD Pipeline Builds and Statuses on a Jira Issue panel**

After a minute or two you will be notified that it is installed and you should select 'Get Started'

## Lab 9 Building an App in Jenkins (experiment lab)

This one will vary from user to user

<https://www.jenkins.io/doc/tutorials/>

You pick one of the following based on your programming skills and interest:

## Using build tools

The following tutorials show how to use Jenkins to cover the basics of CI/CD concepts based on specific technology stacks.

Choose the tutorial that's relevant to your technology stack or one that you're most familiar with:

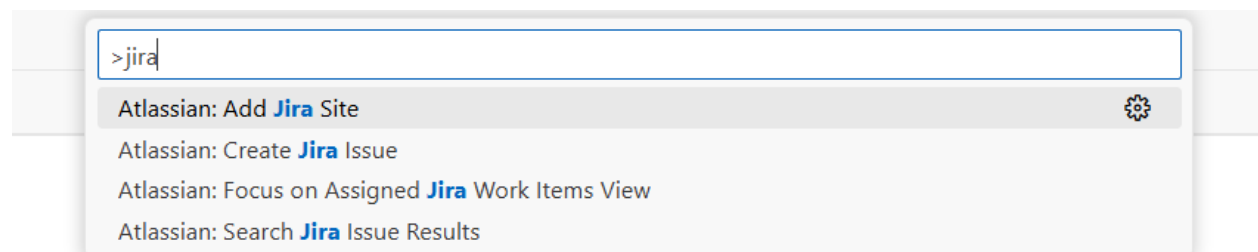
- [Build a Python app with PyInstaller](#)
- [Build a C++ app with Jenkins](#)
- [Build a Java app with Maven](#)
- [Build a Node.js and React app with npm](#)
- [Build a LabVIEW app](#)
- [Build a Jenkins pipeline by using Jenkinsfile Runner GitHub Actions](#)
- [Build a .NET Web App with Jenkins](#)

Only do 1. But when you click you will get step by step instructions.

## Lab 10: Integrate Visual Studio Code with Jira (Recipe Lab)

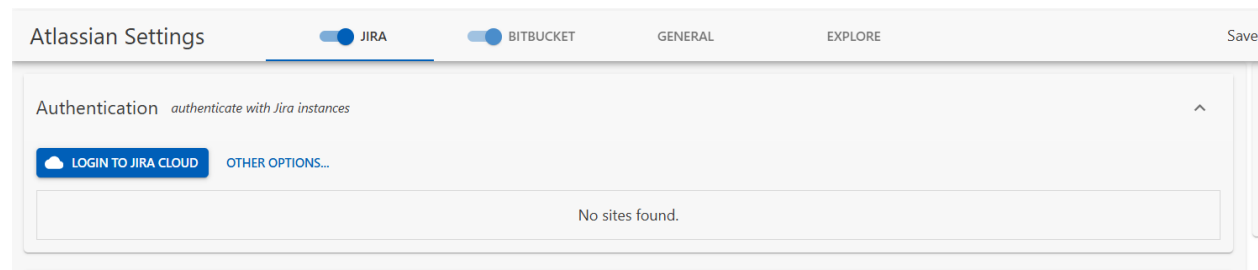
*Step 1: Get Jira in VS Code*

Inside VS Code type ctrl-shift-p to get the command window and type in Jira



Select add Jira site

Choose login to Jira Cloud



You will be prompted to accept

---

# Atlascode Integration is requesting access to your Atlassian account.

Use app on

chuckeasttom.atlassian.net



In Jira, it would like to:

Manage

› jira-project

View

› jira-user, jira-work

Update

› jira-work

---

By accepting this app, you:

- Grant the app access to your data in all places you can access where the app is installed.
- Agree to the app's [privacy policy](#) and [terms of use](#).

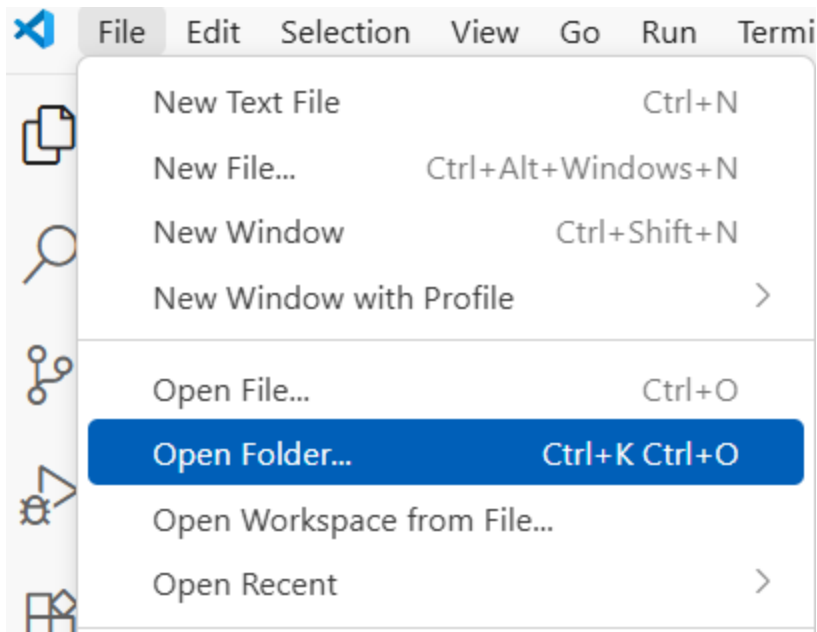
Accept

Cancel

## Lab 11: Test WebGoat in VS Code

*Step 1: Open Web Goat*

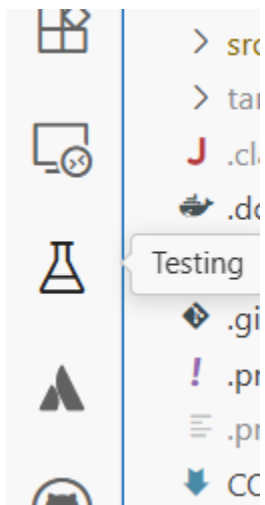
Choose open folder



And navigate to WebGoat

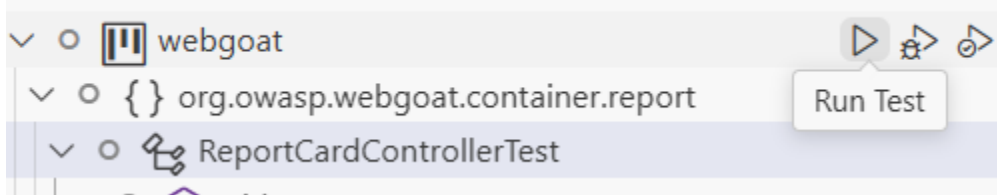
*Step 2: Run Basic Tests*

Click on the flask icon

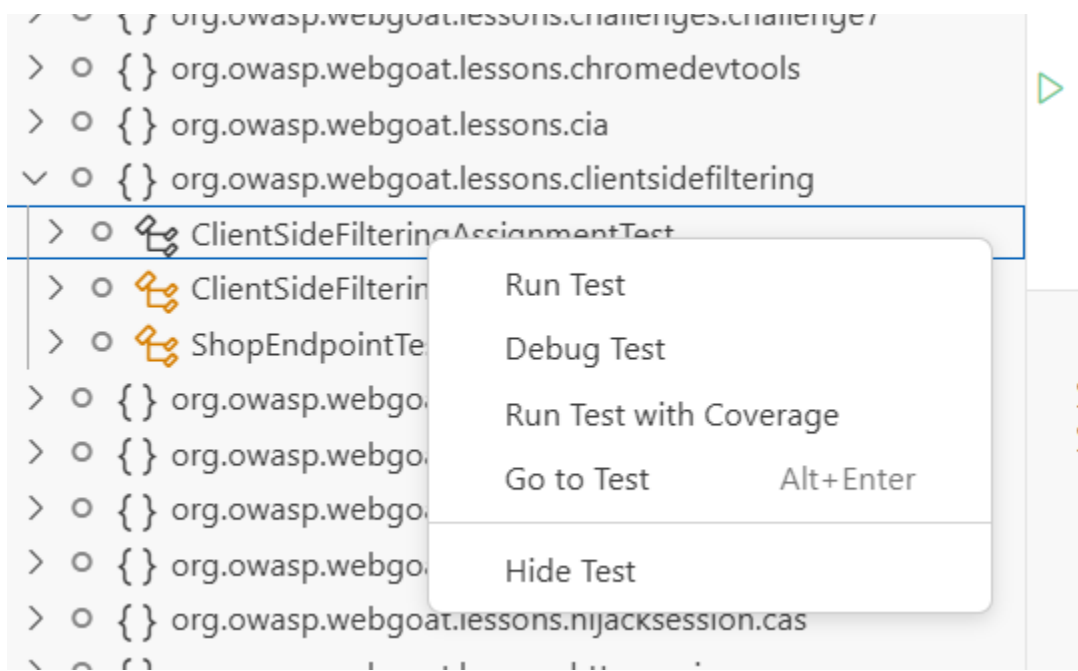


Click on run tests

No test results yet.



You can also right click on sections and run individual tests



Take time to run a few tests.

### **Lab 12: OWASP ZAP**

Download OWASP ZAP and scan some small website. You can scan [www.ChuckEasttom.com](http://www.ChuckEasttom.com) if you wish. The software is in your class materials but also can be found at <https://www.zaproxy.org/download/>

### **Lab 13: Testing in Visual Studio**

Use Visual Studio (not VS Code) to open web goat

Analyze ->Run Code Analysis on Solution

Analyze -> Calculate Code Metrics

### **Bit Bucket Labs**

#### **Lab 14 – BitBucket Initial Setup (Recipe Lab)**

Estimated Time: 20 minutes

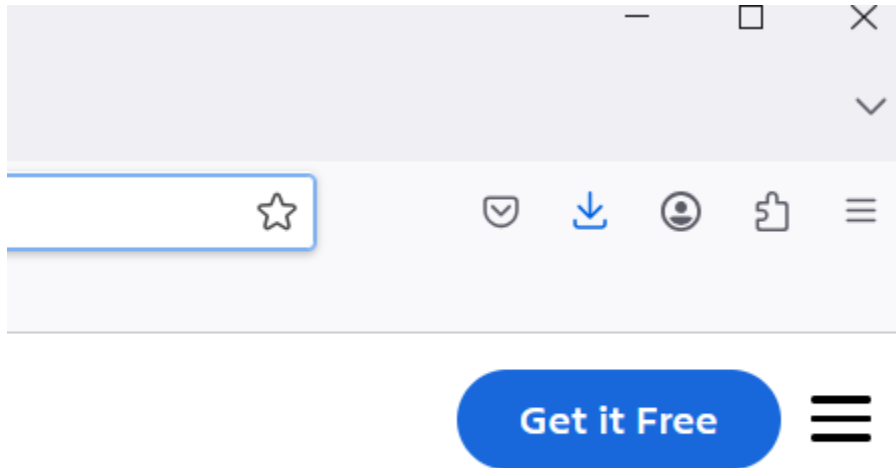
We will be integrating various tools with BitBucket. So, our first goal is to setup a free account.

*Step 1: Navigate to URL*

Navigate to <https://bitbucket.org/>

*Step 2: Get the Free version*

Click on 'Get It Free' in the upper right hand corner:



*Step 3: Create an account*

Then select Create an Account



Log in to continue

Enter your email

Remember me 

Continue

Or continue with:



[Can't log in?](#) • [Create an account](#)

*Step 4: Create a workspace*

Create a workspace

# Let's name your site, Chuck

## Work email

chuckeasttom@gmail.com

[Sign in with a different Atlassian account](#)

## Your new workspace

bitbucket.org/ jediacademy

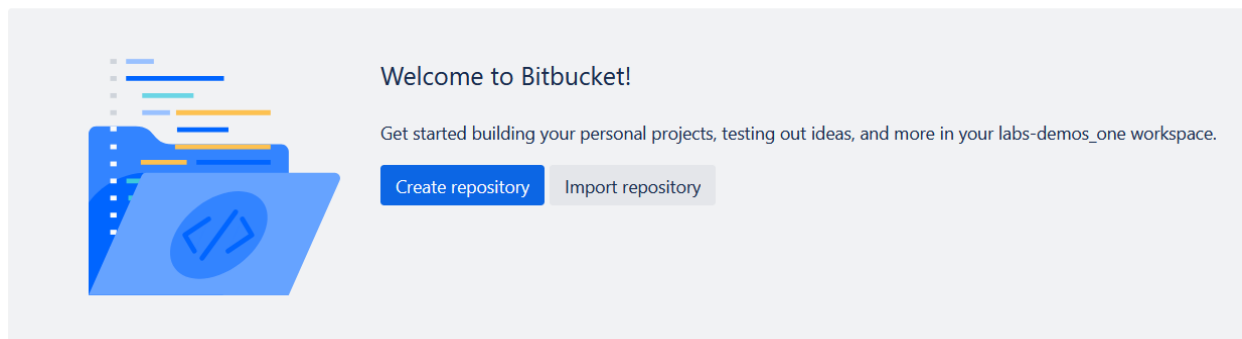
The workspace ID can only contain lowercase letters, numbers, dashes, and underscores.

I agree to the [Atlassian Customer Agreement](#), which incorporates by reference the [AI Product-Specific Terms](#), and acknowledge the [Privacy Policy](#).

**Agree and create workspace**

*Step 5: Create a repository*

Then create a repository

A screenshot of the Bitbucket welcome screen. On the left, there is a blue folder icon with a code symbol inside. To the right of the folder, the text reads "Welcome to Bitbucket!". Below this, it says "Get started building your personal projects, testing out ideas, and more in your labs-demos\_one workspace." At the bottom, there are two buttons: "Create repository" (highlighted in blue) and "Import repository" (in a light gray box).

Welcome to Bitbucket!

Get started building your personal projects, testing out ideas, and more in your labs-demos\_one workspace.

[Create repository](#) [Import repository](#)

## Create a new repository

[Import repository](#)

Workspace  labs-demos\_one

Repository name\*

Access level **Private repository**

Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.

Include a README? **Yes, with a tutorial (for beginners)** ▼

Default branch name

Include .gitignore? **Yes (recommended)** ▼

[> Advanced settings](#)

## Create a new repository

[Import repository](#)

Workspace  labs-demos\_one

Project\*  ▼

Repository name\*

Access level **No matches found**

Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.

- kyber
- Source
- Commits
- Branches
- Pull requests
- Pipelines
- Deployments
- Jira issues
- Security
- Downloads
- Repository settings

labs-demos\_one / Demoproject1

## kyber

Invite Clone

Here's where you'll find this repository's source files. To give your users an idea of what they'll find here, [add a description to your repository](#).

main Files Filter files

Name	Size	Last commit	Message
.gitignore	624 B	6 seconds ago	Initial commit
README.md	2.56 KB	6 seconds ago	Initial commit

### README.md

**Edit a file, create a new file, and clone from Bitbucket in under 2 minutes**

When you're done, you can delete the content in this README and update the file with details for others getting started with your repository.

## Lab 15: Synch Jenkins and BitBucket (Recipe Lab)

### Step 1: Go to Plugins


1. In Jenkins, go to **Jenkins > Manage Jenkins > Manage Plugins**.
2. Select the **Available** tab.
3. Search for BitBucket Server Integration.
4. Tick the checkbox.

You will need to reboot Jenkins after the installation


### Step 2: Visit Jenkins - System

Now navigate to Manage Jenkins > System


#### System Configuration




**System**  
Configure global settings and paths.




**Tools**  
Configure tools, their locations and automatic installers.




**Plugins**  
Add, remove, disable or enable plugins that can extend the functionality of Jenkins.



**Nodes**  
Add, remove, control and monitor the various nodes that Jenkins runs jobs on.



**Clouds**  
Add, remove, and configure cloud instances to provision agents on-demand.



**Appearance**  
Configure the look and feel of Jenkins

Now you will scroll down until you see BitBucket integration. If you don't see it you need to go back to the main Jenkins page and update Jenkins.

Step 3:

The download of the app could take a long time. Then you can get the remaining instructions here:

<https://plugins.jenkins.io/atlassian-bitbucket-server-integration/>

### **Lab 16 – Setup Project in Bit Bucket and integrate with Eclipse (Recipe Lab)**

*Step 1: Navigate to URL*

Navigate to <https://bitbucket.org/>

*Step 2: Create a Repository*

Create a repository:



Bitbucket

Your work

Pull requests

Repositories


Projects

More

Create

labs-demos\_one

### Recent repositories +



kyber  
labs-demos\_one

- Repository
- Project
- Snippet

### Pull requests

Make sure to select 'With Readme..'

