



Week 13 : SOFTWARE DEVELOPMENT TOOLS AND ENVIRONMENTS

LAB 01 : Run Jenkinfile using SCM From Github

In Jenkins, "SCM" stands for Source Code Management. SCM in Jenkins refers to the process of managing and integrating source code from version control systems into Jenkins projects.

Tuchsanai / DevTools

Code Issues Pull requests Actions Projects Security Insights Settings

main DevTools / 03_Jenkins / week13 / 01_Run_Jenkinfile_From_Github_WithSCM /

Tuchsanai d c89be10 · now History

Name	Last commit message	Last commit date
JENKINFILE	d	now
readme.md	ww	1 minute ago
status.py	ww	1 minute ago
readme.md		

Tuchsanai / DevTools

Type ⌘ to search

Code Issues Pull requests Actions Projects Security Insights Settings

dev DevTools / 03_Jenkins / week13 / 01_Run_Jenkinfile_From_Github / status.py

Tuchsanai s

Code Blame 14 lines (12 loc) · 286 Bytes

```
1 import os
2
3 def main():
4     # Displaying status
5     print("System Status:")
6     print("-----")
7     print("Operating System:", os.name)
8     print("\n")
9     print("Environment Variables:", os.environ)
10    print("\n")
11    print("finished.")
12
13 if __name__ == "__main__":
14     main()
```

main DevTools / 03_Jenkins / week13 / 01_Run_Jenkinfile_From_Github / JENKINFILE

Tuchsanai aa 65dd21b · 6 minutes ago History

29 lines (27 loc) · 814 Bytes

Code Blame Raw ⌂ ⌂ ⌂ ⌂ ⌂

```
1 pipeline {
2     agent any
3
4     stages {
5         stage('Check Python Installation') {
6             steps {
7                 script {
8                     // Check if Python is installed
9                     def pythonInstalled = sh(script: "which python3", returnStatus: true) == 0
10                    if (!pythonInstalled) {
11                        // Install Python if not installed
12                        sh 'sudo apt update'
13                        sh 'sudo apt install -y python3'
14                    }
15                }
16            }
17        }
18
19        stage('Run Python Script') {
20            steps {
21                script {
22                    // Run Python script with only os library
23                    sh 'python3 status.py'
24                    sh 'ls -l'
25                }
26            }
27        }
28    }
29 }
```

1

Jenkins

Dashboard >

+ New Item

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue

No builds in the queue.

Build Executor Status

1 Idle

2 Idle

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

2

Jenkins

Search (⌘+K)

Tuchsanai Ploysuwan

Dashboard > All >

Enter an item name

jenkins_file

» Required field

3

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.

OK

Build Triggers

Configure

General

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

Advanced Project Options

Pipeline

Advanced Project Options

Advanced ▾

Pipeline

Definition

- Pipeline script
- Pipeline script from SCM

Script ?

1

try sample Pipeline... ▾

4

DevTools / 03_Jenkins / week13
/ 01_Run_Jenkinfile_From_Github_WithSCM
/ JENKINFILE

Tuchsanai d c89be10 · 1 minute ago History

30 lines (27 loc) · 894 Bytes

Code Blame

```
1 pipeline {  
2     agent any  
3  
4     stages {  
5         stage('Check Python Installation') {  
6             steps {  
7                 script {  
8                     // Check if Python is installed  
9                     def pythonInstalled = sh(script: "which python3", returnStatus: true) == 0  
10                    if (!pythonInstalled) {  
11                        // Install Python if not installed  
12                        sh 'sudo apt update'  
13                        sh 'sudo apt install -y python3'  
14                    }  
15                }  
16            }  
17        }  
18  
19        stage('Run Python Script') {  
20            steps {  
21                script {  
22                    // Run Python script with only os library  
23                    sh 'ls -l'  
24                    sh 'python3 ./03_Jenkins/week13/01_Run_Jenkinfile_From_Github_WithSCM/status.py'  
25                }  
26            }  
27        }  
28    }  
29}  
30}
```

