



**SoftLeader**  
Softleader Technology Corp.

[www.softleader.com.tw](http://www.softleader.com.tw)



# Jenkins

[www.softleader.com.tw](http://www.softleader.com.tw)





- 01 Jenkins 本機安裝
- 02 Jenkins 操作教學
- 03 Jenkins pipeline撰寫



# Jenkins

# Jenkins介紹

**Jenkins** 是一套的可擴充套件的用於自動化部署的開源 **CI/CD** 工具。

**Jenkins** 是用 **Java** 編寫的，是在 **MIT** 許可下發布的。它有一組強大的功能，可以將軟體的構建、測試、部署、整合和釋出等相關任務自動化。

這款用於測試的自動化 **CI/CD** 工具可以在 **macOS**、**Windows** 和各種 **UNIX** 版本（例如 **OpenSUSE**、**Ubuntu**、**Red Hat** 等）系統上使用。

除了通過本地安裝包安裝，它還可以在任何安裝過 **Java** 執行時環境（**Java Runtime Environment**，**JRE**）的機器上單獨安裝或者作為一個 **Docker** 安裝



# Jenkins

本機安裝教學

# 以docker 安裝 Jenkins

## 1. 安裝 Docker

Switch to Linux Containers

<https://docs.docker.com/desktop/windows/install/>

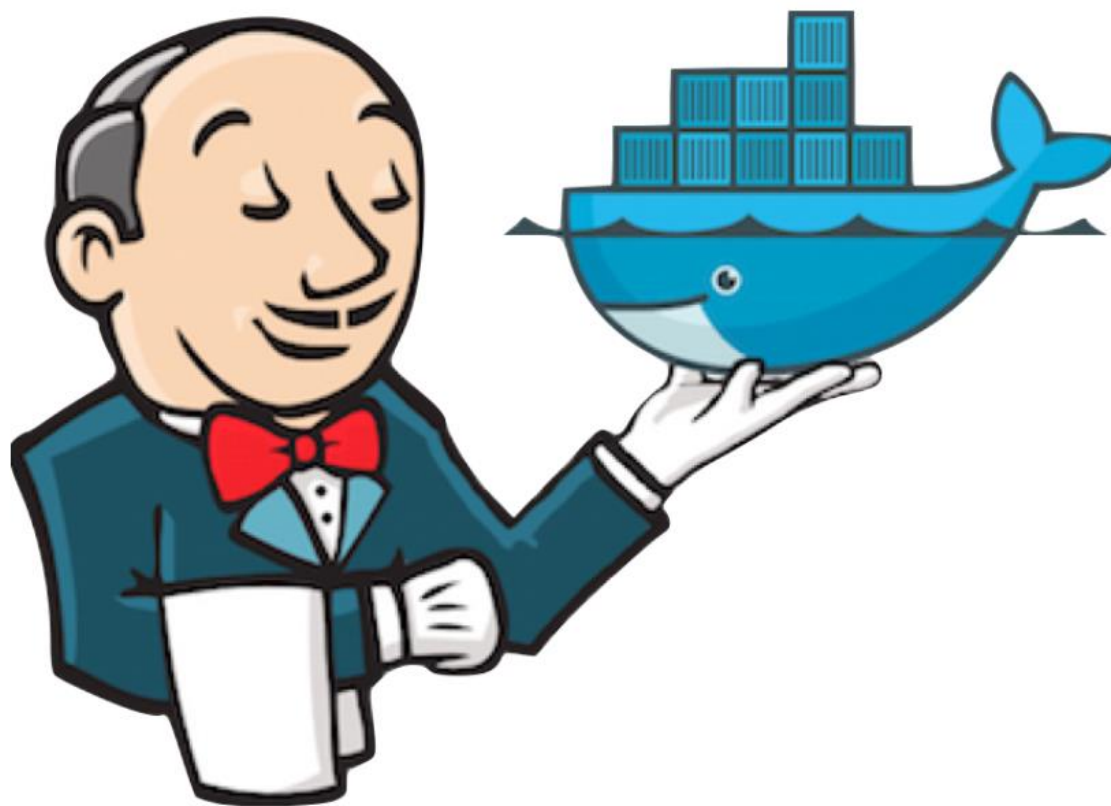
## 2. 選擇Jenkins image，並下載

`docker pull jenkins/jenkins: 2.325-jdk8`

## 3. 啟動Jenkins

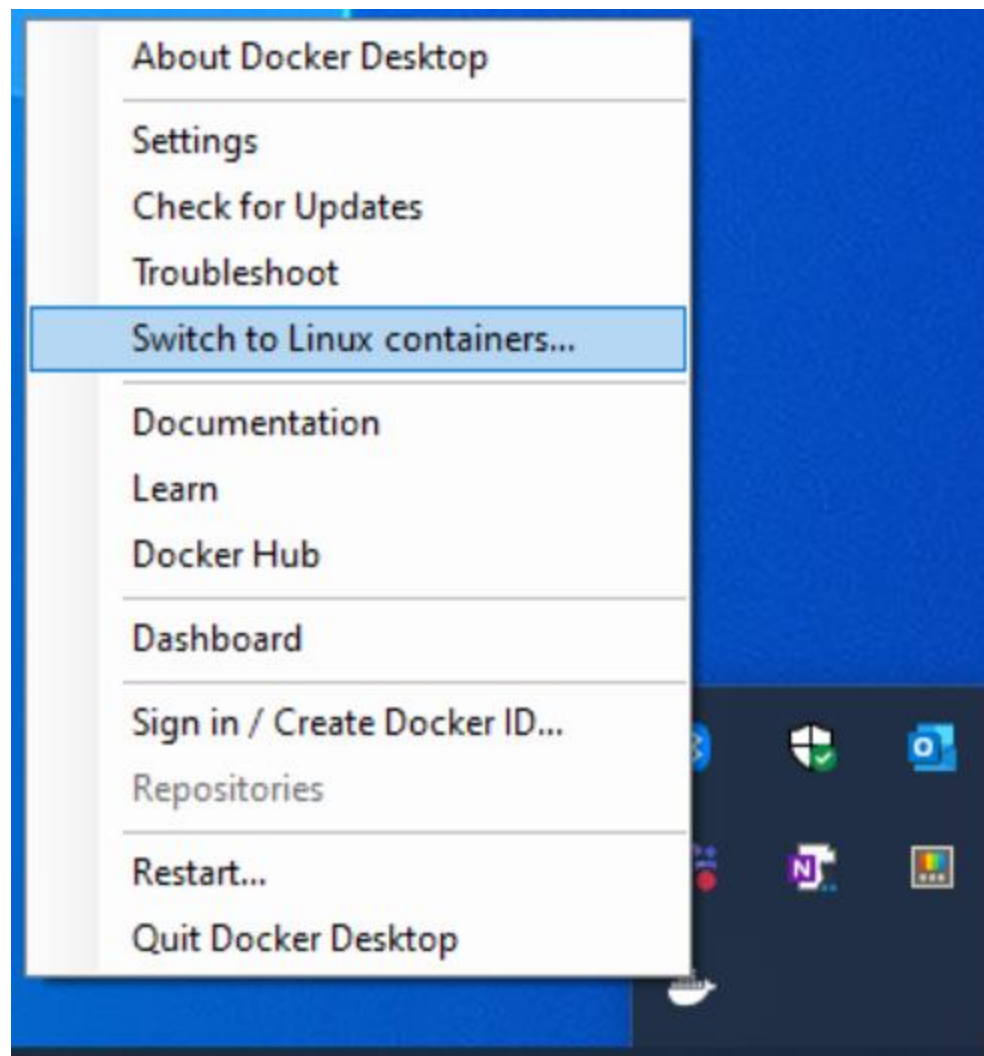
- 以docker指令直接啟動
- 以docker-compose 啟動

<https://www.jenkins.io/doc/book/installing/>



[https://hub.docker.com/\\_/jenkins?tab=tags](https://hub.docker.com/_/jenkins?tab=tags)