## Create a new repository

[Import repository](#)

Workspace  labs-demos\_one

Project\*  Demoproject1  ▼

Repository name\*

Access level  Private repository

Uncheck to make this repository public. Public repositories typically contain open-source code and can be viewed by anyone.

Include a README?  ▼

Default branch name

Include .gitignore?  ▼

› [Advanced settings](#)

[Create repository](#)

[Cancel](#)

*Step 3: View Project*

You should see something like this:

labs-demos\_one / Demoproject1

## web-goat

Here's where you'll find this repository's source files. To give your users an idea of what they'll find here, [add a description to your repository](#).

main Files Filter files

/

Name	Size	Last commit	Message
.gitignore	624 B	15 seconds ago	Initial commit
README.md	2.56 KB	15 seconds ago	Initial commit

### README.md

**Edit a file, create a new file, and clone from Bitbucket in under 2 minutes**

When you're done, you can delete the content in this README and update the file with details for others getting started with your repository.

*We recommend that you open this README in another tab as you perform the tasks below. You can watch our video for a full demo of all the steps in this tutorial. C new tab to avoid leaving Bitbucket.*

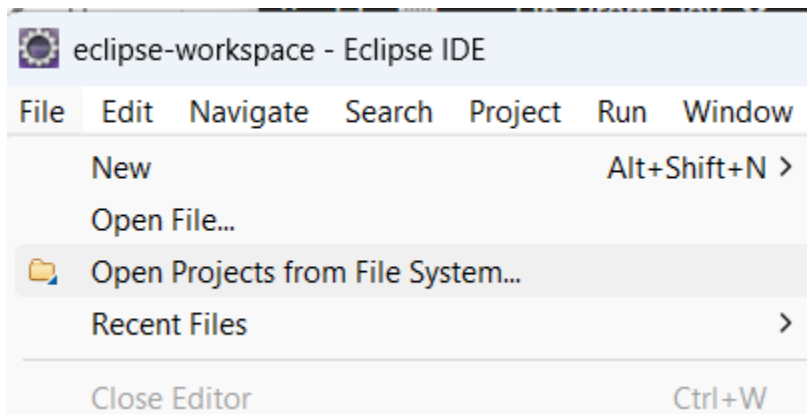
### Step 4: Open Eclipse

Minimize Bit Bucket and open Eclipse

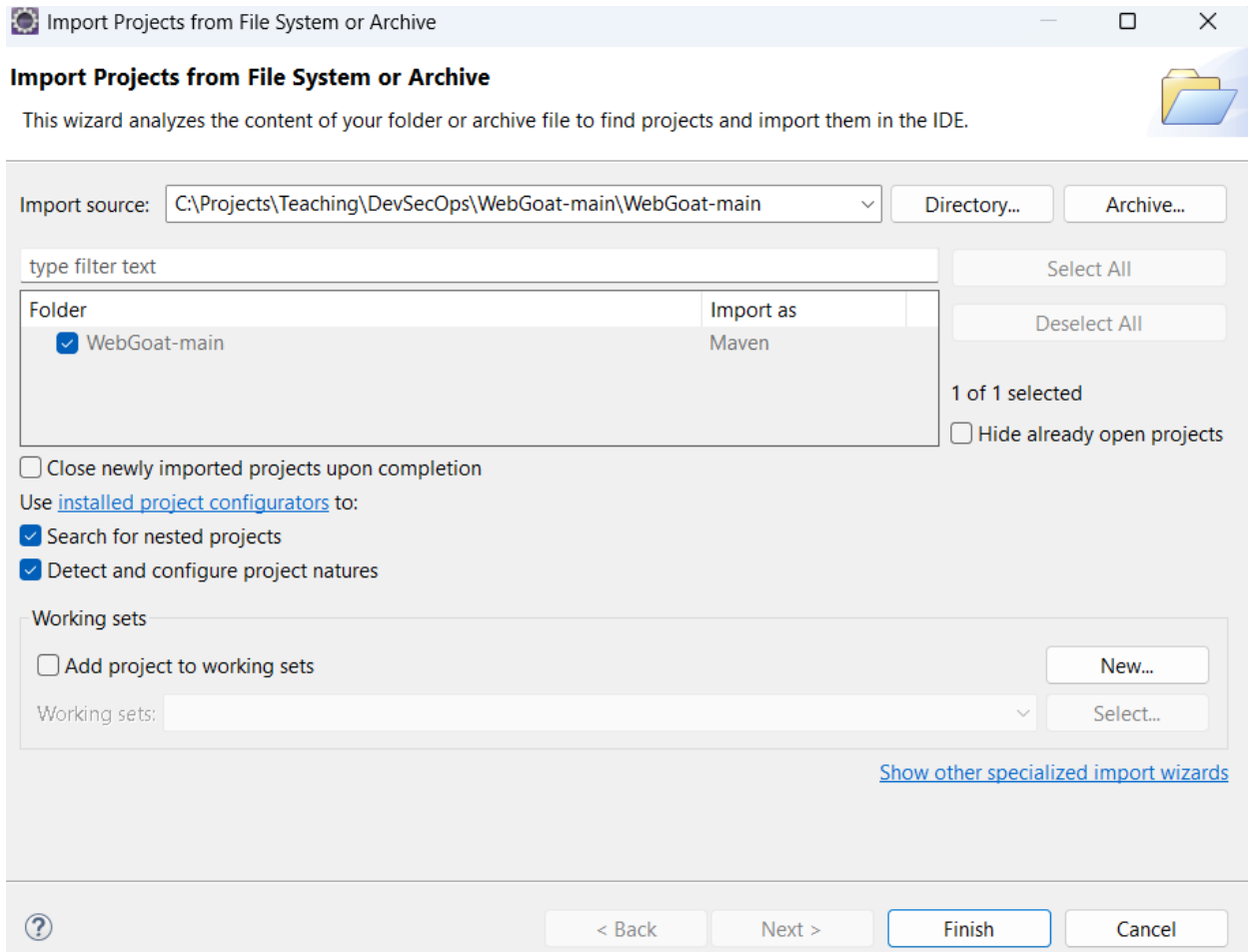
Note: Windows defender notices are common with Eclipse. Don't be concerned.

Step 5:

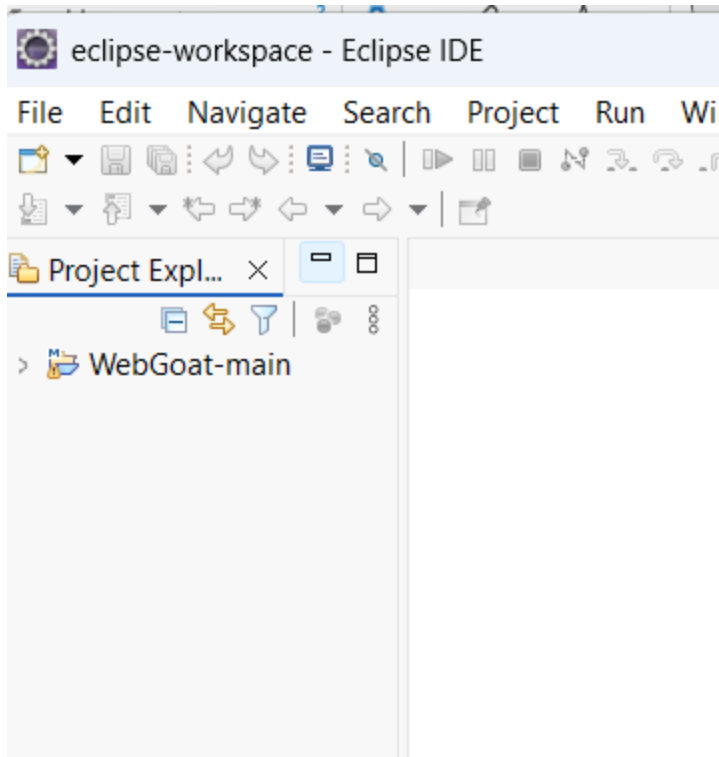
Go to file-> Open Projects from the file system



Navigate to wherever you put WebGoat in lab 5

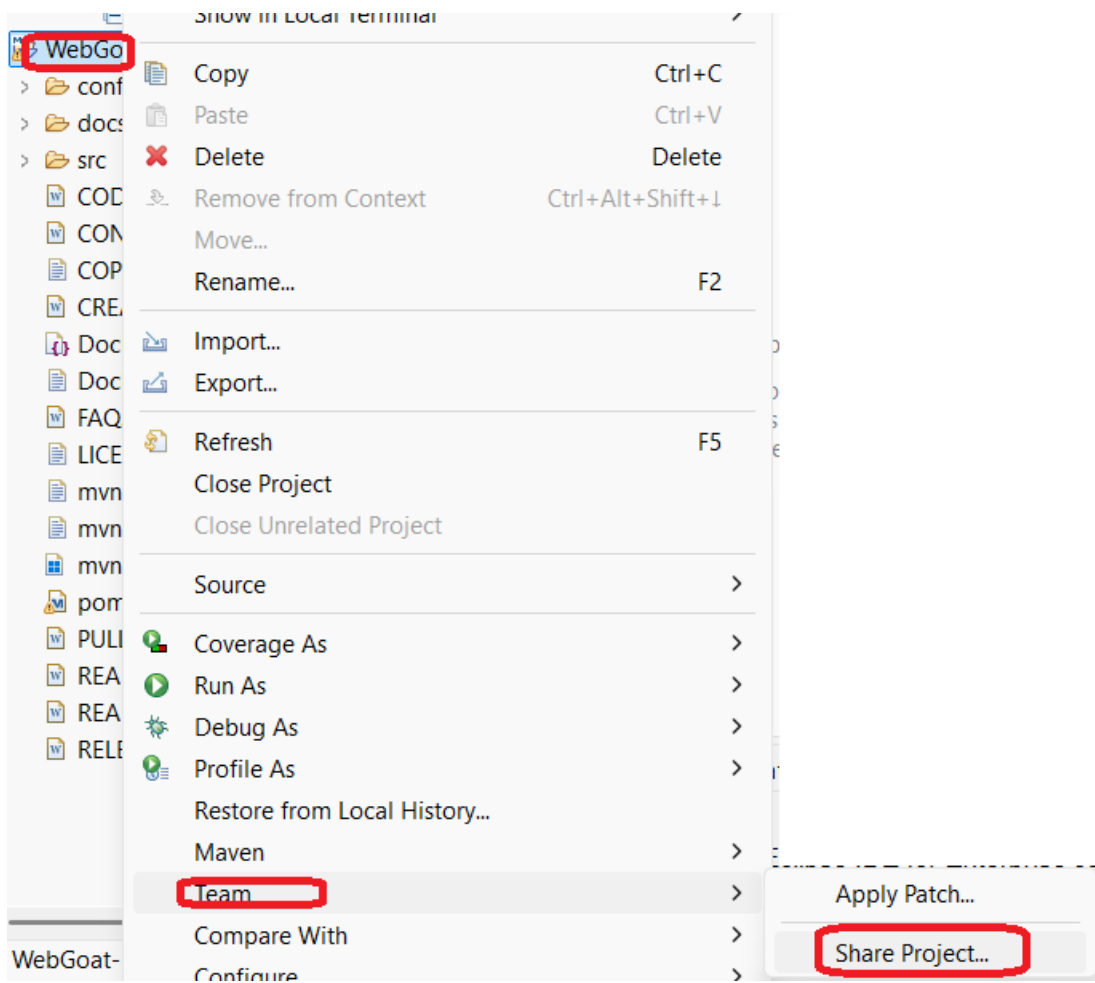


Now click finish, and you should see this:



*Step 5: Share Project*

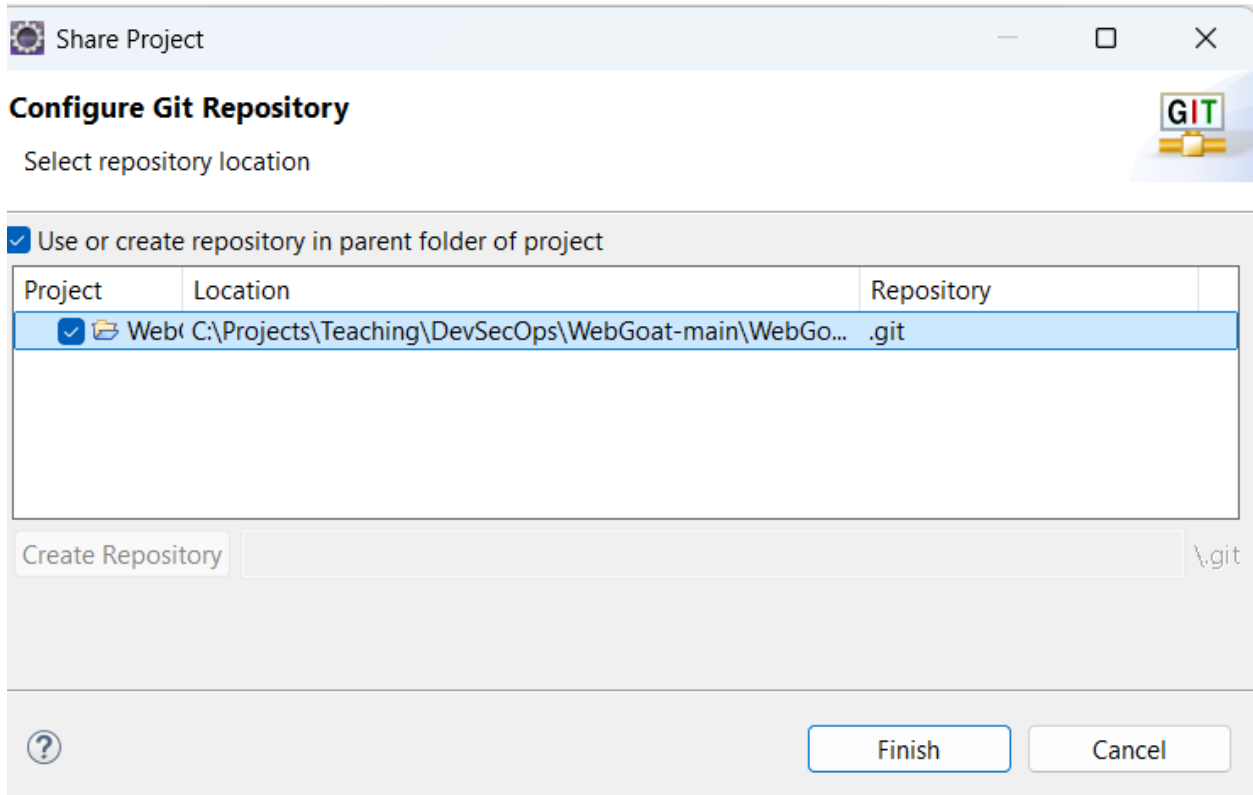
Now right click on the project, choose *teams* and *share project*