Configure

General Build Triggers Advanced Project Options Pipeline

Build Triggers

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

Advanced Project Options

Copy path of Jenkinfile

Pipeline Definition Pipeline script from SCM

SCM

- None
- Git

Script Path

/03_Jenkins/week13/01_Run_Jenkinfile_From_Github_WithSCM/JENKINFILE

Lightweight checkout

Pipeline Syntax

Save Apply

7

Repository URL ?
`https://github.com/Tuchsanai/DevTools.git`

Credentials ?
tuchsanai/******** (github account)

+ Add ▾

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?
*/dev

Add Branch

Repository browser ?
(Auto)

Additional Behaviours

Add ▾

Script Path ?
./03_Jenkins/week13/01_Run_Jenkinfile_From_Github_WithSCM/JENKINFILE

Save Apply

8 # If privated repo, you need credentials

9 # Selected dev branch

10

Jenkins

Dashboard > jenkins_file >

jenkins_file

- Status
- </> Changes
- Build Now
- Configure
- Delete Pipeline
- Full Stage View
- Rename
- Pipeline Syntax

Stage View

Declarative: Checkout SCM	Check Python Installation	Run Python Script
Average stage times: (Average full run time: ~3s) #3 n.w. 27 10:44 2 commits	1s 394ms 982ms	1s 394ms 982ms

Permalinks

- Last build (#3), 2 min 15 sec ago
- Last stable build (#3), 2 min 15 sec ago
- Last successful build (#3), 2 min 15 sec ago
- Last completed build (#3), 2 min 15 sec ago

Dashboard > jenkins_file > #3

Console Output

Started by user Tuchsanai Ploysuwan
Obtained ./.jenkins/week13/01_Run_Jenkinfile_From_Github/JENKINFILE from git https://github.com/Tuchsanai/DevTools.git
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/lib/jenkins/workspace/jenkins_file
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
The recommended git tool is: git
using credential id_github
> git rev-parse --resolve-git-dir /var/lib/jenkins/workspace/jenkins_file/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url https://github.com/Tuchsanai/DevTools.git # timeout=10
Fetching upstream changes from https://github.com/Tuchsanai/DevTools.git
> git --version # timeout=10
git version # git version 2.34.1
using GIT_ASKPASS to set credentials github account
> git fetch --tags --force --progress -- https://github.com/Tuchsanai/DevTools.git +refs/heads/*:refs/remotes/origin/* # timeout=10
[Pipeline] { (Check Python Installation)
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ which python3
/usr/bin/python3
[Pipeline] }
[Pipeline] // stage
[Pipeline] withEnv
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Run Python Script)
[Pipeline] script
[Pipeline] {
[Pipeline] sh
+ pwd
/var/lib/jenkins/workspace/jenkins_file
[Pipeline] sh
+ python3 ./03_Jenkins/week13/01_Run_Jenkinfile_From_Github/status.py
System Status:
Operating System: posix