# 切换linux containers





# docker-compose.yml

version: '3.8'

services:

jenkins:

image: jenkins/jenkins:2.325-jdk8

restart: always

ports:

- 8080:8080

- 50000:50000

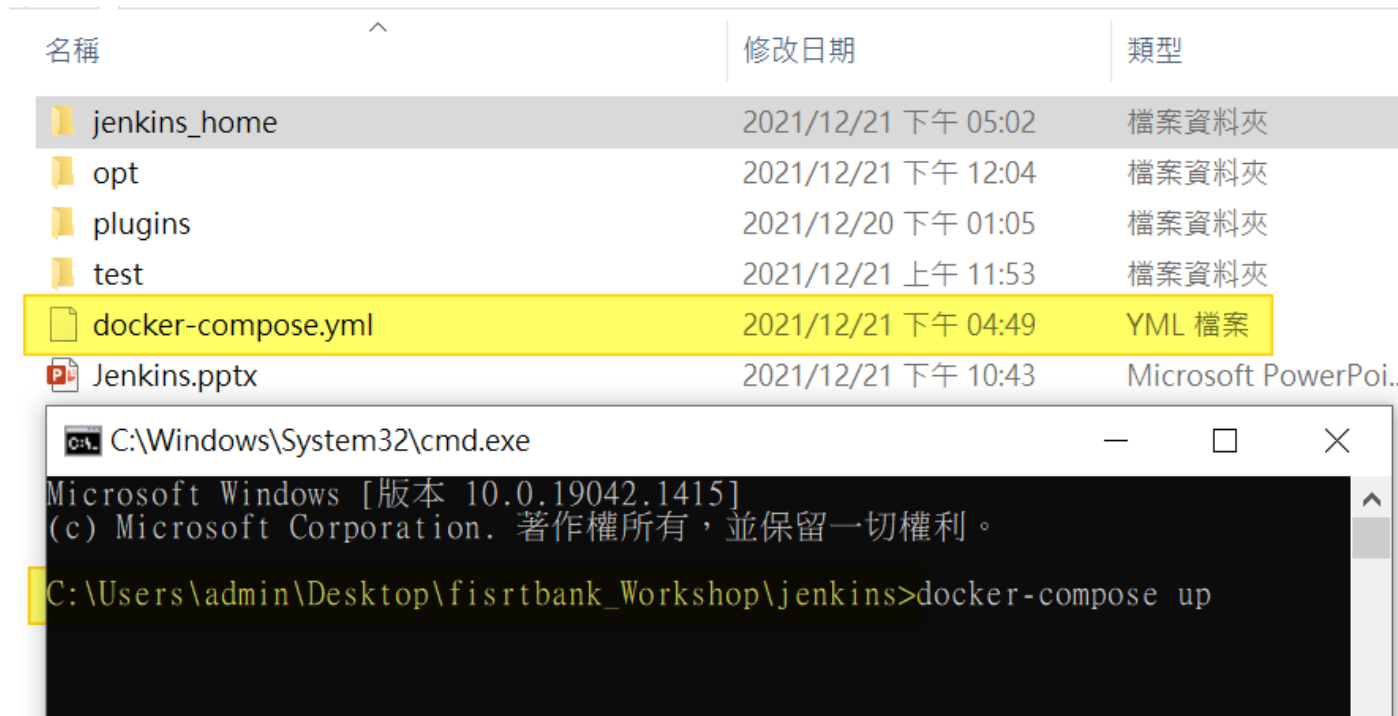
volumes:

- C:\Users\admin\Desktop\fisrtbank\_Workshop\Jenkins\Jenkins\_home:/var/jenkins\_home

