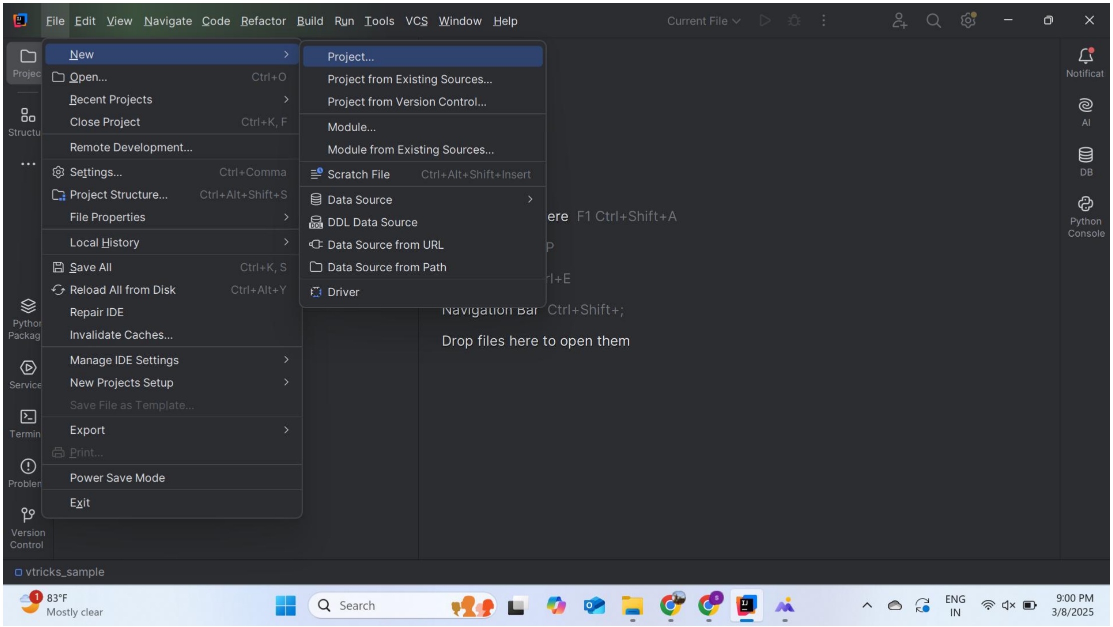
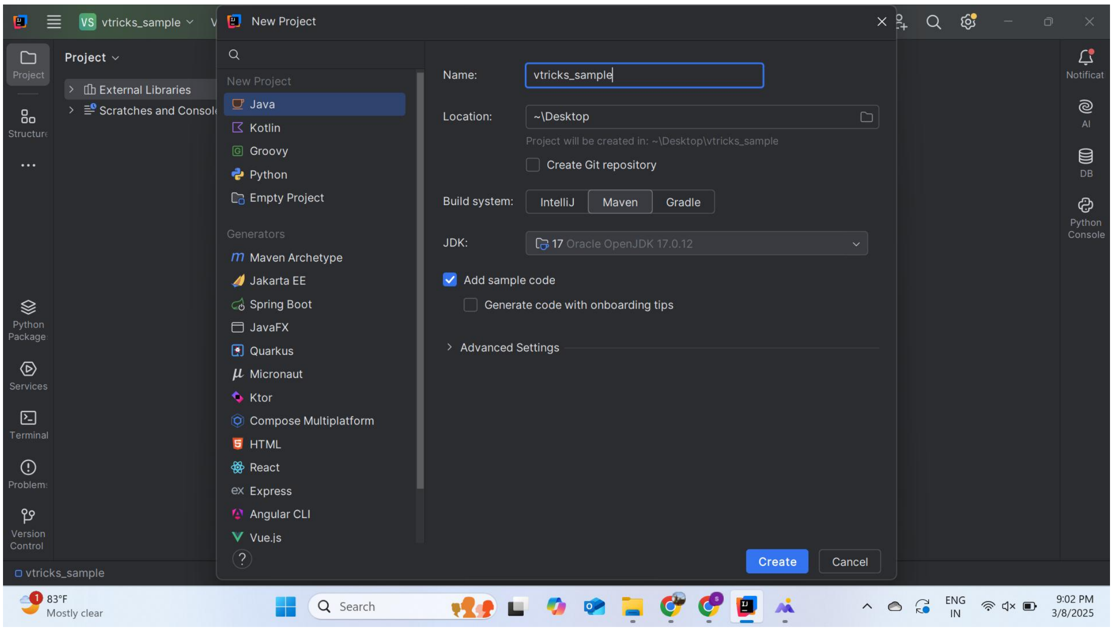
Experiment 10 : Creating Build Pipelines: Building a Maven/Gradle Project with Azure Pipelines, Integrating Code Repositories (e.g., GitHub, Azure Repos), Running Unit Tests and Generating Reports

In intellij, open new project.



Provide the project name and choose build system as maven



Name: vtricks_sample

Location: ~\Desktop

Project will be created in: ~\Desktop\vtricks_sample

□ Create Git repository

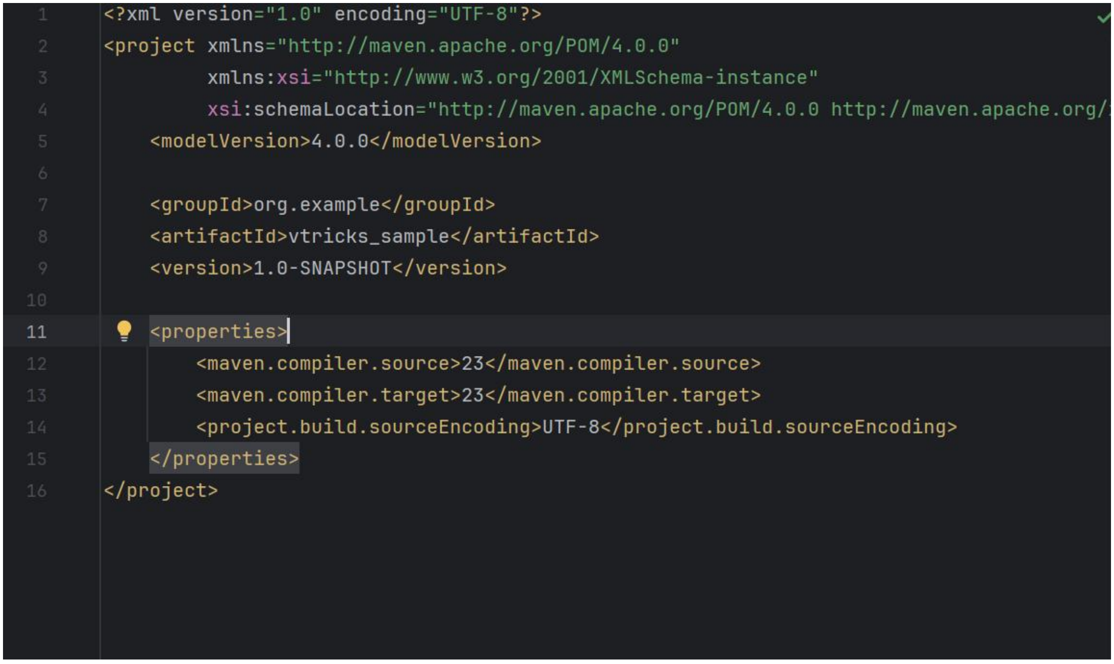
Build system: IntelliJ [✓ Maven] Gradle

JDK: 17 Oracle OpenJDK 17.0.12

☑ Add sample code

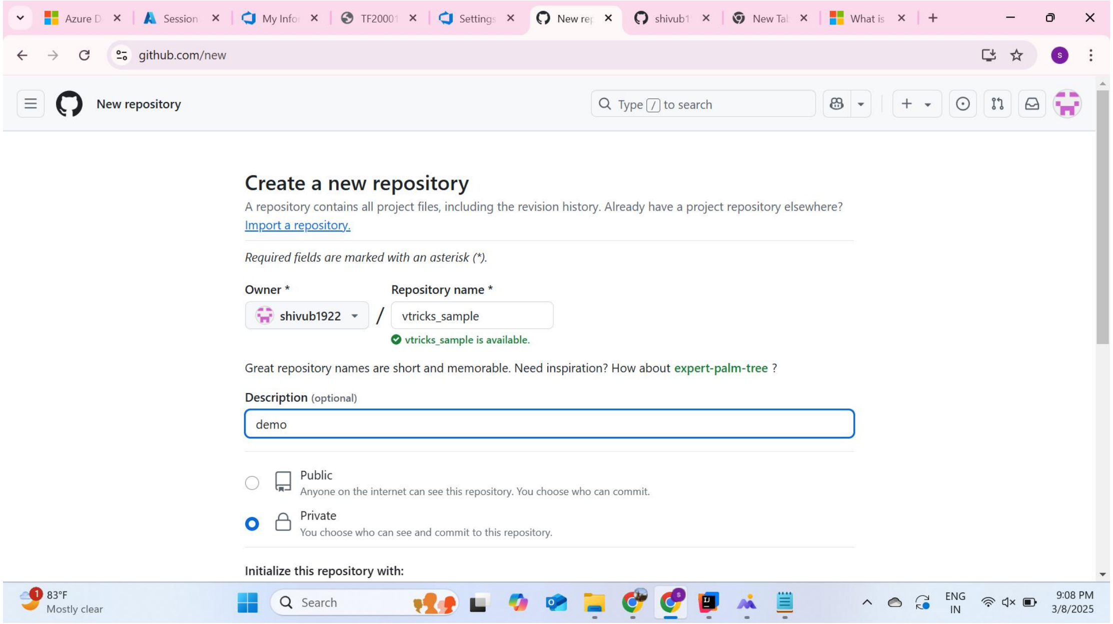
□ Generate code with onboarding tips

You will be getting a default pom.xm l and main.java file



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

    <groupId>org.example</groupId>
    <artifactId>vtricks_sample</artifactId>
    <version>1.0-SNAPSHOT</version>

