NAME: K. Manoj Kumar

REGISTER NUMBER: 23MIS0159

COURSE NAME: Agile Development Process and DevOps Lab

COURSE CODE: ISWE406P

SLOT: L51+L52

TASK 1: Jenkins Familiarization Objective:

Understand Jenkins UI and basic navigation

Tasks:

1. Open Jenkins Dashboard in browser

2. Identify:

o Dashboard

o Manage Jenkins

o New Item

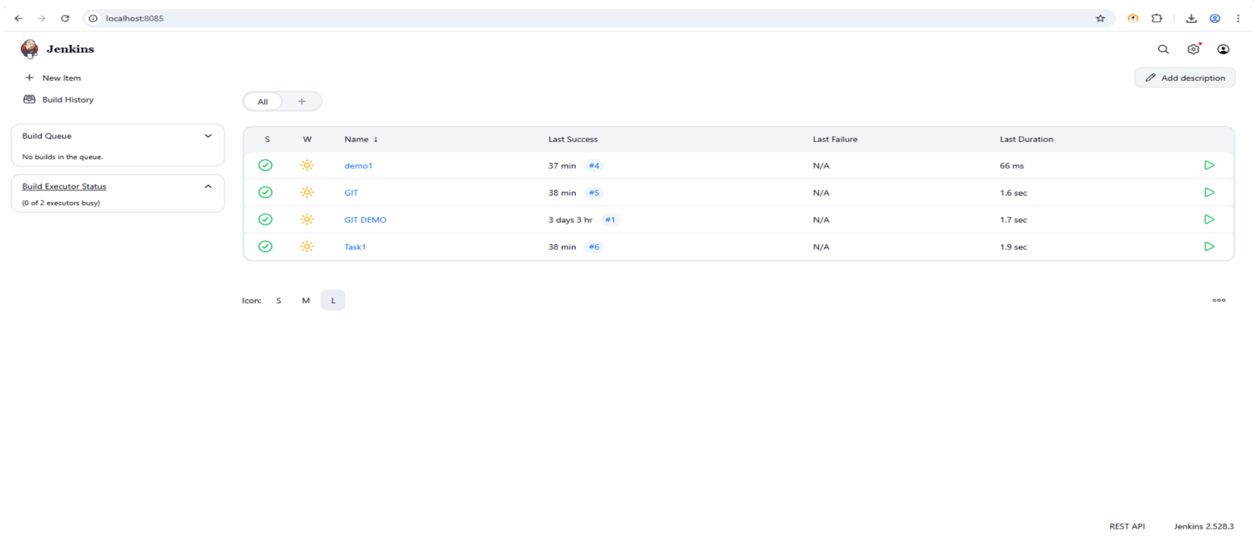
o Build History

3. Check Jenkins version

Expected Output:

Screenshot or note of Jenkins version

DASHBOARD, Jenkins Version:



Build Queue

No builds in the queue.

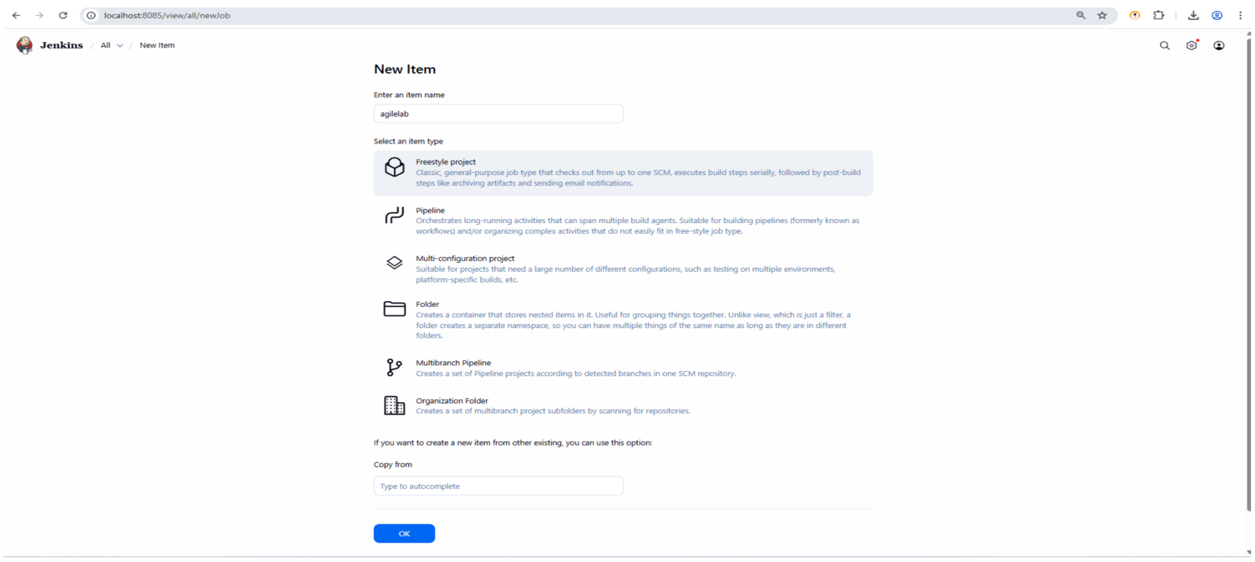
Build Executor Status

(0 of 2 executors busy)

S	W	Name ↓	Last Success	Last Failure	Last Duration
✓	☀	demo1	37 min #4	N/A	66 ms
✓	☀	GIT	38 min #5	N/A	1.6 sec
✓	☀	GIT DEMO	3 days 3 hr #1	N/A	1.7 sec
✓	☀	Task1	38 min #6	N/A	1.9 sec

REST API Jenkins 2.528.3

NEWITEMS:



localhost:8085/view/all/newJob

Jenkins / All / New Item

New Item

Enter an item name

agilelab

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.
- **Folder** — Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.
- **Multibranch Pipeline** — Creates a set of Pipeline projects according to detected branches in one SCM repository.
- **Organization Folder** — Creates a set of multibranch project subfolders by scanning for repositories.

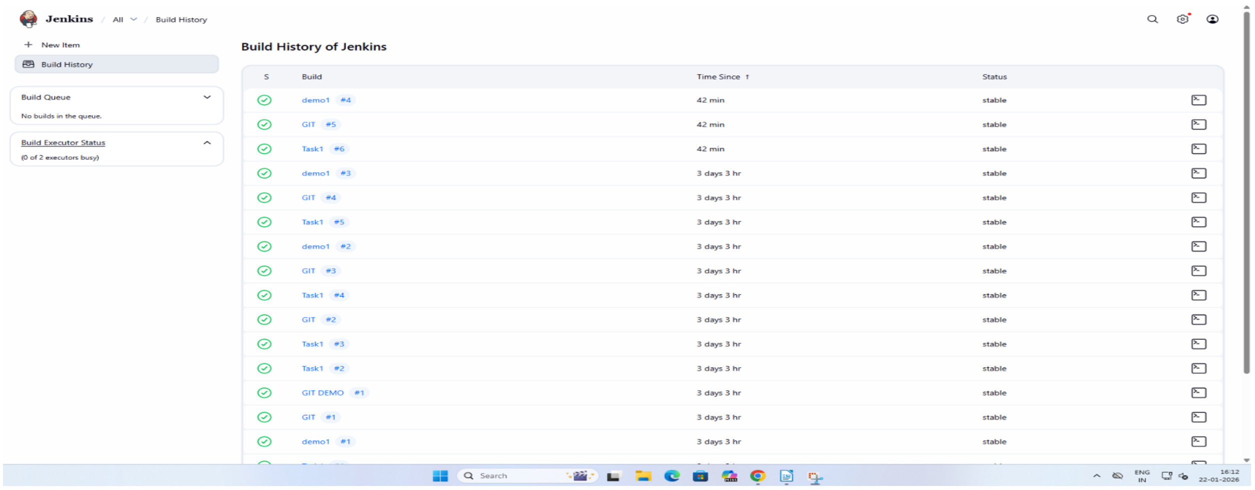
If you want to create a new item from other existing, you can use this option:

Copy from

Type to autocomplete

OK

Build History:



Jenkins / All / Build History

New Item
Build History

Build Queue
No builds in the queue.

Build Executor Status
(0 of 2 executors busy)

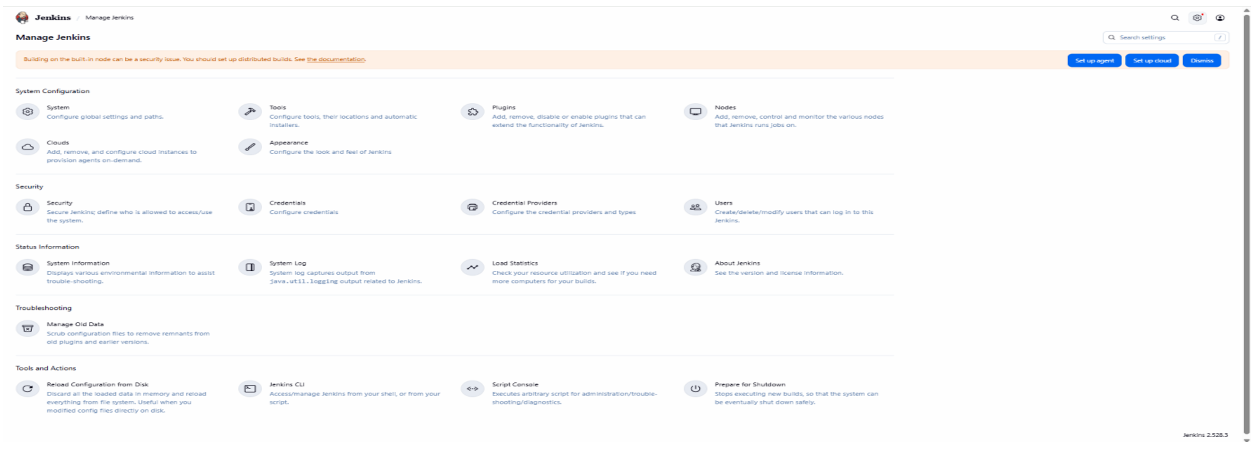
Build History of Jenkins

S	Build	Time Since ↑	Status
✓	demo1 #4	42 min	stable
✓	GIT #5	42 min	stable
✓	Task1 #6	42 min	stable
✓	demo1 #3	3 days 3 hr	stable
✓	GIT #4	3 days 3 hr	stable
✓	Task1 #5	3 days 3 hr	stable
✓	demo1 #2	3 days 3 hr	stable
✓	GIT #3	3 days 3 hr	stable
✓	Task1 #4	3 days 3 hr	stable
✓	GIT #2	3 days 3 hr	stable
✓	Task1 #3	3 days 3 hr	stable
✓	Task1 #2	3 days 3 hr	stable
✓	GIT DEMO #1	3 days 3 hr	stable
✓	GIT #1	3 days 3 hr	stable
✓	demo1 #1	3 days 3 hr	stable

Search

ENG IN 16:12 22-01-2026

MANAGE JENKINS:



Jenkins / Manage Jenkins

Manage Jenkins

Search settings

Building on the built-in node can be a security issue. You should set up distributed builds. See the documentation.

Set up agent | Set up cloud | Dismiss

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

Security

- **Security** — Secure Jenkins; define who is allowed to access/use the system.
- **Credentials** — Configure credentials
- **Credential Providers** — Configure the credential providers and types
- **Users** — Create/delete/modify users that can log in to this Jenkins.

Status Information

- **System Information** — Displays various environmental information to assist trouble-shooting.
- **System Log** — System log captures output from `java.util.logging` output related to Jenkins.
- **Load Statistics** — Check your resource utilization and see if you need more computers for your builds.
- **About Jenkins** — See the version and license information.

Troubleshooting

- **Manage Old Data** — Scrub configuration files to remove remnants from old plugins and earlier versions.

Tools and Actions

- **Reload Configuration from Disk** — Discard all the loaded data in memory and reload everything from file system. Useful when you modified config files directly on disk.
- **Jenkins CLI** — Access/manage Jenkins from your shell, or from your script.
- **Script Console** — Executes arbitrary script for administration/trouble-shooting/diagnostics.
- **Prepare for Shutdown** — Stops executing new builds, so that the system can be eventually shut down safely.

Jenkins 2.528.3

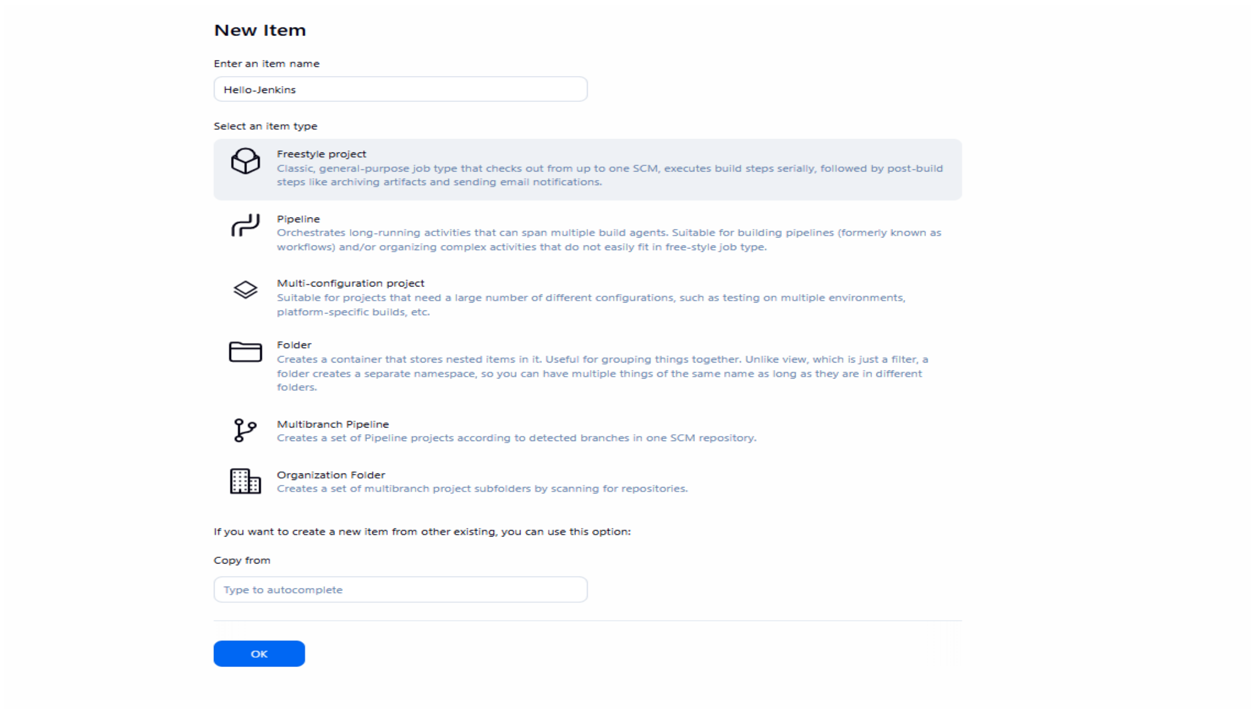
TASK 2: Create First Freestyle Job

Objective:

Create and run a Jenkins job

Tasks:

1. **Create a Freestyle project named Hello-Jenkins**



New Item

Enter an item name

Hello-Jenkins

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.

Folder
Creates a container that stores nested items in it. Useful for grouping things together. Unlike view, which is just a filter, a folder creates a separate namespace, so you can have multiple things of the same name as long as they are in different folders.

Multibranch Pipeline
Creates a set of Pipeline projects according to detected branches in one SCM repository.

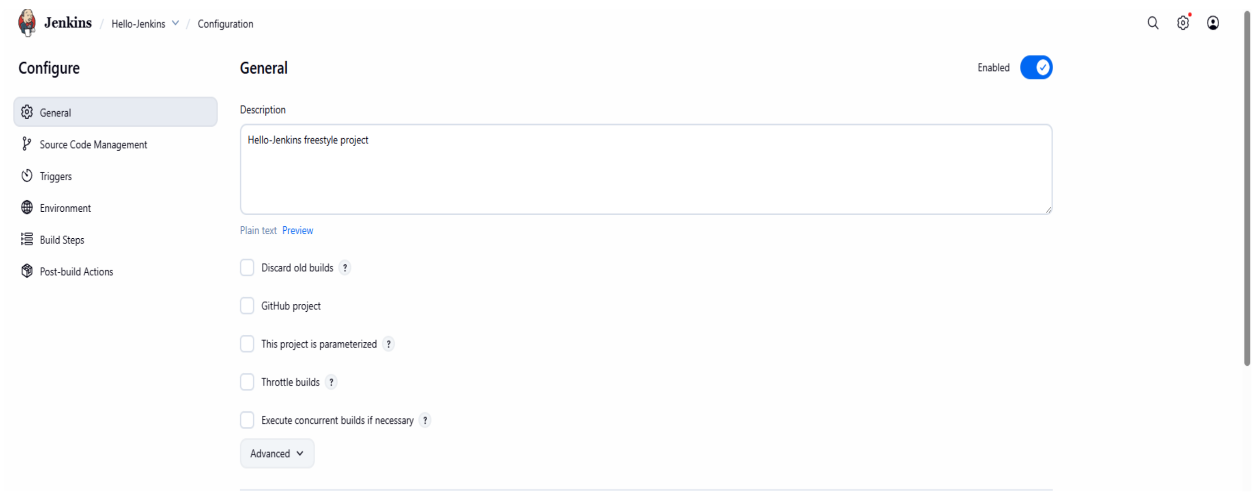
Organization Folder
Creates a set of multibranch project subfolders by scanning for repositories.

If you want to create a new item from other existing, you can use this option:

Copy from

Type to autocomplete

2. Add a description



General

Enabled

Description

Hello-Jenkins freestyle project

Discard old builds

GitHub project

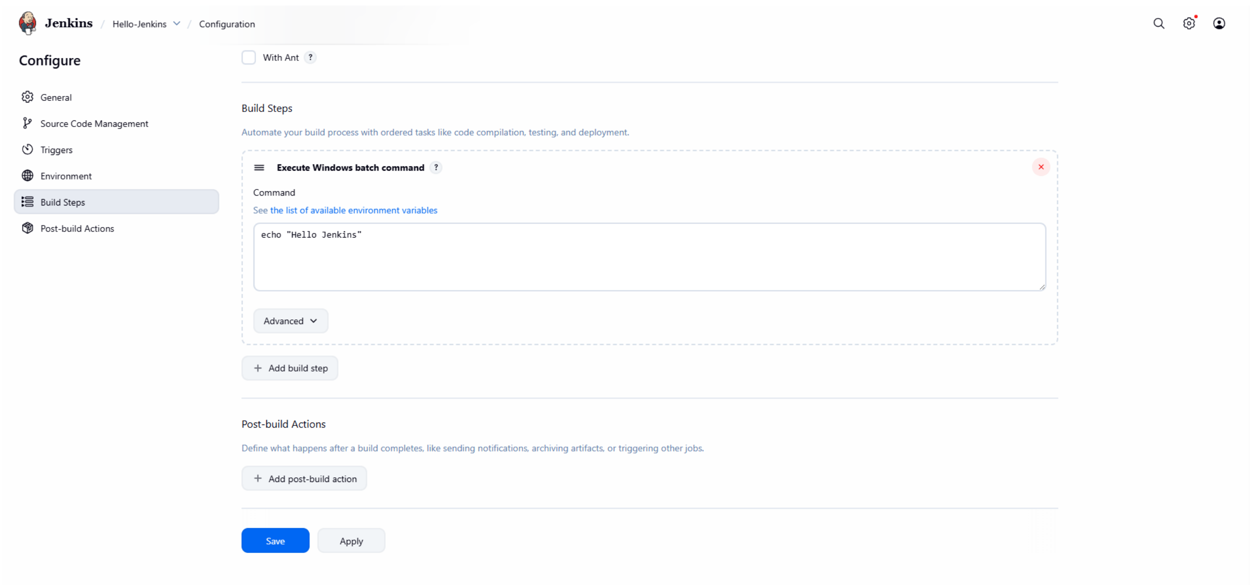
This project is parameterized

Throttle builds

Execute concurrent builds if necessary

-

3. Add build step: o Execute shell / Windows batch command o Print "Hello Jenkins"



Build Steps

Automate your build process with ordered tasks like code compilation, testing, and deployment.