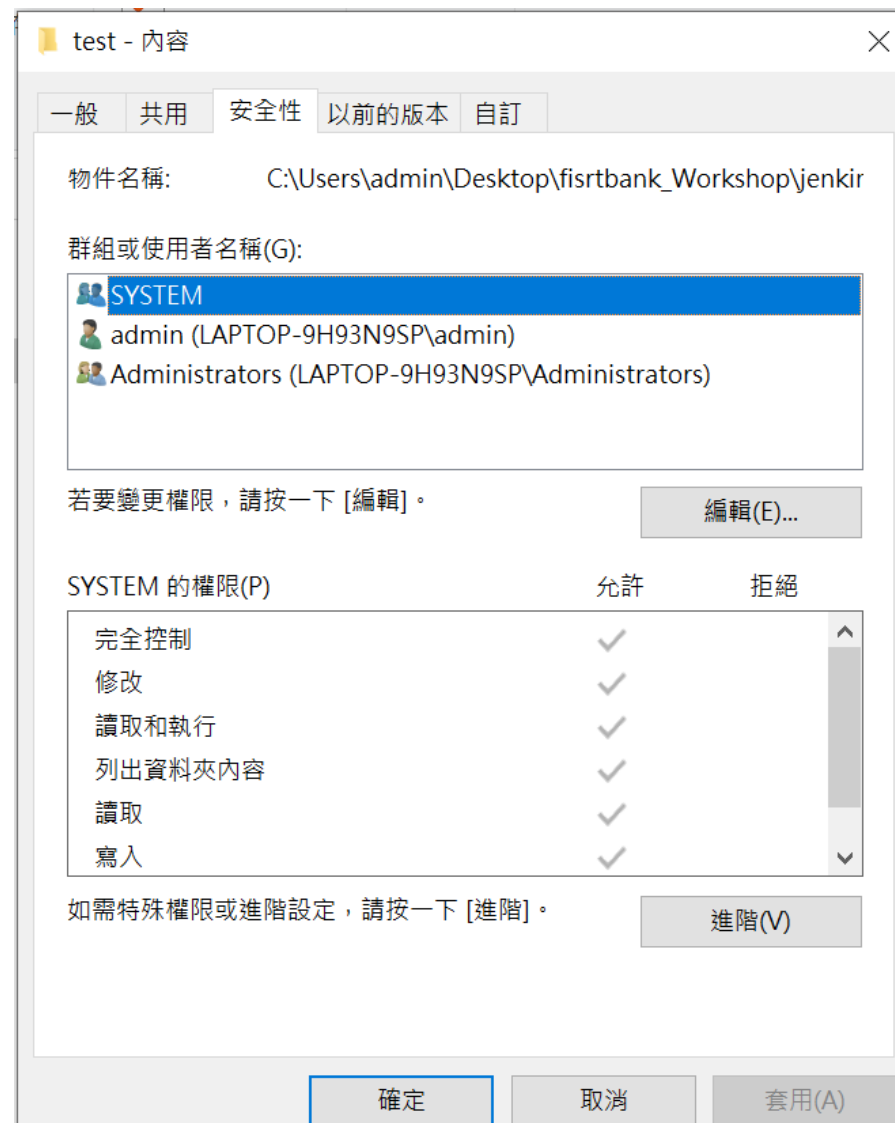
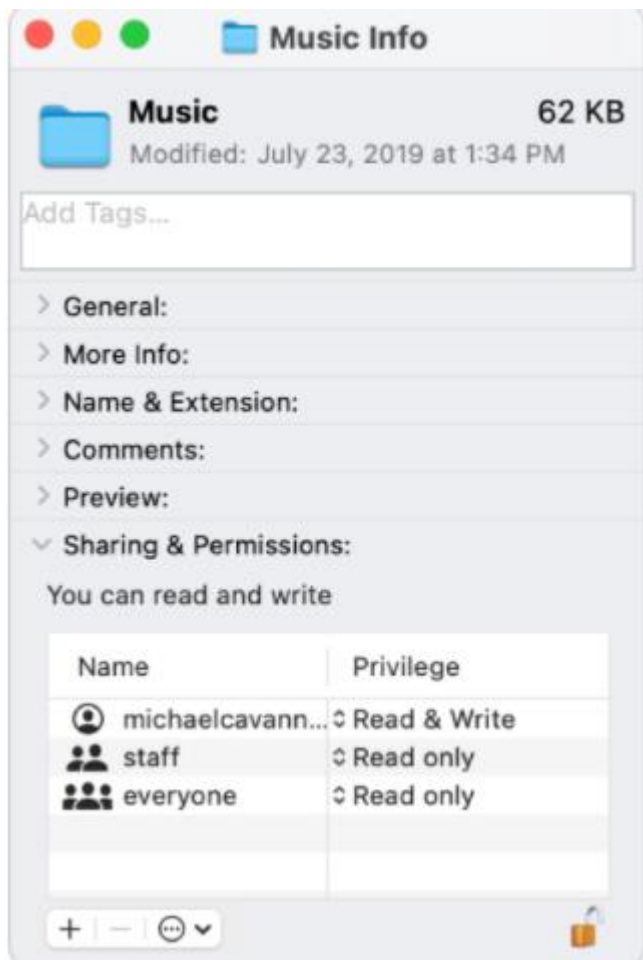
在docker-compose.yml 相同路徑中，下指令