    <properties>
        <maven.compiler.source>23</maven.compiler.source>
        <maven.compiler.target>23</maven.compiler.target>
        <project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>
    </properties>
</project>
```

After that create a git repo with the same name as of your project name



Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? Import a repository.

Required fields are marked with an asterisk (*).

Owner *: shivub1922 / Repository name *: vtricks_sample

vtricks_sample is available.

Great repository names are short and memorable. Need inspiration? How about expert-palm-tree ?

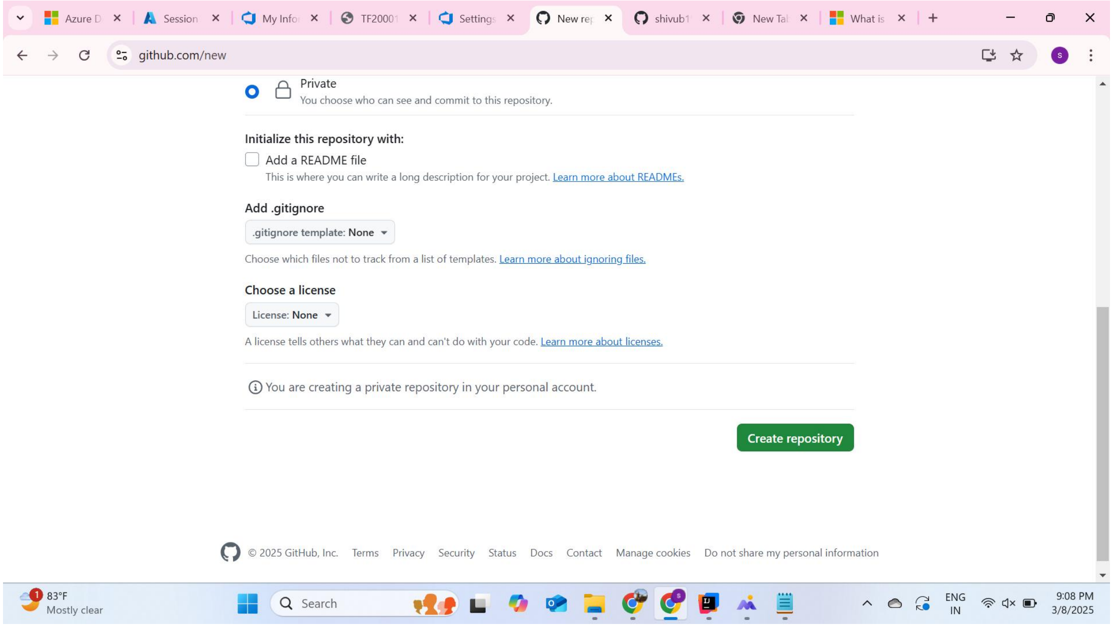
Description (optional): demo

○ Public
Anyone on the internet can see this repository. You choose who can commit.

● Private
You choose who can see and commit to this repository.

Initialize this repository with:

Create the repo click on the Create repository



● Private
You choose who can see and commit to this repository.

Initialize this repository with:

□ Add a README file
This is where you can write a long description for your project. Learn more about READMEs.

Add .gitignore: .gitignore template: None

Choose which files not to track from a list of templates. Learn more about ignoring files.

Choose a license: License: None

A license tells others what they can and can't do with your code. Learn more about licenses.

You are creating a private repository in your personal account.

© 2025 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

After creating the git repository, you need to give some commands.

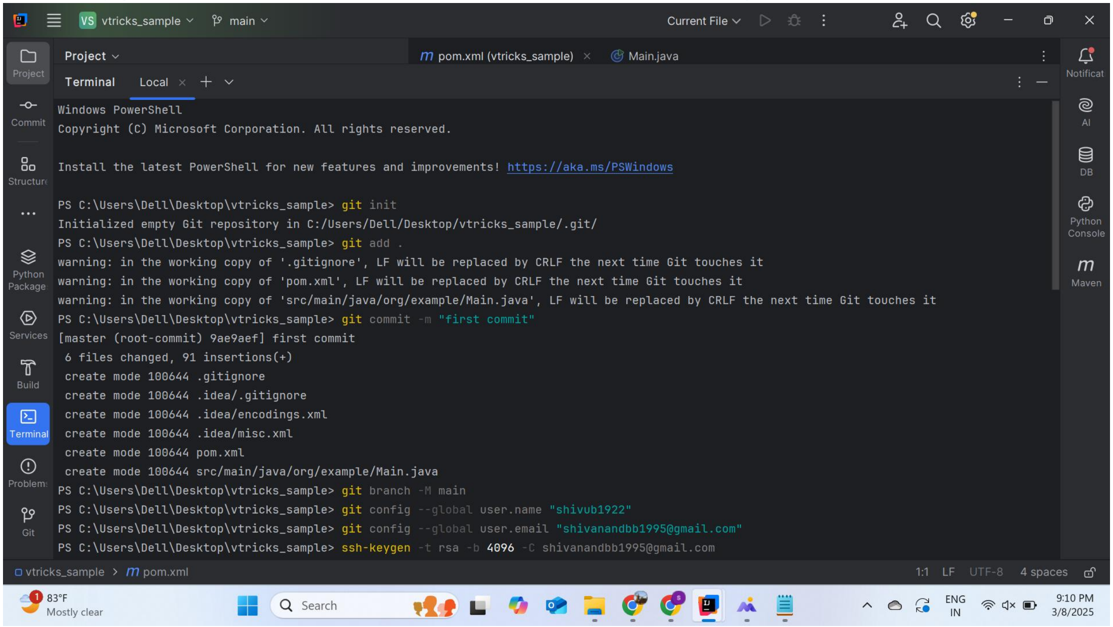
git init --to initialize the git

git add . -- add all files to the git

git commit -m “first commit” -- to commit the added files

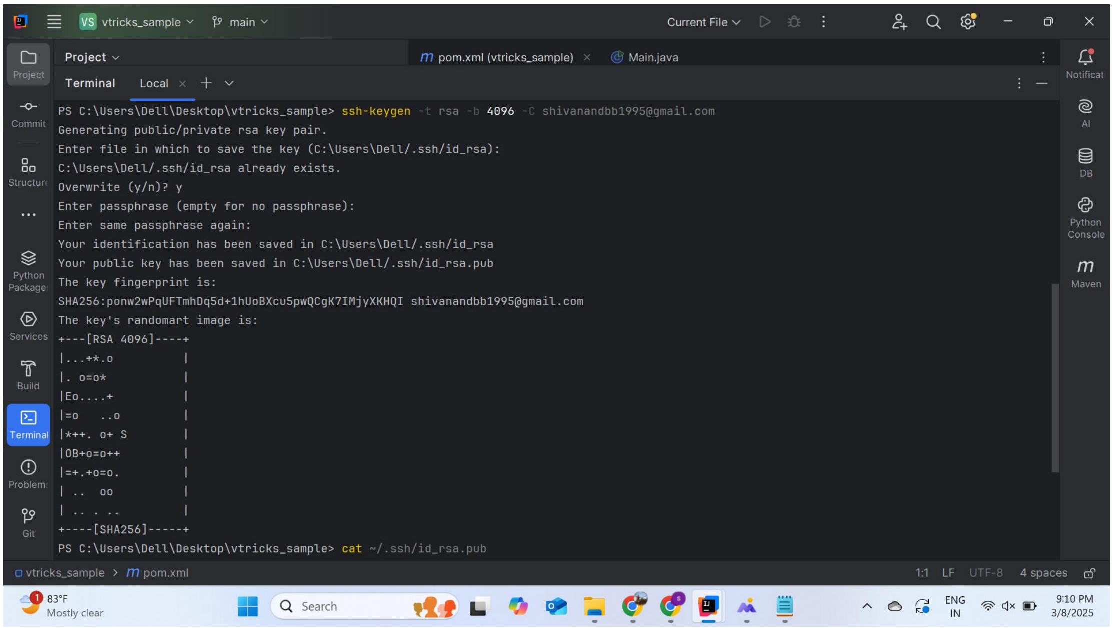
git branch -M main -- to switchto branch main

git config --global user.name “username”   -- to configure your git account using
git config --global user.email “email-id”
ssh-keygen -t rsa -b 4096 -C <email-id>   --to generate the ssh key



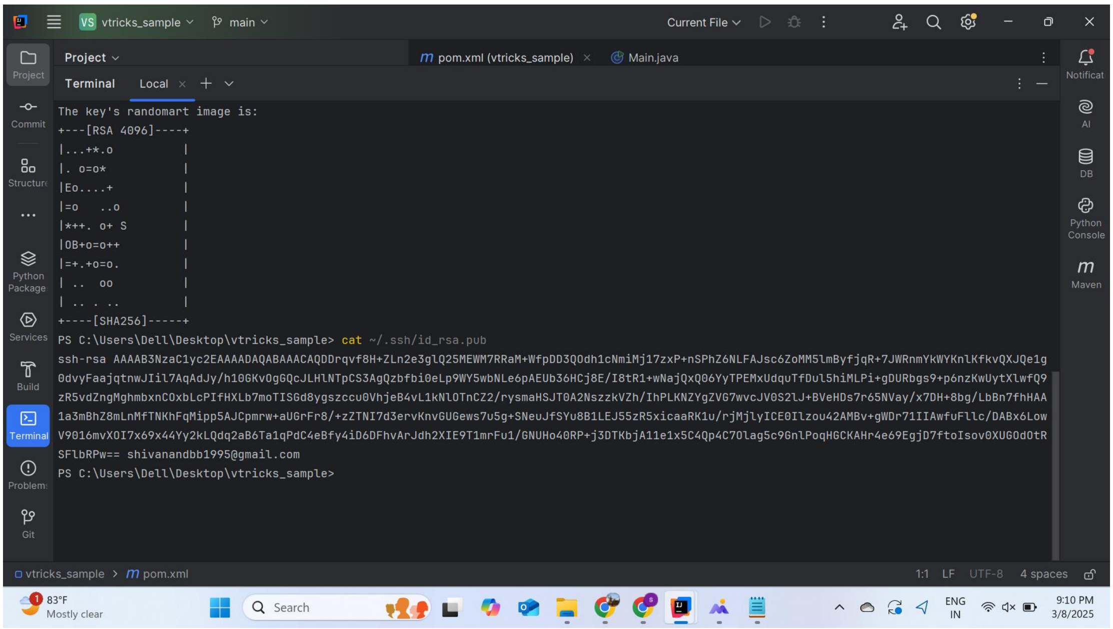
```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Dell\Desktop\vtricks_sample> git init
Initialized empty Git repository in C:/Users/Dell/Desktop/vtricks_sample/.git/
PS C:\Users\Dell\Desktop\vtricks_sample> git add .
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'pom.xml', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'src/main/java/org/example/Main.java', LF will be replaced by CRLF the next time Git touches it
PS C:\Users\Dell\Desktop\vtricks_sample> git commit -m "first commit"
[master (root-commit) 9ae9aef] first commit
 6 files changed, 91 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 .idea/.gitignore
 create mode 100644 .idea/encodings.xml
 create mode 100644 .idea/misc.xml
 create mode 100644 pom.xml
 create mode 100644 src/main/java/org/example/Main.java
PS C:\Users\Dell\Desktop\vtricks_sample> git branch -M main
PS C:\Users\Dell\Desktop\vtricks_sample> git config --global user.name "shivub1922"
PS C:\Users\Dell\Desktop\vtricks_sample> git config --global user.email "shivanandbb1995@gmail.com"
PS C:\Users\Dell\Desktop\vtricks_sample> ssh-keygen -t rsa -b 4096 -C shivanandbb1995@gmail.com
```

press ENTER for all questions



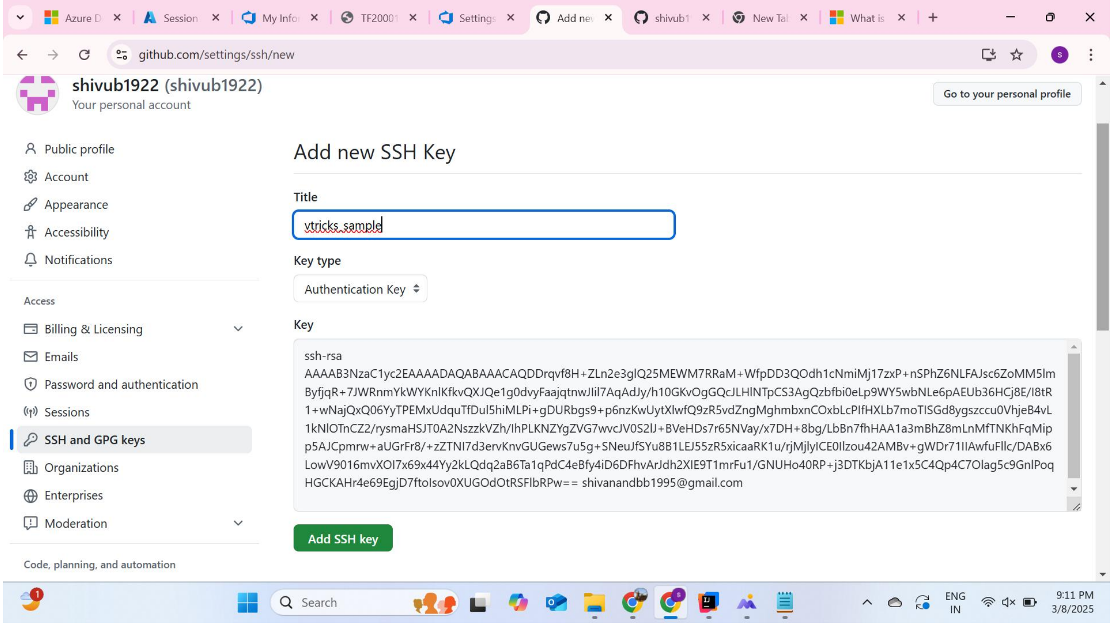
```
PS C:\Users\Dell\Desktop\vtricks_sample> ssh-keygen -t rsa -b 4096 -C shivanandbb1995@gmail.com
Generating public/private rsa key pair.
Enter file in which to save the key (C:\Users\Dell/.ssh/id_rsa):
C:\Users\Dell/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in C:\Users\Dell/.ssh/id_rsa
Your public key has been saved in C:\Users\Dell/.ssh/id_rsa.pub
The key fingerprint is:
SHA256:ponw2wPqUFTmhDq5d+1hUoBXcu5pwQCgK7IMjyXKHQI shivanandbb1995@gmail.com
The key's randomart image is:
+---[RSA 4096]----+
|...+*.o          |
|. o=o*           |
|Eo....+          |
|=o   ..o         |
|*++. o+ S        |
|OB+o=o++         |
|=+.+o=o.         |
| ..   oo         |
| .. . ..         |
+----[SHA256]-----+
PS C:\Users\Dell\Desktop\vtricks_sample> cat ~/.ssh/id_rsa.pub
```

$ cat ~/.ssh/id_rsa.pub (copy the printed SSH key)



