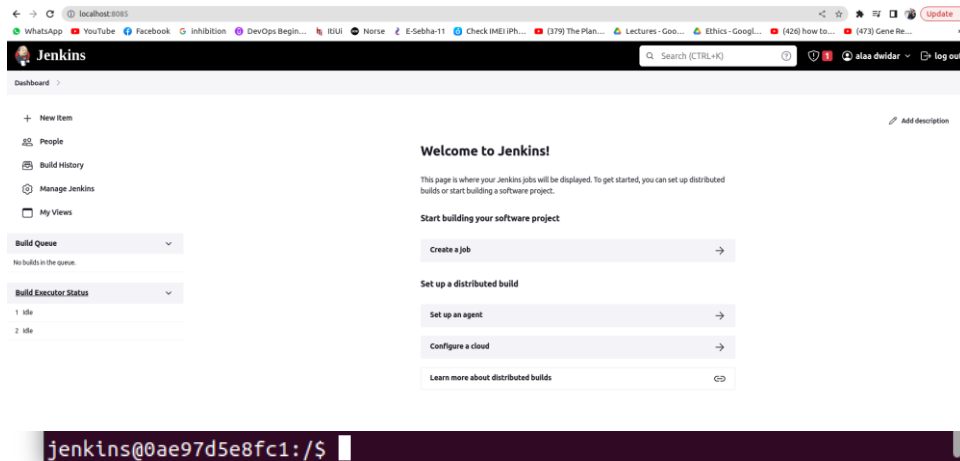


1- install jenkins with docker image

```
docker run -p 8085:8080 -d jenkins/jenkins:its
```

```
docker ps
```

```
docker exec -it 0ae97d5e8fc1 bash
```



2- install role based authorization plugin

### Plugins



### Configure Global Security

#### Authentication

☐ Disable remember me

Security Realm

Jenkins' own user database

☐ Allow users to sign up

Authorization

Role-Based Strategy

### 3- create new user

## Create User

Username

Password

Confirm password

Full name





E-mail address

Create User

## Users

+ Create User

These users can log into Jenkins. This is a sub set of [this list](#), which also contains auto-created users who really just made some commits on some projects and have no direct Jenkins access.

User ID	Name	
 <a href="#">Alaadwidar</a>	alaa dwidar	
 <a href="#">alaaismail208</a>	alaa ismail	 

4- create read role and assign it to the new user

### Manage Roles

Global roles

Role	Overall	Credentials	Agent	Job	Run	View	SCM
admin	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Developers	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Role to add

Developers

### Assign Roles

Global roles

User/group	admin	Developers
alaa dwidar	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Anonymous	<input type="checkbox"/>	<input type="checkbox"/>
alaa ismail	<input checked="" type="checkbox"/>	<input type="checkbox"/>

User/group to add

alaaismail208


Add


5- create free style pipeline and link it to private git repo(inside it create directory and create file with "hello world")


Enter an item name


project-1


» Required field


 **Freestyle project**  
This is the central feature of Jenkins. Jenkins will build your project, combining any SCM with any build system, and this can be even used for something other

 **Pipeline**  
Orchestrates long-running activities that can span multiple build agents. Suitable for building pipelines (formerly known as workflows) and/or organizing com


 **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 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

## Source Code Management

☐ None

☒ Git ?

Repositories ?

Repository URL ?

https://github.com/AlaaiDwidar/pipeline1.git

Credentials ?

alaaidwidar/\*\*\*\*\*

+ Add

Advanced...

Add Repository

Branches to build ?

Branch Specifier (blank for 'any') ?