*Step 6: Configure Git repository*

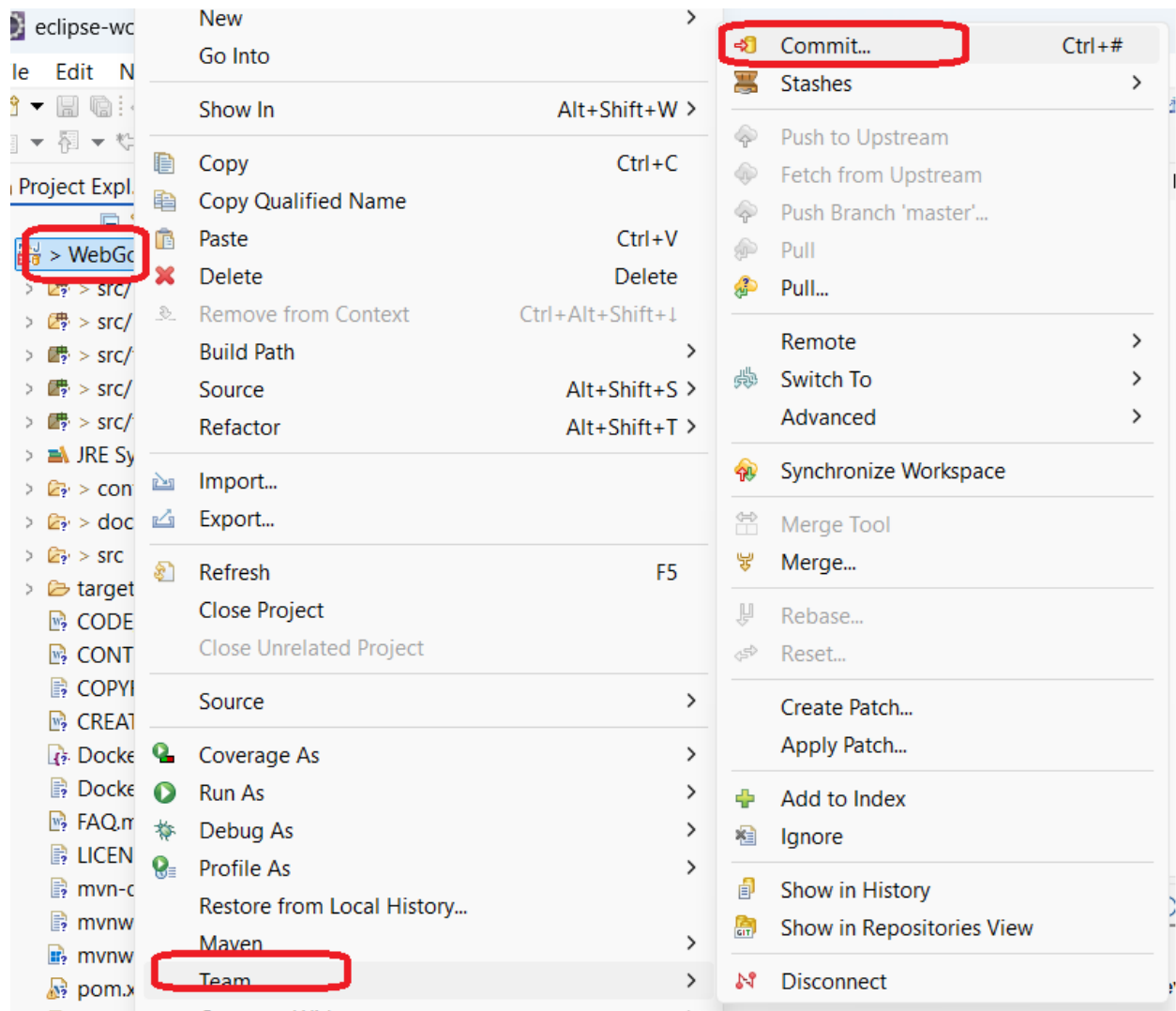
Now choose

Use or create repository in parent folder of project and click create repository then click finish.

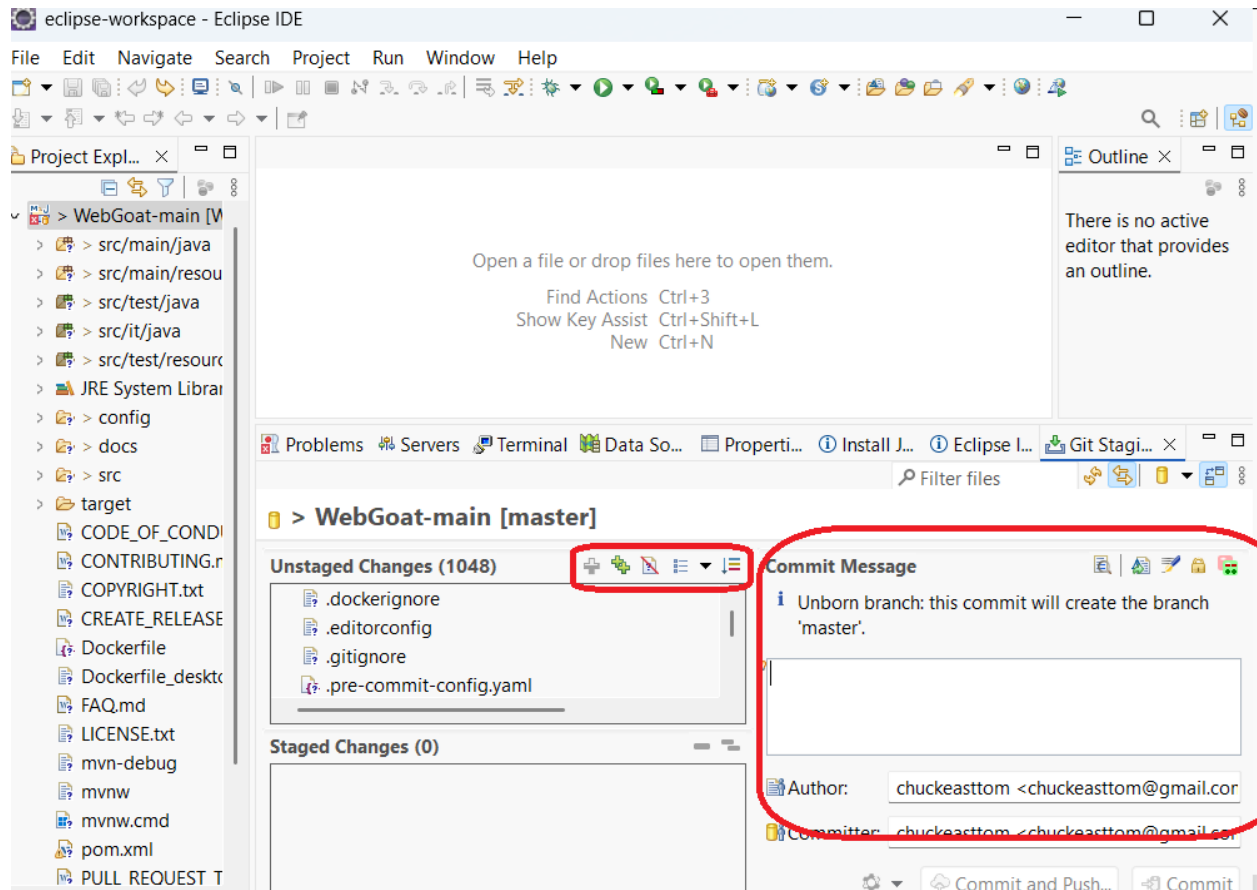


#### *Step 7: Commit*

Now right click on project, choose *team* and *commit*

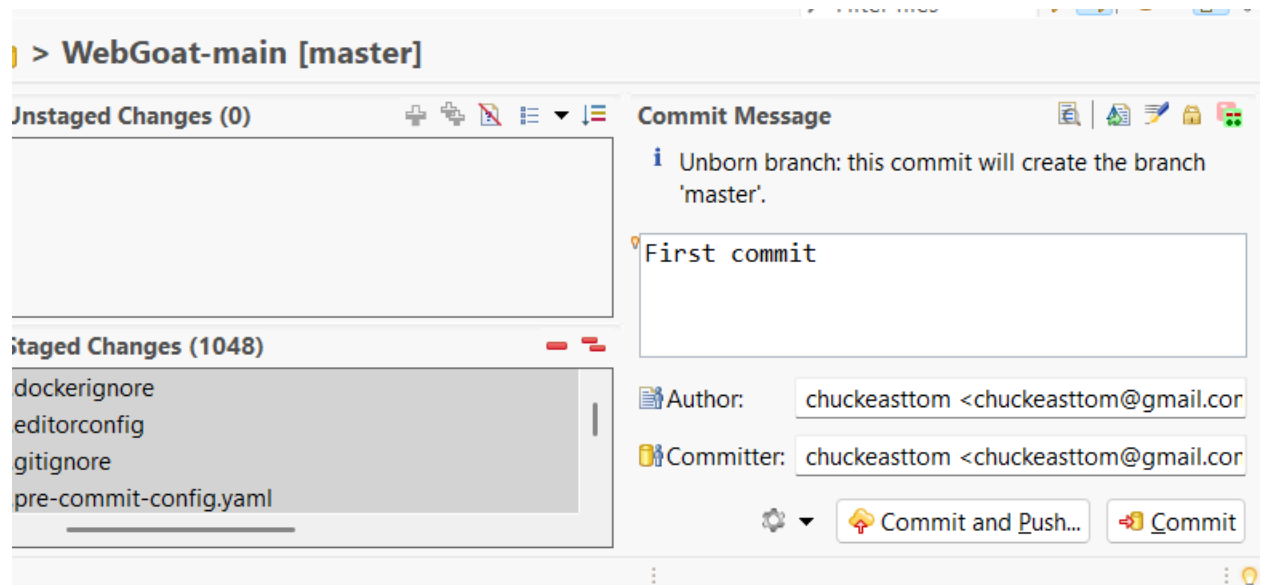


You will now see something new at the bottom:



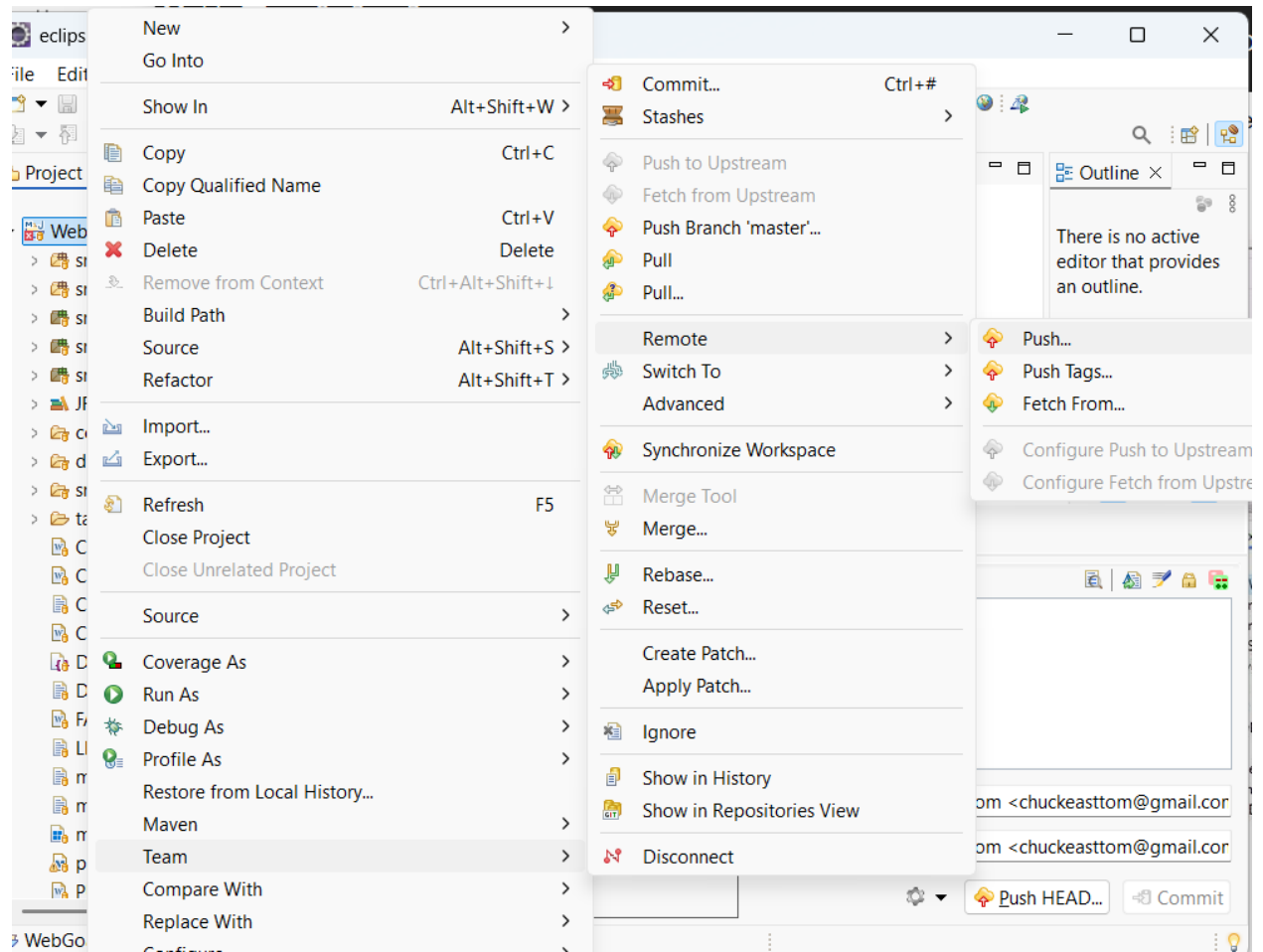
You will first click on the double + signs

Then in the commit message type 'first commit' and hit the commit button



*Step 8: Push*

Now right click on the project, choose *team > remote > push*



You should see the following screen:

Push to Another Repository

### Destination Git Repository

Enter the location of the destination repository.

**Location**

URI:  Local Folder...

Host:

Repository path:

**Connection**

Protocol:

Port:

**Authentication**

User:

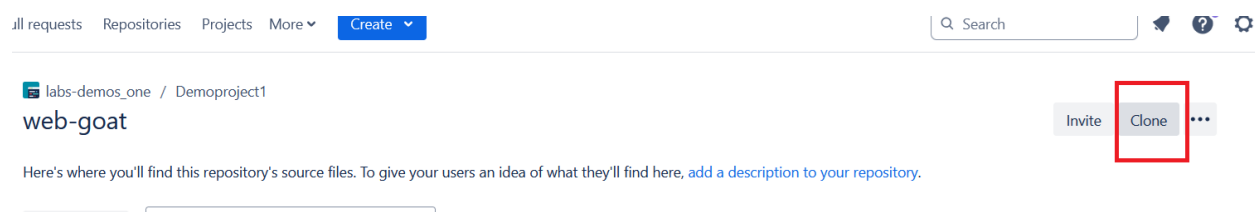
Password:

Store in Secure Store

#### Step 9: View code in BitBucket

Now we will leave that open, go back to your browser and BitBucket to get the information for the remote repository.

In BitBucket we will be in the source code that we created and now we will click on the *clone* button



Copy that path and past it in a text editor

---

## Clone this repository

HTTPS ▾

```
git clone https://chuckeasttom@bitbucket.org/labs-demos_one/web-goat.git
```



**Sourcetree** is a free Git client for Windows.

Clone in Sourcetree

**VS Code** is a source-code editor developed by Microsoft.

Clone in VS Code

Close


You can now close that window and click on the settings icon and choose *personal bitbucket settings*

Q Search



## Settings





### Atlassian Admin

-  **User management**  
Add users, groups, and manage access requests.

### Bitbucket Administration

-  **Workspace settings**  
Manage your workspace settings and integrations.

### Personal Settings

-  **Atlassian account settings**  
Manage your language, time zone, and other profile information. 
-  **Personal Bitbucket settings**  
Manage your email notification and other Bitbucket settings.
-  **Labs**  
Help us test these new or experimental features before they are released. Reme...

On the left you will see App passwords



Chuck Easttom

General

Account settings

Email aliases

Notifications

Access Management

App authorizations

App passwords

Security

GPG keys

SSH keys

Two-step verification

Audit log

## Account settings

### General

Avatar



Name Chuck Easttom

Update your avatar, name, email address, and password or de

### Bitbucket profile settings

Username `chuckeasttom` [\(change\)](#)

Website

Language English

Help [translate Bitbucket](#) into your language.

[Delete account](#)

### Preferences

Click on app passwords then click create app password. You will give this token quite a bit of privileges and give it a name, shown here:

## Details

Label\*

## Permissions

**Account**  Email  
 Read  
 Write

**Workspace membership**  Read  
 Write

**Projects**  Read  
 Write  
 Admin

**Repositories**  Read  
 Write  
 Admin  
 Delete

**Pull requests**  Read  
 Write

**Issues**  Read  
 Write

**Wikis**  Read and write

**Snippets**  Read  
 Write

**Webhooks**  Read and write

**Pipelines**  Read  
 Write  
 Edit variables

**Runners**  Read  
 Write

Now click create

You will get a long password that you should copy to the same text file you put the URL in.

*Step 10: Return to Eclipse*

Now it is back to Eclipse to put in this information

Push to Another Repository

### Destination Git Repository

Enter the location of the destination repository.

Location

URI:  Local Folder...

Host:

Repository path:

Connection

Protocol:

Port:

Authentication

User:

Password:

Store in Secure Store

When you click next, you will be prompted for the password you copied to the text file

Login

Repository

User

Password

Store in Secure Store

Now you type *refs/heads/master* in three location shown here:

Push to: [https://chuckeasttom@bitbucket.org/labs-demos\\_one/web-goat.git](https://chuckeasttom@bitbucket.org/labs-demos_one/web-goat.git)

### Push Ref Specifications

Select refs to push.