```
The key's randomart image is:
+---[RSA 4096]----+
|...+*.o          |
|. o=o*           |
|Eo....+          |
|=o   ..o         |
|*++. o+ S        |
|OB+o=o++         |
|=+.+o=o.         |
| ..  oo          |
| .. . ..         |
+----[SHA256]-----+
PS C:\Users\Dell\Desktop\vtricks_sample> cat ~/.ssh/id_rsa.pub
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAACAQDDrqvf8H+ZLn2e3glQ25MEWM7RRaM+WfpDD3QOdh1cNmiMj17zxP+nSPhZ6NLFAJsc6ZoMM5lmByfjqR+7JWRnmYkWYKnlKfkvQXJQe1g
0dvyFaajqtnwJIil7AqAdJy/h10GKvOgGQcJLHlNTpCS3AgQzbfbi0eLp9WY5wbNLe6pAEUb36HCj8E/I8tR1+wNajQxQ06YyTPEMxUdquTfDul5hiMLPi+gDURbgs9+p6nzKwUytXlwfQ9
zR5vdZngMghmbxnCOxbLcPIfHXLb7moTISGd8ygszccu0VhjeB4vL1kNlOTnCZ2/rysmaHSJT0A2NszzkVZh/IhPLKNZYgZVG7wvcJV0S2lJ+BVeHDs7r65NVay/x7DH+8bg/LbBn7fhHAA
1a3mBhZ8mLnMfTNKhFqMipp5AJCpmrw+aUGrFr8/+zZTNI7d3ervKnvGUGews7u5g+SNeuJfSYu8B1LEJ55zR5xicaaRK1u/rjMjlyICE0Ilzou42AMBv+gWDr71IIAwfuFllc/DABx6Low
V9016mvXOI7x69x44Yy2kLQdq2aB6Ta1qPdC4eBfy4iD6DFhvArJdh2XIE9T1mrFu1/GNUHo40RP+j3DTKbjA11e1x5C4Qp4C7Olag5c9GnlPoqHGCKAHr4e69EgjD7ftoIsov0XUGOdOtR
SFlbRPw== shivanandbb1995@gmail.com
PS C:\Users\Dell\Desktop\vtricks_sample>
```

1. Now go to the SSH and GPG section in the GitHub settings option.
2. Click on the New SSH key button.
Choose a suitable name and paste the SSH key into the provided space.



github.com/settings/ssh/new

shivub1922 (shivub1922)
Your personal account

Add new SSH Key

Title: vtricks_sample

Key type: Authentication Key

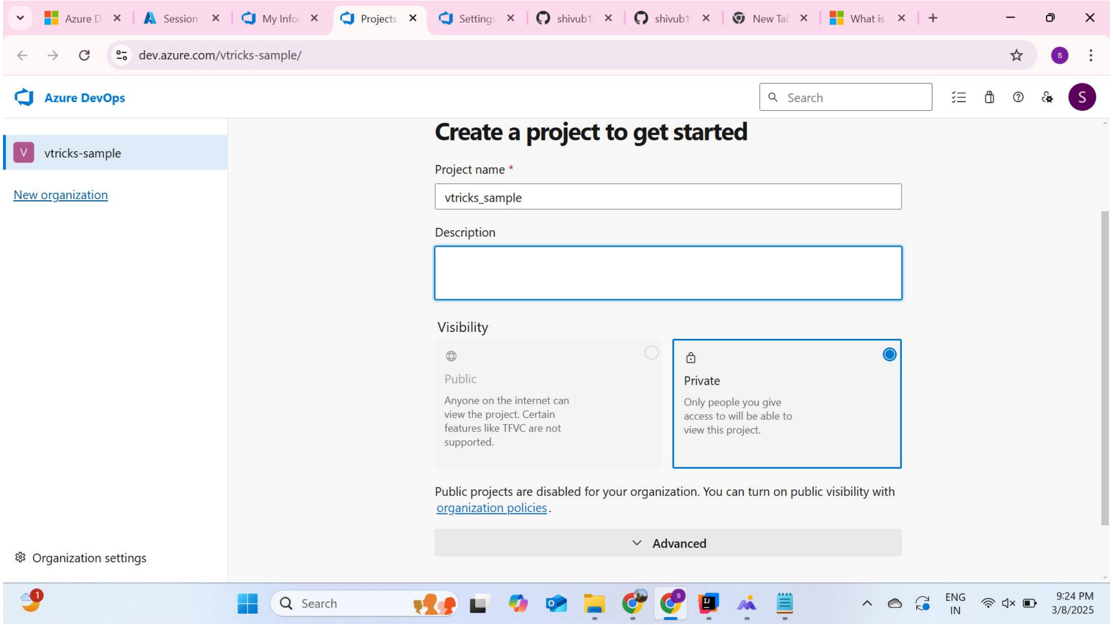
Key