Command

See the list of available environment variables

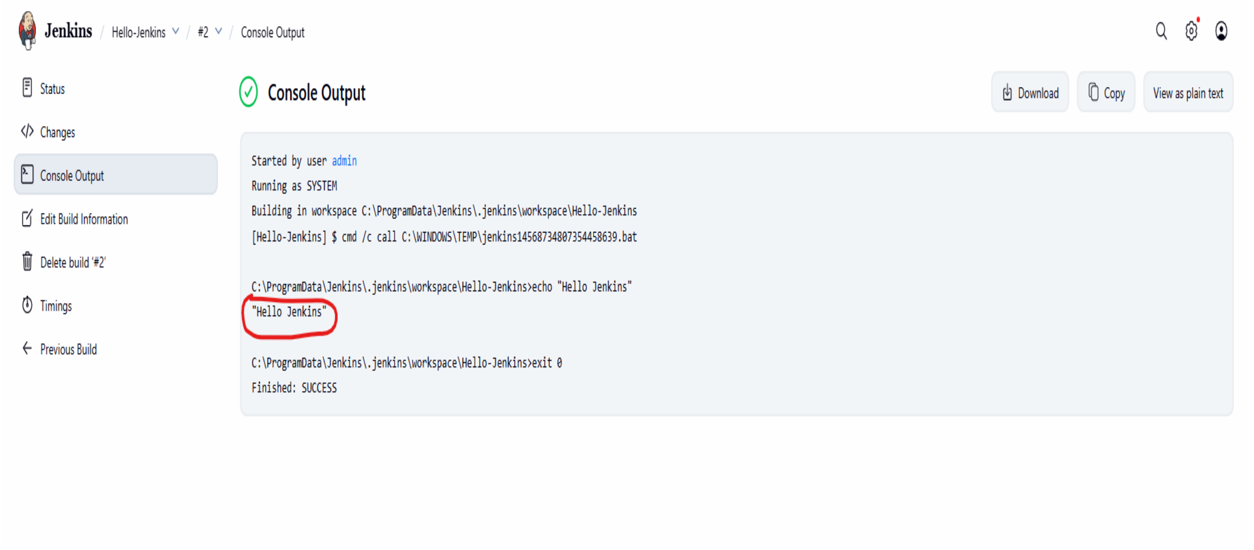
```
echo "Hello Jenkins"
```

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

4. Build the job manually

Expected Output: Console output showing message



```
Started by user admin
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins1456873480735445863 9.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>echo "Hello Jenkins"
"Hello Jenkins"

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>exit 0
Finished: SUCCESS
```

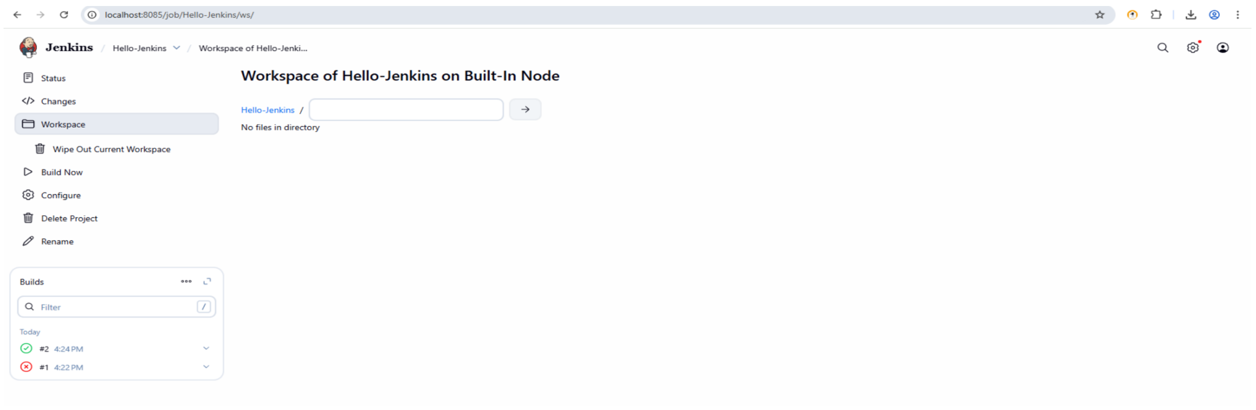
-

TASK 3: Jenkins Workspace & Commands

Objective: Understand workspace usage

Tasks:

1. Navigate to job workspace

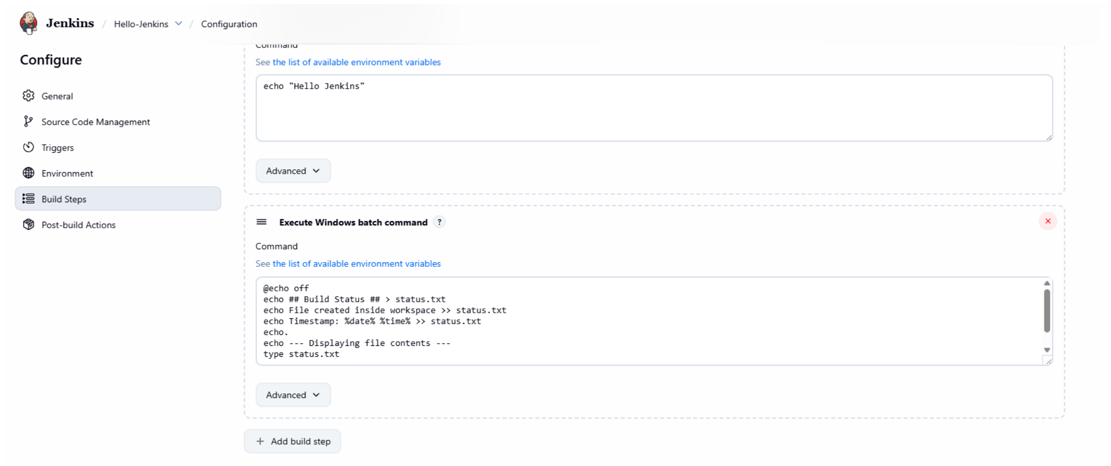

localhost:8085/job/Hello-Jenkins/ws/
Jenkins / Hello-Jenkins / Workspace of Hello-Jenki...
Status
Changes
Workspace
Wipe Out Current Workspace
Build Now
Configure
Delete Project
Rename
Builds
Filter
Today
#2 4:24 PM
#1 4:22 PM

Workspace of Hello-Jenkins on Built-In Node

Hello-Jenkins /

No files in directory

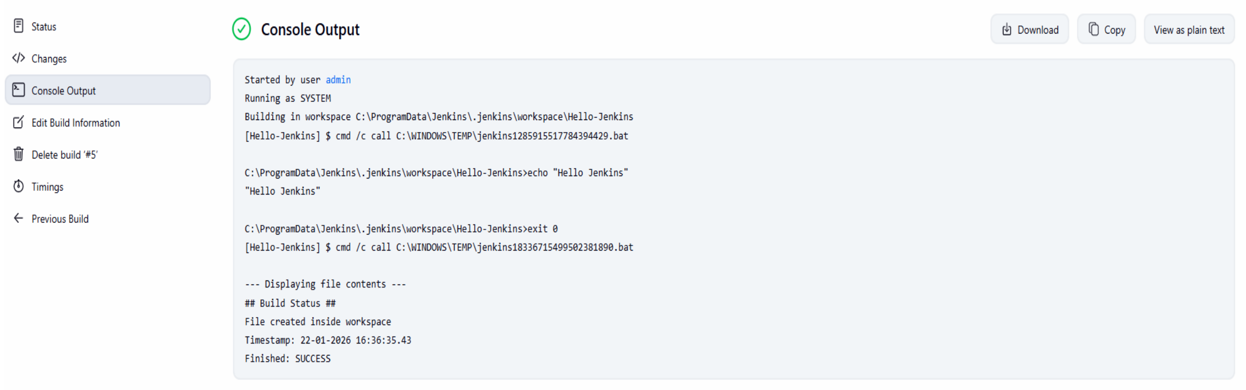
2. Create a text file using build step


Jenkins / Hello-Jenkins / Configuration
Configure
General
Source Code Management
Triggers
Environment
Build Steps
Post-build Actions
See the list of available environment variables
Advanced
Execute Windows batch command
Command
See the list of available environment variables
Advanced
Add build step

```
echo "Hello Jenkins"
```

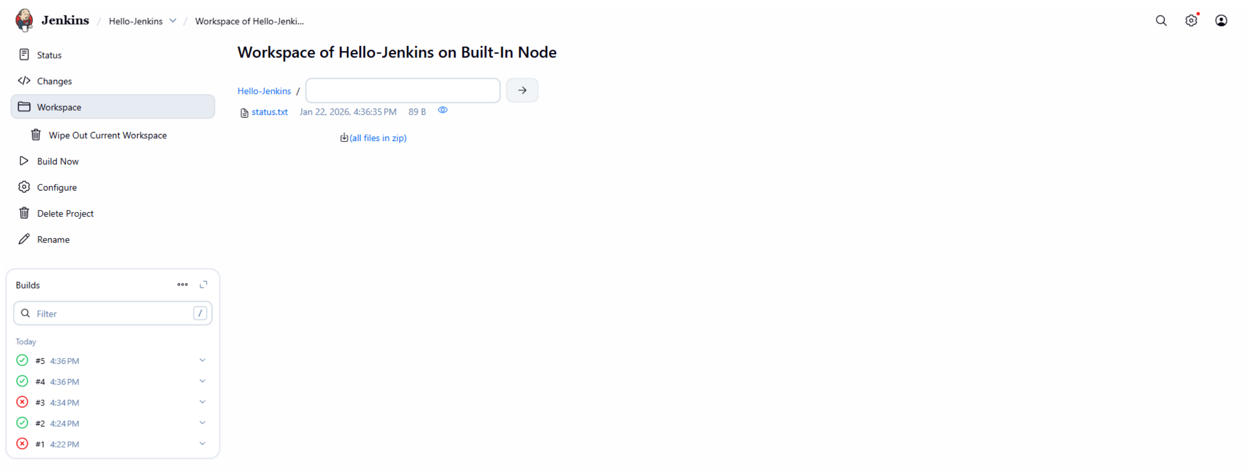
```
@echo off
echo ## Build Status ## > status.txt
echo File created inside workspace >> status.txt
echo Timestamp: %date% %time% >> status.txt
echo.
echo --- Displaying file contents ---
type status.txt
```

3. Display file contents in console



```
Started by user admin
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins12859155177843944429.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>echo "Hello Jenkins"
"Hello Jenkins"

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>exit 0
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins18336715499502381890.bat

--- Displaying file contents ---
## Build Status ##
File created inside workspace
Timestamp: 22-01-2026 16:36:35.43
Finished: SUCCESS
```

Expected Output: File created inside workspace


Jenkins / Hello-Jenkins / Workspace of Hello-Jenki...
Status
Changes
Workspace
Wipe Out Current Workspace
Build Now
Configure
Delete Project
Rename
Builds
Filter

Workspace of Hello-Jenkins on Built-In Node

Hello-Jenkins /

status.txt Jan 22, 2026, 4:36:35 PM 89 B

(all files in zip)

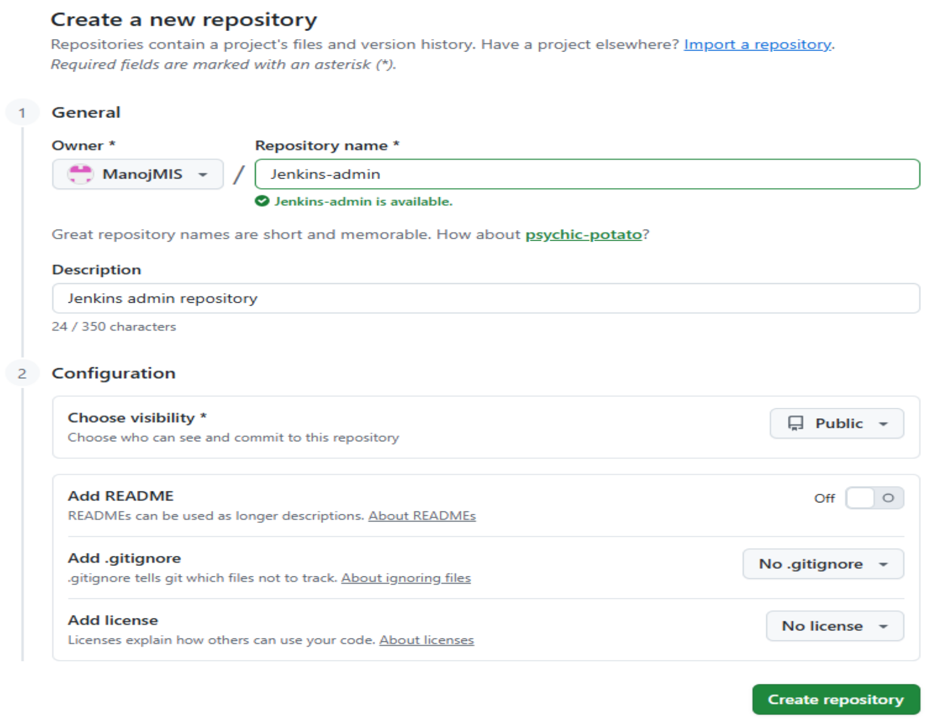
Today

- ✓ #5 4:36 PM
- ✓ #4 4:36 PM
- ✗ #3 4:34 PM
- ✓ #2 4:24 PM
- ✗ #1 4:22 PM

TASK 4: Git Integration

Objective: Integrate Jenkins with GitHub

Tasks: 1. Create a GitHub repository with sample code



Create a new repository

Repositories contain a project's files and version history. Have a project elsewhere? Import a repository.
Required fields are marked with an asterisk (*).

1 General

Owner *: ManojMIS / Repository name *: Jenkins-admin

Jenkins-admin is available.

Great repository names are short and memorable. How about psychic-potato?

Description: Jenkins admin repository

24 / 350 characters

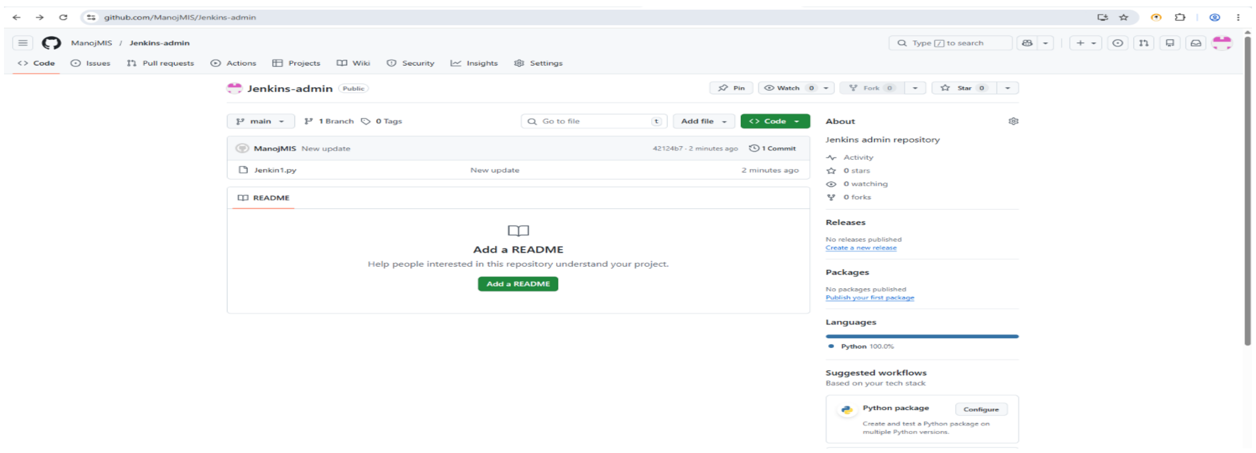
2 Configuration

Choose visibility *: Public
Choose who can see and commit to this repository

Add README: Off
READMEs can be used as longer descriptions. About READMEs

Add .gitignore: No .gitignore
.gitignore tells git which files not to track. About ignoring files

Add license: No license
Licenses explain how others can use your code. About licenses


github.com/ManojMIS/Jenkins-admin

ManojMIS / Jenkins-admin

Jenkins-admin Public

main · 1 Branch · 0 Tags

ManojMIS New update · 42124b7 · 2 minutes ago · 1 Commit

Jenkin1.py · New update · 2 minutes ago

README

Add a README

Help people interested in this repository understand your project.

About

Jenkins admin repository

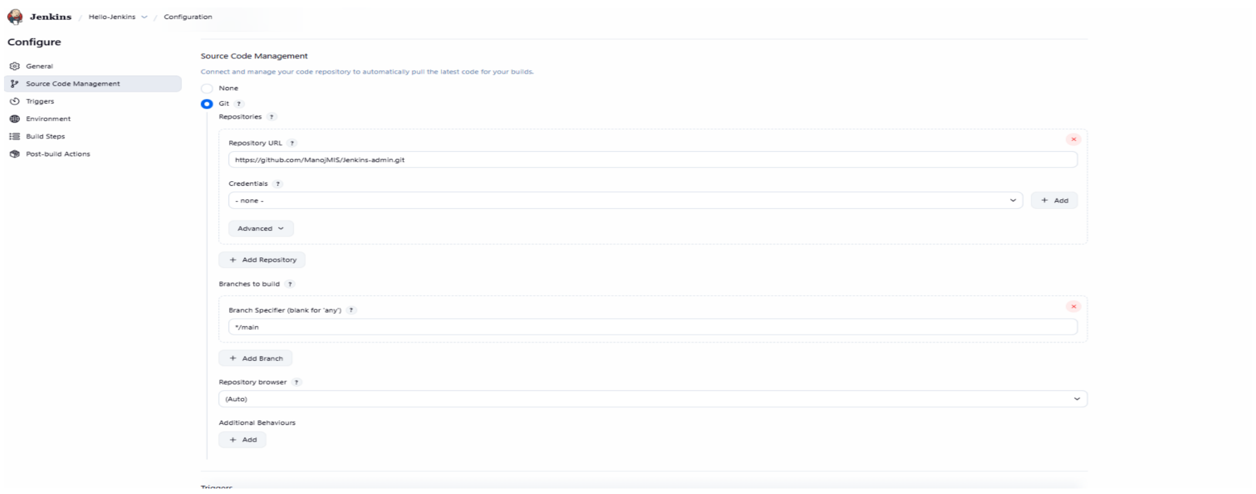
Activity · 0 stars · 0 watching · 0 forks

Releases: No releases published

Packages: No packages published

Languages: Python 100.0%

2. Configure Git in Jenkins


Jenkins / Hello-Jenkins / Configuration

Configure

Source Code Management

Connect and manage your code repository to automatically pull the latest code for your builds.

None [✓ Git]

Repositories

Repository URL: https://github.com/ManojMIS/Jenkins-admin.git

Credentials: - none -

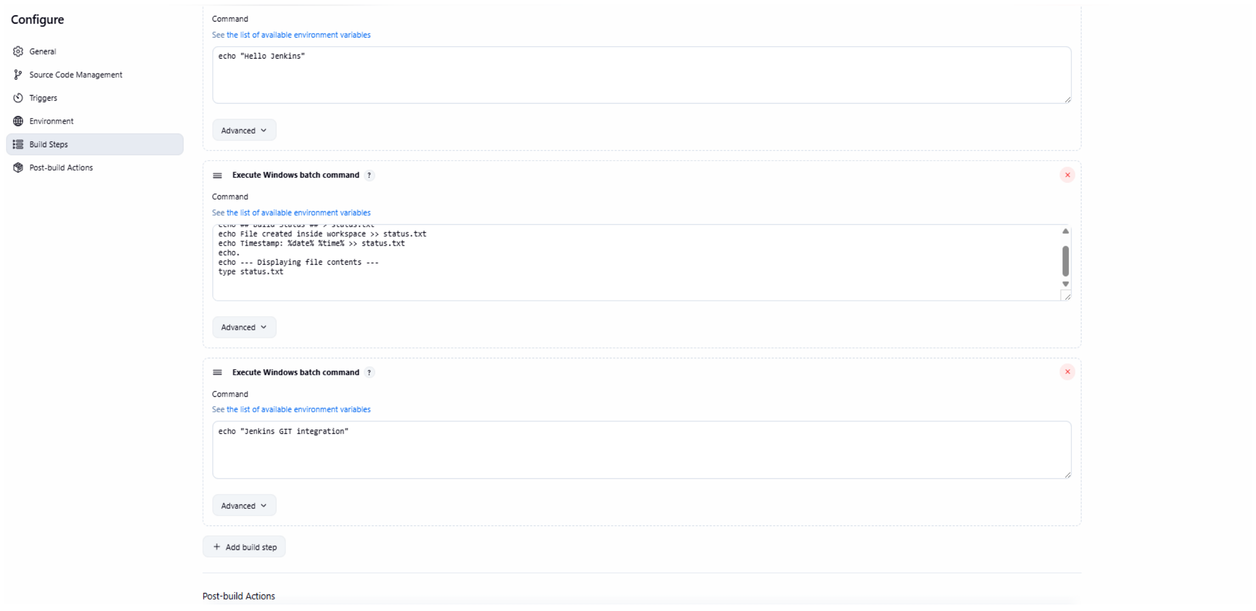
Branches to build

Branch Specifier (blank for 'any'): */main

Repository browser: (Auto)

Additional Behaviours

3. Add Git repository URL in job


Configure

Command

```
echo "Hello Jenkins"
```

Execute Windows batch command

Command

```
echo ## Build Status ## > status.txt
echo File created inside workspace >> status.txt
echo Timestamp: %date% %time% >> status.txt
echo.
echo --- Displaying file contents ---
type status.txt
```