Specifications for push

Source ref:  Destination ref:

Remote ref to delete:

Mode	Source Ref	Destination Ref	Force Update	Remove
			<input checked="" type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>

Next click *Add Spec*

Now check the *force update* and *click next*

### Push Ref Specifications

Select refs to push.



#### Specifications for push

Source ref:  Destination ref:

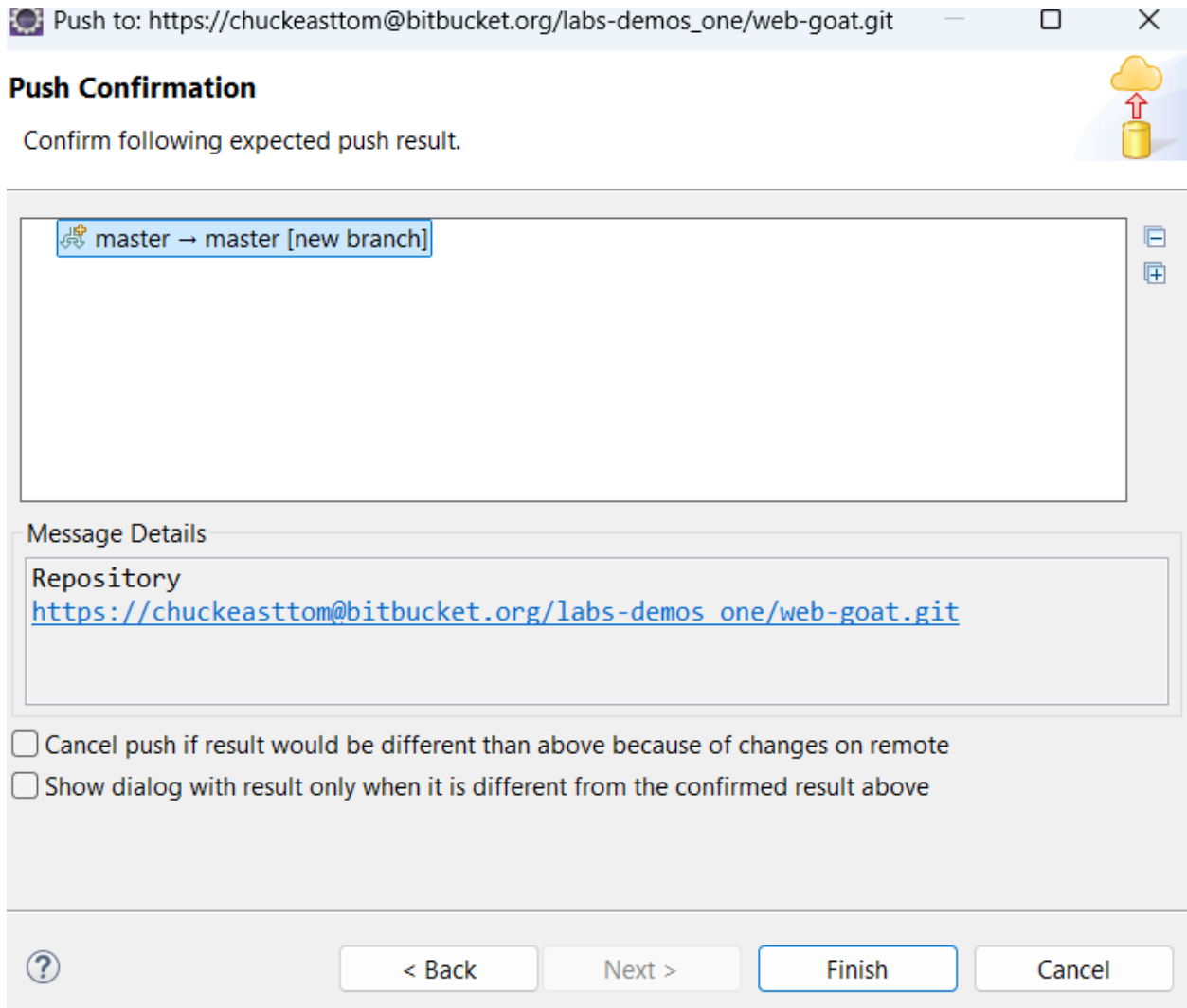
Remote ref to delete:

Mode	Source Ref	Destination Ref	Force Update	Remove
+ Update	refs/heads/master	refs/heads/master	<input checked="" type="checkbox"/>	

+ Add Spec  
✖ Add Spec  
Force Update All  
Remove All

Add predefined specification

You may be prompted for the password again. Now just highlight master->master



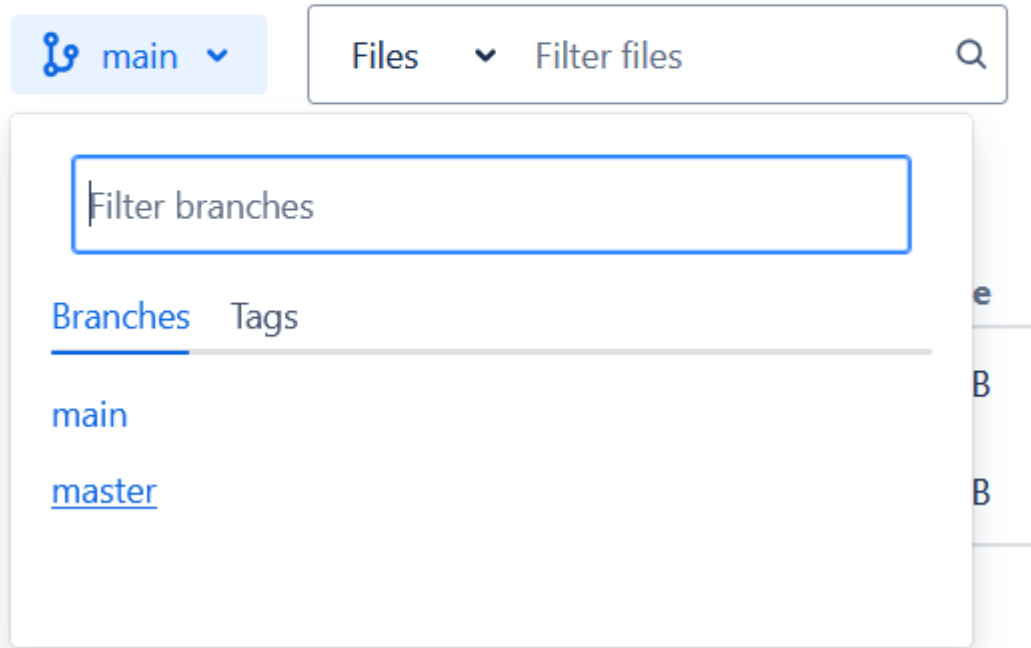
And click finish.

Now it is back to BitBucket. Make sure you select 'master' as the branch

labs-demos\_one / Demoproject1

# web-goat

Here's where you'll find this repository's source files. To give your u



The screenshot shows a GitHub repository interface. At the top, there is a breadcrumb trail: "labs-demos\_one / Demoproject1". Below this is the repository name "web-goat". A message reads: "Here's where you'll find this repository's source files. To give your u". Below the message, there is a navigation bar with a "main" branch selected, a "Files" dropdown menu, and a "Filter files" search box. A dropdown menu for "Filter branches" is open, showing a search input field with the text "Filter branches". Below the search field, there are two tabs: "Branches" (selected) and "Tags". Under the "Branches" tab, two branches are listed: "main" and "master".

And you will see your code

labs-demos\_one / Demoproject1

## web-goat

Here's where you'll find this repository's source files. To give your users an idea of what they'll find here, [add a description to y](#)

master Files Filter files

/

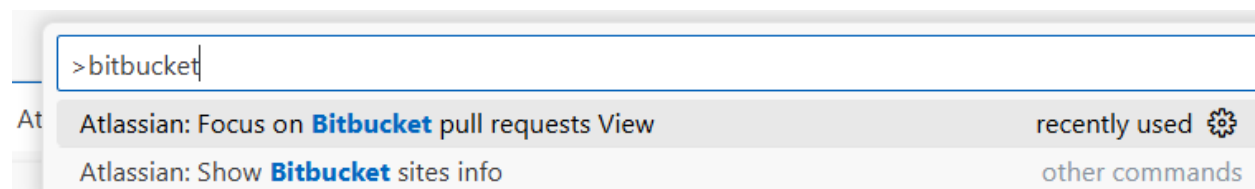
Name	Size	Last commit	Message
.github		19 minutes ago	First commit
.mvn		19 minutes ago	First commit
config		19 minutes ago	First commit
docs		19 minutes ago	First commit
src		19 minutes ago	First commit
.dockerignore	30 B	19 minutes ago	First commit
.editorconfig	341 B	19 minutes ago	First commit
.gitignore	1.37 KB	19 minutes ago	First commit

You have now synched your local project in Eclipse with the Project in BitBucket.

### Lab 17: Integrate Bitbucket with VS Code

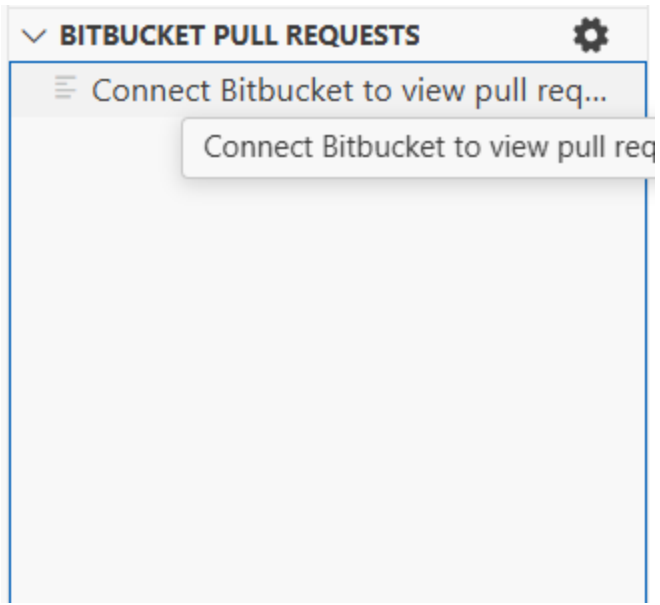
*Step 1: Get Bitbucket in VS Code*

In VS Code type ctrl-shift-p to get the command window, then type in bitbucket



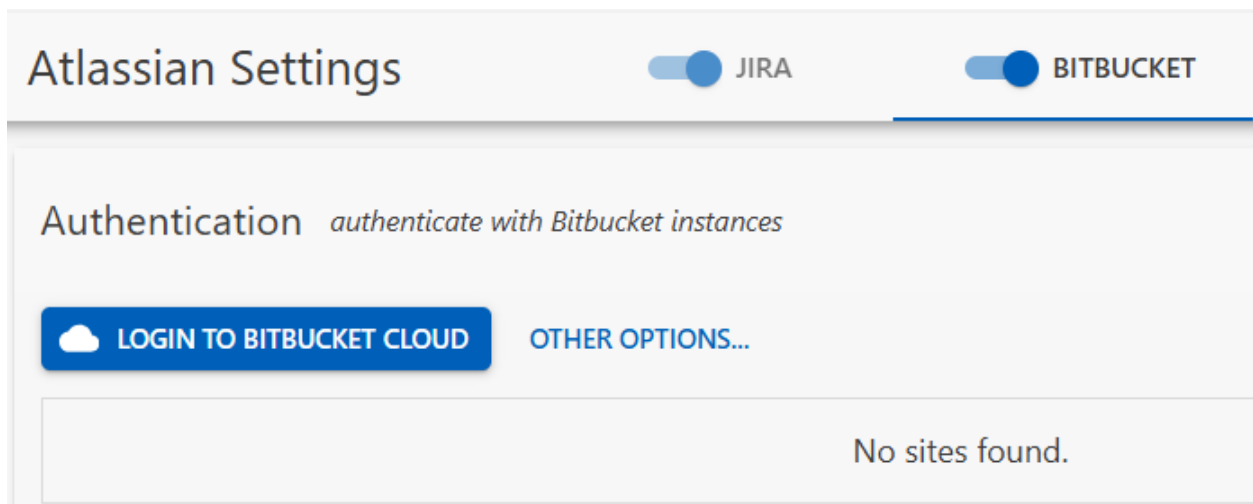
Choose Focus on Bitbucket pull requests view

You will then see this screen



*Step 2: Connect to Bitbucket*

Click on the Connect Bitbucket to view...



Click on Login to Bitbucket Cloud

You will be asked to confirm access:

# Confirm access to your account

---

Atlascode Integration (<http://127.0.0.1>) is requesting access to the following:

Read your account information

Read and modify your repositories' issues

Access your repositories' build pipelines

Read your workspace's project settings and read repositories contained within your workspace's projects

Read and modify your repositories and their pull requests

Read and modify your snippets

Read your team membership information

By installing the App you agree to the [privacy policy](#) provided by Atlascode Integration.