```
ssh-rsa
AAAAB3NzaC1yc2EAAAADAQABAAACAQDDrqvf8H+ZLn2e3glQ25MEWM7RRaM+WfpDD3QOdh1cNmiMj17zxP+nSPhZ6NLFAJsc6ZoMM5lm
ByfjqR+7JWRnmYkWYKnlKfkvQXJQe1g0dvyFaajqtnwJIil7AqAdJy/h10GKvOgGQcJLHlNTpCS3AgQzbfbi0eLp9WY5wbNLe6pAEUb36HCj8E/I8tR
1+wNajQxQ06YyTPEMxUdquTfDul5hiMLPi+gDURbgs9+p6nzKwUytXlwfQ9zR5vdZngMghmbxnCOxbLcPIfHXLb7moTISGd8ygszccu0VhjeB4vL
1kNlOTnCZ2/rysmaHSJT0A2NszzkVZh/IhPLKNZYgZVG7wvcJV0S2lJ+BVeHDs7r65NVay/x7DH+8bg/LbBn7fhHAA1a3mBhZ8mLnMfTNKhFqMip
p5AJCpmrw+aUGrFr8/+zZTNI7d3ervKnvGUGews7u5g+SNeuJfSYu8B1LEJ55zR5xicaaRK1u/rjMjlyICE0Ilzou42AMBv+gWDr71IIAwfuFllc/DABx6
LowV9016mvXOI7x69x44Yy2kLQdq2aB6Ta1qPdC4eBfy4iD6DFhvArJdh2XIE9T1mrFu1/GNUHo40RP+j3DTKbjA11e1x5C4Qp4C7Olag5c9GnlPoq
HGCKAHr4e69EgjD7ftoIsov0XUGOdOtRSFlbRPw== shivanandbb1995@gmail.com
```

3. Now get back to the IDE's terminal and type the following command
4. git remote set-url origin <ur.git repo url>
5. git remote -v
6. git push --set-upstream origin main

7. git push

Open the azure devops and click on the create project, provide the project name



Create a project to get started

Project name *: vtricks_sample

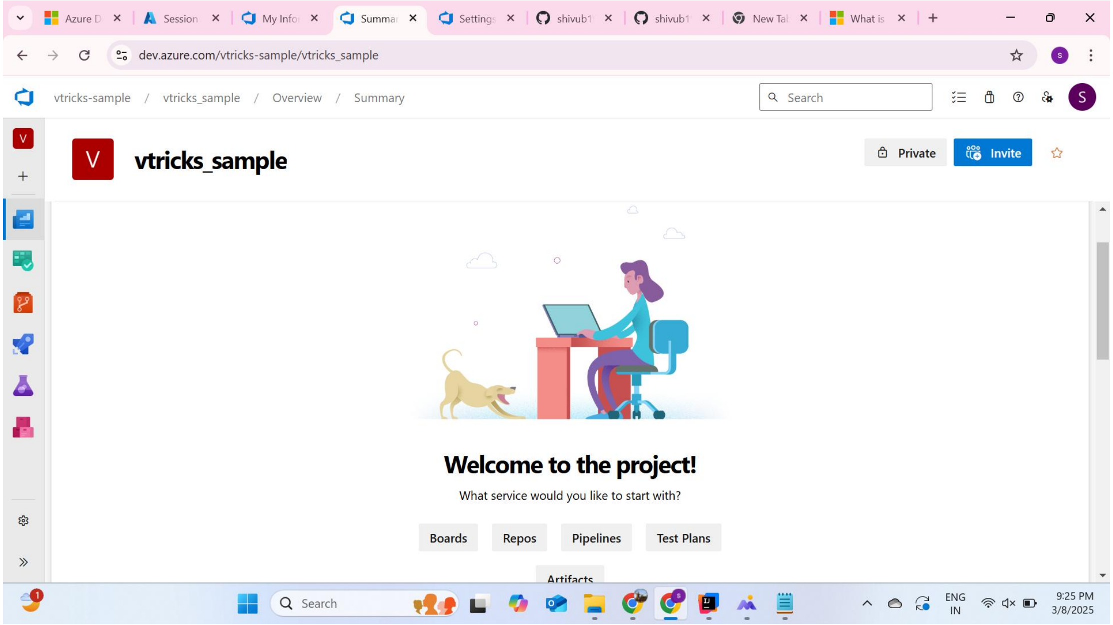
Description:

Visibility: ○ Public ● Private

Public projects are disabled for your organization. You can turn on public visibility with organization policies.

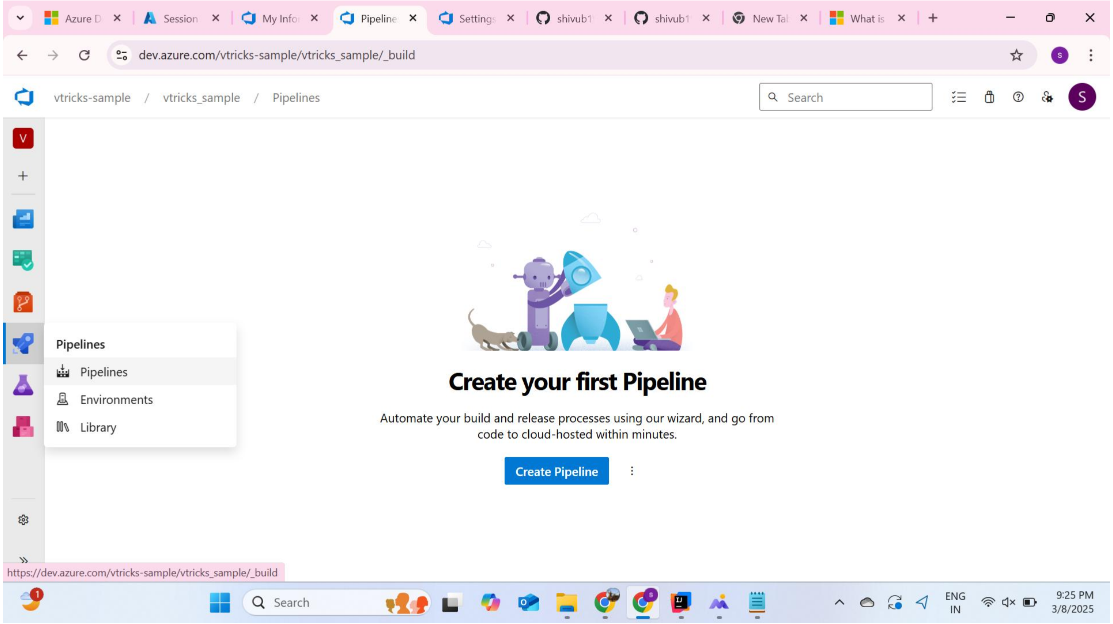
dev.azure.com/vtricks-sample/
Azure DevOps
vtricks-sample
New organization
Public
Anyone on the internet can view the project. Certain features like TFVC are not supported.
Private
Only people you give access to will be able to view this project.
Advanced
Organization settings

You will be logged in to the project



dev.azure.com/vtricks-sample/vtricks_sample
vtricks-sample / vtricks_sample / Overview / Summary
vtricks_sample