Execute Windows batch command

Command

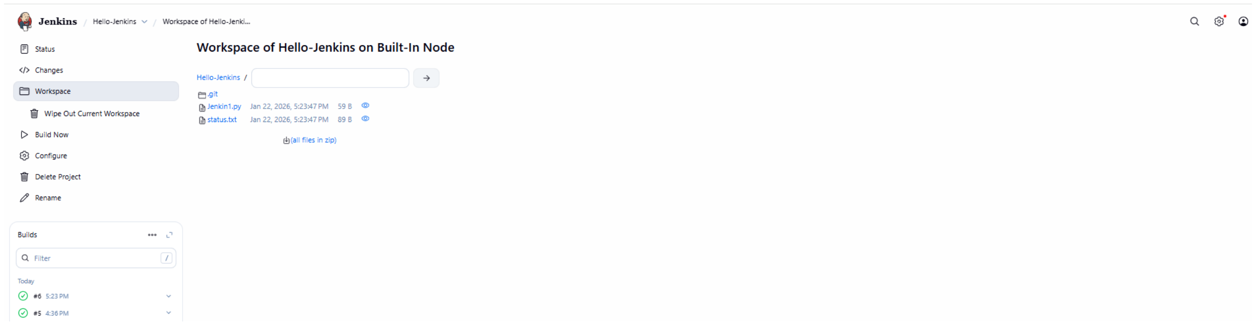
```
echo "Jenkins GIT integration"
```

Add build step

Post-build Actions

4. Build and verify code checkout

Expected Output: Source code visible in workspace



Workspace of Hello-Jenkins on Built-In Node

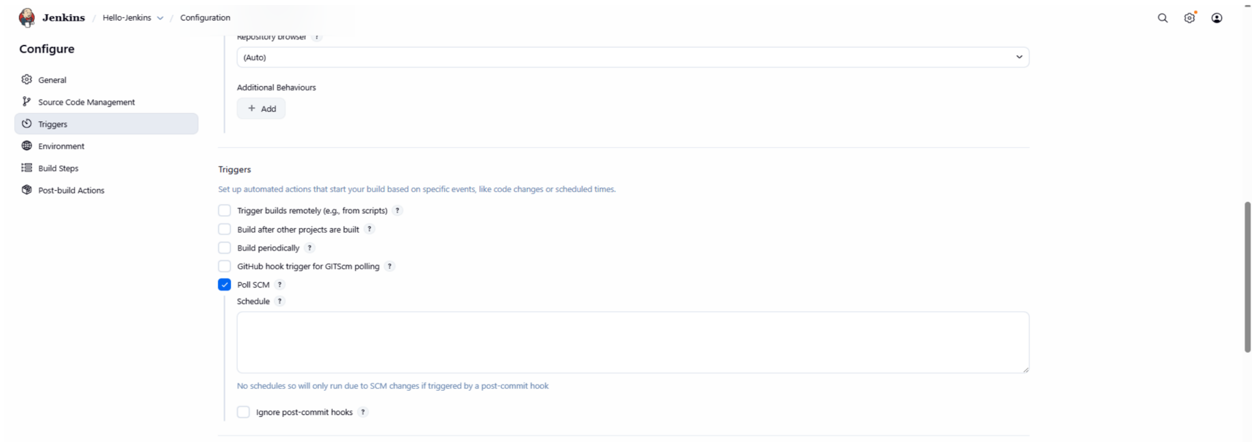
```
.git
Jenkin1.py   Jan 22, 2026, 5:23:47 PM   59 B
status.txt   Jan 22, 2026, 5:23:47 PM   89 B
(all files in zip)
```

Builds: #6 5:23 PM, #5 4:36 PM

TASK 5: Poll SCM Trigger

Objective: Automatically trigger builds on code change

Tasks: 1. Enable Poll SCM



Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

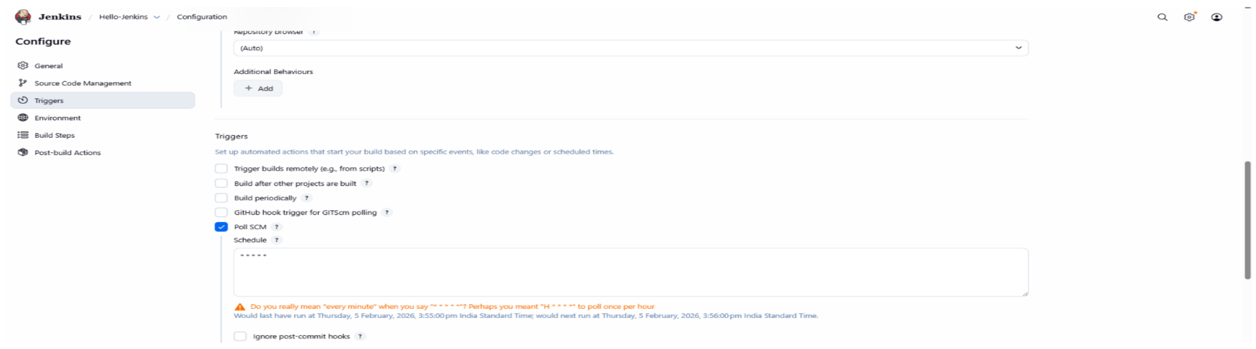
- [ ] Trigger builds remotely (e.g., from scripts)
- [ ] Build after other projects are built
- [ ] Build periodically
- [ ] GitHub hook trigger for GITScm polling
- [x] Poll SCM

Schedule

No schedules so will only run due to SCM changes if triggered by a post-commit hook

- [ ] Ignore post-commit hooks

2. Set schedule: * * * * *



Triggers

Set up automated actions that start your build based on specific events, like code changes or scheduled times.

- [ ] Trigger builds remotely (e.g., from scripts)
- [ ] Build after other projects are built
- [ ] Build periodically
- [ ] GitHub hook trigger for GITScm polling
- [x] Poll SCM

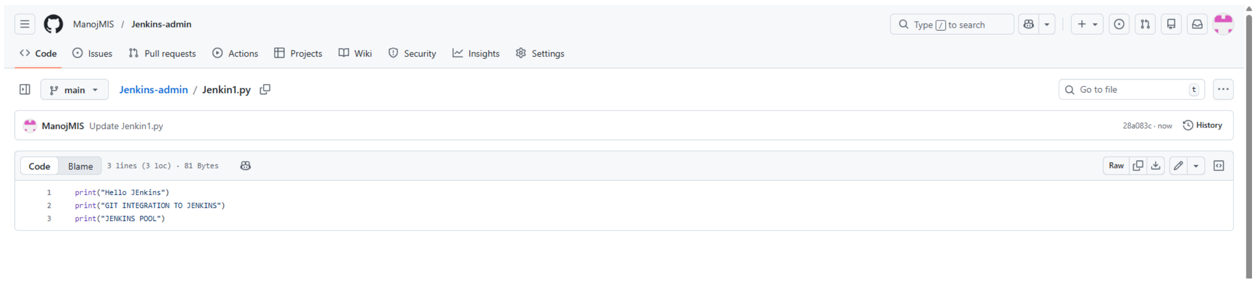
Schedule

```
* * * * *
```

Do you really mean "every minute" when you say "* * * * *"? Perhaps you meant "H * * * *" to poll once per hour
Would last have run at Thursday, 5 February, 2026, 3:55:00 pm India Standard Time; would next run at Thursday, 5 February, 2026, 3:56:00 pm India Standard Time.

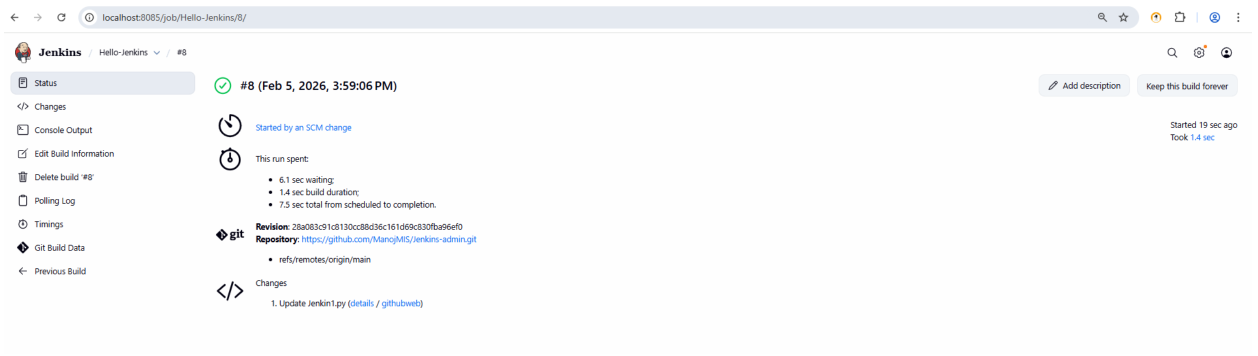
- [ ] Ignore post-commit hooks

3. Modify GitHub file and commit


ManojMIS / Jenkins-admin
Code Issues Pull requests Actions Projects Wiki Security Insights Settings
main
Jenkins-admin / Jenkin1.py
Go to file
ManojMIS Update Jenkin1.py
28a083c · now
History
Code Blame 3 lines (3 loc) · 81 Bytes
Raw

```
print("Hello Jenkins")
print("GIT INTEGRATION TO JENKINS")
print("JENKINS POOL")
```

4. Observe automatic build Expected Output: • Build triggered without manual action



#8 (Feb 5, 2026, 3:59:06 PM)

Started by an SCM change

This run spent:

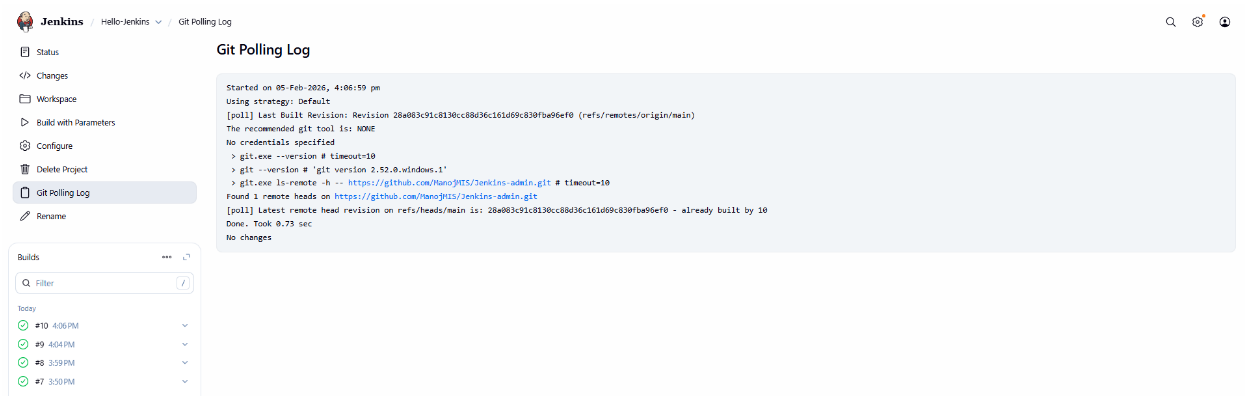
- 6.1 sec waiting;
- 1.4 sec build duration;
- 7.5 sec total from scheduled to completion.

Revision: 28a083c91c8130cc88d36c161d69c830fba96ef0
Repository: https://github.com/ManojMIS/Jenkins-admin.git

- refs/remotes/origin/main

Changes

1. Update Jenkin1.py (details / githubweb)



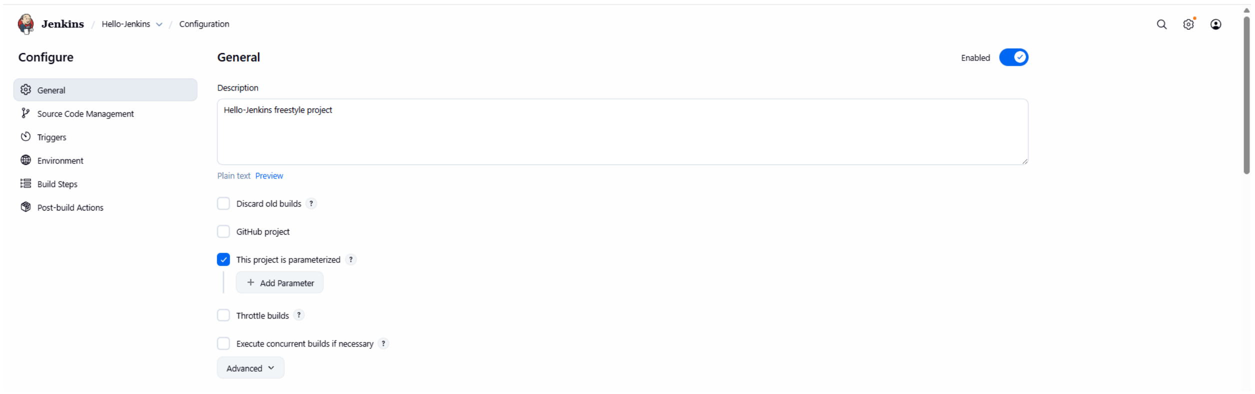
Git Polling Log

```
Started on 05-Feb-2026, 4:06:59 pm
Using strategy: Default
[poll] Last Built Revision: Revision 28a083c91c8130cc88d36c161d69c830fba96ef0 (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
 > git.exe --version # timeout=10
 > git --version # 'git version 2.52.0.windows.1'
 > git.exe ls-remote -h -- https://github.com/ManojMIS/Jenkins-admin.git # timeout=10
Found 1 remote heads on https://github.com/ManojMIS/Jenkins-admin.git
[poll] Latest remote head revision on refs/heads/main is: 28a083c91c8130cc88d36c161d69c830fba96ef0 - already built by 10
Done. Took 0.73 sec
No changes
```

TASK 6: Parameterized Build

Objective: Use parameters in Jenkins job

Tasks: 1. Enable parameterized build



Jenkins / Hello-Jenkins / Configuration

Configure

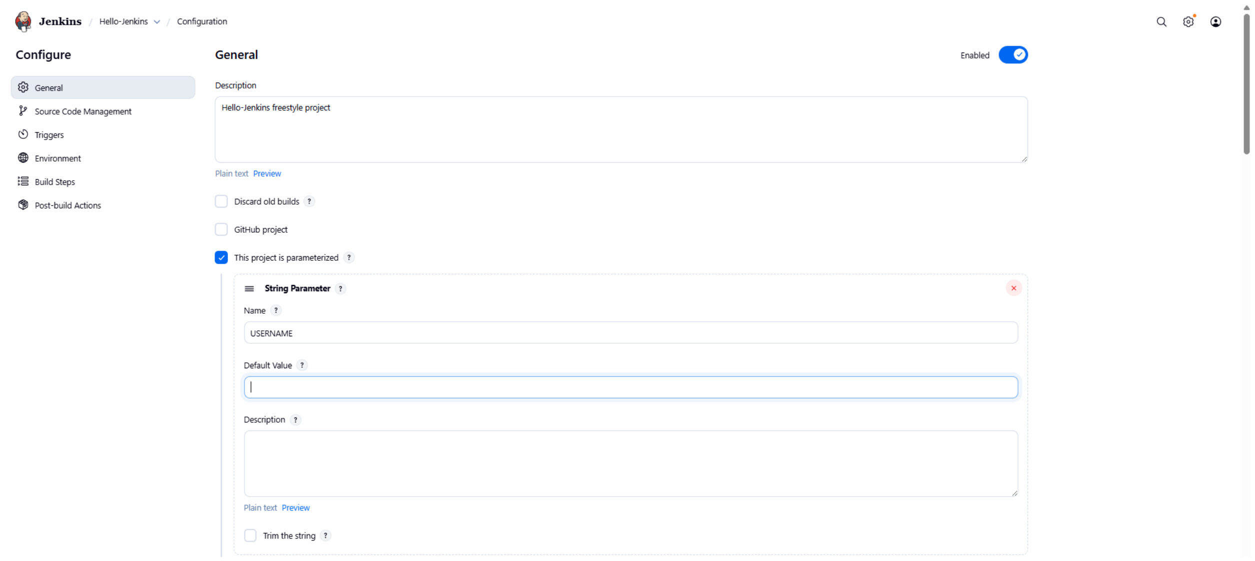
General

Enabled

Description: Hello-Jenkins freestyle project

- [ ] Discard old builds
- [ ] GitHub project
- [x] This project is parameterized
- [ ] Throttle builds
- [ ] Execute concurrent builds if necessary

2. Add String parameter USERNAME



Jenkins / Hello-Jenkins / Configuration

Configure

General

Enabled

Description: Hello-Jenkins freestyle project

- [ ] Discard old builds
- [ ] GitHub project
- [x] This project is parameterized

String Parameter

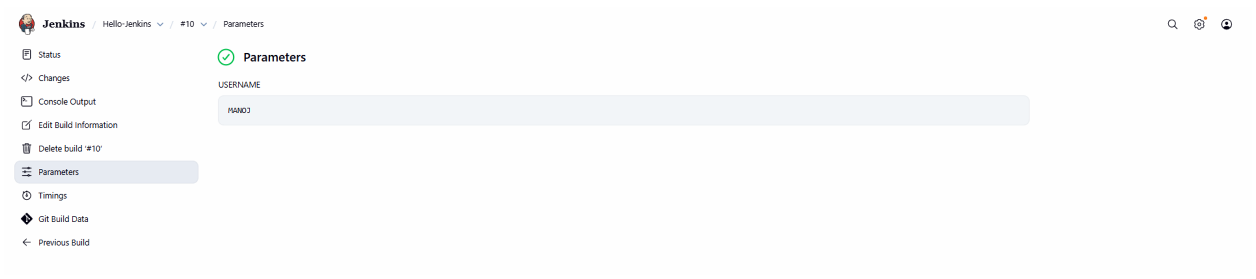
Name: USERNAME

Default Value:

Description:

- [ ] Trim the string

3. Print parameter value in build step Expected Output: • Console output showing parameter value



Parameters

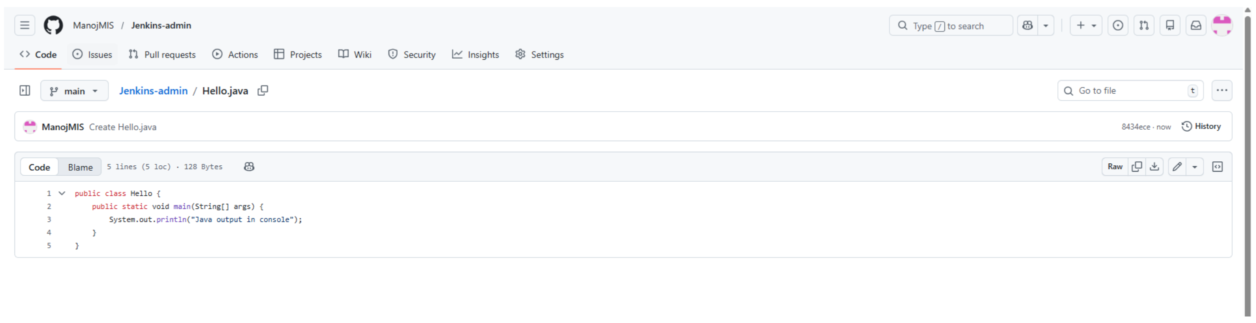
USERNAME

```
MANOJ
```

TASK 7: Java Build Using Jenkins

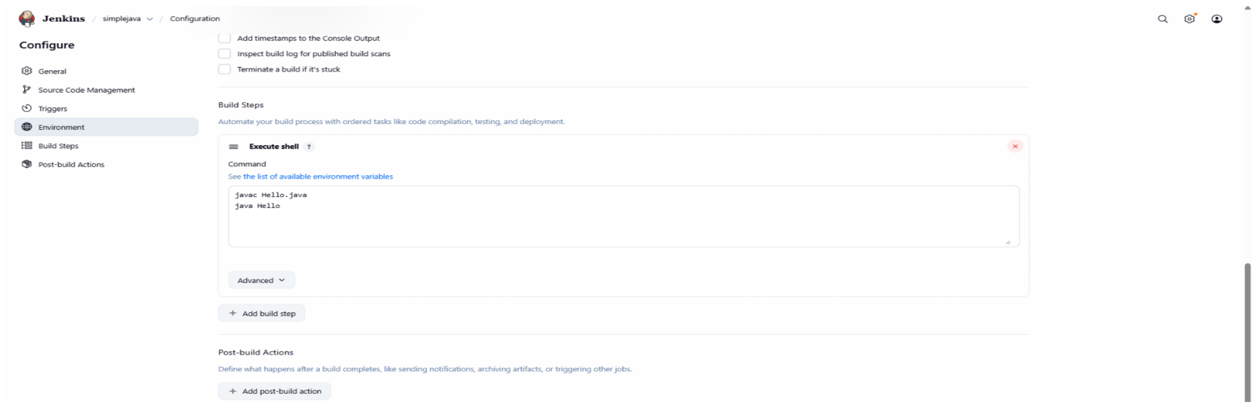
Objective: Compile Java program using Jenkins

Tasks: 1. Create simple Hello.java



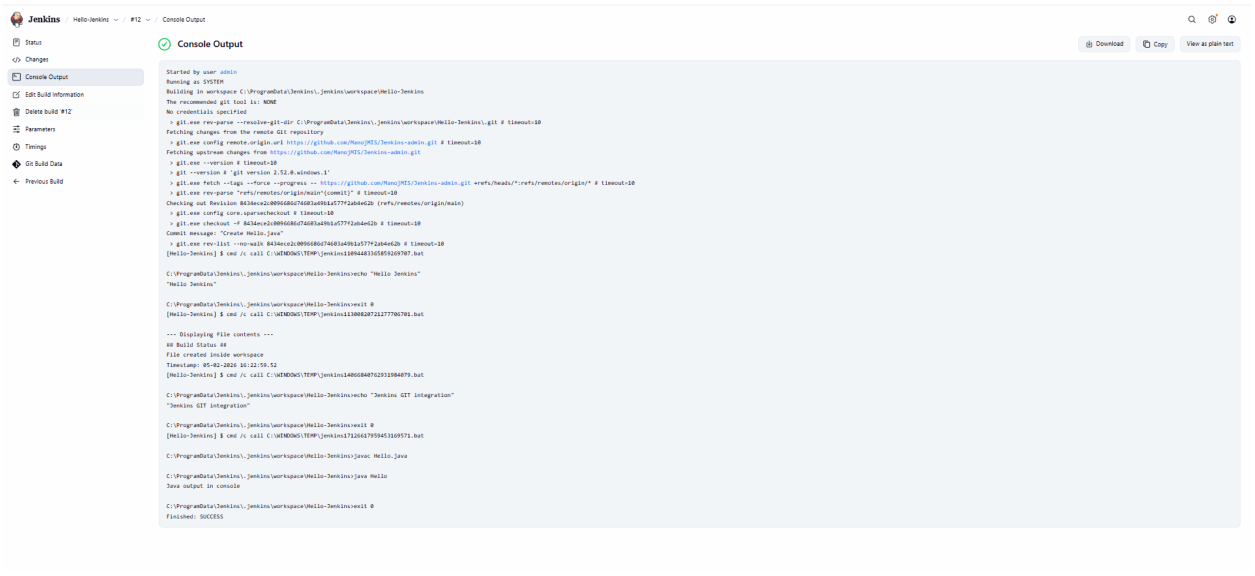
```java
public class Hello {
    public static void main(String[] args) {
        System.out.println("Java output in console");
    }
}
```