Environment Variables: environ({'JENKINS_HOME': '/var/lib/jenkins', 'USER': 'jenkins', 'CI': 'true', 'RUN_CHANGES_DISPLAY_URL': 'http://54.251.188.144:8080/job/jenkins_file/3/display/redirect?page=changes', 'NODE_LABELS': 'built-in', 'HUDSON_URL': 'http://54.251.188.144:8080', 'GIT_COMMIT': 'ad4c6fcbe6618509afecc13d26c1406fd5dbf3', 'HUDSON_COOKIE': '49a06cf3-5d30-418d-b9f6-8226', 'HUDSON_SERVER_COOKIE': 'urable-d5f3f7cf430a25df33ccb3cf7f939bc2452d2ea1b2523a2a26025dc82651f', 'NOTIFY_SOCKET': '/run/systemd/notify', 'SYSTEM_EXEC_PID': '8226', 'WORKSPACE': '/var/lib/jenkins/workspace/jenkins_file', 'LOGNAME': 'jenkins', 'NODE_NAME': 'built-in', 'JOURNAL_STREAM': '8:46337', 'RUN_ARTIFACTS_DISPLAY_URL': 'http://54.251.188.144:8080/job/jenkins_file/3/display/redirect?page=artifacts', 'STAGE_NAME': 'Run Python Script', 'EXECUTOR_NUMBER': '0', 'GIT_BRANCH': 'origin/dev', 'RUN_TESTS_DISPLAY_URL': 'http://54.251.188.144:8080/job/jenkins_file/3/display/redirect?page=tests', 'BUILD_DISPLAY_NAME': '#3', 'HUDSON_HOME': '/var/lib/jenkins', 'JOB_BASENAME': 'jenkins_file', 'PATH': '/usr/local/bin:/usr/bin:/bin:/sbin:/bin:/snap/bin', 'INVOCATION_ID': '3dd1667cf5244899512e0c5fa349e4', 'BUILD_ID': '3', 'BUILD_TAG': 'jenkins-jenkins_file-3', 'LANG': 'C.UTF-8', 'JENKINS_URL': 'http://54.251.188.144:8080', 'JOB_URL': 'http://54.251.188.144:8080/job/jenkins_file/', 'GIT_URL': 'https://github.com/Tuchsanai/DevTools.git', 'HUDSON_SERVER_COOKIE': '25d157583dcefe', 'JOB_DISPLAY_URL': 'http://54.251.188.144:8080/job/jenkins_file/3/display/redirect', 'HUDSON_COOKIE': '25d157583dcefe', 'JOB_NAME': 'jenkins_file', 'PWD': '/var/lib/jenkins/workspace/jenkins_file', 'GIT_PREVIOUS_COMMIT': '32de38e6ab198bf44721d9a241c74dc5f92a427', 'WORKSPACE_TMP': '/var/lib/jenkins/workspace/jenkins_file@tmp'})
finished.

LAB 02 : Run Jenkinfile without SCM From Github

Jenkins

Dashboard >

+ New Item 1

People

Build History

Project Relationship

Check File Fingerprint

Manage Jenkins

My Views

Build Queue No builds in the queue.

Build Executor Status 1 Idle
2 Idle

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 monitor icon

Configure a cloud cloud icon

Learn more about distributed builds ?

Enter an item name

No_SCM » Required field

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 2
 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.

OK

Configure**General** Advanced Project Options**Pipeline**

- Throttle builds ?
- Build Triggers**
- Build after other projects are built ?
- Build periodically ?
- GitHub hook trigger for GITScm polling ?
- Poll SCM ?
- Quiet period ?
- Trigger builds remotely (e.g., from scripts) ?

Advanced Project Options

Advanced ▾

Pipeline**Definition**

Pipeline script

```

1> pipeline {
2>   agent any
3>
4>   stages {
5>     stage('copy repository') {
6>       steps {
7>         sh // Copy the repository
8>       }
9>     }
10>    stage('Check Python Installation') {
11>      steps {
12>        sh
13>      }
14>    }
15>  }
16>
```

Use Groovy Sandbox ?

Copy code

3

Jenkins file with GitHub**- detail in slides**

```

pipeline {
  agent any
  stages {
    stage('copy repository') {
      steps {
        sh // Copy the repository
        checkout scmGit(branches: [[name: './dev']], extensions: [], userRemoteConfigs: [[credentialsId: 'g
      }
    }
    stage('Check Python Installation') {
      steps {
        script {
          // Check if Python is installed
          def pythonInstalled = sh(script: "which python3", returnStatus: true) == 0
          if (!pythonInstalled) {
            // Install Python if not installed
            sh 'sudo apt update'
            sh 'sudo apt install -y python3'
          }
        }
      }
    }
    stage('Run Python Script') {
      steps {
        script {
          // Run Python script with only os library
          sh 'pwd'
          sh 'python3 ./03_Jenkins/week13/02_Run_Jenkinfile_From_Github_withoutSCM/status.py'
          sh 'ls -l'
        }
      }
    }
  }
}
```