Private
Invite
Welcome to the project!
What service would you like to start with?
Boards
Repos
Pipelines
Test Plans
Artifacts

Click on the pipelines

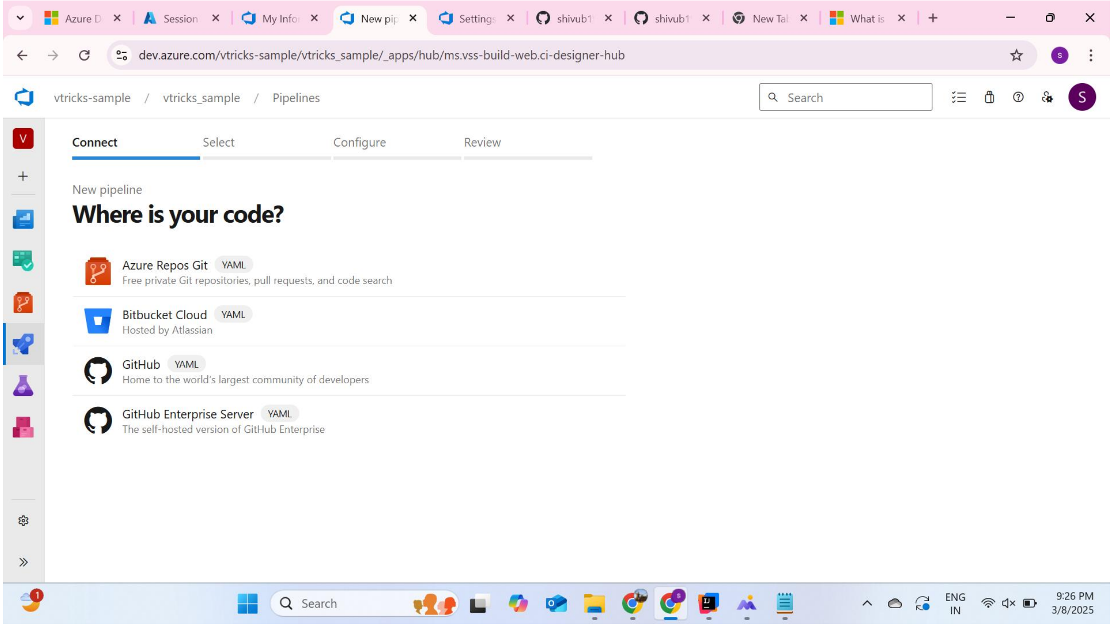


dev.azure.com/vtricks-sample/vtricks_sample/_build

Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

Choose github YAML file



dev.azure.com/vtricks-sample/vtricks_sample/_apps/hub/ms.vss-build-web.ci-designer-hub

New pipeline

Where is your code?

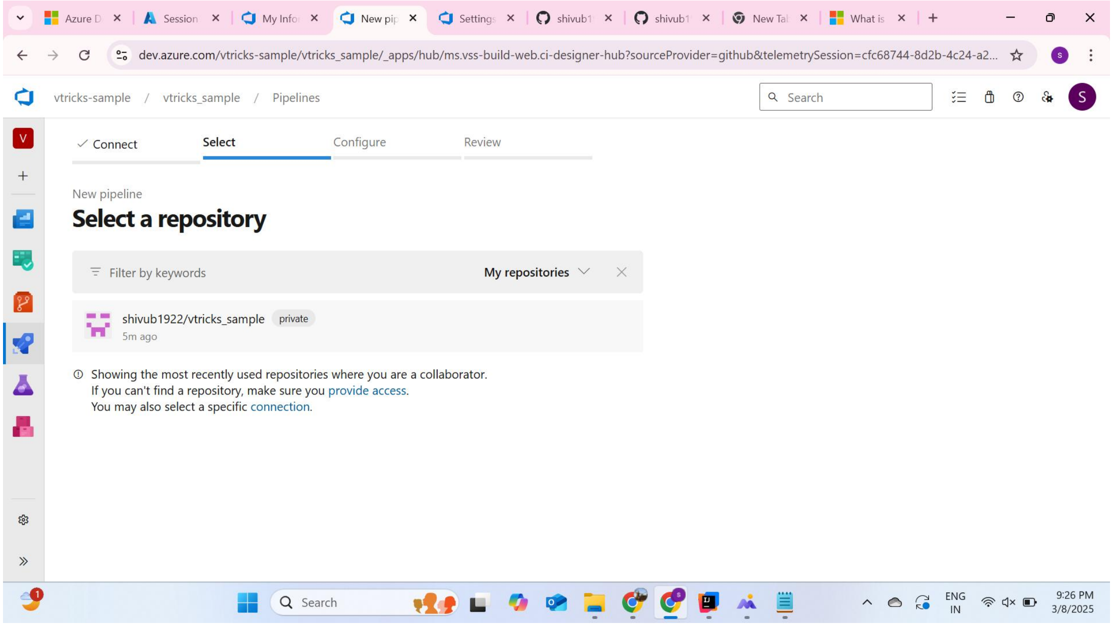
Azure Repos Git — Free private Git repositories, pull requests, and code search

Bitbucket Cloud — Hosted by Atlassian

GitHub — Home to the world's largest community of developers

GitHub Enterprise Server — The self-hosted version of GitHub Enterprise

Then you will be prompted to your repo in the github, choose the necessary git repo



dev.azure.com/vtricks-sample/vtricks_sample/_apps/hub/ms.vss-build-web.ci-designer-hub?sourceProvider=github&telemetrySession=cfc68744-8d2b-4c24-a2...

vtricks-sample / vtricks_sample / Pipelines

✓ Connect | Select | Configure | Review

New pipeline

Select a repository

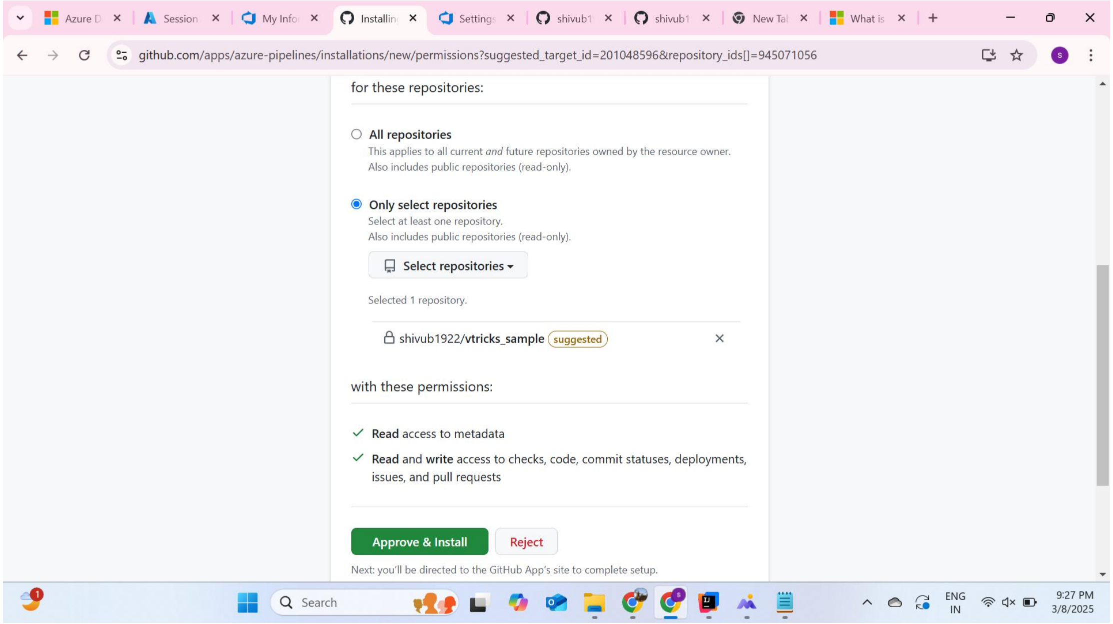
Filter by keywords | My repositories

shivub1922/vtricks_sample private
5m ago

Showing the most recently used repositories where you are a collaborator.
If you can't find a repository, make sure you provide access.
You may also select a specific connection.

9:26 PM 3/8/2025

Click on approve and install



github.com/apps/azure-pipelines/installations/new/permissions?suggested_target_id=201048596&repository_ids[]=945071056

for these repositories:

○ **All repositories**
This applies to all current *and* future repositories owned by the resource owner.
Also includes public repositories (read-only).

● **Only select repositories**