開啟container  
docker-compose up -d

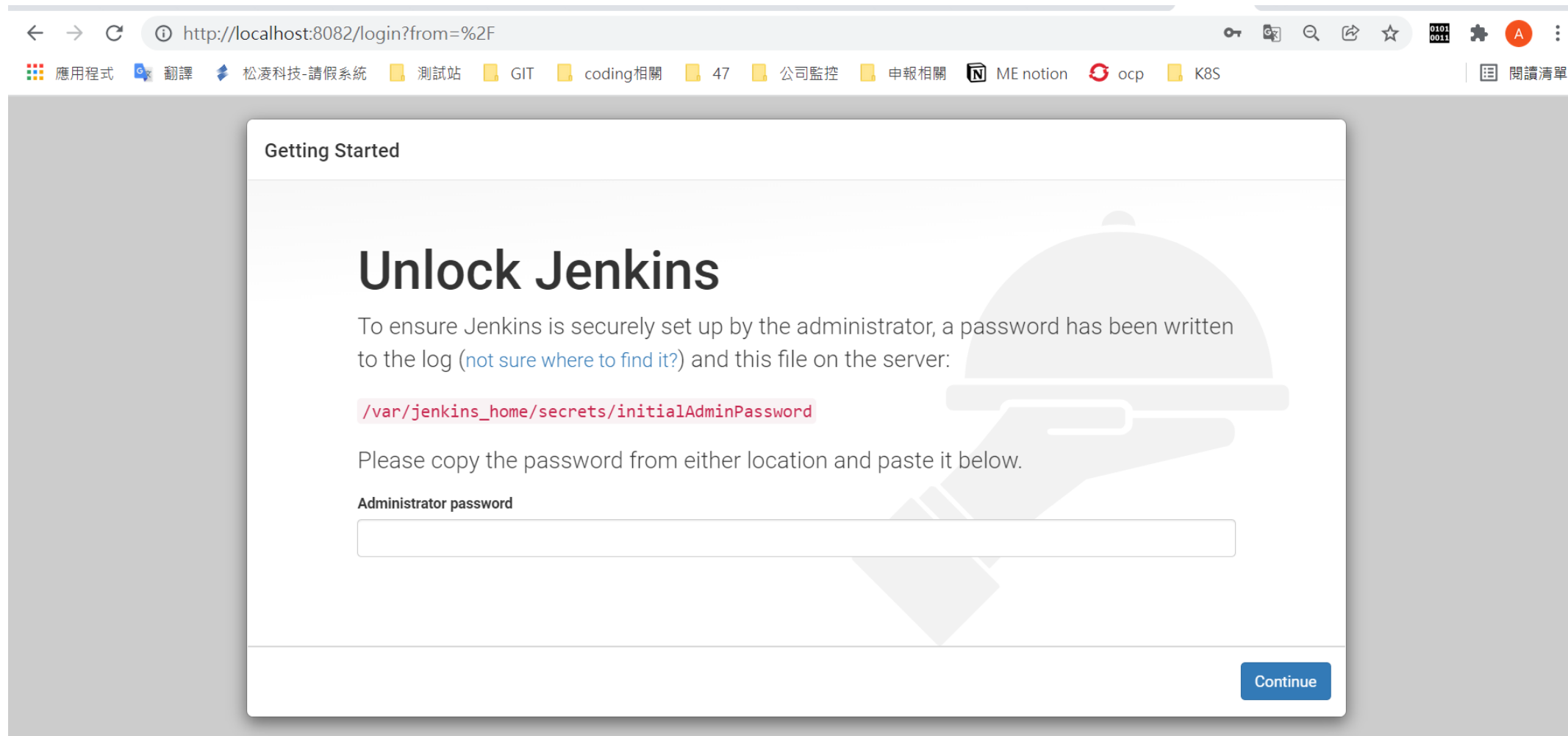
移除container  
docker-compose down



# 資料夾權限問題



# 密碼可在log或者 /var/jenkins\_home/secrets/initialAdminPassword找到



# 查看container的log

## docker logs -f \${container ID}

```
C:\Users\admin\Desktop\fisrtbank_Workshop\jenkins>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
8ec225875b61   jenkins/jenkins:2.325-jdk8         "/sbin/tini -- /usr/..." 37 seconds ago Up 34 seconds 0.0.0.0:8082->8080/tcp, :::8082->8080/tcp, 0.0.0.0:50002->50000/tcp, :::50002->50000/tcp   jenkins_jenkins_1

C:\Users\admin\Desktop\fisrtbank_Workshop\jenkins>docker logs -f 8ec225875b61
Running from: /usr/share/jenkins/jenkins.war
webroot: EnvVars.masterEnvVars.get("JENKINS_HOME")
2021-12-21 08:58:13.448+0000 [id=1] INFO org.eclipse.jetty.util.log.Log#initialized: Logging initialized @1641ms to org.eclipse.jetty.util.log.JavaUtilLog
2021-12-21 08:58:13.977+0000 [id=1] INFO winstone.Logger#logInternal: Beginning extraction from war file
2021-12-21 08:58:14.062+0000 [id=1] WARNING o.e.j.s.handler.ContextHandler#setContextPath: Empty contextPath