4

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various steps. Pick a step you are interested in from the list, configure it, click **Generate Pipeline Script**, and you will see a Pipeline Script statement that would call the step with that configuration. You may copy and paste the whole statement into your script, or pick up just the options you care about. (Most parameters are optional and can be omitted in your script, leaving them at default values.)

Steps

Sample Step

archiveArtifacts: Archive the artifacts

archiveArtifacts

Files to archive ?

Advanced ▾

Generate Pipeline Script

Global Variables

There are many features of the Pipeline that are not steps. These are often exposed via global variables, which are not supported by the snippet generator. See the [Global Variables Reference](#) for details.

5

Back

Overview

This **Snippet Generator** will help you learn the Pipeline Script code which can be used to define various steps. Pick a step you are interested in from the list, configure it, click **Generate Pipeline Script**, and you will see a Pipeline Script statement that would call the step with that configuration. You may copy and paste the whole statement into your script, or pick up just the options you care about. (Most parameters are optional and can be omitted in your script, leaving them at default values.)

Steps

Sample Step

- ✓ archiveArtifacts: Archive the artifacts
bat: Windows Batch Script
build: Build a job
- catchError: Catch error and set build result to failure**
- checkout: Check out from version control**
- cleanup: Delete workspace when build is done
- configFileProvider: Provide Configuration files
- deleteDir: Recursively delete the current directory from the workspace
- dir: Change current directory
- echo: Print Message
- emailExt: Extended Email
- emailExtRecipients: Extended Email Recipients
- error: Error signal
- fileExists: Verify if file exists in workspace
- findBuildScans: Find published build scans
- fingerprint: Record fingerprints of files to track usage
- git: Git
- input: Wait for interactive input
- isUnix: Checks if running on a Unix-like node
- junit: Archive JUnit-formatted test results
- library: Load a library on the fly

6

checkout ?

SCM

Git

Repositories ?

Repository URL ?
https://github.com/Tuchsanai/DevTools.git

Credentials ?
tuchsanai/*****

+ Add -

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?
*/dev

Add Branch

Repository browser ?
(Auto)

Additional Behaviours

Add ▾

Include in polling? ?

Include in changelog? ?

Generate Pipeline Script

Push button

7

If privated repo, you need credentials

8

9

checkout ?

SCM

Git

Repositories ?

Repository URL ?
https://github.com/Tuchsanai/DevTools.git

Credentials ?
tuchsanai/*****

+ Add -

Advanced ▾

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?
*/dev

Add Branch

Repository browser ?
(Auto)

Additional Behaviours

Add ▾

Include in polling? ?

Include in changelog? ?

Generate Pipeline Script

COPY

10

```
checkout scmGit(branches: [[name: '**/dev']], extensions: [], userRemoteConfigs: [[credentialsId: 'github', url: 'https://github.com/Tuchsanai/DevTools.git']])
```

Configure

Preserve stashes from completed builds [?](#)

This project is parameterized [?](#)

General

Advanced Project Options

Pipeline

Build Triggers

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

Advanced Project Options

Advanced [▼](#)

Pipeline

Definition

Pipeline script

```
1v pipeline {  
2v   agent any  
3v  
4v   stages {  
5v     stage('copy repository') {  
6v       steps {  
7v         checkout scmGit(branches: [[name: '*dev']], extensions: [], userRemoteConfigs: [])  
8v       }  
9v     }  
10v    // Copy the repository  
11v    checkout scmGit(branches: [[name: '*dev']], extensions: [], userRemoteConfigs: [])  
12v  }  
13v}  
14v  
15v
```