Grant access

Cancel

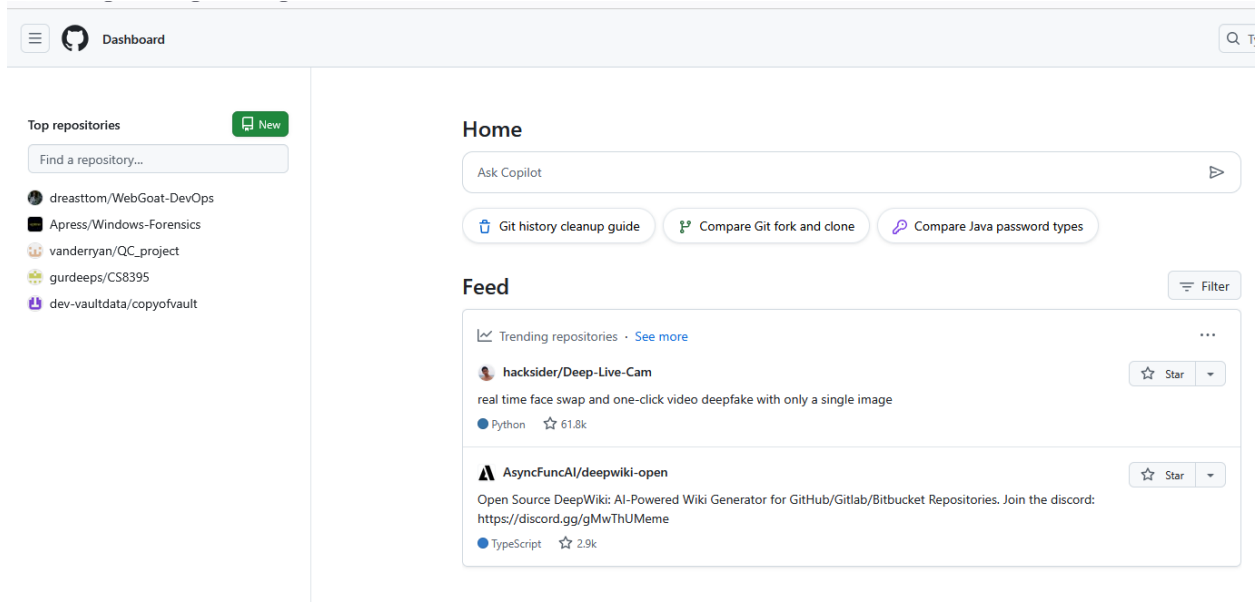
## GitHub Labs

### Lab 18 – GitHub Initial Setup (Recipe Lab)

Estimated time 10 minutes

*Step 1: Navigate to URL*

<https://github.com/>

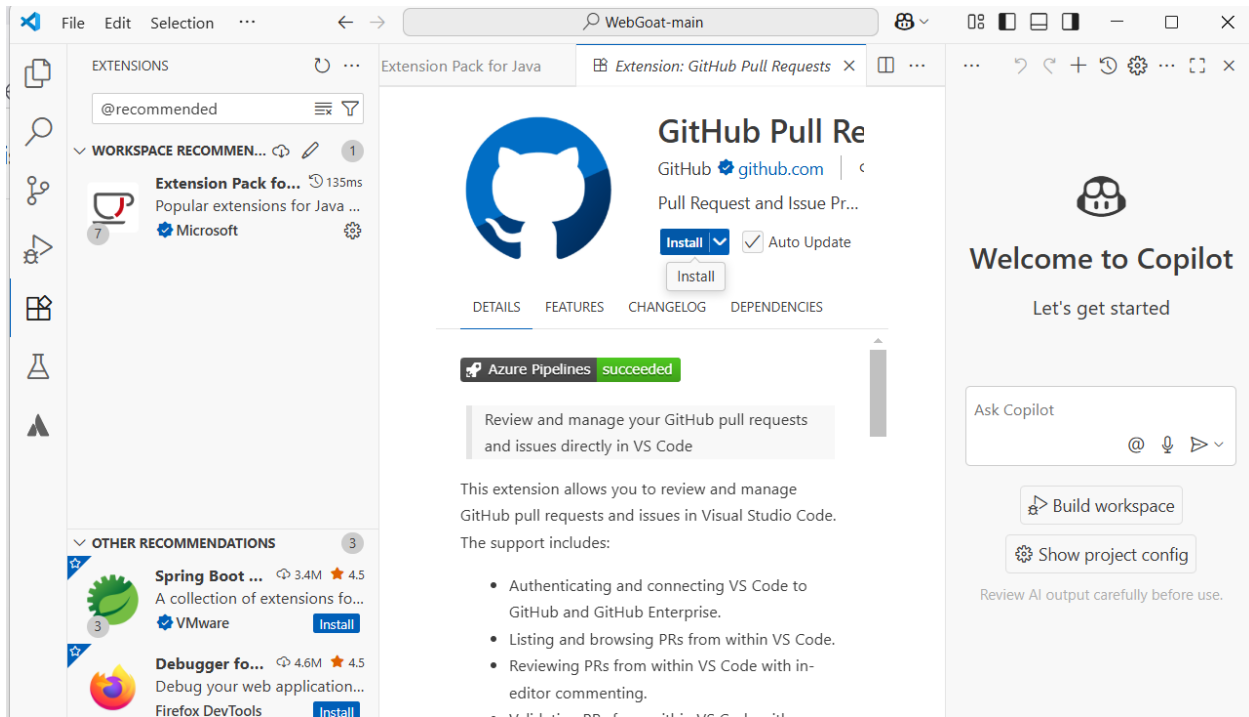


## Lab 19: Integrate Visual Studio Code with Github (Recipe Lab)

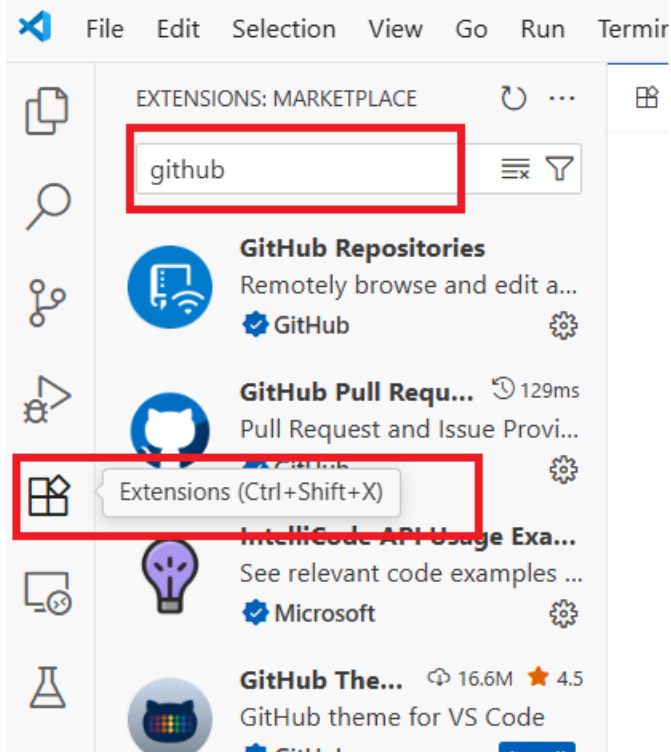
### Step 1 Setup Project

If Visual Studio code is not already installed, check to see if the installation file is there. If not you can download from <https://code.visualstudio.com/download>

### *Step 2: Install the GitHub extensions*

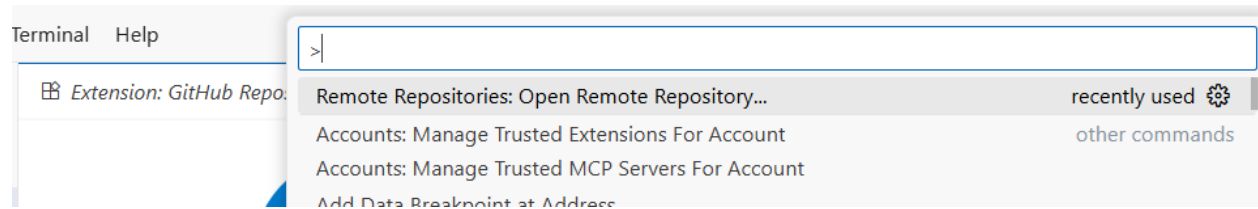


Note you can add a number of extensions by searching for them:

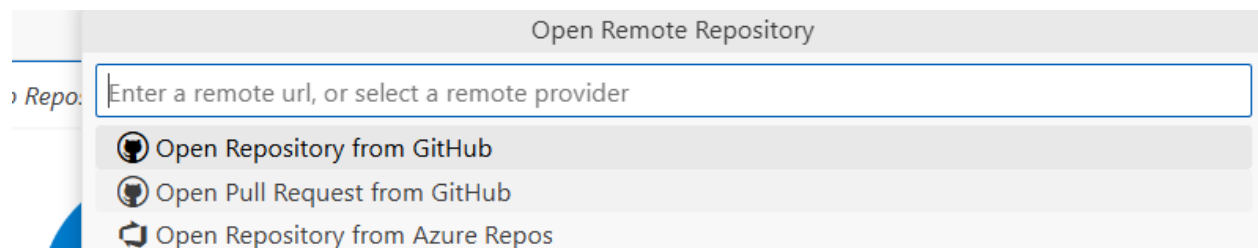


### Step 3 Connect to Github

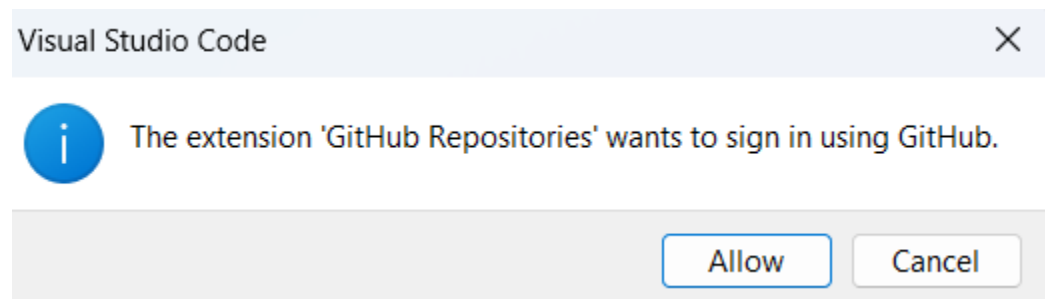
Press F1 to open the command window and enter Remote Repositories: Open Remote Repository.



Then select open repository from Github



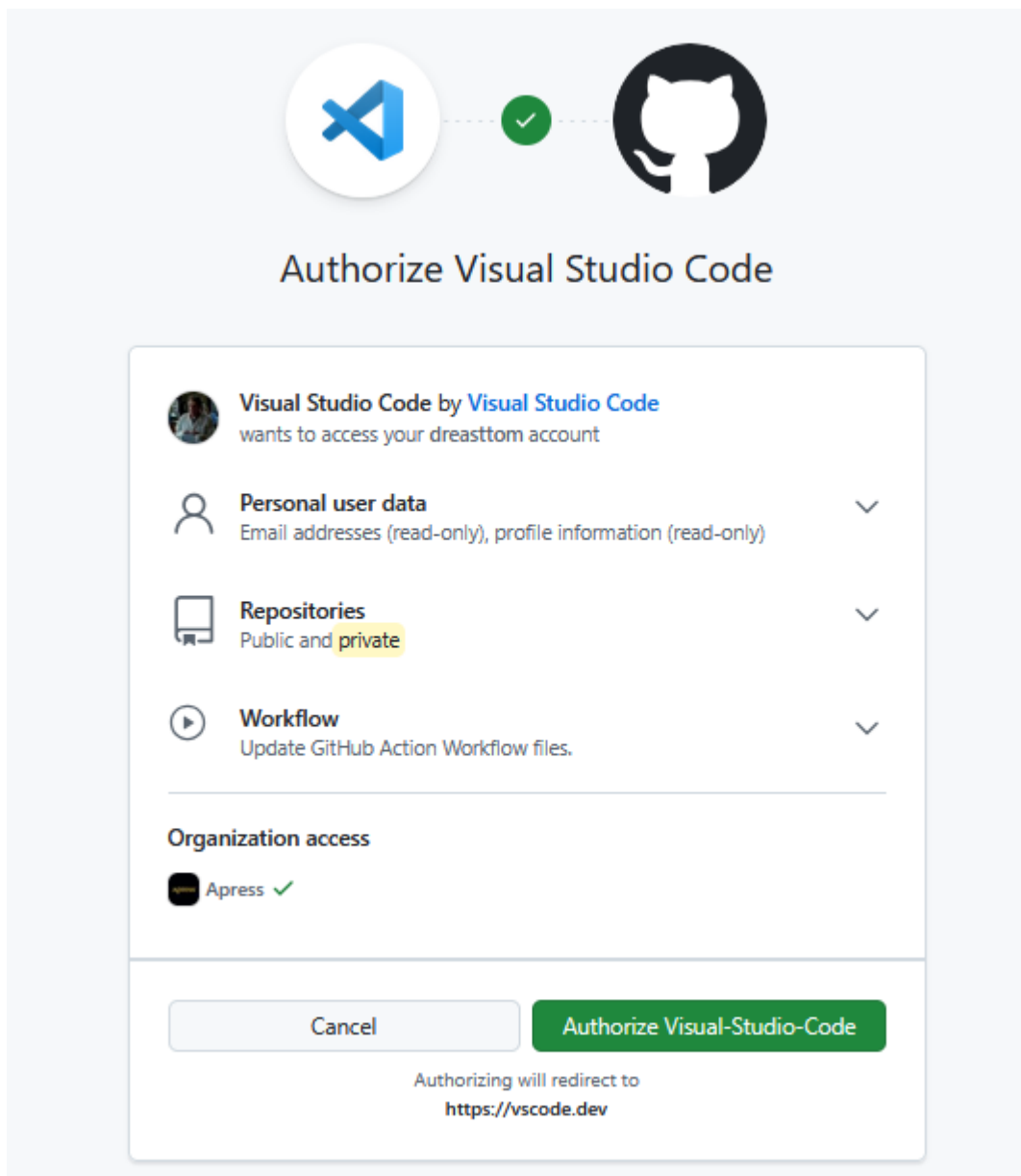
You will be asked to sign in to Github



If you click allow, you will be taken to github and asked to authorize Visual Studio Code

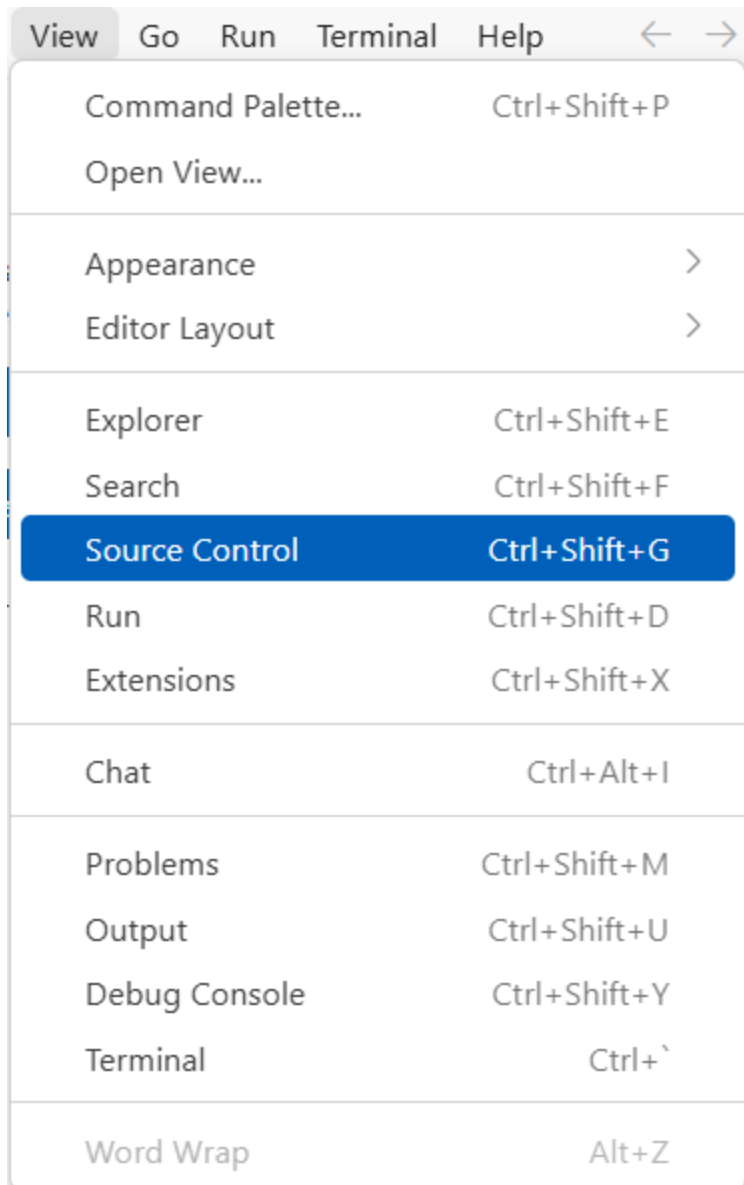
If you wish to go further beyond this lab,

<https://code.visualstudio.com/docs/sourcecontrol/github>



Now press Ctrl-Shift-P to get the command palette

Now you will go to View > Source Control



You should see the following screen. If you don't that means you have a folder open and need to close it.

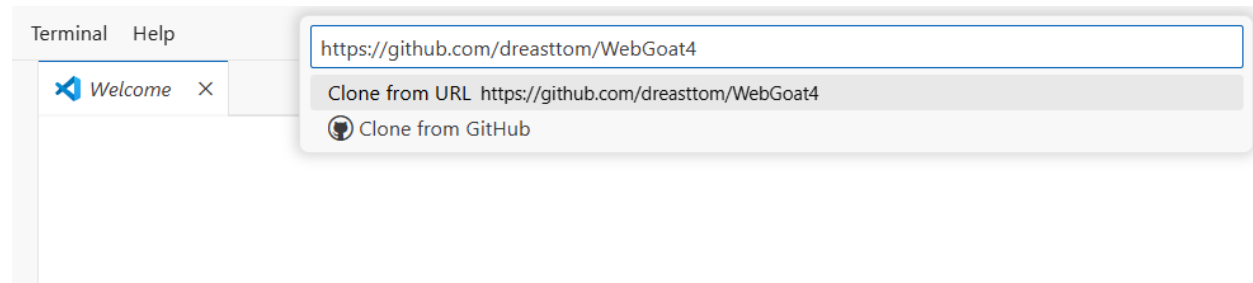
In order to use Git features, you can open a folder containing a Git repository or clone from a URL.