2. Compile using javac


Jenkins / simplejava / Configuration
Configure
General
Source Code Management
Triggers
Environment
Build Steps
Post-build Actions
Add timestamps to the Console Output
Inspect build log for published build scans
Terminate a build if it's stuck
Build Steps
Automate your build process with ordered tasks like code compilation, testing, and deployment.
Execute shell
Command
See the list of available environment variables
Advanced
Add build step
Post-build Actions
Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.
Add post-build action

```
javac Hello.java
java Hello
```

3. Run Java program Expected Output: • Java output in console



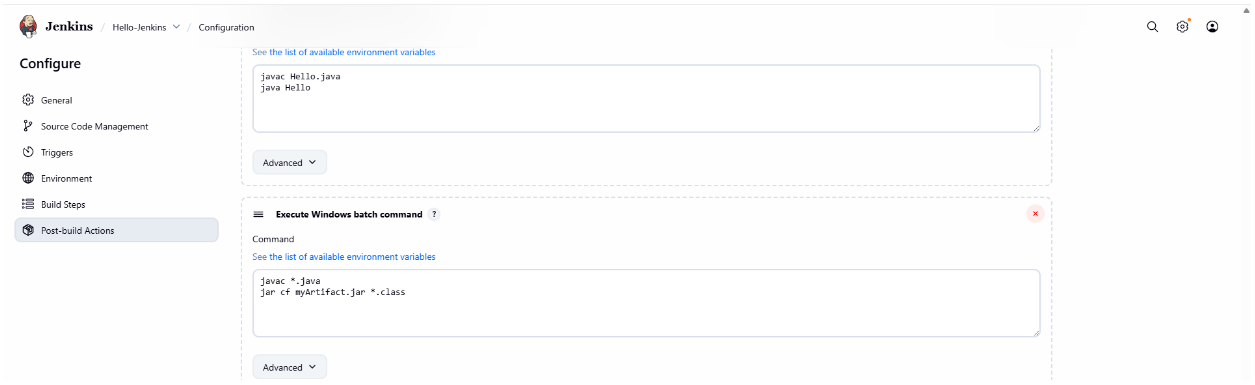
Console Output

```
Started by user admin
Running as SYSTEM
Building in workspace C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins
The recommended git tool is: NONE
No credentials specified
 > git.exe rev-parse --resolve-git-dir C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins\.git # timeout=10
Fetching changes from the remote Git repository
 > git.exe config remote.origin.url https://github.com/ManojMIS/Jenkins-admin.git # timeout=10
Fetching upstream changes from https://github.com/ManojMIS/Jenkins-admin.git
 > git.exe --version # timeout=10
 > git --version # 'git version 2.52.0.windows.1'
 > git.exe fetch --tags --force --progress -- https://github.com/ManojMIS/Jenkins-admin.git +refs/heads/*:refs/remotes/origin/* # timeout=10
 > git.exe rev-parse "refs/remotes/origin/main^{commit}" # timeout=10
Checking out Revision 8434ece2c0096686d74603a49b1a577f2ab4e62b (refs/remotes/origin/main)
 > git.exe config core.sparsecheckout # timeout=10
 > git.exe checkout -f 8434ece2c0096686d74603a49b1a577f2ab4e62b # timeout=10
Commit message: "Create Hello.java"
 > git.exe rev-list --no-walk 8434ece2c0096686d74603a49b1a577f2ab4e62b # timeout=10
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins1109448336585926970 7.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>echo "Hello Jenkins"
"Hello Jenkins"

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>exit 0
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins11300820721277706701.bat

--- Displaying file contents ---
## Build Status ##
File created inside workspace
Timestamp: 05-02-2026 16:22:59.52
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins14066840762931984079.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>echo "Jenkins GIT integration"
"Jenkins GIT integration"

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>exit 0
[Hello-Jenkins] $ cmd /c call C:\WINDOWS\TEMP\jenkins17126617959453169571.bat

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>javac Hello.java

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>java Hello
Java output in console

C:\ProgramData\Jenkins\.jenkins\workspace\Hello-Jenkins>exit 0
Finished: SUCCESS
```

TASK 8: Archive Artifacts

Objective: Store build outputs

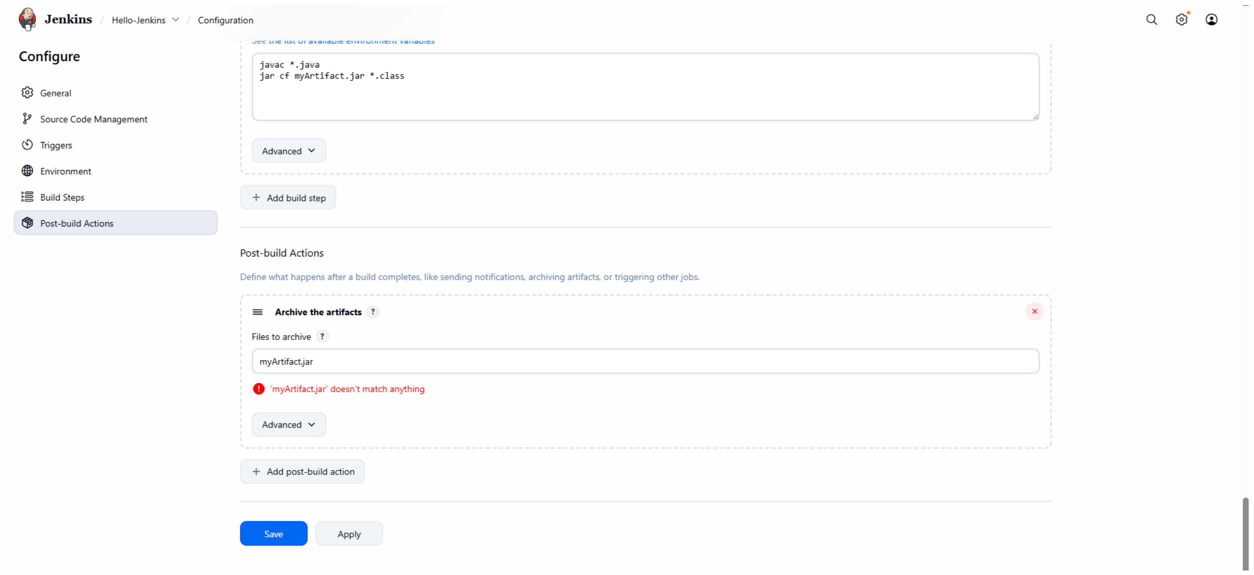
Tasks: 1. Generate .class or .jar file



```
javac Hello.java
java Hello
```

```
javac *.java
jar cf myArtifact.jar *.class
```

2. Archive artifacts in post-build action



```
javac *.java
jar cf myArtifact.jar *.class
```

Advanced

Add build step

Post-build Actions

Define what happens after a build completes, like sending notifications, archiving artifacts, or triggering other jobs.

Archive the artifacts

Files to archive: myArtifact.jar

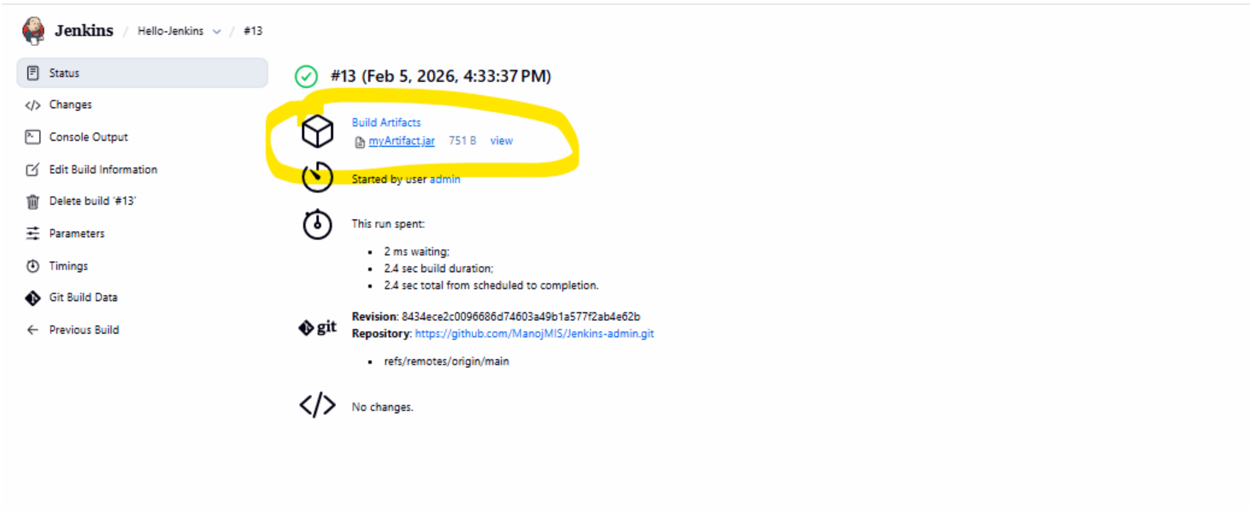
'myArtifact.jar' doesn't match anything

Advanced

Add post-build action

Save Apply

3. Download artifact from Jenkins UI Expected Output: • Artifact available for download



#13 (Feb 5, 2026, 4:33:37 PM)

Build Artifacts
myArtifact.jar 751 B view

Started by user admin

This run spent:

- 2 ms waiting;
- 2.4 sec build duration;
- 2.4 sec total from scheduled to completion.

Revision: 8434ece2c0096686d74603a49b1a577f2ab4e62b
Repository: https://github.com/ManojMIS/Jenkins-admin.git

- refs/remotes/origin/main

No changes.

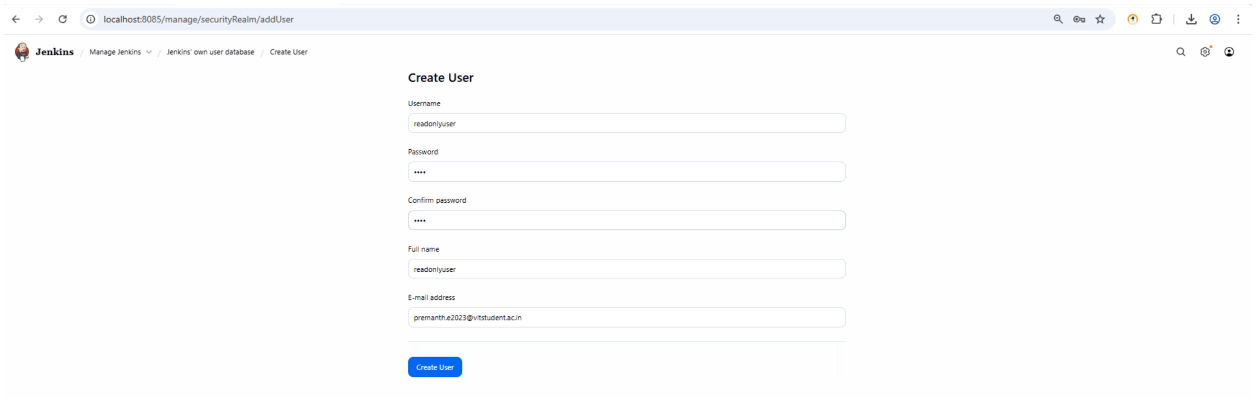
TASK 9: Users & Roles

Objective: Manage Jenkins users

Tasks: 1. Create two users

2. Assign read-only permission to one user

3. Assign build permission to another user Expected Output: ● Permission differences verified


localhost:8085/manage/securityRealm/addUser

Jenkins / Manage Jenkins / Jenkins' own user database / Create User

Create User

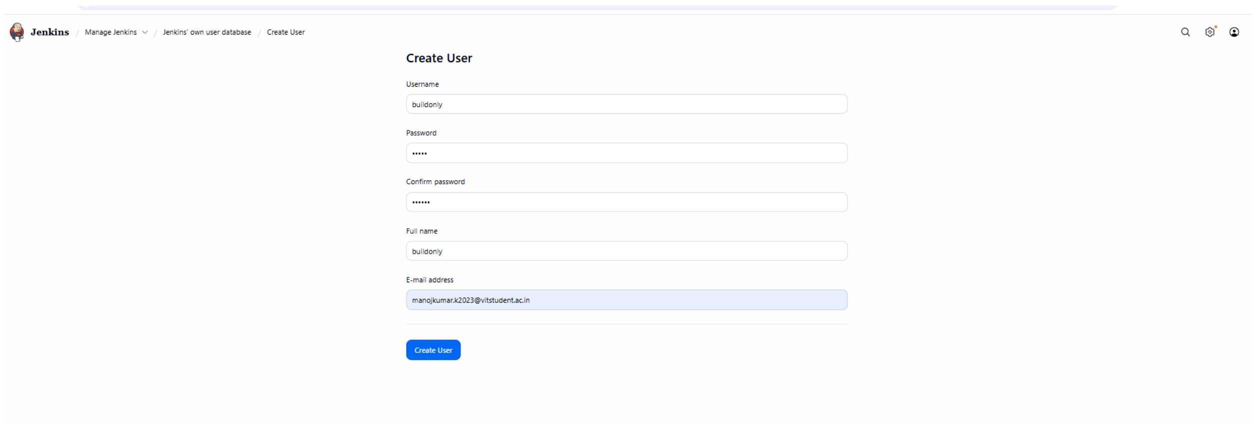
Username: readonlyuser

Password: ••••

Confirm password: ••••

Full name: readonlyuser

E-mail address: premanth.e2023@vitstudent.ac.in


Jenkins / Manage Jenkins / Jenkins' own user database / Create User

Create User

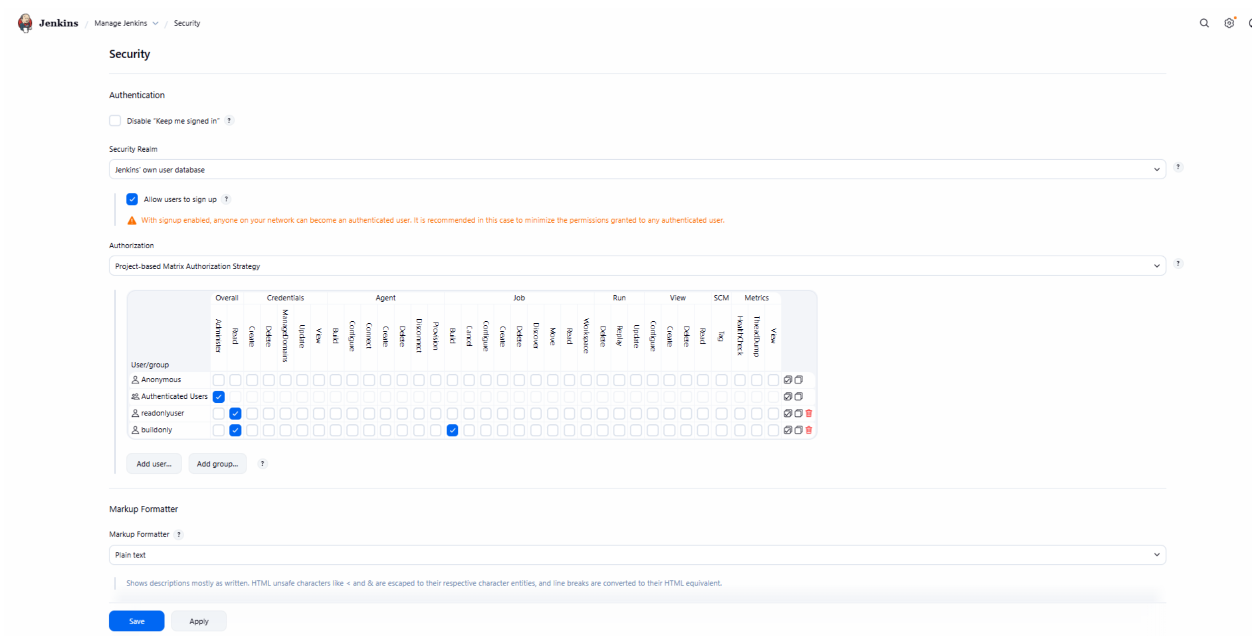
Username: buildonly

Password: •••••

Confirm password: ••••••

Full name: buildonly

E-mail address: manojkumar.k2023@vitstudent.ac.in



Jenkins / Manage Jenkins / Security

Security

Authentication

☐ Disable "Keep me signed in" ?

Security Realm

Jenkins' own user database

✓ Allow users to sign up ?

⚠ With signup enabled, anyone on your network can become an authenticated user. It is recommended in this case to minimize the permissions granted to any authenticated user.

Authorization

Project-based Matrix Authorization Strategy

	Overall		Credentials					Agent							Job									Run			View				SCM	Metrics		
User/group	Administer	Read	Create	Delete	ManageDomains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect	Provision	Build	Cancel	Configure	Create	Delete	Discover	Move	Read	Workspace	Delete	Replay	Update	Configure	Create	Delete	Read	Tag	HealthCheck	ThreadDump	View
Anonymous	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Authenticated Users	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
readonlyuser	☐	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
buildonly	☐	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Add user... Add group... ?

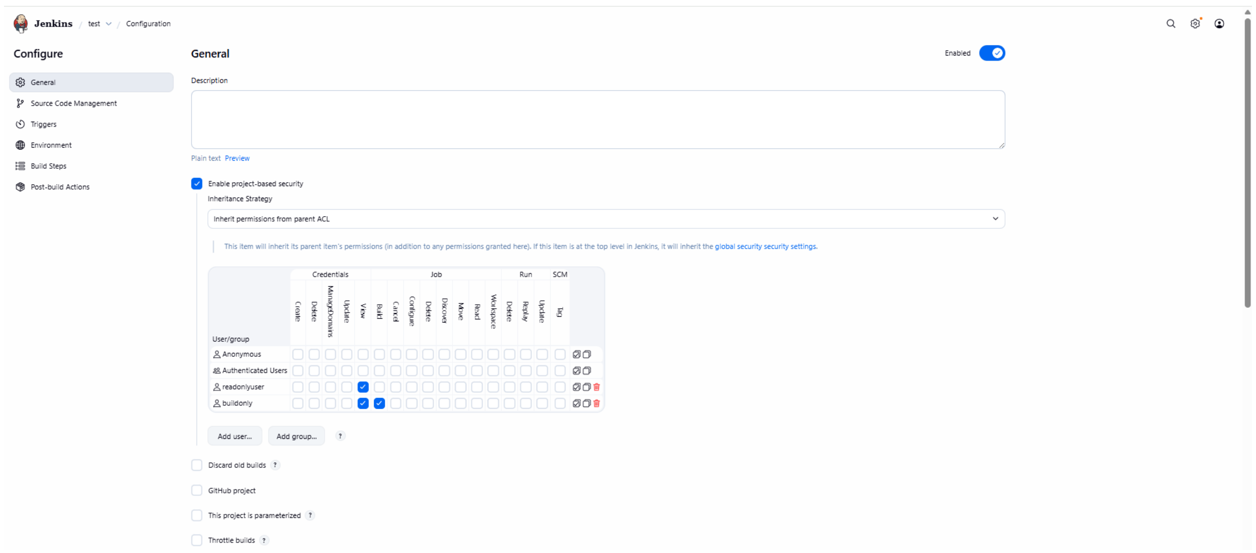
Markup Formatter

Markup Formatter ?

Plain text

Shows descriptions mostly as written. HTML unsafe characters like < and & are escaped to their respective character entities, and line breaks are converted to their HTML equivalent.

Save Apply



Jenkins / test / Configuration

Configure

General

Enabled

Description

Plain text Preview

✓ Enable project-based security

Inheritance Strategy

Inherit permissions from parent ACL

This item will inherit its parent item's permissions (in addition to any permissions granted here). If this item is at the top level in Jenkins, it will inherit the global security security settings.

	Credentials					Job								Run			SCM
User/group	Create	Delete	ManageDomains	Update	View	Build	Cancel	Configure	Delete	Discover	Move	Read	Workspace	Delete	Replay	Update	Tag
Anonymous	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
Authenticated Users	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
readonlyuser	☐	☐	☐	☐	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐
buildonly	☐	☐	☐	☐	✓	✓	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐	☐

Add user... Add group... ?

☐ Discard old builds ?

☐ GitHub project

☐ This project is parameterized ?

☐ Throttle builds ?