Use Groovy Sandbox [?](#)

Pipeline Syntax

Save

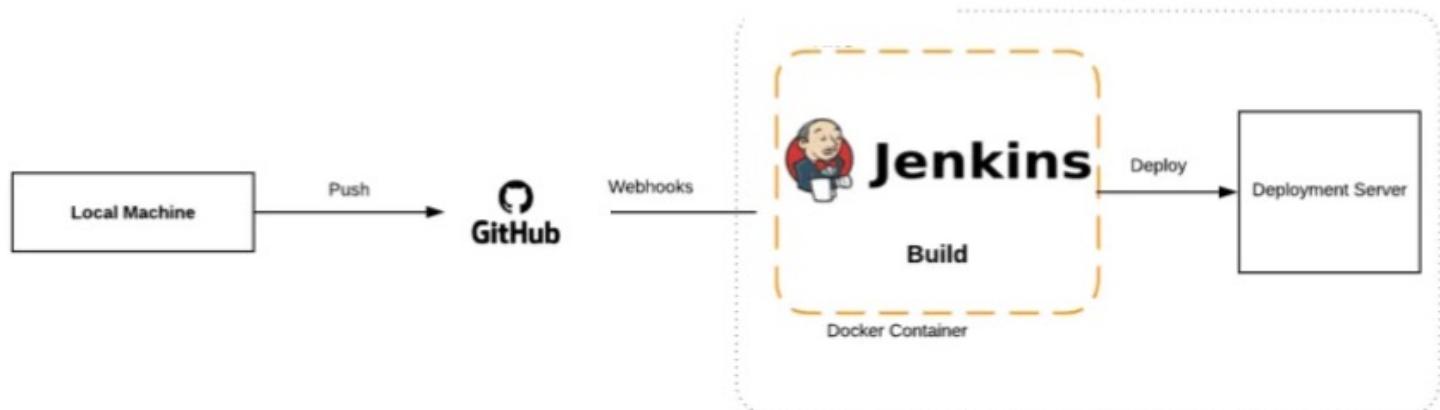
Apply

Console Output

```
Started by user tp  
[Pipeline] Start of Pipeline  
[Pipeline] node  
Running on Jenkins in /var/lib/jenkins/workspace/No_SCM  
[Pipeline] stage  
[Pipeline] { (copy repository)  
[Pipeline] checkout  
The recommended git tool is: NONE  
using credential github  
> git rev-parse --resolve-dir /var/lib/jenkins/workspace/No_SCM.git # timeout=10  
Fetching changes from the remote Git repository  
> git config remote.origin.url https://github.com/Tuchsnaai/DevTools.git # timeout=10  
Fetching upstream changes from https://github.com/Tuchsnaai/DevTools.git  
> git fetch --tags --force --progress -- https://github.com/Tuchsnaai/DevTools.git +refs/heads/*:refs/remotes/origin/* # timeout=10  
Checking out Revision 777695d8493e3e2f982eb6cede9ab80461af605 (refs/remotes/origin/dev)  
> git checkout -f 777695d8493e3e2f982eb6cede9ab80461af605 # timeout=10  
Commit message: "delete files"  
First time build. Skipping changelog.  
[Pipeline] // stage  
[Pipeline] { (Check Python Installation)  
[Pipeline] script  
[Pipeline] sh  
+ which python3  
/usr/bin/python3  
[Pipeline] // stage  
[Pipeline] { (Run Python Script)  
[Pipeline] script  
[Pipeline] sh  
+ which python3  
/var/lib/jenkins/workspace/No_SCM  
[Pipeline] sh  
+ python3 /03_Jenkins/week13/02_Run_Jenkinsfile_From_Github_withoutSCM/status.py  
System Status:  
-----  
Operating System: posix  
  
Environment Variables: environ({'JENKINS_HOME': '/var/lib/jenkins', 'USER': 'jenkins', 'CI': 'true', 'RUN_CHANGES_DISPLAY_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/display/redirect?page=changes', 'NODE_LABELS': 'built-in', 'HUDSON_URL': 'http://175.41.181.239.8080/', 'HOME': '/var/lib/jenkins', 'BUILD_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/', 'HUDSON_COOKIE': 'a7b58642-e0f8-4b73-9815-31ababfe139f', 'JENKINS_SERVER_COOKIE': 'duration=586d74ee4c1e7d4ccf88cf3ed72fb78e91ca847f638989c02a53e38619', 'NOTIFY_SOCKET': '/run/systemd/notify', 'SYSTEMD_EXEC_PID': '7568', 'WORKSPACE': '/var/lib/jenkins/workspace/No_SCM', 'LOGNAME': 'jenkins', 'NODE_NAME': 'built-in', 'JOURNAL_STREAM': '8:44809', 