Select at least one repository.
Also includes public repositories (read-only).

Select repositories ▾

Selected 1 repository.

shivub1922/**vtricks_sample** suggested ✕

with these permissions:

✓ **Read** access to metadata

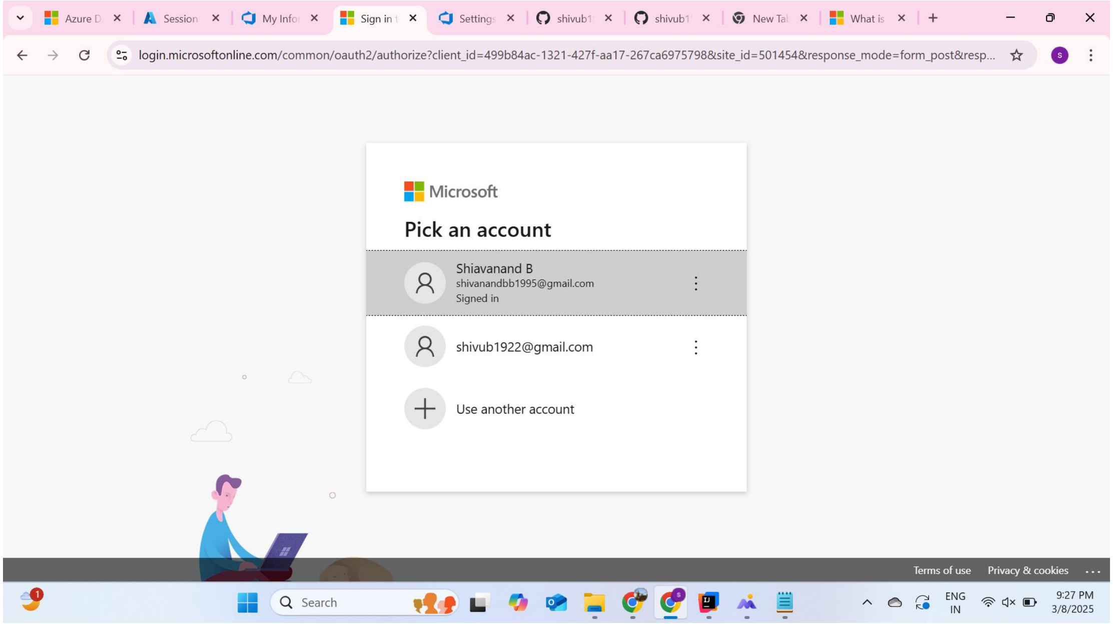
✓ **Read** and **write** access to checks, code, commit statuses, deployments, issues, and pull requests

Approve & Install | Reject

Next: you'll be directed to the GitHub App's site to complete setup.

9:27 PM 3/8/2025

Choose the azure account


login.microsoftonline.com/common/oauth2/authorize?client_id=499b84ac-1321-427f-aa17-267ca6975798&site_id=501454&response_mode=form_post&resp...

Microsoft

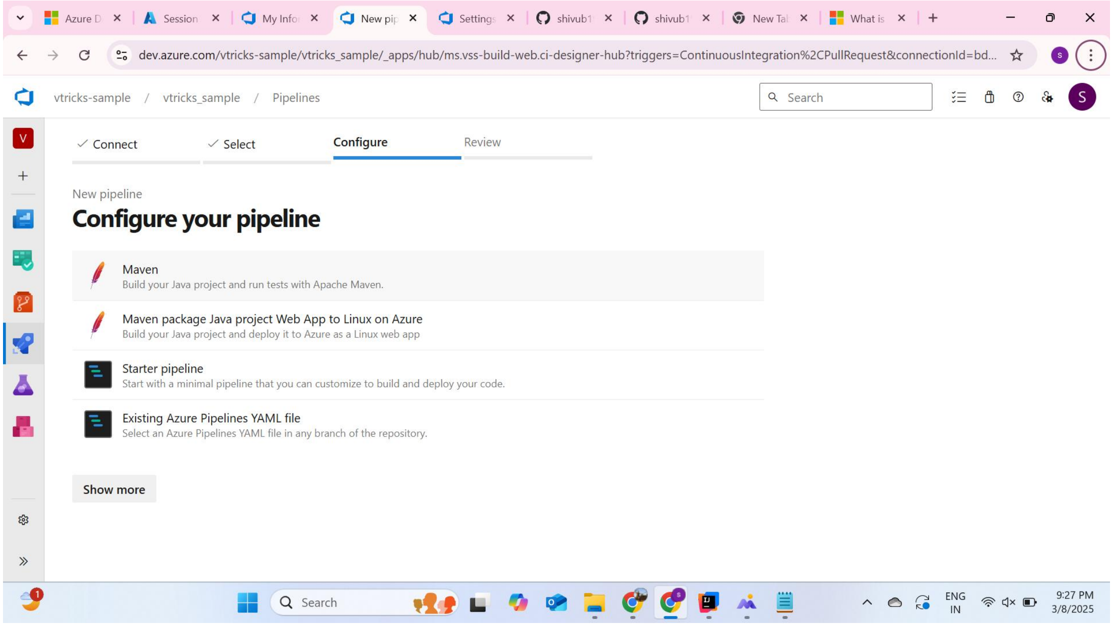
Pick an account

Shiavanand B
shivanandbb1995@gmail.com
Signed in

shivub1922@gmail.com

Use another account

Choos ethe build system as maven


dev.azure.com/vtricks-sample/vtricks_sample/_apps/hub/ms.vss-build-web.ci-designer-hub?triggers=ContinuousIntegration%2CPullRequest&connectionId=bd...

vtricks-sample / vtricks_sample / Pipelines

✓ Connect ✓ Select Configure Review

New pipeline

Configure your pipeline

Maven
Build your Java project and run tests with Apache Maven.

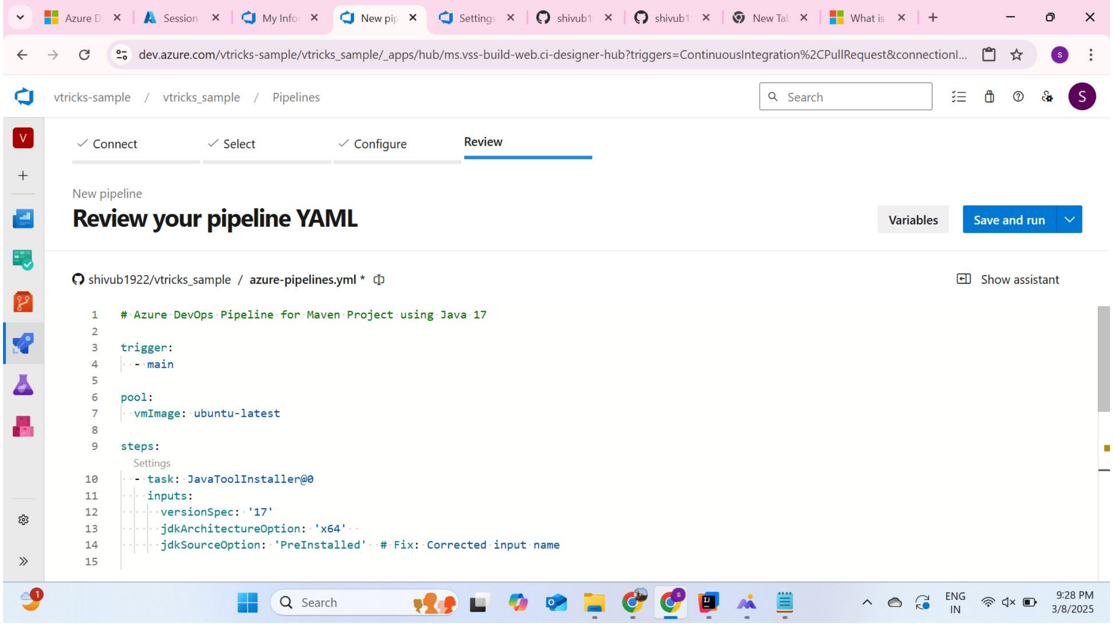
Maven package Java project Web App to Linux on Azure
Build your Java project and deploy it to Azure as a Linux web app

Starter pipeline
Start with a minimal pipeline that you can customize to build and deploy your code.

Existing Azure Pipelines YAML file
Select an Azure Pipelines YAML file in any branch of the repository.

Show more

Then it will create the azurepipeline.yaml file automatically



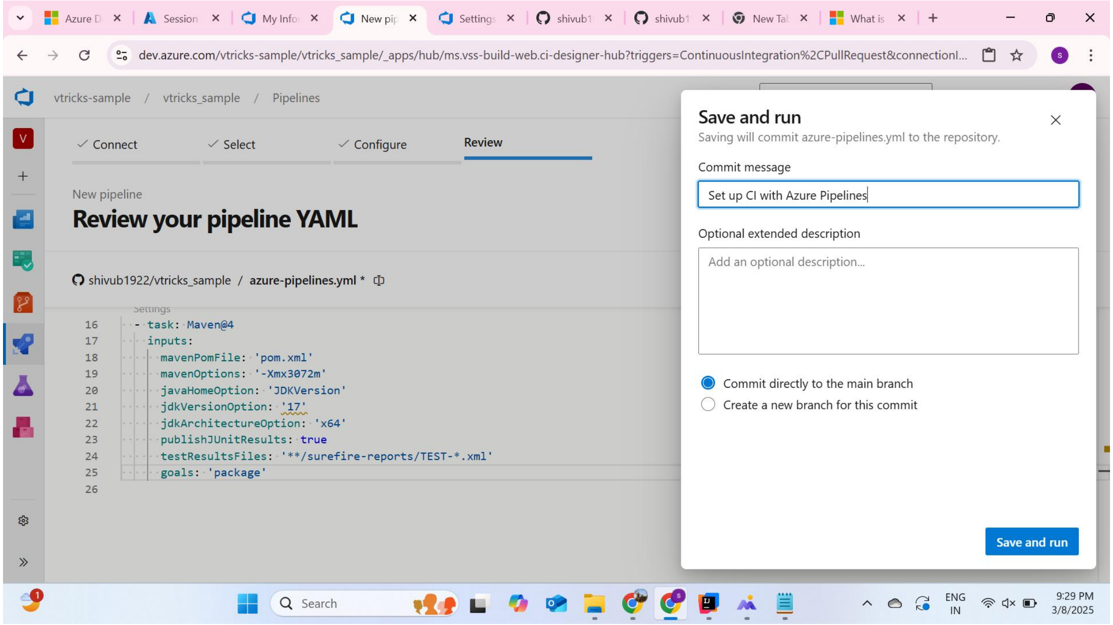
New pipeline

Review your pipeline YAML

shivub1922/vtricks_sample / azure-pipelines.yml *