2021-12-21 08:58:14.222+0000 [id=1] INFO org.eclipse.jetty.server.Server#doStart: jetty-9.4.43.v20210629; built: 2021-06-30T11:07:22.254Z; git: 526006ecfa3af7f1a23a288e2bef7ea9dd7e8; jvm 1.8.0_312-b07
2021-12-21 08:58:19.019+0000 [id=1] INFO o.e.j.w.StandardDescriptorProcessor#visitServlet: NO JSP Support for /, did not find org.eclipse.jetty.jsp.JettyJspServlet
2021-12-21 08:58:19.177+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: DefaultSessionIdManager workerName=node0
2021-12-21 08:58:19.177+0000 [id=1] INFO o.e.j.s.s.DefaultSessionIdManager#doStart: No SessionScavenger set, using defaults
2021-12-21 08:58:19.179+0000 [id=1] INFO o.e.j.server.session.HouseKeeper#startScavenging: node0 Scavenging every 660000ms
2021-12-21 08:58:21.144+0000 [id=1] INFO hudson.WebAppMain#contextInitialized: Jenkins home directory: /var/jenkins_home found at: EnvVars.masterEnvVars.get("JENKINS_HOME")
```

# plugin 安裝

## Customize Jenkins

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


### Install suggested plugins

Install plugins the Jenkins community finds most useful.

### Select plugins to install

Select and install plugins most suitable for your needs.