'RUN_ARTIFACTS_DISPLAY_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/display/redirect?page=artifacts', 'STAGE_NAME': 'Run Python Script', 'EXECUTOR_NUMBER': '1', 'RUN_DISPLAY_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/display/redirect?page=tests', 'BUILD_DISPLAY_NAME': '#2', 'HUDSON_HOME': '/var/lib/jenkins', 'JOB_BASE_NAME': 'No_SCM', 'PATH': '/usr/local/bin:/usr/local/bin:/usr/sbin:/usr/bin:/snap/bin', 'INVOCATION_ID': 'aca235982c43aa08d8f67ec08ae', 'BUILD_URL': 'http://175.41.181.239.8080/job/No\_SCM/', 'BUILD_NUMBER': '2', 'JENKINS_NODE_COOKIE': '3b108ed5-a913-4708-9984-2549fe7ea95', 'SHELL': '/bin/sh', 'JOE_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/display/redirect', 'HUDSON_SERVER_COOKIE': 'a8de3eef92178a', 'JOB_DISPLAY_URL': 'http://175.41.181.239.8080/job/No\_SCM/2/display/redirect', 'JOB_NAME': 'No_SCM', 'PWD': '/var/lib/jenkins/workspace/No_SCM', 'WORKSPACE_TMP': '/var/lib/jenkins/workspace/No_SCM/tmp'}
```

```
finished.  
[Pipeline] sh  
+ ls -l  
total 28  
drwxr-xr-x 6 jenkins jenkins 4096 Feb 28 08:53 @_GIT  
drwxr-xr-x 3 jenkins jenkins 4096 Feb 28 08:53 @_Google Cloud  
drwxr-xr-x 4 jenkins jenkins 4096 Feb 28 08:53 @_Docker  
drwxr-xr-x 3 jenkins jenkins 4096 Feb 28 08:53 @_Artifactory  
drwxr-xr-x 2 jenkins jenkins 4096 Feb 28 08:53 @_Jenkins Project  
-rw-r--r-- 1 jenkins jenkins 1695 Feb 28 08:53 README.md
```

LAB 03 : Github Webhook with Jenkinfile using SCM



Delivery (CD).

Configure with github

A screenshot of a GitHub repository page for 'Tuchsanai / DevTools'. The top navigation bar includes links for Code, Issues, Pull requests, Actions, Projects, Security, Insights, and Settings. The Settings link is circled in red and has a large red number '1' above it. Below the navigation bar, there's a header for 'DevTools' with options to Unpin, Unwatch, Fork, Star, and Edit. The main content area shows a list of commits from the 'main' branch, with a commit by 'Tuchsanai' dated 36a75ee · 6 hours ago. It also displays sections for About (no description, website, or topics provided), Releases (none), Packages (none), and Languages (none). A green button labeled 'Code' is visible at the bottom.