DESCRIPTION:PIPELINES WERE NOT TAUGHT SO MAM ANNOUNCED TO DO TASKS TILL 9 and from 10 to 15 related to pipeline not required

TASK 10: Simple Jenkins Pipeline

Objective: Create basic pipeline

Tasks: 1. Create Pipeline job

2. Write pipeline with stages: o Checkout o Build o Test

3. Run pipeline Expected Output: • Pipeline stage view

TASK 11: Jenkinsfile from Git

Objective: Pipeline as Code

Tasks: 1. Create Jenkinsfile in Git repo

2. Configure pipeline from SCM

3. Trigger build Expected Output: • Pipeline executed from Git

TASK 12: Post-Build Actions

Objective: Handle build result

Tasks: 1. Add post section

2. **Print message on success/failure Expected Output: • Appropriate message displayed**

TASK 13: Trigger Job from Another Job

Objective: Job chaining

Tasks: 1. Create Job-A and Job-B

3. **Configure Job-B to trigger after Job-A Expected Output: • Job-B triggered automatically**

TASK 14: Workspace Cleanup

Objective: Manage disk usage

Tasks: 1. Install Workspace Cleanup plugin

2. **Clean workspace before build Expected Output: • Workspace cleared before execution TASK**

15: Mini CI Project

Objective: Implement basic CI flow

Tasks: 1. Git commit → Jenkins build

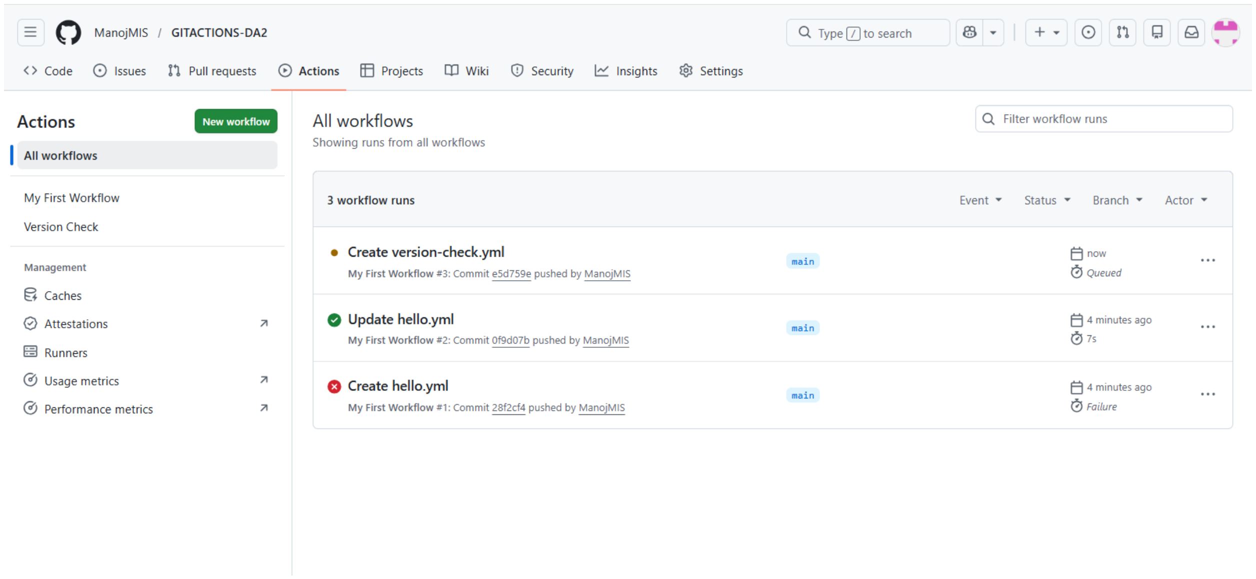
2. Compile code

3. Archive artifacts

4. Fail build on error Expected Output: • Automated CI pipeline

TOOL:GIT ACTIONS

1.)GIT ACTIONS DASBOARD:


ManojMIS / GITACTIONS-DA2

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

Actions — New workflow

All workflows
My First Workflow
Version Check

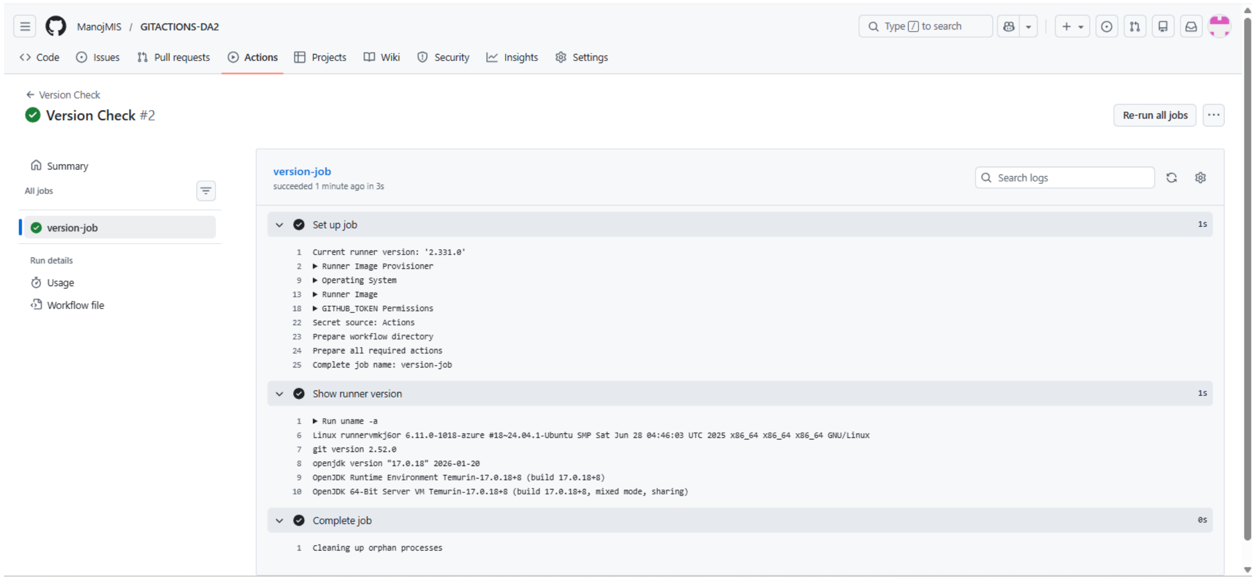
Management
Caches
Attestations
Runners
Usage metrics
Performance metrics

All workflows
Showing runs from all workflows

3 workflow runs — Event · Status · Branch · Actor

- Create version-check.yml — My First Workflow #3: Commit e5d759e pushed by ManojMIS — main — now — Queued
- Update hello.yml — My First Workflow #2: Commit 0f9d07b pushed by ManojMIS — main — 4 minutes ago — 7s
- Create hello.yml — My First Workflow #1: Commit 28f2cf4 pushed by ManojMIS — main — 4 minutes ago — Failure

VERSION AND OUTPUT:


ManojMIS / GITACTIONS-DA2

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

← Version Check
Version Check #2 — Re-run all jobs

Summary
All jobs
version-job
Run details
Usage
Workflow file

version-job
succeeded 1 minute ago in 3s

Set up job — 1s

```
Current runner version: '2.331.0'
▶ Runner Image Provisioner
▶ Operating System
▶ Runner Image
▶ GITHUB_TOKEN Permissions
Secret source: Actions
Prepare workflow directory
Prepare all required actions
Complete job name: version-job
```

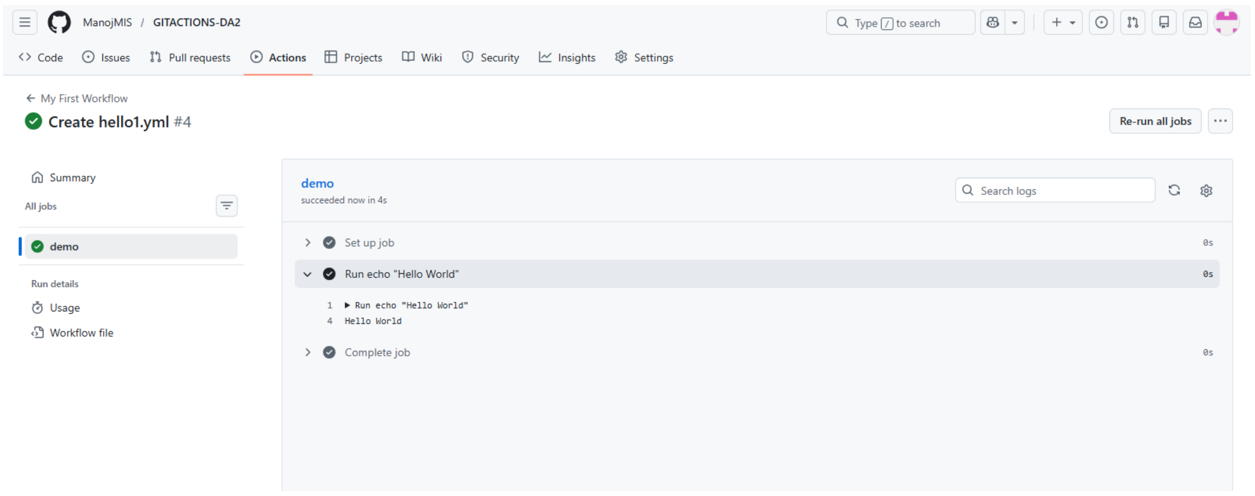
Show runner version — 1s

```
▶ Run uname -a
Linux runnervmkj6or 6.11.0-1018-azure #18~24.04.1-Ubuntu SMP Sat Jun 28 04:46:03 UTC 2025 x86_64 x86_64 x86_64 GNU/Linux
git version 2.52.0
openjdk version "17.0.18" 2026-01-20
OpenJDK Runtime Environment Temurin-17.0.18+8 (build 17.0.18+8)
OpenJDK 64-Bit Server VM Temurin-17.0.18+8 (build 17.0.18+8, mixed mode, sharing)
```

Complete job — 0s

```
Cleaning up orphan processes
```

2.)First Freestyle job created and Output:



ManojMIS / GITACTIONS-DA2

My First Workflow

Create hello1.yml #4

demo
succeeded now in 4s

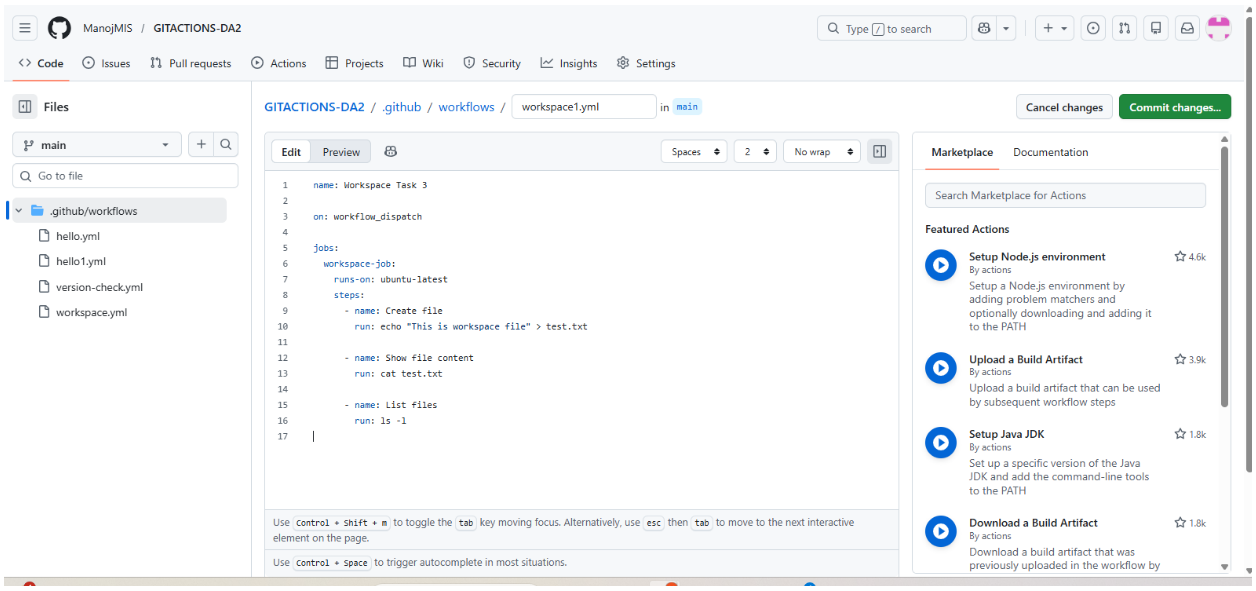
Set up job 0s

Run echo "Hello World" 0s

```
▶ Run echo "Hello World"
Hello World
```

Complete job 0s

3.)Text File Created:



ManojMIS / GITACTIONS-DA2

GITACTIONS-DA2 / .github / workflows / workspace1.yml in main

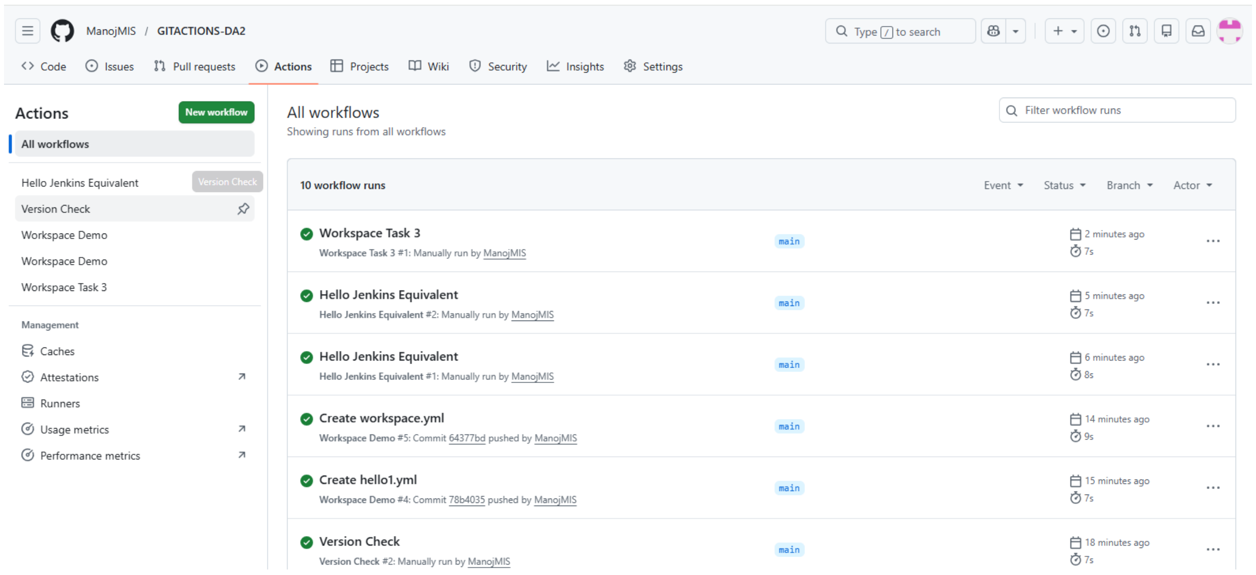
.github/workflows: hello.yml, hello1.yml, version-check.yml, workspace.yml

```
name: Workspace Task 3

on: workflow_dispatch

jobs:
  workspace-job:
    runs-on: ubuntu-latest
    steps:
      - name: Create file
        run: echo "This is workspace file" > test.txt

      - name: Show file content
        run: cat test.txt

      - name: List files
        run: ls -l
```

Use control + shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Use control + space to trigger autocomplete in most situations.

Job Workspace:



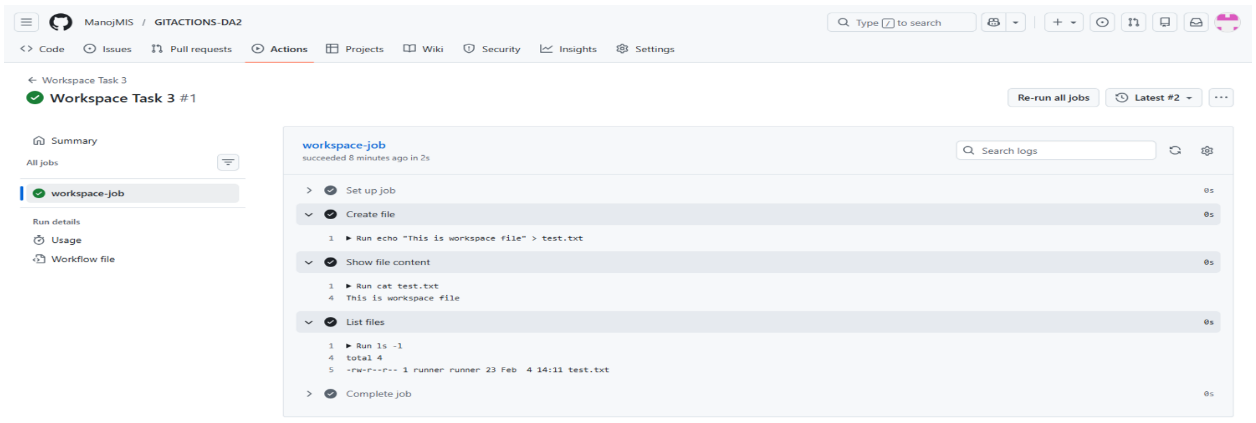
ManojMIS / GITACTIONS-DA2

All workflows
Showing runs from all workflows

10 workflow runs

Run	Branch	When	Duration
Workspace Task 3 Workspace Task 3 #1: Manually run by ManojMIS	main	2 minutes ago	7s
Hello Jenkins Equivalent Hello Jenkins Equivalent #2: Manually run by ManojMIS	main	5 minutes ago	7s
Hello Jenkins Equivalent Hello Jenkins Equivalent #1: Manually run by ManojMIS	main	6 minutes ago	8s
Create workspace.yml Workspace Demo #5: Commit 64377bd pushed by ManojMIS	main	14 minutes ago	9s
Create hello1.yml Workspace Demo #4: Commit 78b4035 pushed by ManojMIS	main	15 minutes ago	7s
Version Check Version Check #2: Manually run by ManojMIS	main	18 minutes ago	7s

OUTPUT:



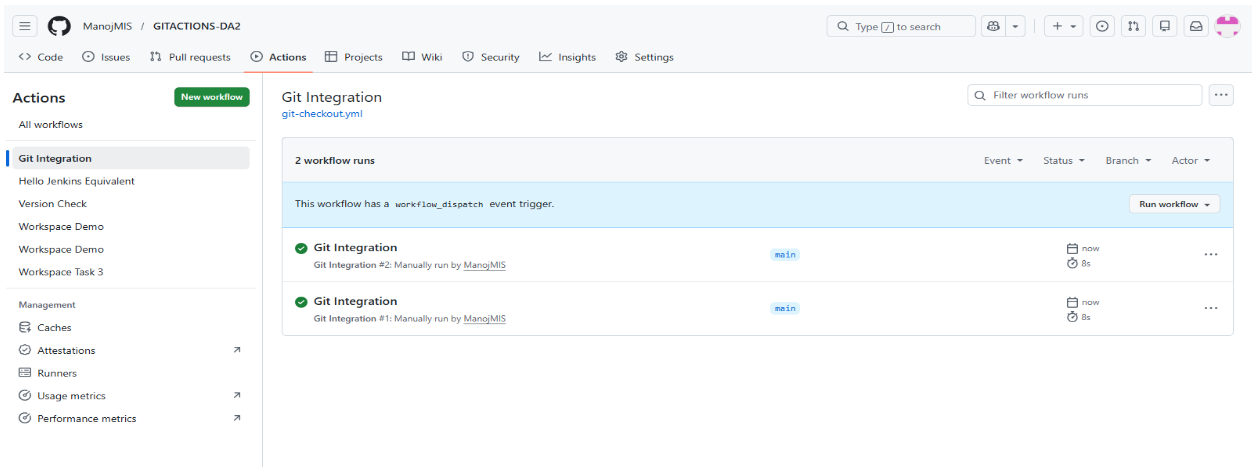
ManojMIS / GITACTIONS-DA2

Workspace Task 3 #1

workspace-job
succeeded 8 minutes ago in 2s

```
Set up job                                           0s
Create file                                          0s
  ▶ Run echo "This is workspace file" > test.txt
Show file content                                    0s
  ▶ Run cat test.txt
  This is workspace file
List files                                           0s
  ▶ Run ls -l
  total 4
  -rw-r--r-- 1 runner runner 23 Feb  4 14:11 test.txt
Complete job                                         0s
```

4.)**Git Integration Workflow created:**



Git Integration
git-checkout.yml

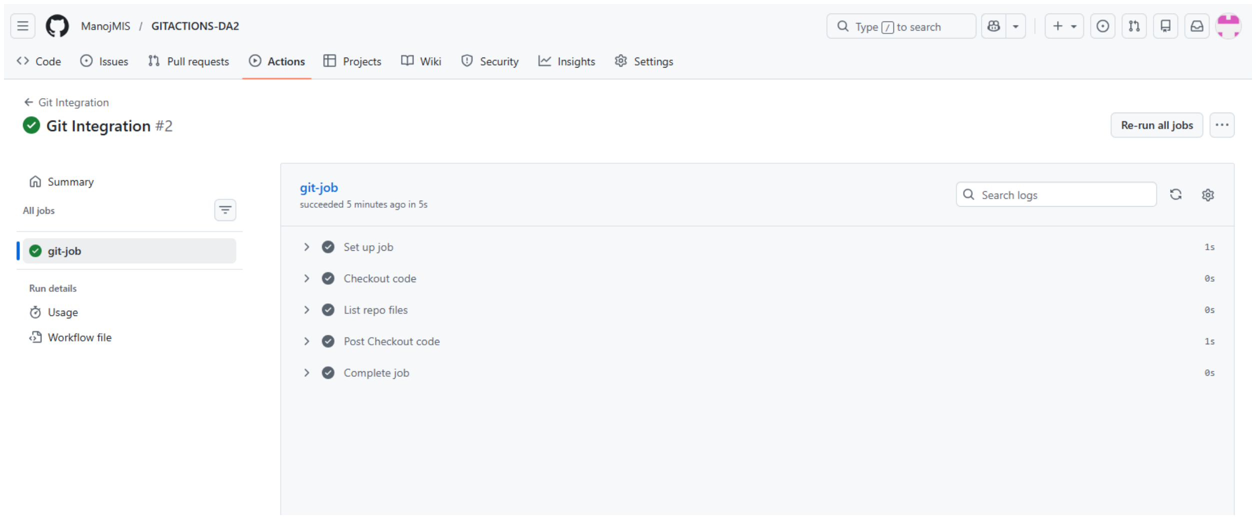
2 workflow runs

This workflow has a `workflow_dispatch` event trigger.