Open Folder

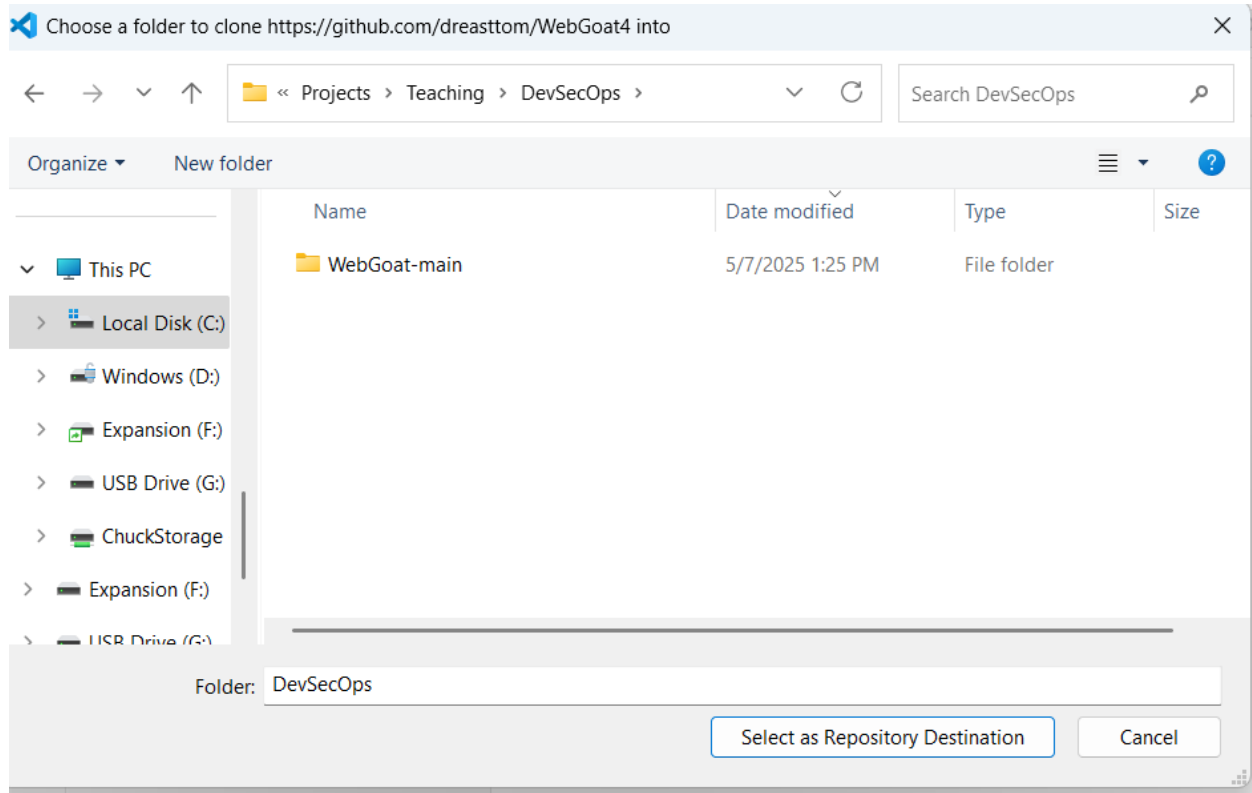
Clone Repository

To learn more about how to use Git and source control in VS Code [read our docs](#).

Click 'Clone Repository' and then type in the URL for your GitHub repository



You will be prompted to provide a destination for the clone

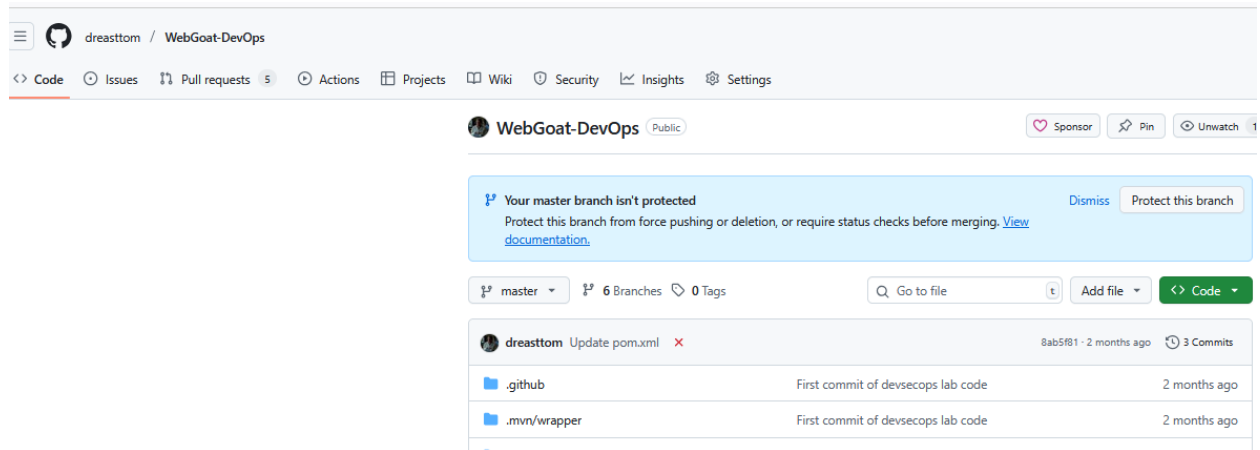


You have now linked that repository to your Visual Studio Code

## Lab 20: Viewing Activities in GitHub

### Step 1: Login into a GitHub repository

You will login to a repository you created in a previous lab.



### Step 2: Check Status

First click on pull requests

Navigation: Pull requests (5), Actions, Projects, Wiki, Security, Insights, Settings

Label issues and pull requests for new contributors  
Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)

Filters: is:pr is:open | Labels: 11

5 Open | 2 Closed | Author | Label | Projects | Milestone

- Bump docker/build-push-action from 2.7.0 to 6.18.0** × dependencies github\_actions  
#7 opened on May 28 by dependabot (bot)
- Bump softprops/action-gh-release from 1 to 2** × dependencies github\_actions  
#5 opened on May 4 by dependabot (bot)
- Bump docker/setup-buildx-action from 1 to 3** × dependencies github\_actions  
#4 opened on May 4 by dependabot (bot)
- Bump docker/setup-qemu-action from 1.1.0 to 3.6.0** × dependencies github\_actions  
#2 opened on May 4 by dependabot (bot)
- Bump actions/setup-java from 2 to 4** × dependencies github\_actions  
#1 opened on May 4 by dependabot (bot)

From here you can see all pull requests on this particular repository.

### Step 3: Setup Security

Click on security

Navigation: Wiki, Security, Insights, Settings

Overview | Security overview

- Reporting
  - Policy
  - Advisories
- Vulnerability alerts
  - Dependabot
  - Code scanning
  - Secret scanning

**Security policy** • Disabled  
Define how users should report security vulnerabilities for this repository [Set up a security policy](#)

**Security advisories** • Enabled  
View or disclose security advisories for this repository [View security advisories](#)

**Private vulnerability reporting** • Disabled  
Allow users to privately report potential security vulnerabilities [Enable vulnerability reporting](#)

**Dependabot alerts** • Disabled  
Get notified when one of your dependencies has a vulnerability [Enable Dependabot alerts](#)

**Code scanning alerts** • Needs setup  
Automatically detect common vulnerability and coding errors [Set up code scanning](#)

**Secret scanning alerts** • Enabled  
Get notified when a secret is pushed to this repository [View detected secrets](#)

Choose code scanning alerts

**Dependabot alerts • Disabled**

Get notified when one of your dependencies has a vulnerability

Enable Dependabot alerts

**Code scanning alerts • Needs setup**

Automatically detect common vulnerability and coding errors

Set up code scanning

**Secret scanning alerts • Enabled**

Get notified when a secret is pushed to this repository

[View detected secrets](#)

You will see several tools

## Code scanning

Automatically detect common vulnerabilities and coding errors.

### Tools

**CodeQL analysis**

Identify vulnerabilities and errors with [CodeQL](#) for [eligible](#) repositories.

Set up ▾

**Other tools**

Add any third-party code scanning tool.

[Explore workflows](#)

**Copilot Autofix**

Suggest fixes for CodeQL alerts using AI. CodeQL default or advanced setup must be enabled for this feature to work.

Learn more about the [limitations of autofix code suggestions](#)

On

**Copilot Autofix for third-party tools** Beta

Suggest fixes for third-party alerts using AI. Ensure that these tools are properly configured or that an analysis is uploaded for this feature to work. Learn more about the [limitations of autofix code suggestions for third party tools](#)

On

### Protection rules

**Check runs failure threshold**

Select the alert severity level for code scanning check runs to fail. [Create a branch ruleset](#) to prevent a branch from merging when these checks fail.

Security alert severity level: High or higher ▾

Standard alert severity level: Only errors ▾

### Secret Protection

GitHub will always send alerts to partners for detected secrets in public repositories. [Learn more about partner patterns.](#)

Disable

### Push protection

Block commits that contain [supported secrets](#).

Disable

At least setup the default CodeQL analysis. If you feel a bit more adventurous, try the Advanced, or even the ‘other tools’

## Tools

**CodeQL analysis** Set up ▾  
Identify vulnerabilities and errors with [CodeQL](#) for [eligible](#) repositories.

---

**Other tools**  
Add any third-party code scanning tool.

**Default**  
CodeQL will automatically find the best configuration for your repository.

**Advanced**  
Customize your CodeQL configuration via a YAML file checked into the repository.

**Copilot Autofix** On   
Suggest fixes for CodeQL alerts using AI. CodeQL default or advanced setup must be enabled for this feature to work.  
Learn more about the [limitations of autofix code suggestions](#)

---

**Copilot Autofix for third-party tools** Beta On   
Suggest fixes for third-party alerts using AI. Ensure that these tools are properly configured or that an analysis is uploaded for this feature to work. Learn more about the [limitations of autofix code suggestions for third party tools](#)


Even with default you can edit things like schedule

### CodeQL default configuration ×

**Query suites**  
[Group of queries](#) to run against your code.

**Default**  
CodeQL high-precision queries.

**Runner type**  
This is the runner default setup will use to run


 **Standard GitHub runner**

**Scan events**  
These events will trigger a new scan.

**On push and pull requests to**  
master and [protected branches](#)

---

**On a weekly schedule**

 Edit Cancel Enable CodeQL

It will take a few minutes to setup

Once it is setup you can view activity, including scanning logs

### Tools

The screenshot shows the GitHub CodeQL analysis tools interface. It features three main sections: 'CodeQL analysis', 'Other tools', and 'Copilot Autofix'. The 'CodeQL analysis' section includes a description, a link to 'CodeQL for eligible repositories', and a 'Default setup' button with a 'Last scan 6 minutes ago' indicator. The 'Other tools' section prompts the user to 'Add any third-party code scanning tool.'. The 'Copilot Autofix' section describes AI-powered fixes for CodeQL alerts and includes a link to 'limitations of autofix code suggestions'. A 'Copilot Autofix for third-party tools' section is also visible, marked as 'Beta'. A dropdown menu is open over the 'CodeQL analysis' section, listing options: 'View last scan log', 'View Code Scanning alerts', 'View CodeQL configuration', 'Switch to advanced' (with a sub-description: 'Customize your CodeQL configuration via a YAML file checked into the repository.'), and 'Disable CodeQL'.

## Lab 21: GitHub Collaboration

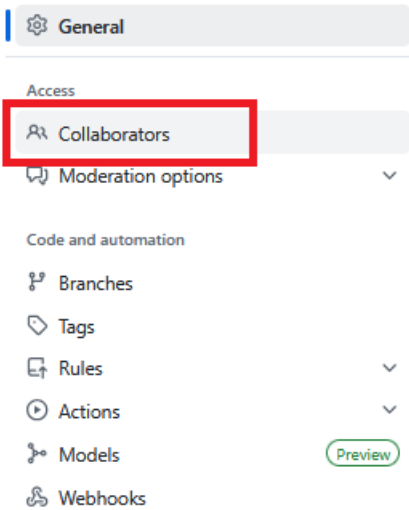
This will require you to work with at least one other student.

### *Step 1: Invite Collaborator*

The host will go to one of their repositories and click on settings

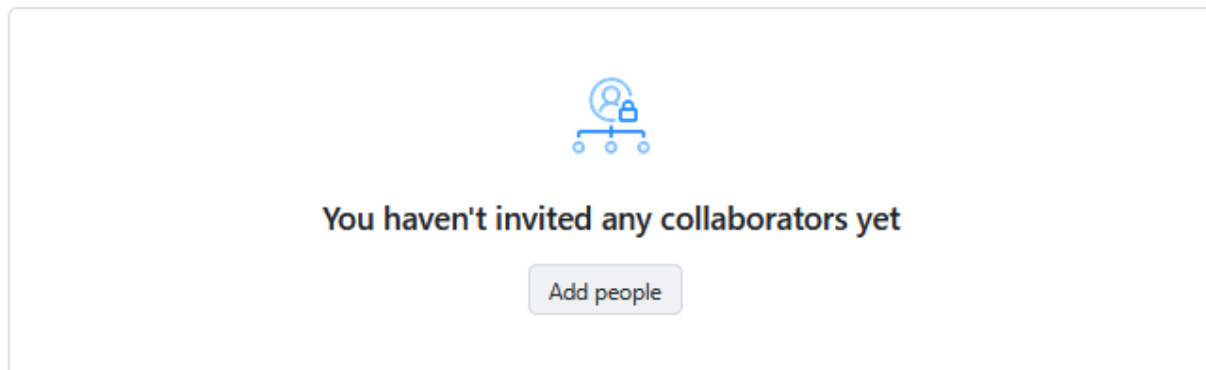
The screenshot shows the GitHub repository navigation bar for 'dreasttom / WebGoat-DevOps'. The navigation items are: '<> Code', 'Issues', 'Pull requests 5', 'Actions', 'Projects', 'Wiki', 'Security 64', 'Insights', and 'Settings'. The 'Settings' item is highlighted with a red rectangular box.

Then select collaborators

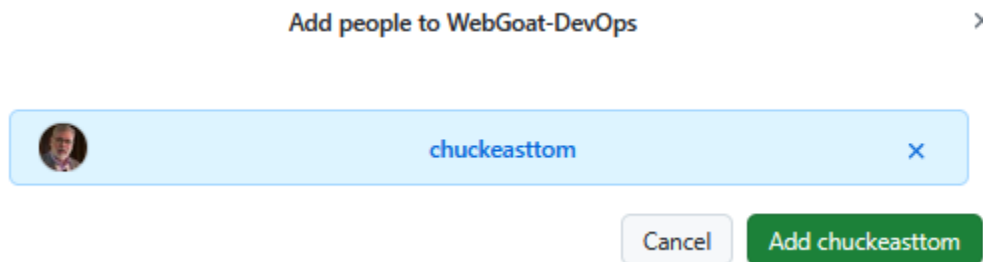


Then select add people

## Manage access



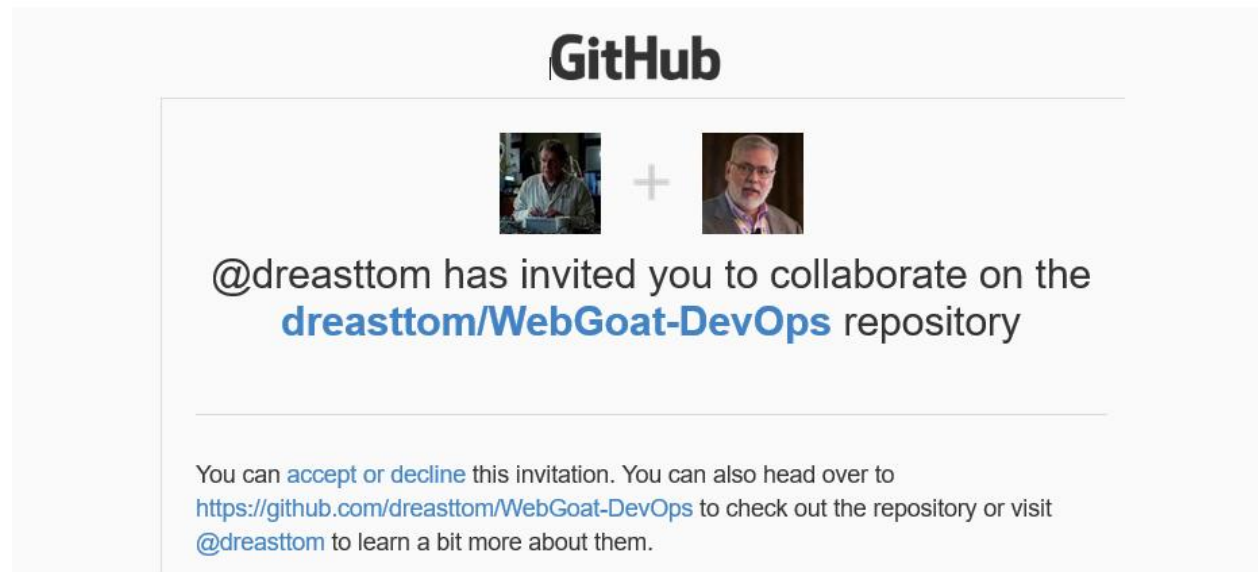
You will use an email or user name that your fellow student provides you. It must be one they have a GitHub account with



Note: while one of my accounts is shown this is just an example. Please do not invite me.