# 若離線安裝 可以將plugin 放在: /jenkins\_home/plugins/內，完成插鍵離線安裝。

 **Jenkins**

搜尋

5

cicduser

登出

Jenkins

外掛程式管理

回到儀表板

管理 Jenkins

filter

更新

可用的

已安裝

進階

安裝	名稱 ↓	版本	Released	已安裝
<input type="checkbox"/>	<b>Branch API</b> api-plugin This plugin provides an API for multiple branch based projects.	2.6.0	1 月 10 天 ago	2.5.9
<input type="checkbox"/>	<b>Credentials</b> api-plugin This plugin allows you to store credentials in Jenkins.	2.3.13	1 月 10 天 ago	2.3.12
<input type="checkbox"/>	<b>Durable Task</b> api-plugin 建置工具 Library offering an extension point for processes which can run outside of Jenkins yet be monitored.	1.35	1 月 8 天 ago	1.34
<input type="checkbox"/>	<b>ECharts API</b> api-plugin Provides ECharts for Jenkins plugins.	4.9.0-1	19 天 ago	4.8.0-2
<input type="checkbox"/>	<b>Email Extension</b> 建置工具 email 建置通知程式 This plugin is a replacement for Jenkins's email publisher. It allows to configure every aspect of email notifications: when an email is sent, who should receive it and what the email says Applying this update will address security vulnerabilities in the currently installed version.	2.77	23 天 ago	2.74

# 設定Admin User

Getting Started

## Create First Admin User

使用者名稱:

密碼:

確認密碼:

全名:



# 設定連線jenkins的介面的URL 要跟剛剛docker-compose.yml 對外port號相同

## Getting Started

### Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the BUILD\_URL environment variable provided to build steps.


The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

Jenkins 2.325

[Not now](#)

[Save and Finish](#)

# 安裝成功後，大家可以先玩玩看介面


 **Jenkins**


搜尋


admin | 登出


Jenkins


開啟自動更新頁面


 新增作業

 使用者

 建置歷程

 管理 Jenkins

 我的視景

 [新增描述](#)

## 歡迎使用Jenkins!

請[建立新工作](#)來開始使用

建置佇列

佇列中沒有建置作業。

建置執行程式狀態

- 閒置
- 閒置



# Jenkins

設定配置




# Jenkins 設定配置

- 預覽基礎設定
- Plugin 安裝
- 設定使用者認證- Jenkins 內部使用者
- 設定專案權限與角色權限



## 基礎設定

# Jenkins 基礎瀏覽

 **Jenkins**

搜尋


1


2


admin


登出


Dashboard


 新增作業


 使用者

 建置歷程

 管理 Jenkins

 我的視景

 Lockable Resources

 New View

建置佇列

佇列中沒有建置作業。

建置執行程式狀態

1 閒置


## 系統屬性

名稱 ↓	值
awt.toolkit	sun.awt.X11.XToolkit
executable-war	/usr/share/jenkins/jenkins.war
file.encoding	UTF-8
file.encoding.pkg	sun.io
file.separator	/
java.awt.graphicsenv	sun.awt.X11GraphicsEnvironment
java.awt.headless	true






# Jenkins 建置歷程




# 查看使用者


 **Jenkins**


Q 搜尋


  1  2  admin  登出


Dashboard


 新增作業


 使用者

 建置歷程

 管理 Jenkins

 我的視景

 Lockable Resources

 New View

建置佇列






佇列中沒有建置作業。

建置執行程式狀態

1 閒置

## 人員

Includes all known "users", including login identities which the current security realm can enumerate, as well as people mentioned in commit messages in recorded changelogs.

使用者 ID	名稱	LAST COMMIT ACTIVITY <sup>1</sup>	模組
 admin	admin	N/A	
 sarah	sarah	N/A	
 SYSTEM	SYSTEM	N/A	
 ally	ally.lee	N/A	
 tina	Tina	N/A	





安裝plugin

# Jenkins 安裝plugin



Jenkins

搜尋



1



1



admin



登出

Dashboard ▸ 外掛程式管理



回到儀表板



管理 Jenkins



更新中心

## Plugin Manager

validating-string-parameter



更新

可用的

已安裝

進階

INSTALL NAME ↓

RELEASED

Validating String Parameter 2.8



其他

Build Parameters

11 月 ago

Adds a new parameter type called Validating String Parameter which supports [regular expression](#) validation of a user entered parameter.