Git Integration — Git Integration #2: Manually run by ManojMIS — main — now — 8s

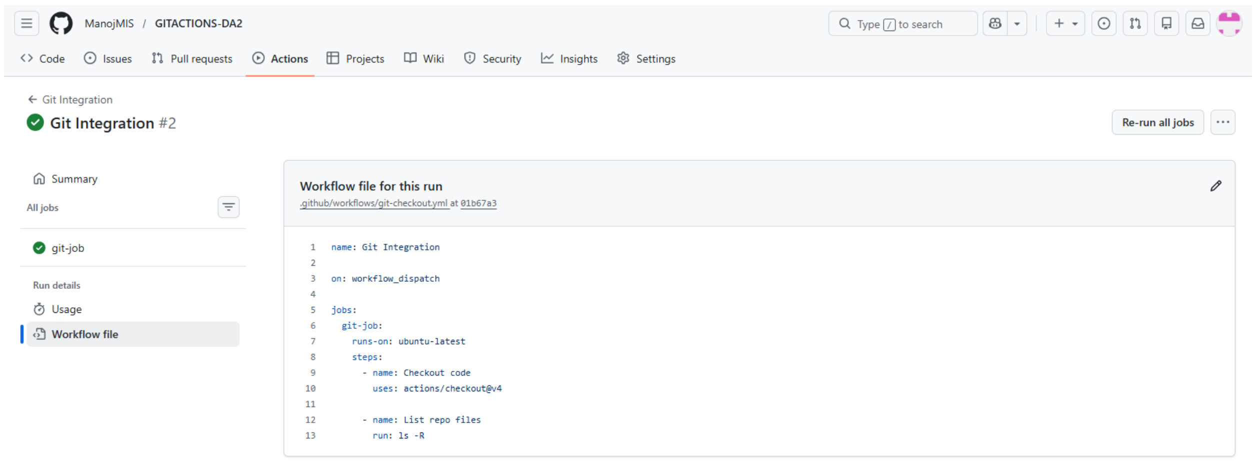
Git Integration — Git Integration #1: Manually run by ManojMIS — main — now — 8s

OUTPUT:



git-job
succeeded 5 minutes ago in 5s

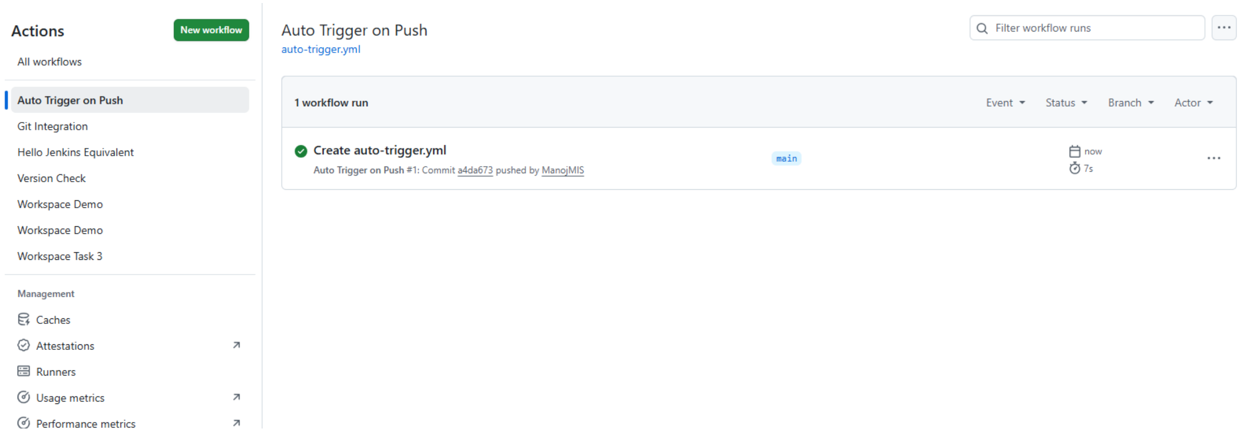
```
Set up job            1s
Checkout code         0s
List repo files       0s
Post Checkout code    1s
Complete job          0s
```



Workflow file for this run
.github/workflows/git-checkout.yml at 01b67a3

```
name: Git Integration

on: workflow_dispatch

jobs:
  git-job:
    runs-on: ubuntu-latest
    steps:
      - name: Checkout code
        uses: actions/checkout@v4

      - name: List repo files
        run: ls -R
```

5.)**<u>Auto Trigger Workflow created:</u>**

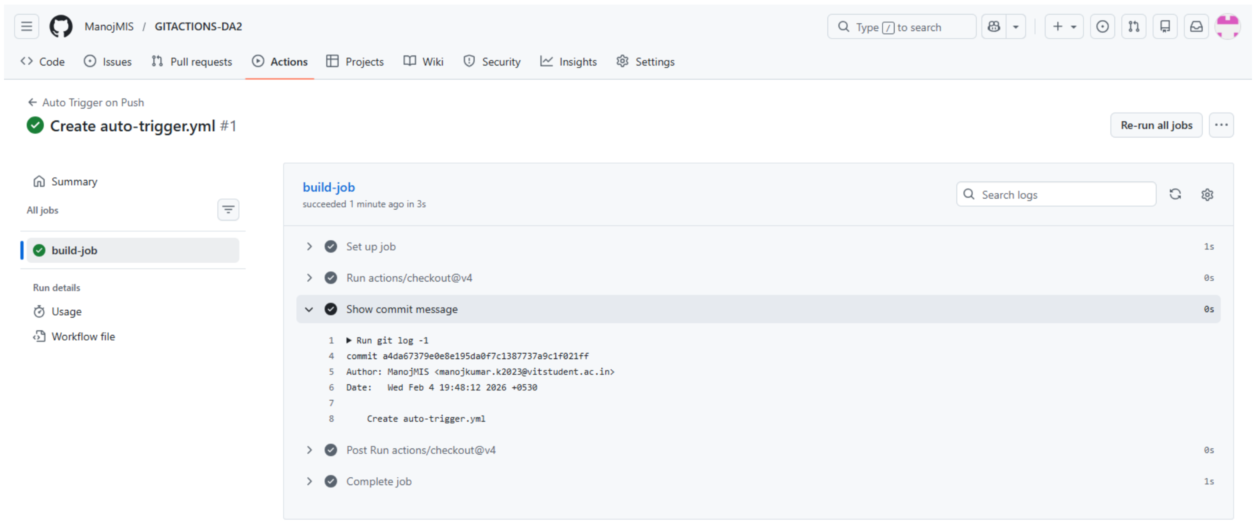


Auto Trigger on Push
auto-trigger.yml

1 workflow run

Create auto-trigger.yml — main — now — 7s
Auto Trigger on Push #1: Commit a4da673 pushed by ManojMIS

<u>OUTPUT:</u>

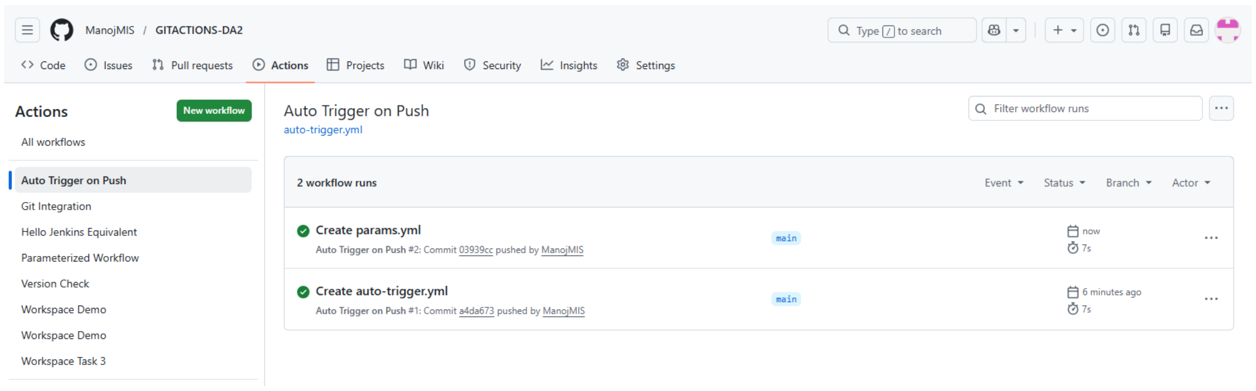


Create auto-trigger.yml #1

build-job
succeeded 1 minute ago in 3s

```
Set up job                    1s
Run actions/checkout@v4       0s
Show commit message           0s
▶ Run git log -1
commit a4da67379e0e8e195da0f7c1387737a9c1f021ff
Author: ManojMIS <manojkumar.k2023@vitstudent.ac.in>
Date:   Wed Feb 4 19:48:12 2026 +0530

    Create auto-trigger.yml
Post Run actions/checkout@v4  0s
Complete job                  1s
```

6.)**<u>Parameterized Workflow Created:</u>**



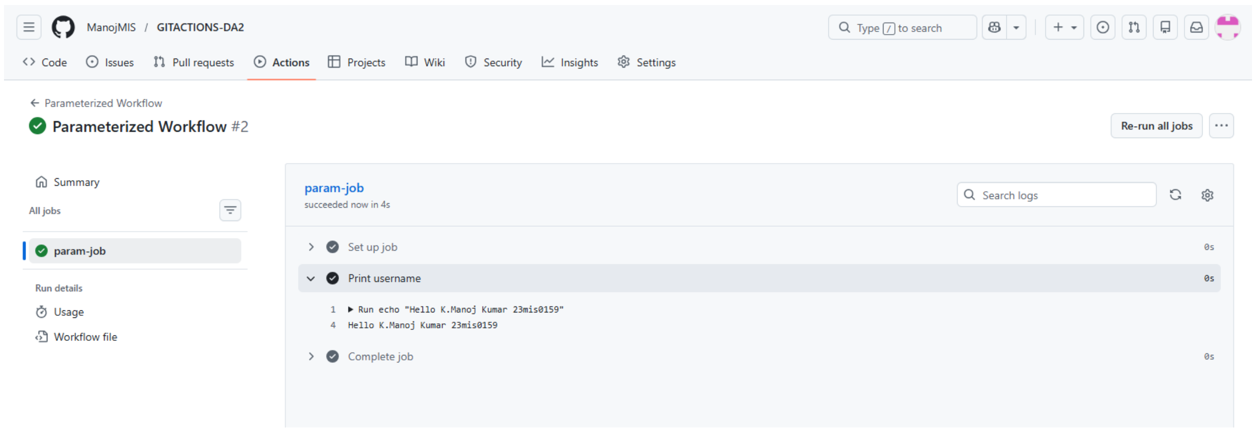
Auto Trigger on Push
auto-trigger.yml

2 workflow runs

Create params.yml
Auto Trigger on Push #2: Commit 03939cc pushed by ManojMIS — main — now — 7s

Create auto-trigger.yml
Auto Trigger on Push #1: Commit a4da673 pushed by ManojMIS — main — 6 minutes ago — 7s

<u>OUTPUT:</u>



Parameterized Workflow #2

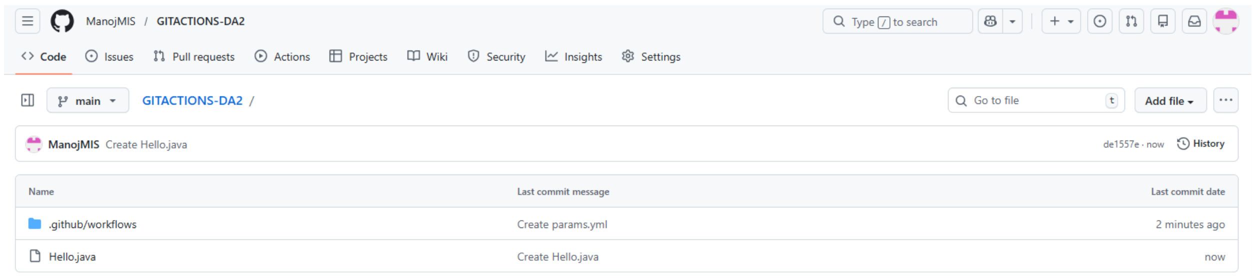
param-job
succeeded now in 4s

Set up job 0s
Print username 0s

```
▶ Run echo "Hello K.Manoj Kumar 23mis0159"
Hello K.Manoj Kumar 23mis0159
```

Complete job 0s

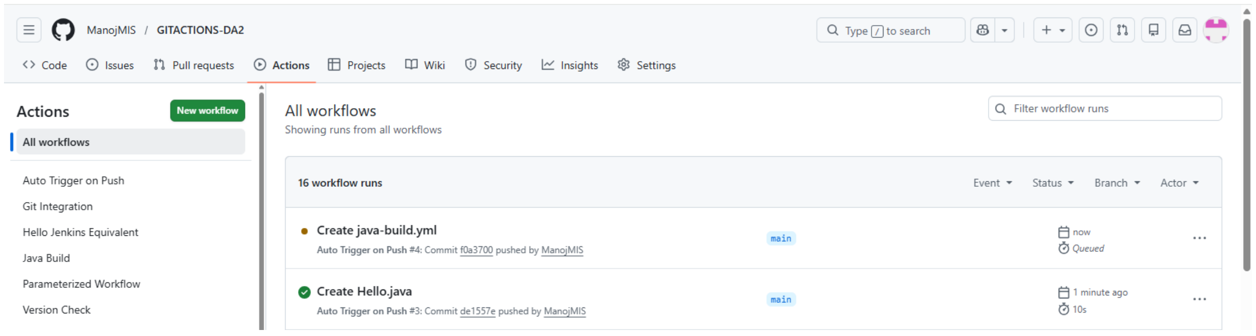
7.)**Hello.java FILE:**



ManojMIS Create Hello.java de1557e · now

Name	Last commit message	Last commit date
.github/workflows	Create params.yml	2 minutes ago
Hello.java	Create Hello.java	now

JAVA workflow created:



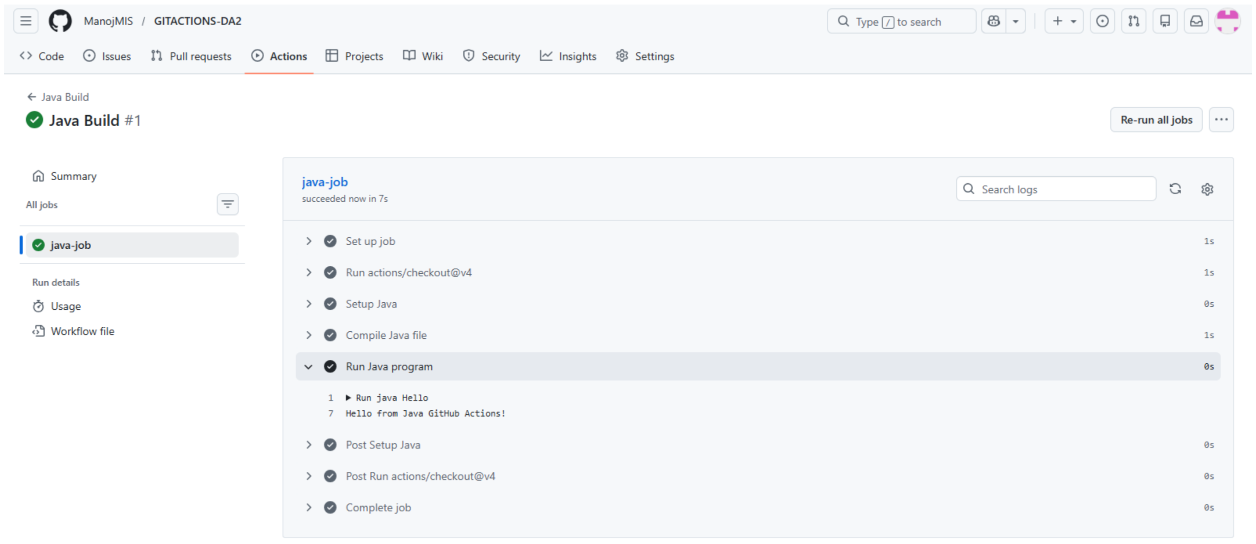
All workflows
Showing runs from all workflows

16 workflow runs

Create java-build.yml
Auto Trigger on Push #4: Commit f0a3700 pushed by ManojMIS · main · now · Queued

Create Hello.java
Auto Trigger on Push #3: Commit de1557e pushed by ManojMIS · main · 1 minute ago · 10s

OUTPUT:

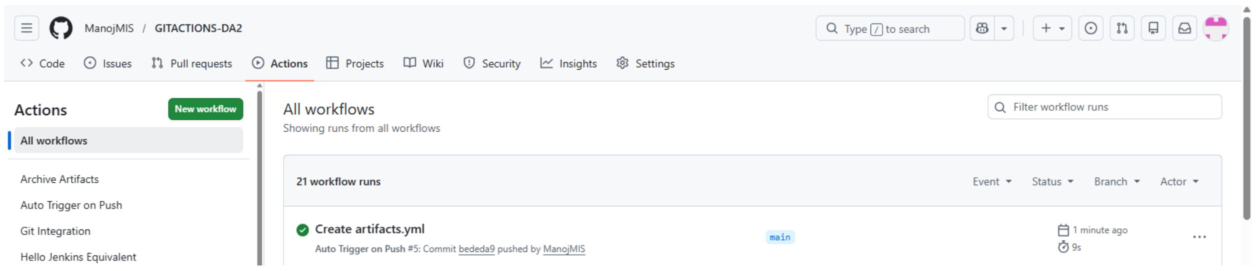


Java Build #1

java-job
succeeded now in 7s

```
Set up job                 1s
Run actions/checkout@v4    1s
Setup Java                 0s
Compile Java file          1s
Run Java program           0s
  1  ▶ Run java Hello
  7  Hello from Java GitHub Actions!
Post Setup Java            0s
Post Run actions/checkout@v4  0s
Complete job               0s
```

8.)Artifacts workflow Created:



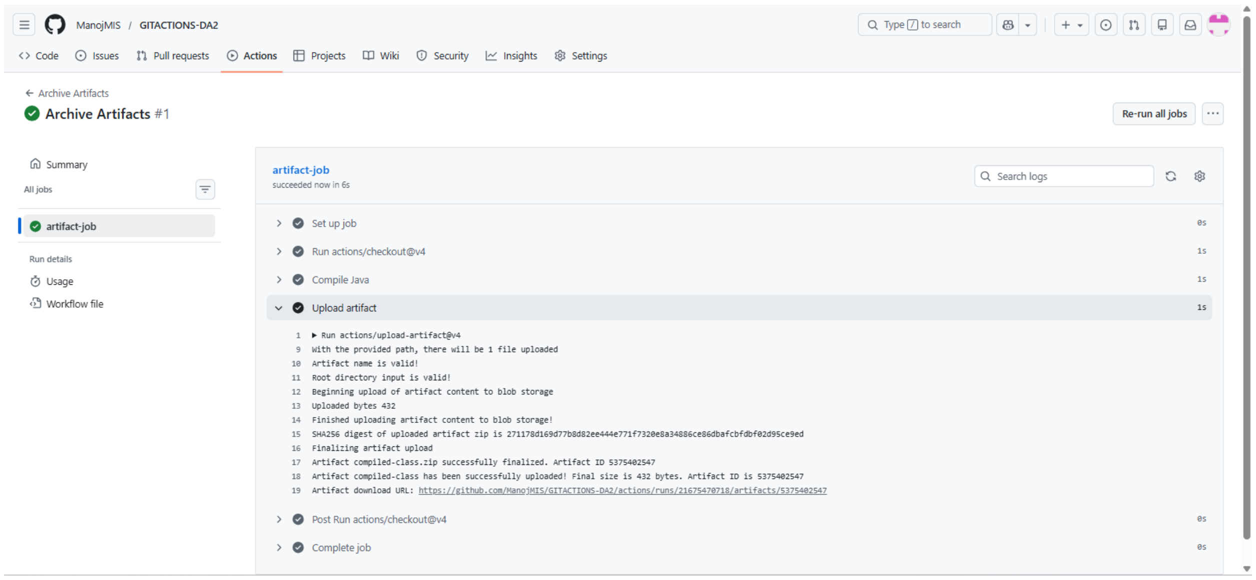
Actions — All workflows: Archive Artifacts, Auto Trigger on Push, Git Integration, Hello Jenkins Equivalent