```
# Azure DevOps Pipeline for Maven Project using Java 17

trigger:
  - main

pool:
  vmImage: ubuntu-latest

steps:
  - task: JavaToolInstaller@0
    inputs:
      versionSpec: '17'
      jdkArchitectureOption: 'x64'
      jdkSourceOption: 'PreInstalled'  # Fix: Corrected input name

```

Then Save and run the pipeline,
Create a commit message and press the run button again.
Click on the created Job from the Jobs table.



New pipeline

Review your pipeline YAML

shivub1922/vtricks_sample / azure-pipelines.yml *

```
  - task: Maven@4
    inputs:
      mavenPomFile: 'pom.xml'
      mavenOptions: '-Xmx3072m'
      javaHomeOption: 'JDKVersion'
      jdkVersionOption: '17'
      jdkArchitectureOption: 'x64'
      publishJUnitResults: true
      testResultsFiles: '**/surefire-reports/TEST-*.xml'
      goals: 'package'

```

Save and run

Saving will commit azure-pipelines.yml to the repository.

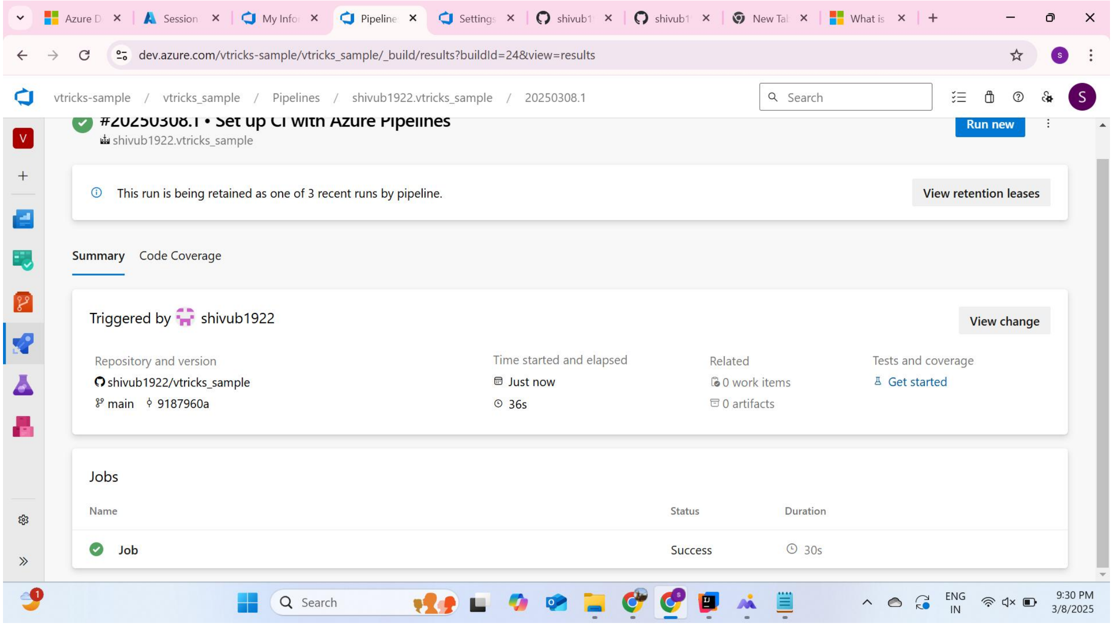
Commit message: Set up CI with Azure Pipelines

Optional extended description: Add an optional description...

● Commit directly to the main branch

○ Create a new branch for this commit

Then it will take a minute to build the pipeline,



dev.azure.com/vtricks-sample/vtricks_sample/_build/results?buildId=24&view=results

vtricks-sample / vtricks_sample / Pipelines / shivub1922.vtricks_sample / 20250308.1

#20250308.1 • Set up CI with Azure Pipelines
shivub1922.vtricks_sample

This run is being retained as one of 3 recent runs by pipeline.

Triggered by shivub1922

Repository and version: shivub1922/vtricks_sample, main, 9187960a

Time started and elapsed: Just now, 36s

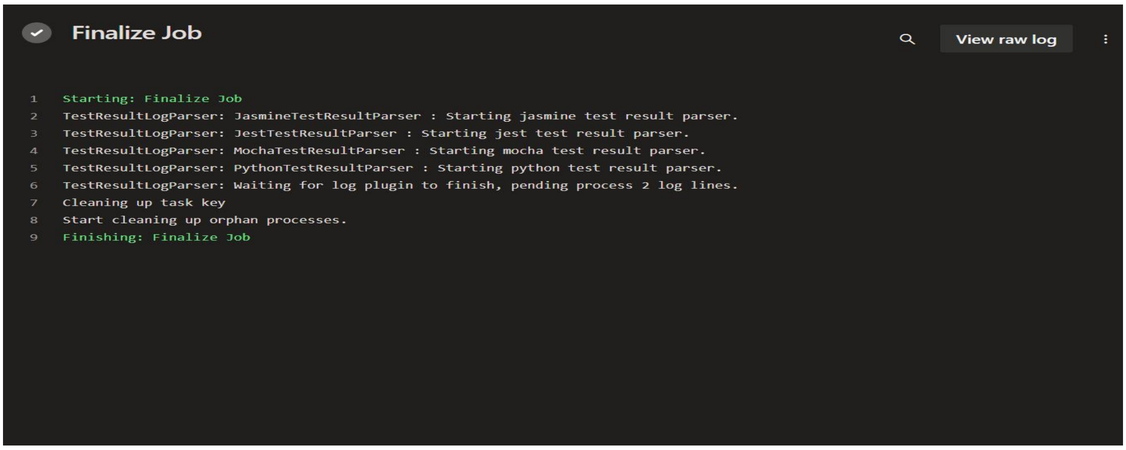
Related: 0 work items, 0 artifacts

Tests and coverage: Get started

Jobs

Name	Status	Duration
Job	Success	30s

A successful job completion would be like this.



Finalize Job

```
Starting: Finalize Job
TestResultLogParser: JasmineTestResultParser : Starting jasmine test result parser.
TestResultLogParser: JestTestResultParser : Starting jest test result parser.
TestResultLogParser: MochaTestResultParser : Starting mocha test result parser.
TestResultLogParser: PythonTestResultParser : Starting python test result parser.
TestResultLogParser: Waiting for log plugin to finish, pending process 2 log lines.
Cleaning up task key
Start cleaning up orphan processes.
Finishing: Finalize Job
```

Running Unit Tests and Generating Reports

with Maven in Azure DevOps When you build your Maven project using Azure Pipelines, the build process usually includes running unit tests with the Maven Surefire plugin. This plugin executes tests (typically written with JUnit) and produces test result files in XML format. Azure Pipelines can then pick up these XML files and present them as part of the build summary. Below are the steps and details to ensure that your unit tests are executed and the reports are published.

1. Maven Surefire Plugin and Test Reports What Happens During the Maven Build?

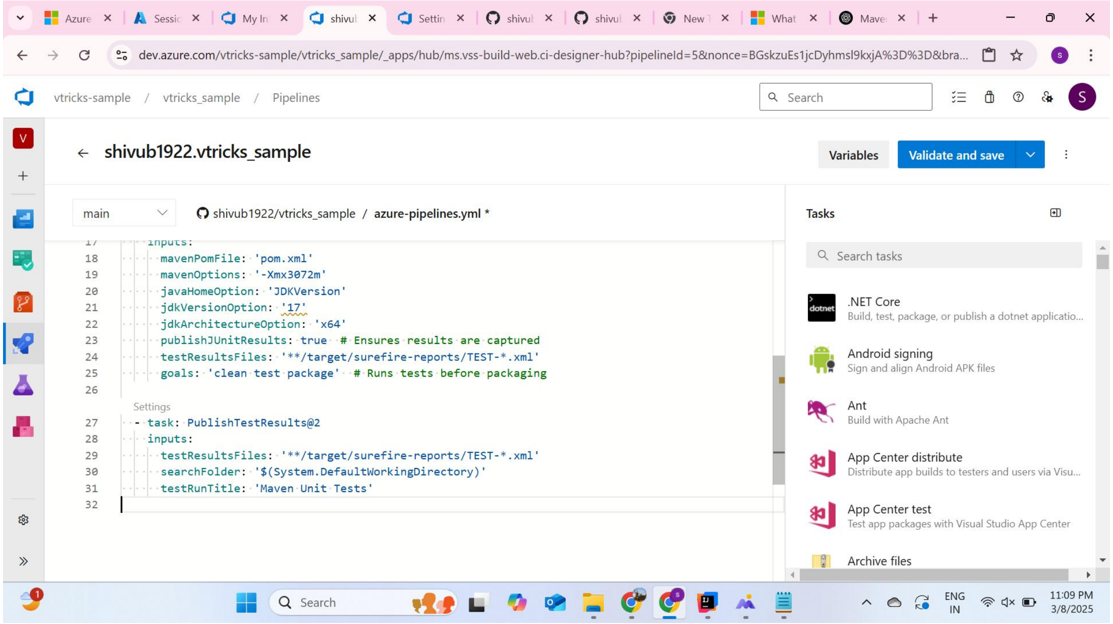
- Maven Surefire Plugin: When you run the command mvn clean package (or mvn test), the Surefire plugin automatically executes the unit tests found in the src/test/java directory.
- Test Report Generation: By default, the Surefire plugin creates XML reports in the target/surefire-reports/ directory. These files usually have names like TEST-.xml.
- Importance of Test Reports: These XML files contain detailed information on test execution, including the number of tests run, passed, failed, and any error messages or stack traces.

2. Configuring Your Azure Pipeline to Publish Test Results After your Maven build runs and tests are executed, you need to add a step in your Azure Pipeline YAML file that locates these test reports and publishes them in Azure DevOps. This is accomplished by using the **PublishTestResults@2** task.

Azurepipeline.yml file to publish the results

```
# Azure DevOps Pipeline for Maven Project using Java 17

trigger:
 - main

pool:
 vmImage: ubuntu-latest

steps:
 - task: JavaToolInstaller@0
   inputs:
     versionSpec: '17'
     jdkArchitectureOption: 'x64'
     jdkSourceOption: 'PreInstalled'

 - task: Maven@4
   inputs:
     mavenPomFile: 'pom.xml'
     mavenOptions: '-Xmx3072m'
```

```
    javaHomeOption: 'JDKVersion'
    jdkVersionOption: '17'
    jdkArchitectureOption: 'x64'
    publishJUnitResults: true  # Ensures results are captured
    testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
    goals: 'clean test package'  # Runs tests before packaging

- task: PublishTestResults@2
  inputs:
    testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
    searchFolder: '$(System.DefaultWorkingDirectory)'
    testRunTitle: 'Maven Unit Tests'
```



```
  inputs:
    mavenPomFile: 'pom.xml'
    mavenOptions: '-Xmx3072m'
    javaHomeOption: 'JDKVersion'
    jdkVersionOption: '17'
    jdkArchitectureOption: 'x64'
    publishJUnitResults: true  # Ensures results are captured
    testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
    goals: 'clean test package'  # Runs tests before packaging