*Step 2: Accept the invitation*

Your colleague will now login into their GitHub. They also probably will receive an email like this:



That person will then see this



[dreasttom](#) invited you to collaborate on  
dreasttom/WebGoat-DevOps

Accept invitation

Decline invitation

 Owners of WebGoat-DevOps will be able to see:

- Your public profile information
- [Certain activity](#) within this repository
- Country of request origin
- Your access level for this repository
- Your IP address

---

Is this user sending spam or malicious content?  
[Block dreasttom](#)

They then accept the invitation and now both of you have access to this Github repository.

## GitLab Labs

### Lab 22 GitLab New project

*Step 1: Start New GitLab project*

Log into GitLab and choose to create a new project:



### Create blank project

Create a blank project to store your files, plan your work, and collaborate on code, among other things.



### Create from template

Create a project pre-populated with the necessary files to get you started quickly.



### Import project

Migrate your data from an external source like GitHub, Bitbucket, or another instance of GitLab.



### Run CI/CD for external repository

Connect your external repository to GitLab CI/CD.

## Select Create From Template

### *Step 2: Use Sample Template*

## Scroll down past numerous templates and select Sample GitLab Project



#### Laravel Framework

A basic folder structure of a Laravel application, to help you get started.



#### NIST 800-53r5

A project containing issues for security and privacy controls published by the U.S. National Institute of Standards and Technology



#### GitLab CI/CD components

A basic folder structure and sample files for a CI/CD components project.



#### HIPAA Audit Protocol

A project containing issues for each audit inquiry in the HIPAA Audit Protocol published by the U.S. Department of Health & Human Services



#### Sample GitLab Project

An example project that shows off the best practices for setting up GitLab for your own organization, including sample issues, merge requests, and milestones

And create a project from this template:

**Template**  
Sample GitLab Project Change template

**Project name**  
TestProject  
Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

**Project URL** https://gitlab.com/ chuckeasttom-group ▼ / **Project slug** testproject


**Project description (optional)**  
Description format

**Visibility Level** [?](#)

- Private**  
Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.
- Internal** [🔒](#)  
The project can be accessed by any logged in user except external users.
- Public** [🌐](#)  
The project can be accessed without any authentication.

Create project Cancel

You will see this progress screen:

 **Import in progress**  
This project is being imported. Do not make any changes to the project until the import is complete.

 **Import in progress**

Please wait while we import the repository for you. Refresh at will.

*Step 3: Review results*

When it is done you will have this screen:

⚠ You can't push or pull repositories using SSH until you add an SSH key to your profile.

Add SSH key

Don't show again

ℹ The project was successfully imported.

T **TestProject** 🔒 Free



Star 0



🔗 master ▾

testproject /



Find file

Edit ▾

Code ▾



**Project information**

🔗 2 Commits

🔗 31 Branches

🔗 0 Tags

📁 176 KIB Project Storage

📄 README

+ Add LICENSE

+ Add CHANGELOG

+ Add CONTRIBUTING

+ Add Kubernetes cluster

+ Set up CI/CD

+ Configure Integrations

🌐 Update README.md  
Administrator authored Feb 15, 2021

27329d3a



History

Name	Last commit	Last update
📄 README.md	Update README.md	4 years ago

📄 README.md

### Sample GitLab Project

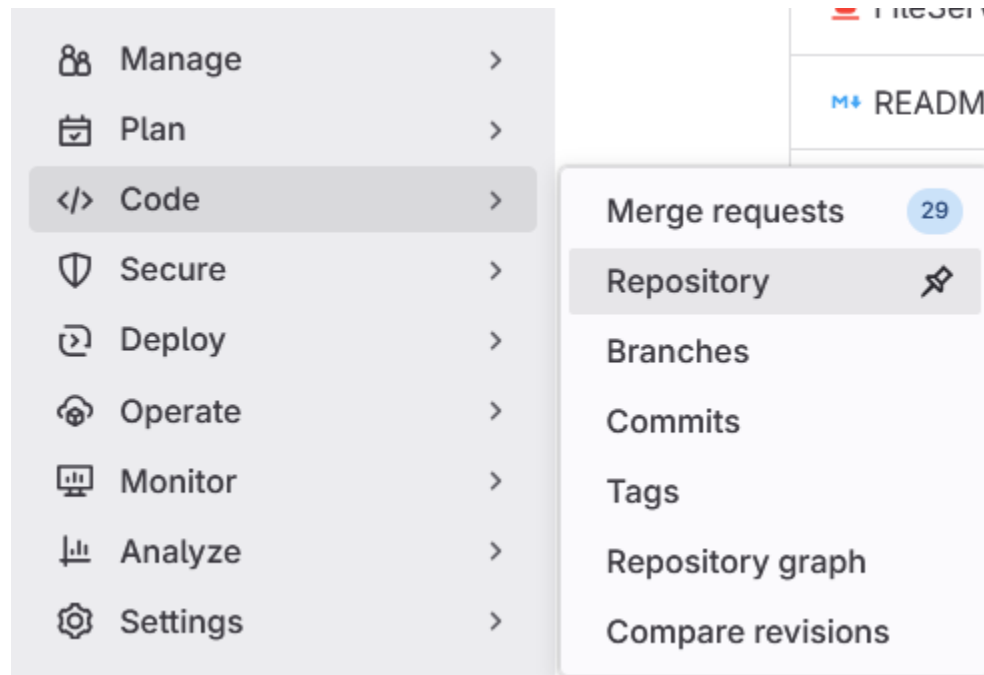
This sample project shows how a project in GitLab looks for demonstration purposes. It contains issues, merge requests and Markdown files in many branches, named and filled with lorem ipsum.

You can look around to get an idea how to structure your project and, when done, you can safely delete this project.

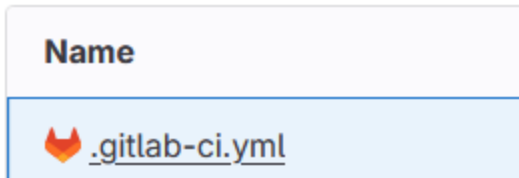
## Lab 23: Create a Pipeline in GitLab

### Step 1: Setup YAML file

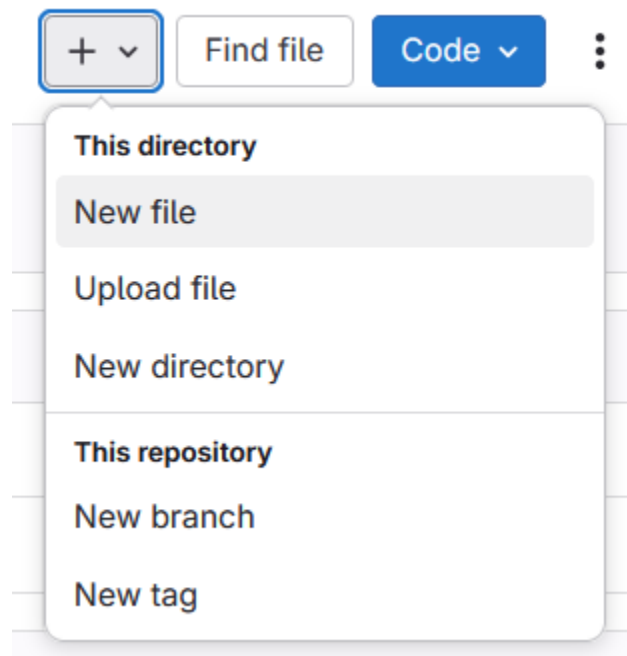
First open a project, then on the left choose Code > Repository



Look for the CI YAML file



If you don't have one, add one



### *Step 2: Setup Pipeline in YAML*

Paste the following code (note this comes from a GitLab tutorial [https://docs.gitlab.com/ci/quick\\_start/#create-a-gitlab-ciyaml-file](https://docs.gitlab.com/ci/quick_start/#create-a-gitlab-ciyaml-file))

build-job:

stage: build

script:

- echo "Hello, \$GITLAB\_USER\_LOGIN!"

test-job1:

stage: test

script:

```
- echo "This job tests something"
```

test-job2:

```
stage: test
```

```
script:
```

```
- echo "This job tests something, but takes more time than test-job1."
```

```
- echo "After the echo commands complete, it runs the sleep command for 20 seconds"
```

```
- echo "which simulates a test that runs 20 seconds longer than test-job1"
```

```
- sleep 20
```

deploy-prod:

```
stage: deploy
```

```
script:
```

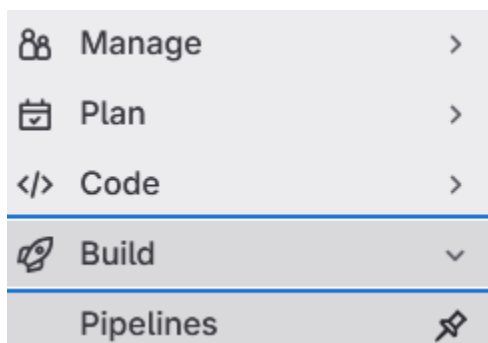
```
- echo "This job deploys something from the $CI_COMMIT_BRANCH branch."
```

```
environment: production
```

### Select **Commit changes**

*Step 3: See your pipeline running*

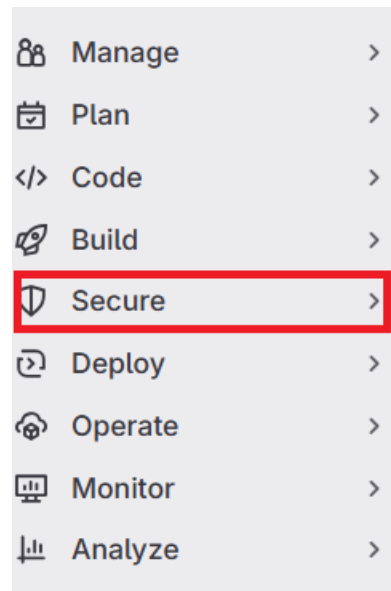
Go to Build > Pipelines. A pipeline with three stages should be displayed:



The project may fail the first time. That is often a problem with copying and pasting the code above in step 2.

## Lab 24: GitLab Secure Scanning

Open your project you created in lab 22 and choose security from the left hand menu:

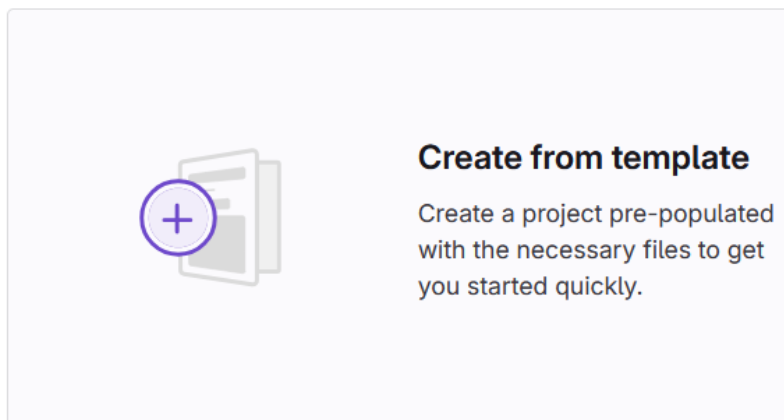


## Lab 25 Experiment With GitLab

### *Step 1 Setup Project in GitLab*

You may have already setup a project in a previous lab.

Projects> New Project use template



Choose Sample GitLab project



### Sample GitLab Project

An example project that shows off the best practices for setting up GitLab for your own organization, including sample issues, merge requests, and milestones

Preview

Use template

#### Project name

AwesomProject

Must start with a lowercase or uppercase letter, digit, emoji, or underscore. Can also contain dots, pluses, dashes, or spaces.

#### Project URL

https://gitlab.com/ chuckedstom-group

#### Project slug

AwesomProject

#### Project description (optional)

Description format

#### Visibility Level

Private

Project access must be granted explicitly to each user. If this project is part of a group, access is granted to members of the group.

Internal

The project can be accessed by any logged in user except external users.

Public

The project can be accessed without any authentication.

Create project

Cancel

It will take a while to import



### Import in progress

This project is being imported. Do not make any changes to the project until the import is complete.

## ○ Import in progress

Please wait while we import the repository for you. Refresh at will.

When it is done, upload a few code files. It really does not matter what they are, you can use WebGoat files if you wish. Or you can push them from Eclipse or Visual Studio if you prefer.

Now your repository should show a commit for each file you uploaded.

### *Step 2: Setup CI/CD*

From the project screen on the right side choose Set up CI/CD

📄 README

+ Add LICENSE

+ Add CHANGELOG

+ [Add CONTRIBUTING](#)

+ Add Kubernetes cluster

+ Set up CI/CD

+ Configure Integrations

---

# Optimize your workflow with CI/CD Pipelines

Create a new `.gitlab-ci.yml` file at the root of the repository to get started.

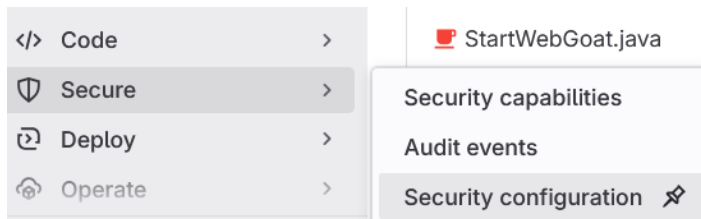
[Configure pipeline](#)

This will create a basic script

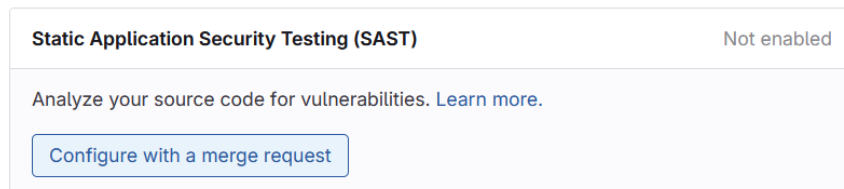
```
19 stages:           # List of stages for jobs, and their order of execution
20 |   - build
21 |   - test
22 |   - deploy
23
24 build-job:        # This job runs in the build stage, which runs first.
25 |   stage: build
26 |   script:
27 |     - echo "Compiling the code..."
28 |     - echo "Compile complete."
29
30 unit-test-job:    # This job runs in the test stage.
31 |   stage: test    # It only starts when the job in the build stage comple
32 |   script:
33 |     - echo "Running unit tests... This will take about 60 seconds."
34 |     - sleep 60
35 |     - echo "Code coverage is 90%"
36
37 lint-test-job:    # This job also runs in the test stage.
38 |   stage: test    # It can run at the same time as unit-test-job (in para
39 |   script:
```