All workflows
Showing runs from all workflows

21 workflow runs

Create artifacts.yml — main — 1 minute ago — 9s
Auto Trigger on Push #5: Commit bededa9 pushed by ManojMIS

OUTPUT:

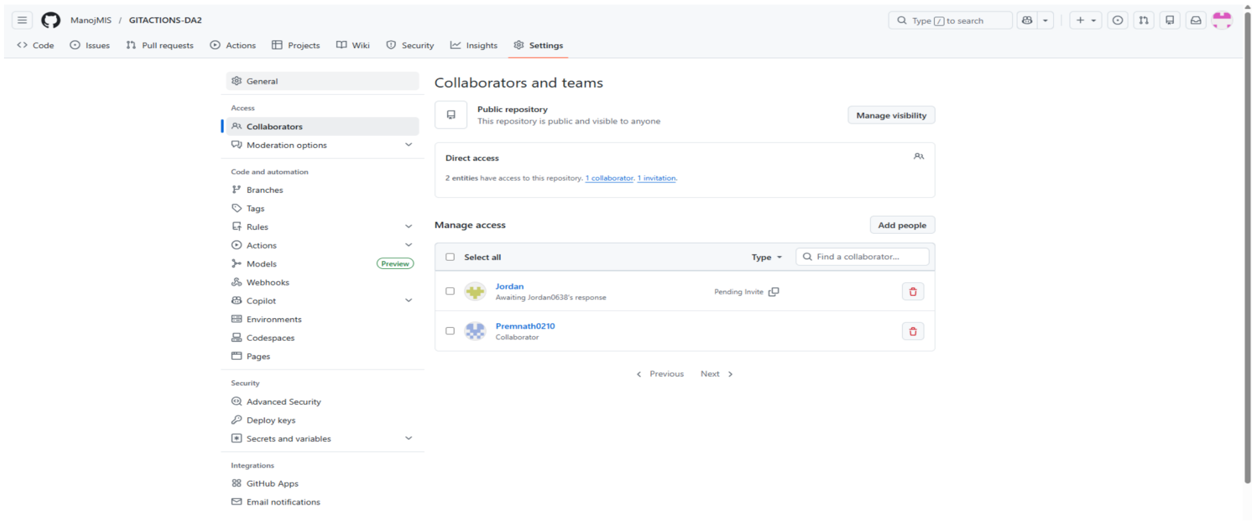


Archive Artifacts #1

artifact-job
succeeded now in 6s

```
Set up job                              0s
Run actions/checkout@v4                 1s
Compile Java                            1s
Upload artifact                         1s
▶ Run actions/upload-artifact@v4
With the provided path, there will be 1 file uploaded
Artifact name is valid!
Root directory input is valid!
Beginning upload of artifact content to blob storage
Uploaded bytes 432
Finished uploading artifact content to blob storage!
SHA256 digest of uploaded artifact zip is 271178d169d77b8d82ee444e771f7320e8a34886ce86dbafcbfdbf02d95ce9ed
Finalizing artifact upload
Artifact compiled-class.zip successfully finalized. Artifact ID 5375402547
Artifact compiled-class has been successfully uploaded! Final size is 432 bytes. Artifact ID is 5375402547
Artifact download URL: https://github.com/ManojMIS/GITACTIONS-DA2/actions/runs/21675470718/artifacts/5375402547
Post Run actions/checkout@v4            0s
Complete job                            0s
```

9.)GIT USERS:



Collaborators and teams

Public repository
This repository is public and visible to anyone

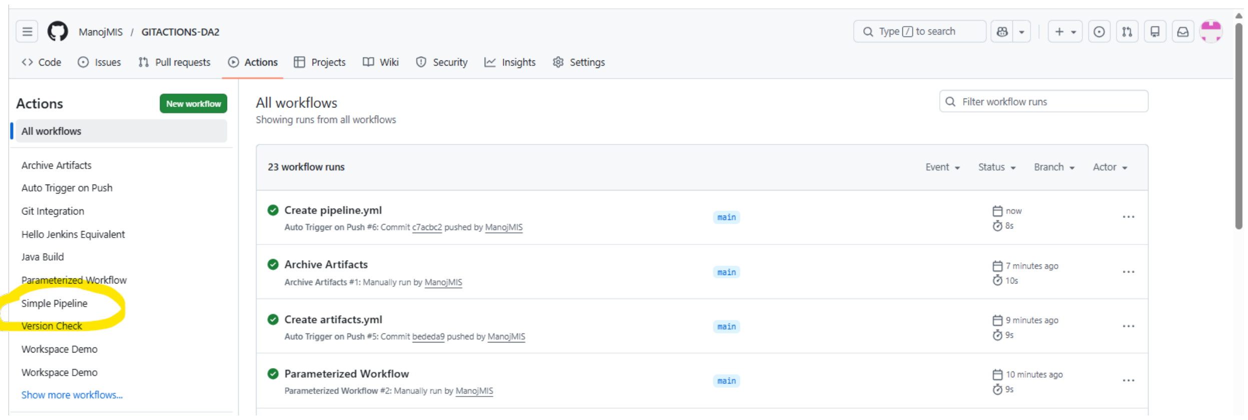
Direct access
2 entities have access to this repository. 1 collaborator. 1 invitation.

Manage access

Jordan	Awaiting Jordan0638's response	Pending Invite
Premnath0210	Collaborator	

10.)

Pipeline workflow created:


ManojMIS / GITACTIONS-DA2

Actions

All workflows
Showing runs from all workflows

23 workflow runs

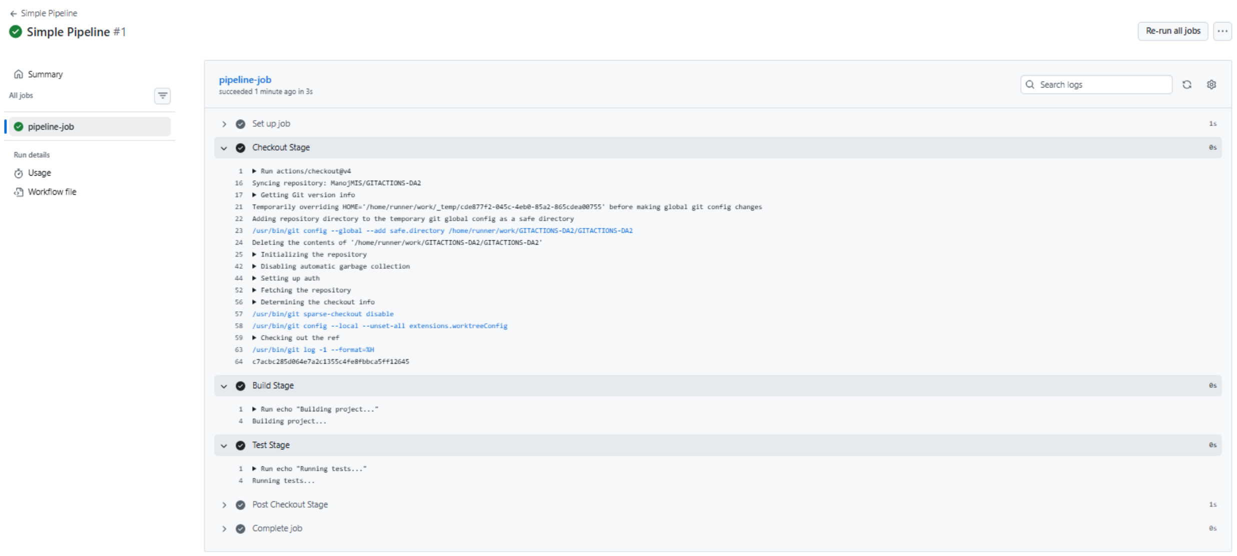
Create pipeline.yml
Auto Trigger on Push #6: Commit c7acbc2 pushed by ManojMIS — main — now — 8s

Archive Artifacts
Archive Artifacts #1: Manually run by ManojMIS — main — 7 minutes ago — 10s

Create artifacts.yml
Auto Trigger on Push #5: Commit bededa9 pushed by ManojMIS — main — 9 minutes ago — 9s

Parameterized Workflow
Parameterized Workflow #2: Manually run by ManojMIS — main — 10 minutes ago — 9s

OUTPUT:


Simple Pipeline #1

pipeline-job
succeeded 1 minute ago in 3s

Set up job — 1s

Checkout Stage — 0s

```
▶ Run actions/checkout@v4
Syncing repository: ManojMIS/GITACTIONS-DA2
▶ Getting Git version info
Temporarily overriding HOME='/home/runner/work/_temp/cde877f2-045c-4eb0-85a2-865cdea00755' before making global git config changes
Adding repository directory to the temporary git global config as a safe directory
/usr/bin/git config --global --add safe.directory /home/runner/work/GITACTIONS-DA2/GITACTIONS-DA2
Deleting the contents of '/home/runner/work/GITACTIONS-DA2/GITACTIONS-DA2'
▶ Initializing the repository
▶ Disabling automatic garbage collection
▶ Setting up auth
▶ Fetching the repository
▶ Determining the checkout info
/usr/bin/git sparse-checkout disable
/usr/bin/git config --local --unset-all extensions.worktreeConfig
▶ Checking out the ref
/usr/bin/git log -1 --format=%H
c7acbc285d064e7a2c1355c4fe8fbbca5ff12645
```

Build Stage — 0s

```
▶ Run echo "Building project..."
Building project...
```

Test Stage — 0s

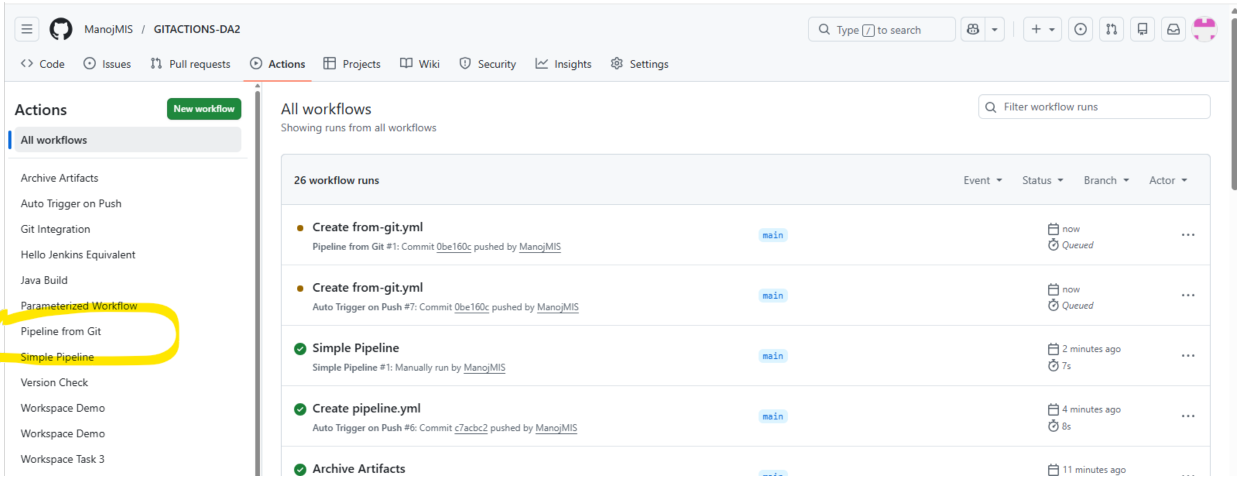
```
▶ Run echo "Running tests..."
Running tests...
```

Post Checkout Stage — 1s

Complete job — 0s

11.)

<u>Pipeline from GIT workflow created:</u>


ManojMIS / GITACTIONS-DA2

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

Actions — New workflow

All workflows
Archive Artifacts
Auto Trigger on Push
Git Integration
Hello Jenkins Equivalent
Java Build
Parameterized Workflow
Pipeline from Git
Simple Pipeline
Version Check
Workspace Demo
Workspace Demo
Workspace Task 3

All workflows
Showing runs from all workflows

26 workflow runs

Create from-git.yml — Pipeline from Git #1: Commit 0be160c pushed by ManojMIS — main — now — Queued

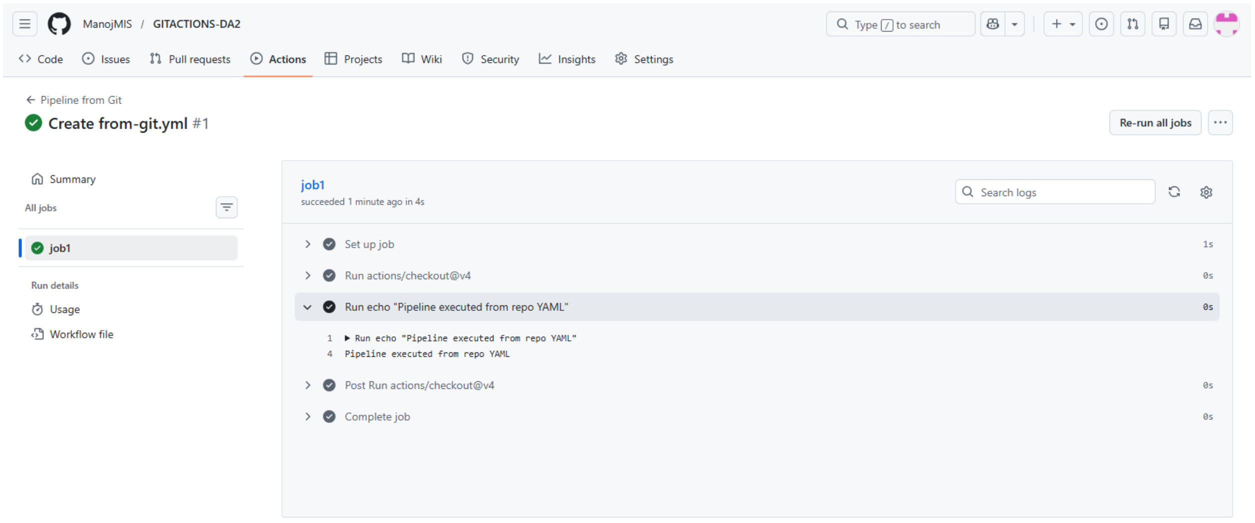
Create from-git.yml — Auto Trigger on Push #7: Commit 0be160c pushed by ManojMIS — main — now — Queued

Simple Pipeline — Simple Pipeline #1: Manually run by ManojMIS — main — 2 minutes ago — 7s

Create pipeline.yml — Auto Trigger on Push #6: Commit c7acbc2 pushed by ManojMIS — main — 4 minutes ago — 8s

Archive Artifacts — 11 minutes ago

<u>OUTPUT:</u>


ManojMIS / GITACTIONS-DA2

Pipeline from Git
Create from-git.yml #1

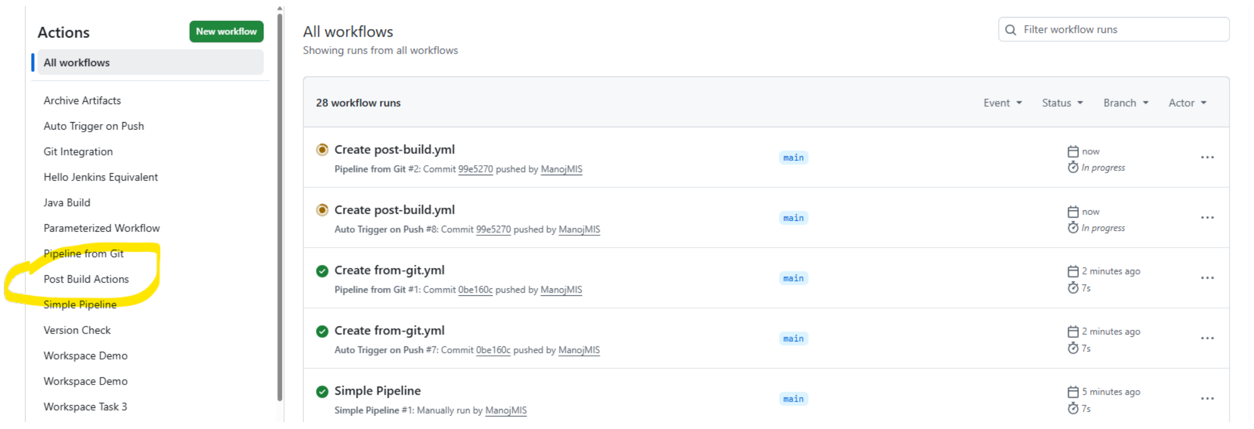
Re-run all jobs

job1
succeeded 1 minute ago in 4s

```
Set up job                                       1s
Run actions/checkout@v4                          0s
Run echo "Pipeline executed from repo YAML"      0s
  ▶ Run echo "Pipeline executed from repo YAML"
  Pipeline executed from repo YAML
Post Run actions/checkout@v4                     0s
Complete job                                     0s
```

12.)

Post-build workflow created:


Actions New workflow

All workflows

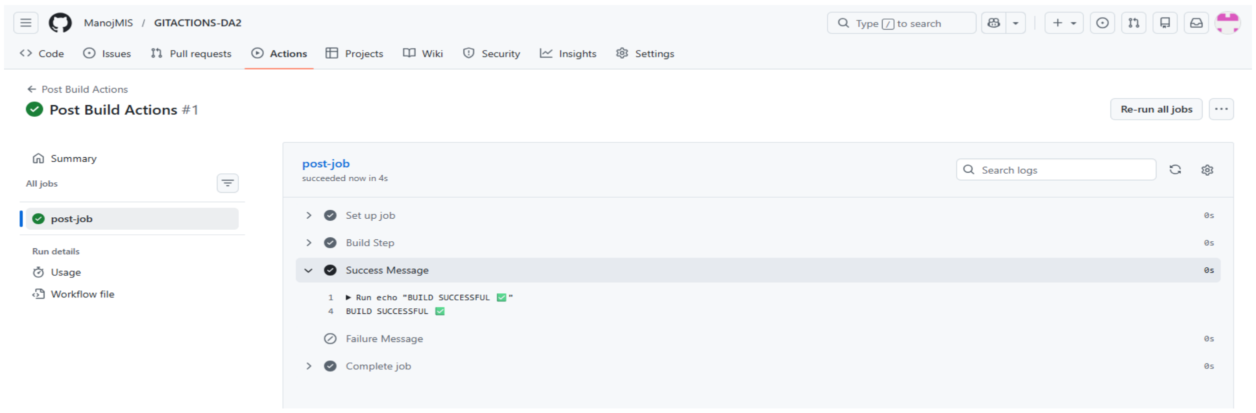
Archive Artifacts
Auto Trigger on Push
Git Integration
Hello Jenkins Equivalent
Java Build
Parameterized Workflow
Pipeline from Git
Post Build Actions
Simple Pipeline
Version Check
Workspace Demo
Workspace Demo
Workspace Task 3

All workflows
Showing runs from all workflows

28 workflow runs

Run	Branch	Time	Status
Create post-build.yml Pipeline from Git #2: Commit 99e5270 pushed by ManojMIS	main	now	In progress
Create post-build.yml Auto Trigger on Push #8: Commit 99e5270 pushed by ManojMIS	main	now	In progress
Create from-git.yml Pipeline from Git #1: Commit 0be160c pushed by ManojMIS	main	2 minutes ago	7s
Create from-git.yml Auto Trigger on Push #7: Commit 0be160c pushed by ManojMIS	main	2 minutes ago	7s
Simple Pipeline Simple Pipeline #1: Manually run by ManojMIS	main	5 minutes ago	7s

OUTPUT:


ManojMIS / GITACTIONS-DA2

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

← Post Build Actions
Post Build Actions #1 — Re-run all jobs

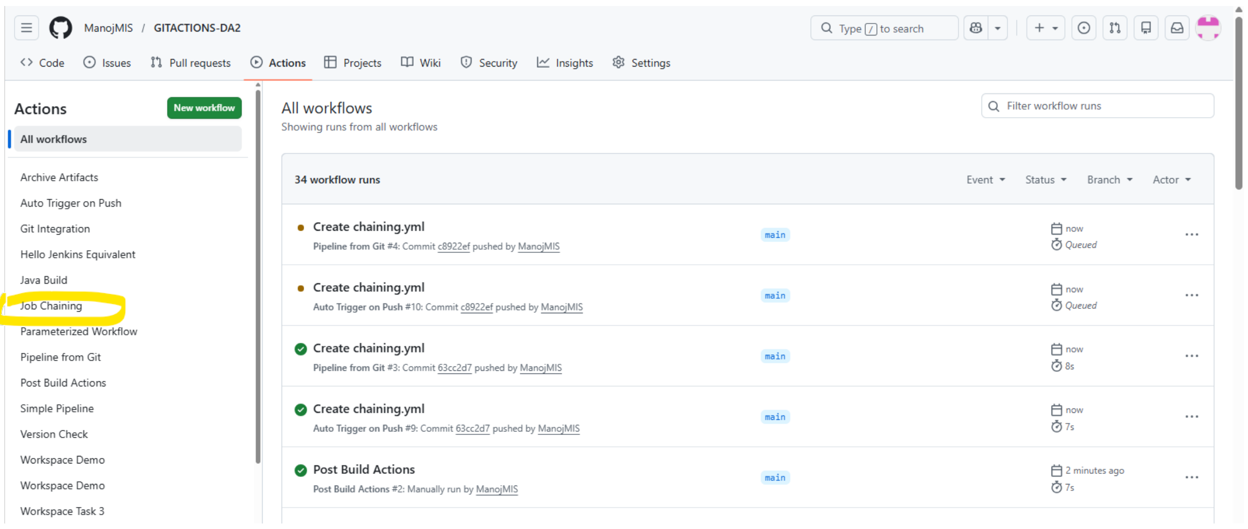
Summary
All jobs
post-job
Run details
Usage
Workflow file

post-job
succeeded now in 4s

```
Set up job            0s
Build Step            0s
Success Message       0s
  ▶ Run echo "BUILD SUCCESSFUL ✅"
  BUILD SUCCESSFUL ✅
Failure Message       0s
Complete job          0s
```

13.)