\*/main

Save

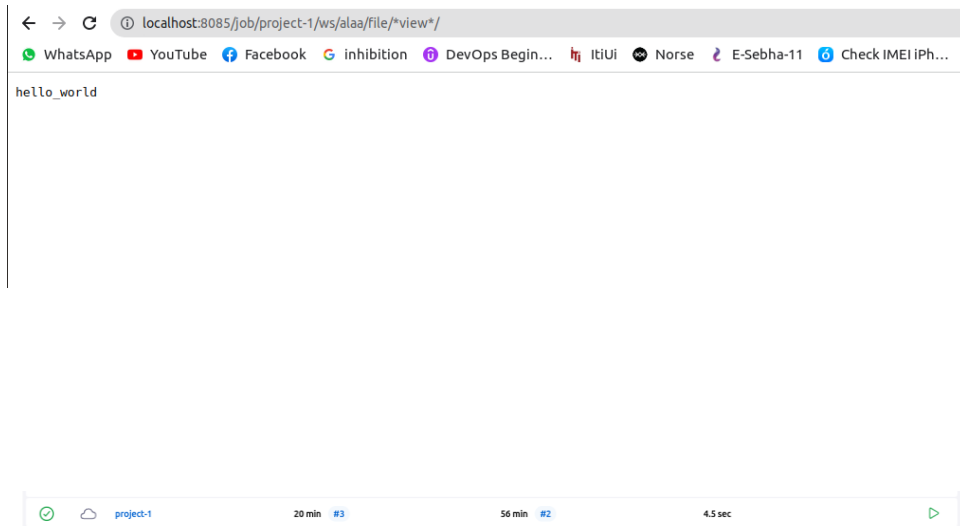
Apply

## Workspace of project-1 on Built-In Node

project-1 / alaa /  →

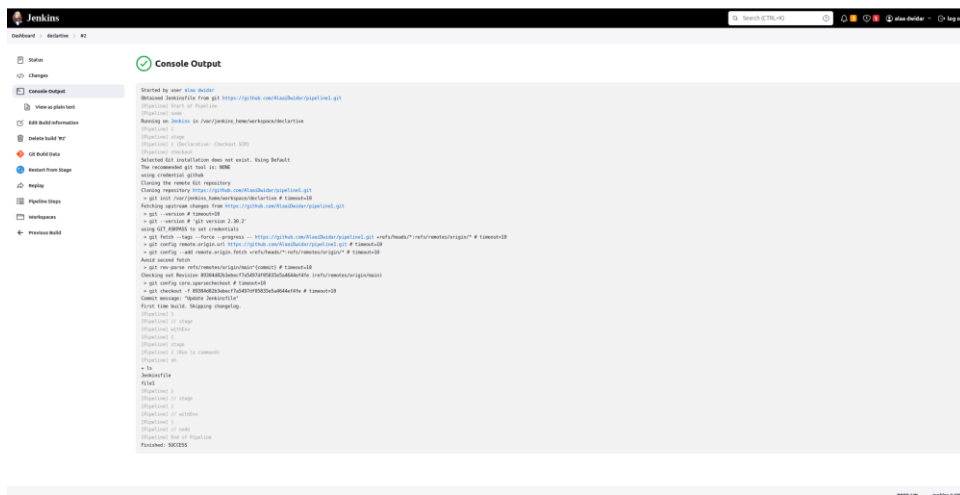
file 4 Feb 2023, 01:21:52 12 B  

 (all files in zip)



## Part 2

## 1- create declarative in jenkins GUI pipeline for your own repo to do "ls"



Jenkins

Dashboard >

+ New Item

People

Build History

Project Relationship

Check File Fingerprint

Add description

S	W	Name :	Last Success	Last Failure	Last Duration
		declarative	2 min 38 sec #2	N/A	5.4 sec

2- create scripted in jenkins GUI pipeline for your own repo to do "ls"

Dashboard > scriptive > #1

Status

Changes

Console Output

View as plain text

Edit build information

Delete build 'W'