*Step 3: Security Configuration*

Open Security Configuration



Click configure SAST

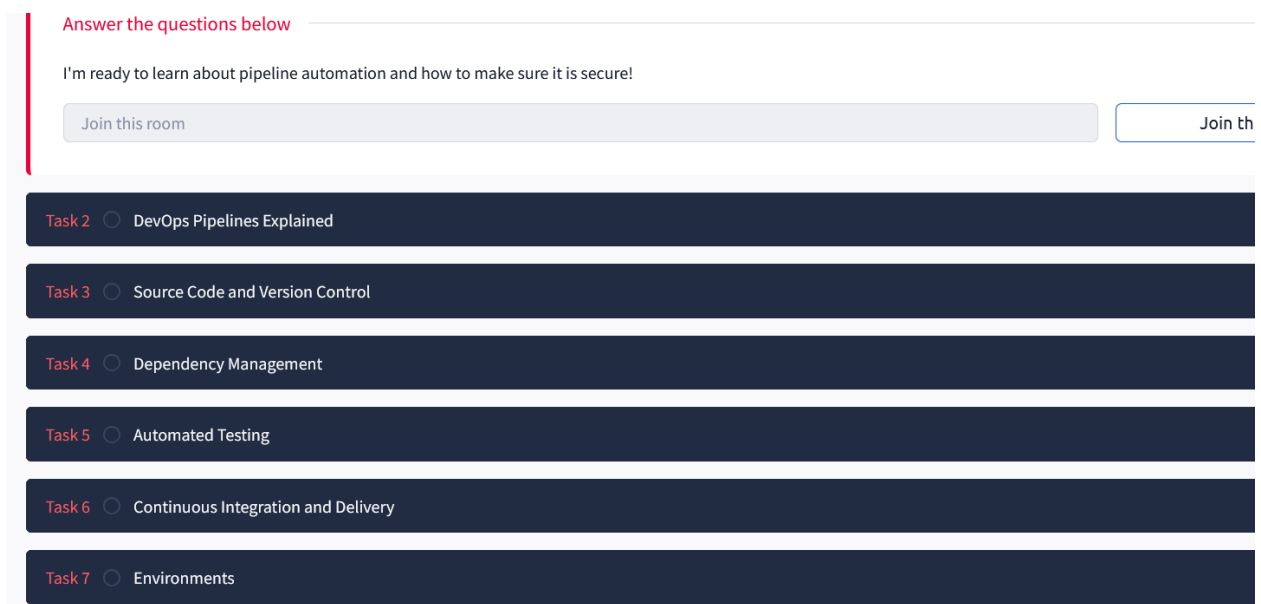


Take a few moments to review the various settings in SAST

## Online Labs (No VM needed)

### Lab 26: HackMe Pipeline Lab

<https://tryhackme.com/room/introtopipelineautomation>



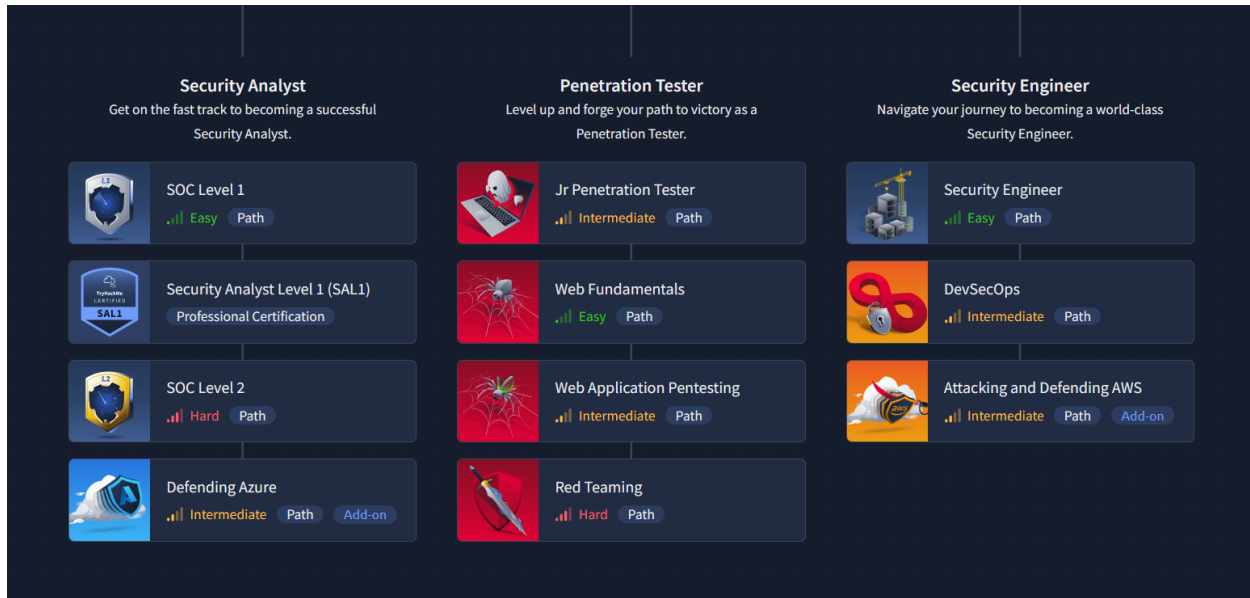
## Lab 27: Hackme DevSecOps Setup (Recipe Lab)

Navigate to

<https://tryhackme.com/path/outline/devsecops>

When you sign up note a) it is free and b) you can skip giving them email and phone number

You will be taken to the welcome page after you sign up. Then choose 'Check out our learning map' and scroll down until you see this:



Choose DevSecOps

There are Seven tasks to perform



Your goal in this first lab is to first get a free account, second navigate to DevSecOps and third to just to task 1 and 2.

### Lab 28: Hackme DevSecOps Setup (Recipe Lab)

Now you return to HackmeDevSecOps <https://tryhackme.com/path/outline/devsecops>

Again navigate until you get to the steps to perform and now finish the remaining five steps:



### Lab 29: Git(Recipe Lab)

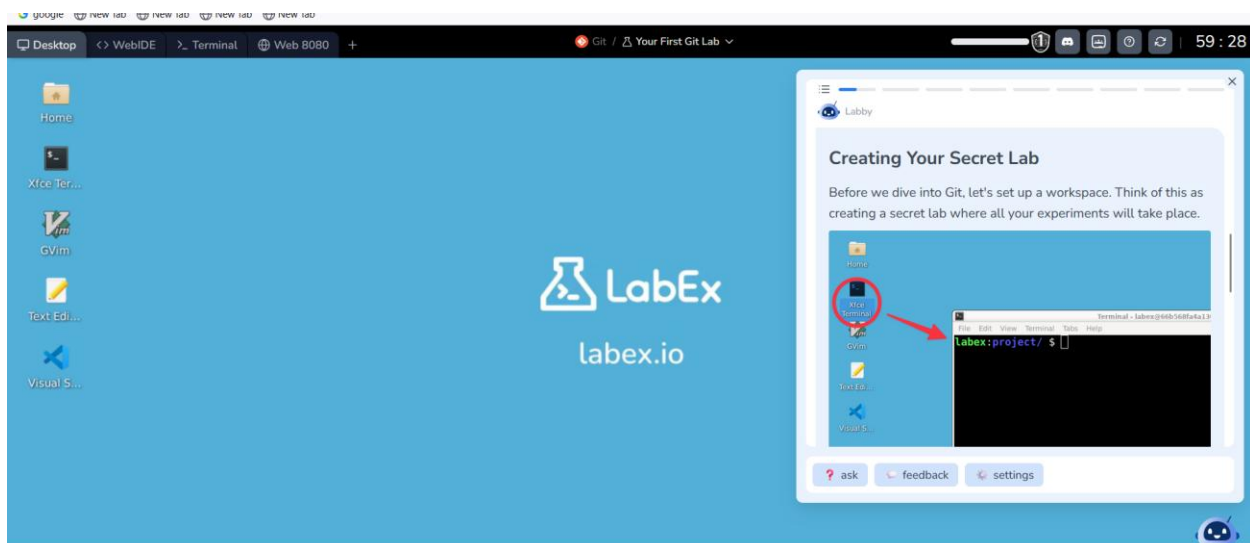
Go to <https://labex.io>

Choose Git

The image shows three skill cards from a learning platform. Each card features a circular icon, a title, a 'Joined' status button, and a progress bar. The first card is for 'DevOps' with a red icon containing 'C', 'O', and 'D', showing '2 / 352 skills', '8 courses', and '53 projects'. The second card is for 'Git' with a red icon containing a white Git logo, showing '0 / 32 skills' and '3 courses'. The third card is for 'Docker' with a blue icon containing a white Docker logo, showing '2 / 32 skills' and '5 courses'.

You will then select *Quick Start with Git*.

Click the 'Start Learning' button on the far right and you will launch a VM




Simply follow the instructions

For most labs on this site, there is a point where you will get to labs only available for premium (i.e. paid) subscriptions. Always stop there.

## Lab 30: Docker (Recipe Lab)

Go to <https://labex.io>

Navigate down to the 'Docker' project.




### Docker

0 / 32 skills 5 courses

Joined

That will take you to this

LEARN PROJECTS PRICING Upgrade




## Docker

Docker is revolutionizing application deployment and scaling. This Skill Tree offers a systematic way to learn containerization with Docker. Ideal for DevOps beginners, it provides a clear roadmap for understanding container creation, management, and orchestration. Hands - on, non - video courses and practical exercises in a Docker playground help you develop real - world skills in containerizing and deploying applications.

32 skills | 5 courses

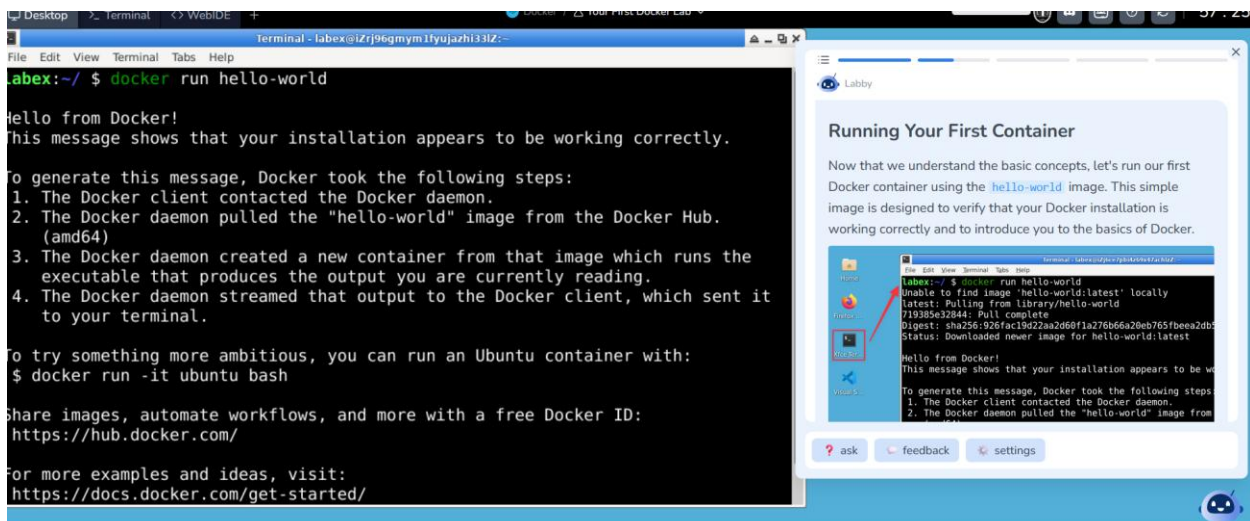
Unlock All Docker Labs

 Quick Start with Docker Start Learning

A VM will launch with instructions on the right:



Now you will walk through the lab instructions it gives. This is the first



For most labs on this site, there is a point where you will get to labs only available for premium (i.e. paid) subscriptions. Always stop there.

## Lab 31: Kubernetes (Recipe Lab)

Go to <https://labex.io>

Navigate to Kubernetes



Kubernetes

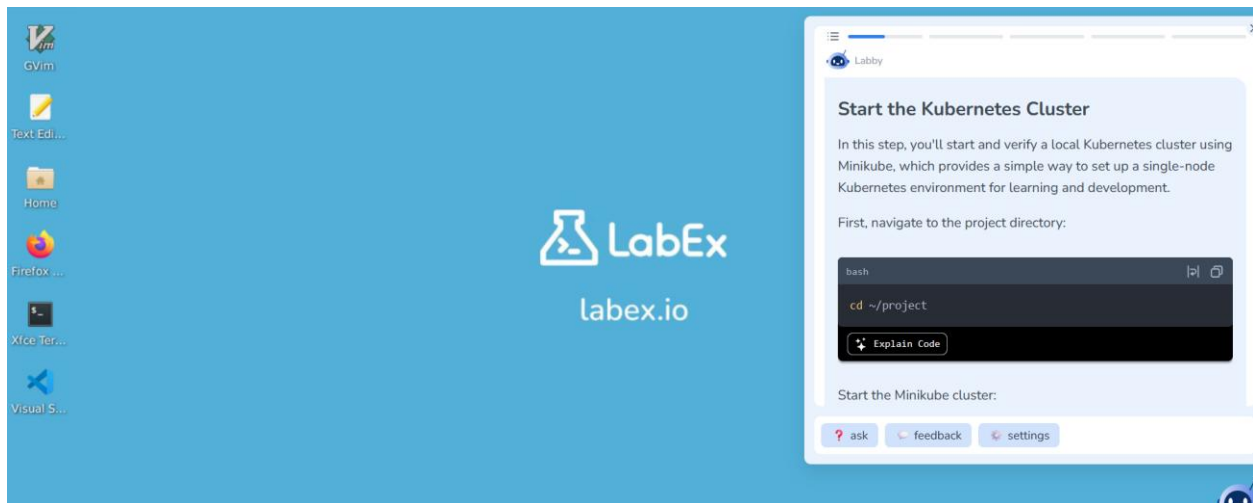
0 / 27 skills 2 courses

Joined

Start that project

Just like the dockers lab there is a VM and step by step instructions

Like before there are instructions for you to follow:



### Lab 32: Another Kubernetes Lab (recipe/experiment)

This site charges for online labs, but they also have free trials. You will go to

<https://kodekloud.com/courses/kubernetes-challenges/>

And select 'Try the Free Kubernetes Lab'



Enroll for Free

Already Subscribed? [Log in](#)

✦ Try the Free Kubernetes Lab

On the right hand side is a terminal window for you to enter commands. On the left is a menu:

Task Hint Solution AI Assistant (Beta) 56:39

Next →

1 / 13

How many pods exist on the system?  
In the current(default) namespace.

2

3

4

The lab is done by answering the questions. Note the 'hint' and 'solution' tabs. If you don't know the answer it is given to you!

### Lab 33: HackMe Kubernetes

<https://tryhackme.com/room/k8sruntimesecurity>

Task 2  Kubernetes Auditing

Task 3  Runtime Security & K8s

Task 4  Falco

Task 5  Falco in Action

### Lab 34: HackMe Intro to IaC

<https://tryhackme.com/room/introtoiac>

- Task 2  IaC - The Concept ▼
- Task 3  IaC - The Tools Part 1 ▼
- Task 4  IaC - The Tools Part 2 📄 ▼
- Task 5  Infrastructure as Code Lifecycle ▼

## Lab 35: Terraform Online Lab (recipe/experiment)

This site sells access to labs, but you will use the ‘try it for free’

<https://kodekloud.com/courses/terraform-for-beginners/>



### Get access to KodeKloud's all courses.

You'll get access to 140+ courses, 1000+ hands-on labs, and 60+ playgrounds.

Subscribe Now

Already Subscribed? [Log in](#)

✦ Try the Free Terraform Lab

Once in there is a terminal window on the right and questions on the left you will answer:

The image shows a task interface on the left and a terminal window on the right. The task interface has a 'Task' tab, a 'Hint' tab, and an 'AI Assistant (Beta)' icon. It displays a task description: 'We have a new configuration directory located at the path /root/terraform-projects/things-to-do. Inspect this directory and find out the number of providers initialized within this directory. Do not run terraform init yet!'. Below the text are five radio button options: 5 (selected), 2, 0, 3, and 4. The terminal window shows a file explorer with a directory structure: 'terraform-projects' containing 'christmas-wishlist', 'provider', 'provider-a', 'provider-b', and 'things-to-do'. The 'things-to-do' directory contains a 'README.md' file. The terminal output shows the user running 'ls' in the directory, resulting in 'main.tf'.

Notice the tab that says 'hint'

### Lab 36: Online OWAS ZAP

There is a limited online version of OWASP ZAP <https://hostedscan.com/owasp-vulnerability-scan>

The image shows a dark-themed website for 'OWASP ZAP ONLINE'. The main heading is 'Website and web application vulnerability scanner'. Below it, there is a section titled 'Run a free scan'. Two input fields are shown, each with a green checkmark to its right. The first input field contains 'www.chuckeasttom.com' and the second contains 'chuckeasttom@gmail.com'.

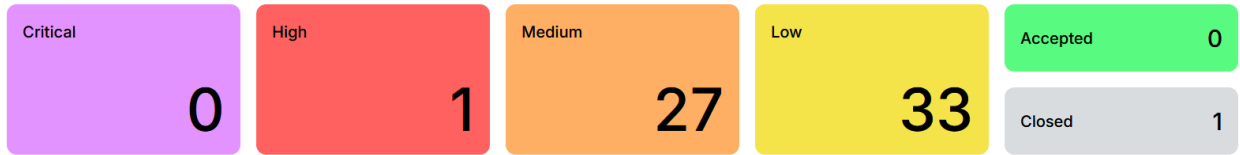
Results look like this

# Dashboard

2 scans in progress | 0 scheduled scans

+ New Scan

## Risks detected Total: 62



### Recent Scans

[See all scans >](#)

OPENVAS www.chuckeasttom.com 0 minutes ago	0%
OWASP ZAP www.chuckeasttom.com 0 minutes ago	Queued
NMAP www.chuckeasttom.com 0 minutes ago	Report

### Recent Risks

[See all risks >](#)

NMAP Open TCP Port: 3306 www.chuckeasttom.com
NMAP Open TCP Port: 21 www.chuckeasttom.com
NMAP Open TCP Port: 465 www.chuckeasttom.com