Chaining workflow created:


ManojMIS / GITACTIONS-DA2

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

Actions New workflow

All workflows

Archive Artifacts
Auto Trigger on Push
Git Integration
Hello Jenkins Equivalent
Java Build
Job Chaining
Parameterized Workflow
Pipeline from Git
Post Build Actions
Simple Pipeline
Version Check
Workspace Demo
Workspace Demo
Workspace Task 3

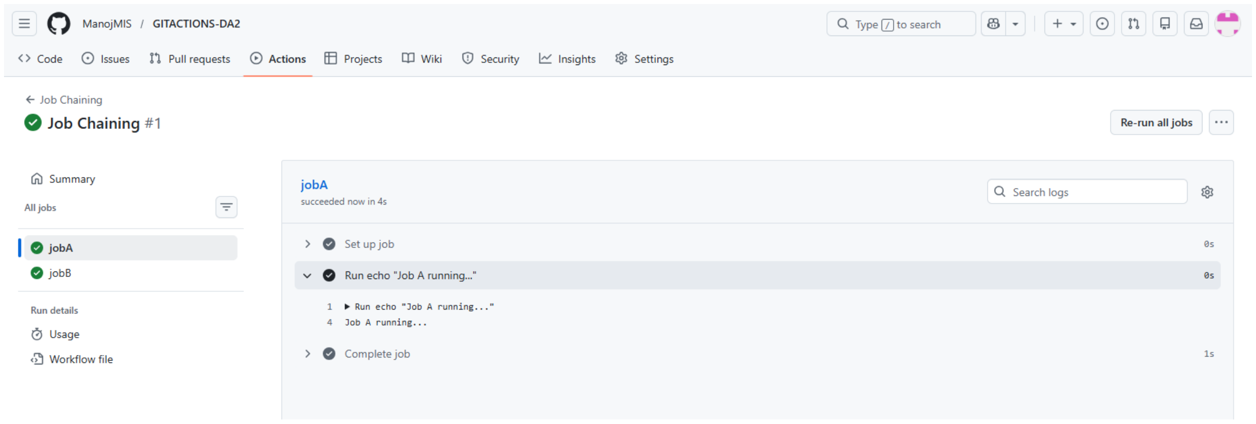
All workflows
Showing runs from all workflows

34 workflow runs

Run	Branch	Time	Status
Create chaining.yml Pipeline from Git #4: Commit c8922ef pushed by ManojMIS	main	now	Queued
Create chaining.yml Auto Trigger on Push #10: Commit c8922ef pushed by ManojMIS	main	now	Queued
Create chaining.yml Pipeline from Git #3: Commit 63cc2d7 pushed by ManojMIS	main	now	8s
Create chaining.yml Auto Trigger on Push #9: Commit 63cc2d7 pushed by ManojMIS	main	now	7s
Post Build Actions Post Build Actions #2: Manually run by ManojMIS	main	2 minutes ago	7s

OUTPUT:

JOB-A:



Job Chaining #1

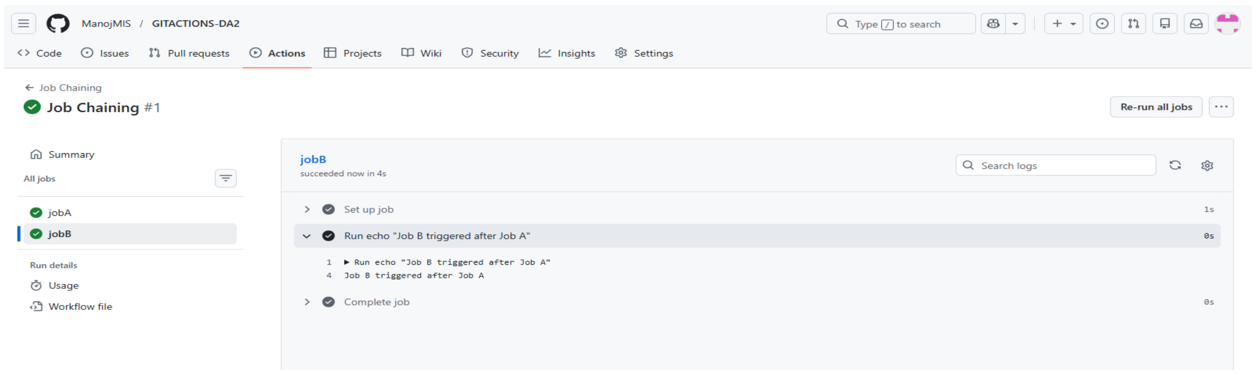
jobA
succeeded now in 4s

Set up job 0s
Run echo "Job A running..." 0s

```
▶ Run echo "Job A running..."
Job A running...
```

Complete job 1s

JOB-B:



Job Chaining #1

jobB
succeeded now in 4s

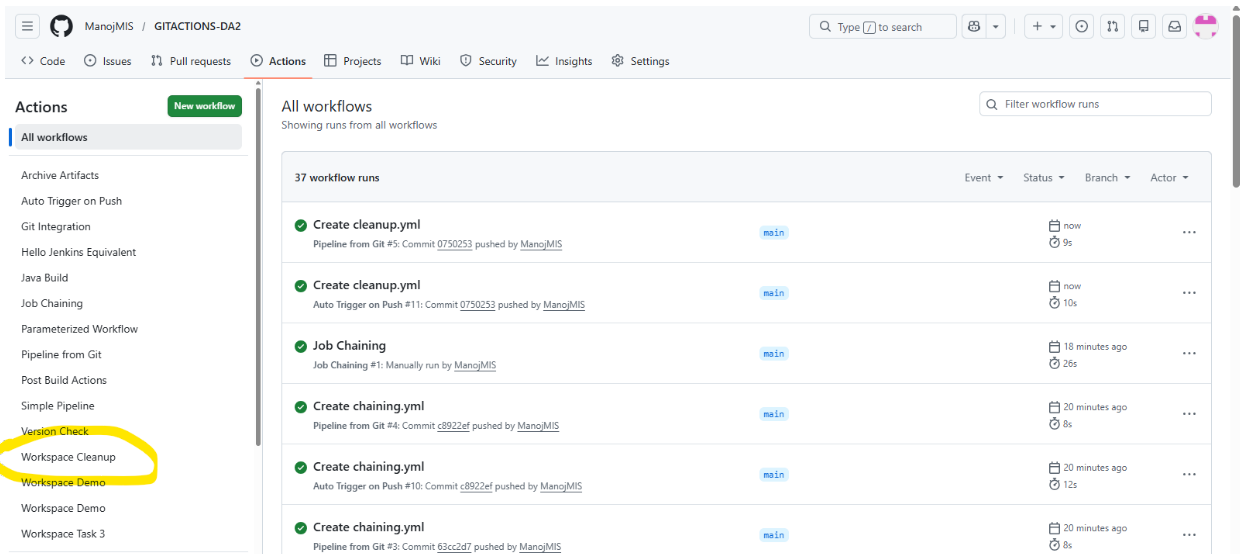
Set up job 1s
Run echo "Job B triggered after Job A" 0s

```
▶ Run echo "Job B triggered after Job A"
Job B triggered after Job A
```

Complete job 0s

14.)

Cleanup Workflow created:



All workflows
Showing runs from all workflows

37 workflow runs

Create cleanup.yml
Pipeline from Git #5: Commit 0750253 pushed by ManojMIS — main — now — 9s

Create cleanup.yml
Auto Trigger on Push #11: Commit 0750253 pushed by ManojMIS — main — now — 10s

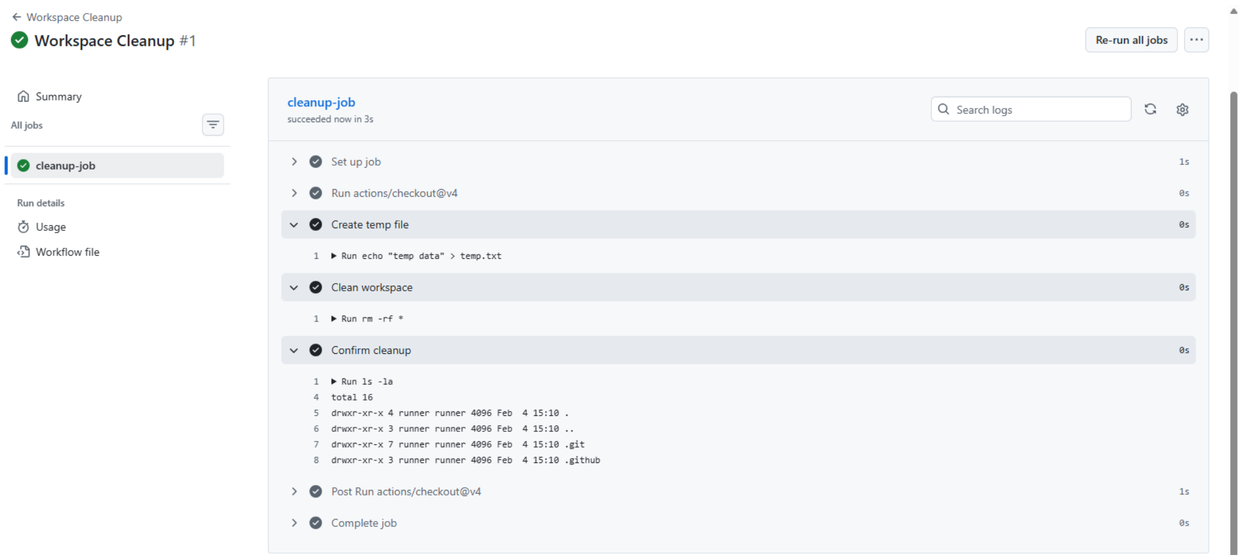
Job Chaining
Job Chaining #1: Manually run by ManojMIS — main — 18 minutes ago — 26s

Create chaining.yml
Pipeline from Git #4: Commit c8922ef pushed by ManojMIS — main — 20 minutes ago — 8s

Create chaining.yml
Auto Trigger on Push #10: Commit c8922ef pushed by ManojMIS — main — 20 minutes ago — 12s

Create chaining.yml
Pipeline from Git #3: Commit 63cc2d7 pushed by ManojMIS — main — 20 minutes ago — 8s

OUTPUT:



← Workspace Cleanup

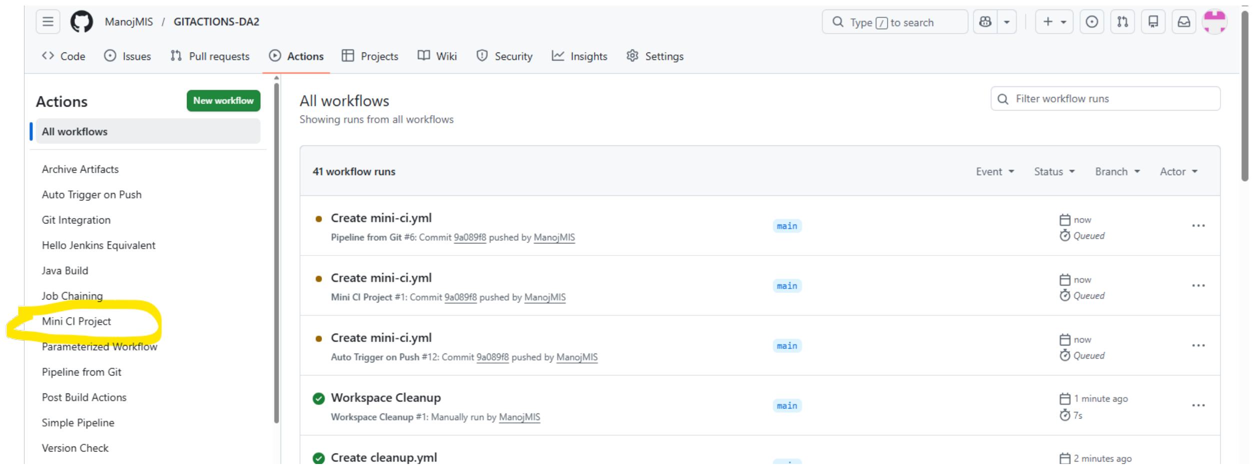
Workspace Cleanup #1

cleanup-job
succeeded now in 3s

```
Set up job                                   1s
Run actions/checkout@v4                      0s
Create temp file                             0s
  ▶ Run echo "temp data" > temp.txt
Clean workspace                              0s
  ▶ Run rm -rf *
Confirm cleanup                              0s
  ▶ Run ls -la
  total 16
  drwxr-xr-x 4 runner runner 4096 Feb  4 15:10 .
  drwxr-xr-x 3 runner runner 4096 Feb  4 15:10 ..
  drwxr-xr-x 7 runner runner 4096 Feb  4 15:10 .git
  drwxr-xr-x 3 runner runner 4096 Feb  4 15:10 .github
Post Run actions/checkout@v4                 1s
Complete job                                 0s
```

15.)

CI workflow created:



ManojMIS / GITACTIONS-DA2

Actions

All workflows

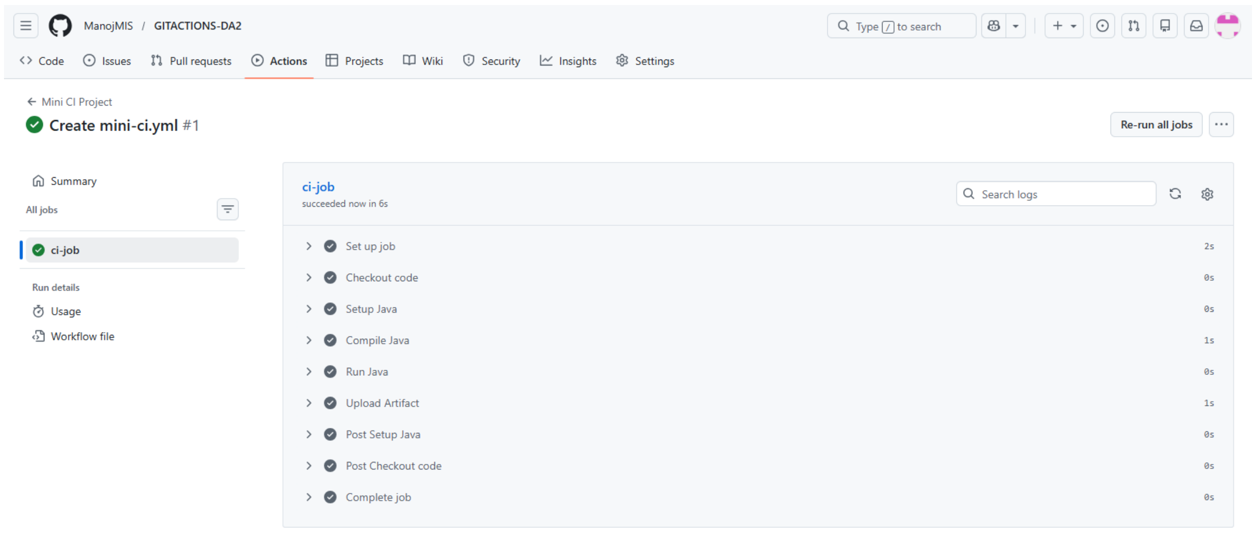
Archive Artifacts
Auto Trigger on Push
Git Integration
Hello Jenkins Equivalent
Java Build
Job Chaining
Mini CI Project
Parameterized Workflow
Pipeline from Git
Post Build Actions
Simple Pipeline
Version Check

All workflows
Showing runs from all workflows

41 workflow runs

Create mini-ci.yml Pipeline from Git #6: Commit 9a089f8 pushed by ManojMIS	main	now Queued
Create mini-ci.yml Mini CI Project #1: Commit 9a089f8 pushed by ManojMIS	main	now Queued
Create mini-ci.yml Auto Trigger on Push #12: Commit 9a089f8 pushed by ManojMIS	main	now Queued
Workspace Cleanup Workspace Cleanup #1: Manually run by ManojMIS	main	1 minute ago 7s
Create cleanup.yml		2 minutes ago

OUTPUT:



ManojMIS / GITACTIONS-DA2

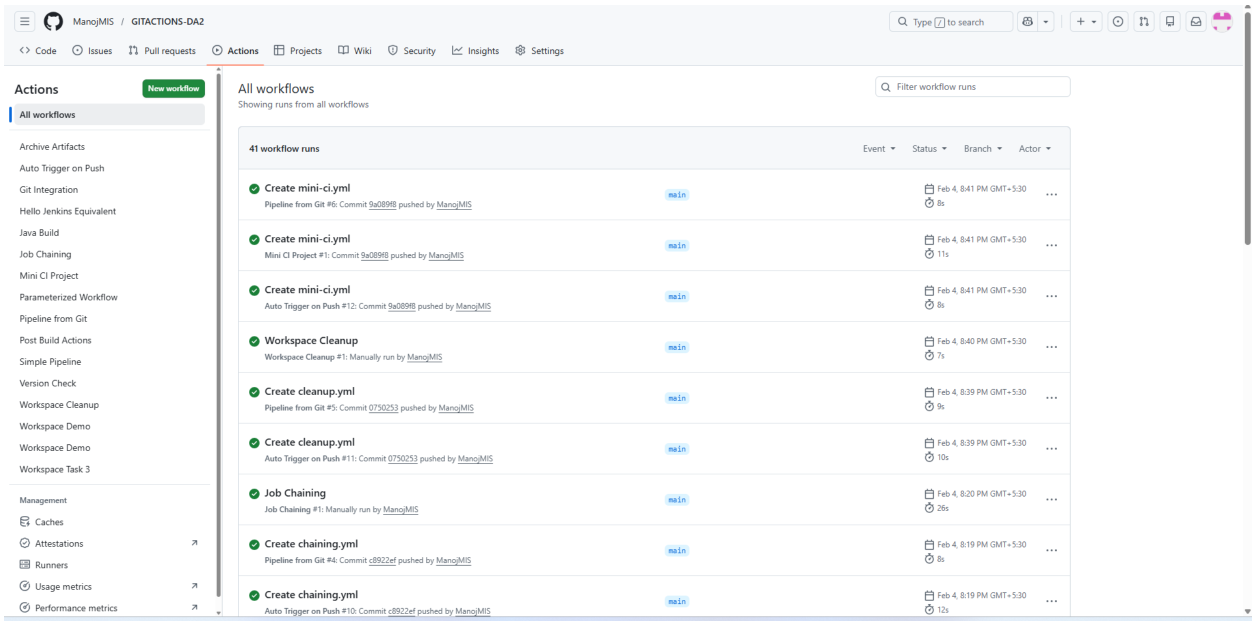
Mini CI Project

Create mini-ci.yml #1

ci-job
succeeded now in 6s

```
Set up job             2s
Checkout code          0s
Setup Java             0s
Compile Java           1s
Run Java               0s
Upload Artifact        1s
Post Setup Java        0s
Post Checkout code     0s
Complete job           0s
```

ALL THE WORKFLOWS CREATED:



ManojMIS / GITACTIONS-DA2

Actions — All workflows

Archive Artifacts, Auto Trigger on Push, Git Integration, Hello Jenkins Equivalent, Java Build, Job Chaining, Mini CI Project, Parameterized Workflow, Pipeline from Git, Post Build Actions, Simple Pipeline, Version Check, Workspace Cleanup, Workspace Demo, Workspace Demo, Workspace Task 3

All workflows
Showing runs from all workflows

41 workflow runs

```
Create mini-ci.yml                                                       main   Feb 4, 8:41 PM GMT+5:30   8s
Pipeline from Git #6: Commit 9a089f8 pushed by ManojMIS
Create mini-ci.yml                                                       main   Feb 4, 8:41 PM GMT+5:30   11s
Mini CI Project #1: Commit 9a089f8 pushed by ManojMIS
Create mini-ci.yml                                                       main   Feb 4, 8:41 PM GMT+5:30   8s
Auto Trigger on Push #12: Commit 9a089f8 pushed by ManojMIS
Workspace Cleanup                                                        main   Feb 4, 8:40 PM GMT+5:30   7s
Workspace Cleanup #1: Manually run by ManojMIS
Create cleanup.yml                                                       main   Feb 4, 8:39 PM GMT+5:30   9s
Pipeline from Git #5: Commit 0750253 pushed by ManojMIS
Create cleanup.yml                                                       main   Feb 4, 8:39 PM GMT+5:30   10s
Auto Trigger on Push #11: Commit 0750253 pushed by ManojMIS
Job Chaining                                                             main   Feb 4, 8:20 PM GMT+5:30   26s
Job Chaining #1: Manually run by ManojMIS
Create chaining.yml                                                      main   Feb 4, 8:19 PM GMT+5:30   8s
Pipeline from Git #4: Commit c8922ef pushed by ManojMIS
Create chaining.yml                                                      main   Feb 4, 8:19 PM GMT+5:30   12s
Auto Trigger on Push #10: Commit c8922ef pushed by ManojMIS
```