- task: PublishTestResults@2
  inputs:
    testResultsFiles: '**/target/surefire-reports/TEST-*.xml'
    searchFolder: '$(System.DefaultWorkingDirectory)'
    testRunTitle: 'Maven Unit Tests'
```

dev.azure.com/vtricks-sample/vtricks_sample/_apps/hub/ms.vss-build-web.ci-designer-hub?pipelineId=5&nonce=BGskzuEs1jcDyhmsl9kxjA%3D%3D&bra...
vtricks-sample / vtricks_sample / Pipelines
Search
shivub1922.vtricks_sample
Variables
Validate and save
main
shivub1922/vtricks_sample / azure-pipelines.yml *
Settings
Tasks
Search tasks
.NET Core
Build, test, package, or publish a dotnet applicatio...
Android signing
Sign and align Android APK files
Ant
Build with Apache Ant
App Center distribute
Distribute app builds to testers and users via Visu...
App Center test
Test app packages with Visual Studio App Center
Archive files
11:09 PM
3/8/2025

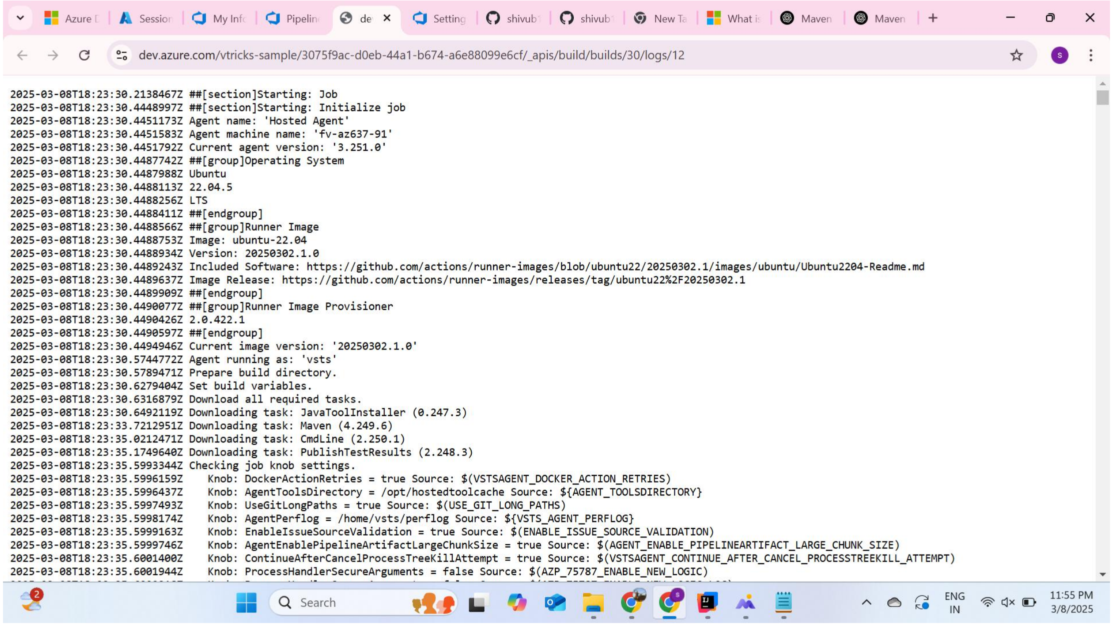
Running and Verifying the Pipeline

After committing the YAML file to your repository, the pipeline is triggered (either automatically or manually):

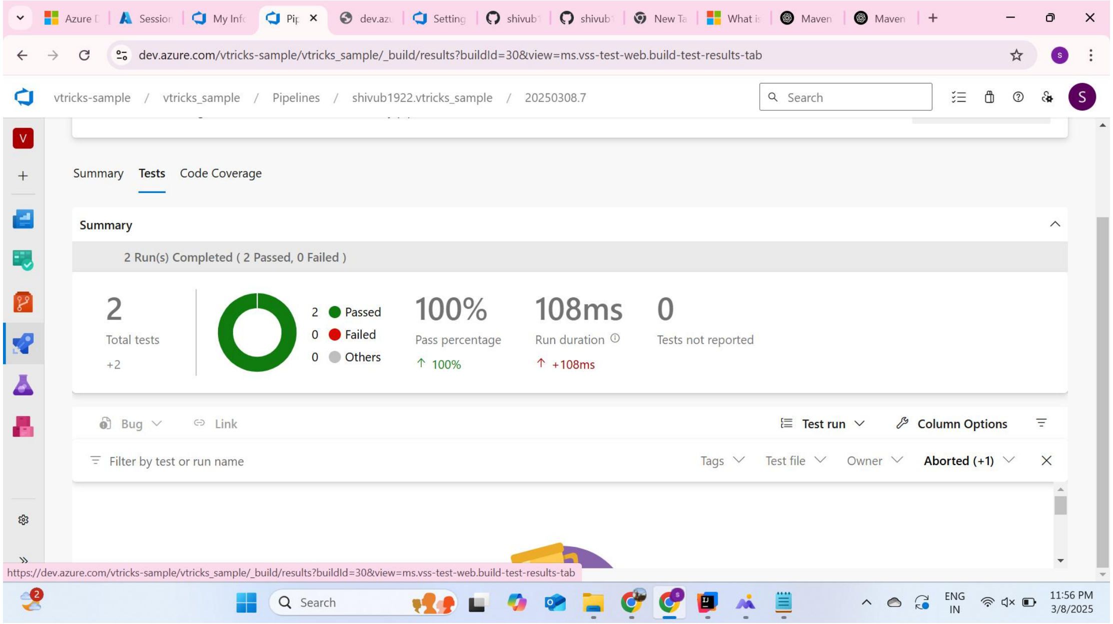
1. Trigger the Pipeline: Once your YAML file is saved in your repository, Azure Pipelines will pick up the changes. If not automatically triggered, you can click “Run pipeline” manually.

2. Monitor the Build Output: Navigate to the “Build” or “Logs” section of the pipeline run. o Confirm that the Maven task logs show the execution of tests and that the Surefire reports are created.

3. Review Test Reports: o Once the build completes, click on the “Tests” tab (often found on the pipeline summary page) to review the detailed test results.



```
2025-03-08T18:23:30.2138467Z ##[section]Starting: Job
2025-03-08T18:23:30.4448997Z ##[section]Starting: Initialize job
2025-03-08T18:23:30.4451173Z Agent name: 'Hosted Agent'
2025-03-08T18:23:30.4451583Z Agent machine name: 'fv-az637-91'
2025-03-08T18:23:30.4451792Z Current agent version: '3.251.0'
2025-03-08T18:23:30.4487742Z ##[group]Operating System
2025-03-08T18:23:30.4487988Z Ubuntu
2025-03-08T18:23:30.4488113Z 22.04.5
2025-03-08T18:23:30.4488256Z LTS
2025-03-08T18:23:30.4488411Z ##[endgroup]
2025-03-08T18:23:30.4488566Z ##[group]Runner Image
2025-03-08T18:23:30.4488753Z Image: ubuntu-22.04
2025-03-08T18:23:30.4488934Z Version: 20250302.1.0
2025-03-08T18:23:30.4489243Z Included Software: https://github.com/actions/runner-images/blob/ubuntu22/20250302.1/images/ubuntu/Ubuntu2204-Readme.md
2025-03-08T18:23:30.4489637Z Image Release: https://github.com/actions/runner-images/releases/tag/ubuntu22%2F20250302.1
2025-03-08T18:23:30.4489909Z ##[endgroup]
2025-03-08T18:23:30.4490077Z ##[group]Runner Image Provisioner
2025-03-08T18:23:30.4490426Z 2.0.422.1
2025-03-08T18:23:30.4490597Z ##[endgroup]
2025-03-08T18:23:30.4494946Z Current image version: '20250302.1.0'
2025-03-08T18:23:30.5744772Z Agent running as: 'vsts'
2025-03-08T18:23:30.5789471Z Prepare build directory.
2025-03-08T18:23:30.6279404Z Set build variables.
2025-03-08T18:23:30.6316879Z Download all required tasks.
2025-03-08T18:23:30.6492119Z Downloading task: JavaToolInstaller (0.247.3)
2025-03-08T18:23:33.7212951Z Downloading task: Maven (4.249.6)
2025-03-08T18:23:35.0212471Z Downloading task: CmdLine (2.250.1)
2025-03-08T18:23:35.1749640Z Downloading task: PublishTestResults (2.248.3)
2025-03-08T18:23:35.5993344Z Checking job knob settings.
2025-03-08T18:23:35.5996159Z    Knob: DockerActionRetries = true Source: $(VSTSAGENT_DOCKER_ACTION_RETRIES)
2025-03-08T18:23:35.5996437Z    Knob: AgentToolsDirectory = /opt/hostedtoolcache Source: ${AGENT_TOOLSDIRECTORY}
2025-03-08T18:23:35.5997493Z    Knob: UseGitLongPaths = true Source: $(USE_GIT_LONG_PATHS)
2025-03-08T18:23:35.5998174Z    Knob: AgentPerflog = /home/vsts/perflog Source: ${VSTS_AGENT_PERFLOG}
2025-03-08T18:23:35.5999163Z    Knob: EnableIssueSourceValidation = true Source: $(ENABLE_ISSUE_SOURCE_VALIDATION)
2025-03-08T18:23:35.5999746Z    Knob: AgentEnablePipelineArtifactLargeChunkSize = true Source: $(AGENT_ENABLE_PIPELINEARTIFACT_LARGE_CHUNK_SIZE)
2025-03-08T18:23:35.6001400Z    Knob: ContinueAfterCancelProcessTreeKillAttempt = true Source: $(VSTSAGENT_CONTINUE_AFTER_CANCEL_PROCESSTREEKILL_ATTEMPT)
2025-03-08T18:23:35.6001944Z    Knob: ProcessHandlerSecureArguments = false Source: $(AZP_75787_ENABLE_NEW_LOGIC)
```



dev.azure.com/vtricks-sample/vtricks_sample/_build/results?buildId=30&view=ms.vss-test-web.build-test-results-tab

vtricks-sample / vtricks_sample / Pipelines / shivub1922.vtricks_sample / 20250308.7

Summary Tests Code Coverage

Summary

2 Run(s) Completed (2 Passed, 0 Failed)

2 Total tests +2	2 ● Passed 0 ● Failed 0 ● Others	100% Pass percentage ↑ 100%	108ms Run duration ↑ +108ms	0 Tests not reported

https://dev.azure.com/vtricks-sample/vtricks_sample/_build/results?buildId=30&view=ms.vss-test-web.build-test-results-tab