A screenshot of the 'General' settings page for the same GitHub repository. The 'General' tab is selected. On the left, there are sections for Access, Collaborators, Moderation options, Code and automation, Branches, Tags, Rules, Actions, and Webhooks. The 'Webhooks' section is circled in red and has a large red number '2' above it. On the right, there are sections for Repository name (set to 'DevTools'), Template repository (unchecked), Require contributors to sign off on web-based commits (unchecked), Default branch (set to 'main'), and Social preview. A URL 'https://github.com/Tuchsanai/DevTools/settings/hooks' is shown at the bottom.

Tuchsanai / DevTools

Code Issues Pull requests Actions Projects Security Insights Settings

Webhooks

3

Add webhook

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Code Issues Pull requests Actions Projects Security Insights Settings

Webhooks / Add webhook

We'll send a POST request to the URL below with details of any subscribed events. You can also specify which data format you'd like to receive (JSON, x-www-form-urlencoded, etc). More information can be found in [our developer documentation](#).

Payload URL *

http://175.41.181.239:8080/github-webhook/

Content type

application/json

Secret

Which events would you like to trigger this webhook?

Just the push event.

Send me everything.

Let me select individual events.

Active

We will deliver event details when this hook is triggered.

Add webhook

4

http://yourJenkinsIP:8080/github-webhook/

<> Code ⏪ Issues ⏪ Pull requests ⏪ Actions ⏪ Projects ⏪ Security ⏪ Insights ⏪ Settings

Okay, that hook was successfully created. We sent a ping payload to test it out! Read more about it at <https://docs.github.com/webhooks/#ping-event>.

Webhooks

Add webhook

General

Access

Collaborators

Moderation options

Code and automation

Branches

Tags

Rules

Actions

Webhooks

Environments

Codespaces

Pages

Security

Code security and analysis

Deploy keys

Secrets and variables

Integrations

GitHub Apps

Email notifications

Webhooks allow external services to be notified when certain events happen. When the specified events happen, we'll send a POST request to each of the URLs you provide. Learn more in our [Webhooks Guide](#).

● <http://175.41.181.239:8080/github...> (push)

Edit Delete

Configure with Jenkins

Not Secure 175.41.181.239:8080/view/all/newJob

Jenkins

Dashboard > All >

Enter an item name

webhook
» Required field

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.

OK

Configure

General

- Do not allow concurrent builds
- Do not allow the pipeline to resume if the controller restarts
- GitHub project
- Pipeline speed/durability override ?
- Preserve stashes from completed builds ?
- This project is parameterized ?
- Throttle builds ?

Build Triggers

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

Advanced Project Options

Advanced ▼

Pipeline

Definition

 Pipeline script Pipeline script from SCMScript ?

1

try sample Pipeline... ▼

The screenshot shows a GitHub repository interface. At the top, there are navigation links: Code, Issues, Pull requests, Actions, Projects, Security, and a dropdown for branches (main). Below the header, a commit card is displayed for a commit made 2 minutes ago by user 'Tuchsanai' with message 'aa'. The commit is associated with the 'main' branch. The repository path is shown as 'DevTools / 03_Jenkins / week13 / 03_Github_webhook /'. A yellow box highlights this path. The main content area lists files: 'Jenkinfile' and 'readme.md', both with last commit messages 'aa' and dates '2 minutes ago'.

The screenshot shows the Jenkins Pipeline configuration for the GitHub webhook. The 'SCM' section is set to 'Git'. The 'Repository URL' is 'https://github.com/Tuchsanai/DevTools.git' and the 'Credentials' are 'tuchsanai/********'. The 'Branch Specifier' is set to '*dev'. The 'Script Path' is '/03_Jenkins/week13/03_Github_webhook/Jenkinfile'. A checkbox for 'Lightweight checkout' is checked. There are 'Save' and 'Apply' buttons at the bottom.

```
git https://github.com/Tuchsanai/DevTools.git  
tuchsanai/********  
*dev  
/03_Jenkins/week13/03_Github_webhook/Jenkinfile
```