Git Build Data

Replay

Pipeline Steps

Workspaces

Console Output

Started by user alaa dwidar

Obtained Jenkinsfile from git https://github.com/AlaaBwider/pipeline1.git

[Pipeline] Start of Pipeline

[Pipeline] node

Running on Jenkins in /var/jenkins\_home/workspace/scriptive

[Pipeline] {

[Pipeline] checkout

Selected git installation does not exist. Using Default

The recommended git tool is: NONE

using credential github

Cloning the remote git repository

Cloning repository https://github.com/AlaaBwider/pipeline1.git

> git init /var/jenkins\_home/workspace/scriptive # timeout=10

Fetching upstream changes from https://github.com/AlaaBwider/pipeline1.git

> git --version # timeout=10

> git --version # 'git version 2.38.2'

using GIT\_ASKPASS to set credentials

> git fetch --tags --force --progress -- https://github.com/AlaaBwider/pipeline1.git +refs/heads/\*:refs/remotes/origin/\* # timeout=10

> git config remote.origin.url https://github.com/AlaaBwider/pipeline1.git # timeout=10

> git config --add remote.origin.fetch +refs/heads/\*:refs/remotes/origin/\* # timeout=10

Avoid second fetch

> git rev-parse refs/remotes/origin/main:commit # timeout=10

Checking out Revision 97388dce130773805c2089f507772b5d0e (refs/remotes/origin/main)

> git config core.sparsecheckout # timeout=10

> git checkout -f 97388dce130773805c2089f507772b5d0e # timeout=10

Commit message: "update Jenkinsfile"

First time build. Skipping changelog.

[Pipeline] sh

+ ls

Jenkinsfile

file

[Pipeline] }

[Pipeline] // node

[Pipeline] End of Pipeline

Finished: SUCCESS

scriptive

1 min 0 sec #1

N/A

3.7 sec

3- create the same with jenkinsfile in your branches as multibranch pipeline

### ✓ Scan Repository Log

```
Started
[Sat Feb 04 02:37:28 UTC 2023] Starting branch indexing...
02:37:29 Connecting to https://api.github.com using alaaidwidar/*****
Examining AlaaDwidar/pipeline1

Checking branches...

Getting remote branches...

Checking branch main

Getting remote pull requests...
'Jenkinsfile' found
Met criteria
Scheduled build for branch: main

1 branches were processed

Checking pull-requests...

0 pull requests were processed

Finished examining AlaaDwidar/pipeline1

[Sat Feb 04 02:37:31 UTC 2023] Finished branch indexing. Indexing took 2.6 sec
Finished: SUCCESS
```

### multibranch

Branches (1) Pull Requests (0)

Disable Multibranch Pipeline

S	W	Name	Last Success	Last Failure	Last Duration
✓	☀	main	1 min 54 sec #1	N/A	5.7 sec

## ✓ Console Output

```
Branch indexing
02:37:37 Connecting to https://api.github.com using alaiidwidar/*****
Obtained Jenkinsfile from 9718dbc8e1497f18056c2d089f5977f72d5d8e6
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/multibranch_main
[Pipeline] {
[Pipeline] checkout
Selected Git installation does not exist. Using Default
The recommended git tool is: NONE
using credential github
Cloning the remote Git repository
Cloning with configured refspecs honored and without tags
Cloning repository https://github.com/Alaiidwidar/pipeline1.git
> git init /var/jenkins_home/workspace/multibranch_main # timeout=10
Fetching upstream changes from https://github.com/Alaiidwidar/pipeline1.git
> git --version # timeout=10
> git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials
> git fetch --no-tags --force --progress -- https://github.com/Alaiidwidar/pipeline1.git +refs/heads/main:refs/remotes/origin/main # timeout=10
> git config remote.origin.url https://github.com/Alaiidwidar/pipeline1.git # timeout=10
> git config --add remote.origin.fetch +refs/heads/main:refs/remotes/origin/main # timeout=10
Avoid second fetch
Checking out Revision 9718dbc8e1497f18056c2d089f5977f72d5d8e6 (main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 9718dbc8e1497f18056c2d089f5977f72d5d8e6 # timeout=10
Commit message: "Update Jenkinsfile"
First time build. Skipping changelog.
[Pipeline] sh
+ ls
Jenkinsfile
file1
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline

Github has been notified of this commit's build result

Finished: SUCCESS
```