Install without restart

Download now and install after restart

Update information obtained: 1 分 0 秒 ago

馬上檢查

# 若離線安裝 plugin :

## Plugin Manager

[更新](#)[可用的](#)[已安裝](#)[進階](#)

### Deploy Plugin

You can select a plugin file from your local system or provide a URL to install a plugin from outside the central plugin repository.

檔案

[選擇檔案](#)

未選擇任何檔案

Or

URL


[Deploy](#)


<https://updates.jenkins-ci.org/download/plugins/>

# 離線安裝，需要注意Jenkins版本與plugin版本的相容性， 或與其他plugin的依賴性

**Dashboard** ▶ **Update Center**

Pipeline: GitHub Groovy Libraries	✓ Already installed
Pipeline: Stage View	✓ Already Installed
Git	✓ Already Installed
SSH Build Agents	✓ Already Installed
Matrix Authorization Strategy	✓ Already Installed
PAM Authentication	✓ Already Installed
LDAP	✓ Already Installed
Email Extension	✓ Already Installed
Mailer	✓ Already Installed
Loading plugin extensions	✓ Success
準備項目	
gitlab-plugin	✓ 成功

 [回到首頁](#)  
(你可以開始使用已安裝好的插件)

 ☐ 當安裝完成且沒有工作正在執行時，重啟Jenkins



## SMTP設定

# SMTP設定: 登入Jenkins --> 系統設定 --> jenkins位置、電子郵件通知



## Welcome to Jenkins!

如果您還沒有帳號，[建一個吧](#)。

Sign in

☐ Keep me signed in

**Jenkins**

Jenkins

 新增作業

 使用者

 建置歷程

 **管理 Jenkins**

 我的視景

 Lockable Resources

 New View

建置佇列

佇列中沒有建置作業。

建置執行程式狀態

1 閒置

2 閒置

### 管理 Jenkins

新版的 Jenkins (2.260) 已經可以[下載](#) ([改版記錄](#))。

1

Warnings have been published for the following currently installed components:

- [Validating String Parameter Plugin 2.4:](#)  
[Stored XSS vulnerability](#)
- [Script Security Plugin 1.74:](#)  
[Sandbox bypass vulnerability](#)
- [Mailer Plugin 1.32:](#)  
[Missing hostname validation](#)
- [Email Extension Plugin 2.74:](#)  
[Missing hostname validation](#)
- [Lockable Resources plugin 2.8:](#)  
[CSRF vulnerability](#)

### System Configuration

 **設定系統**  
設定全域設定及路徑。

2

 **Global Tool Configuration**  
Configure tools, their locations and automatic installers.

# jenkins位置、電子郵件通知

## Jenkins 位置

Jenkins URL

https://192.168.1.47:8443/

系統管理員郵件地址

i079db <i079db@firstbank.com.tw>

- 1.SMTP伺服器
  - 2.允許使用SMTP認證
  - 3.使用者名稱:
  - 4.密碼:
  - 5.是否允許使用SSL
  - 6.SMTP連接埠:
  - 7.預設內容類型: 請選Html
  - 8.預設主旨: 請依照自身需求設定, 這裡我保留預設
- 預設內容: 這邊請設定 `${JELLY_SCRIPT,template="email-template.jelly"}`,  
也就是上一個步驟存放的email template檔名, 要用這個檔來產生電子郵件內容

## 電子郵件通知

SMTP 伺服器

10.14.88.169

預設使用者信箱後綴字串

@firstbank.com.tw

☒ Use SMTP Authentication

使用者名稱

i079db@firstbank.com.tw

密碼

...

使用 SSL

☐

Use TLS

☐

SMTP 連接埠

25

回信信箱

i079db@firstbank.com.tw

字元集

UTF-8

☒ 寄測試信, 看看設定正不正確

測試信收件者

i079db@firstbank.com.tw

儲存

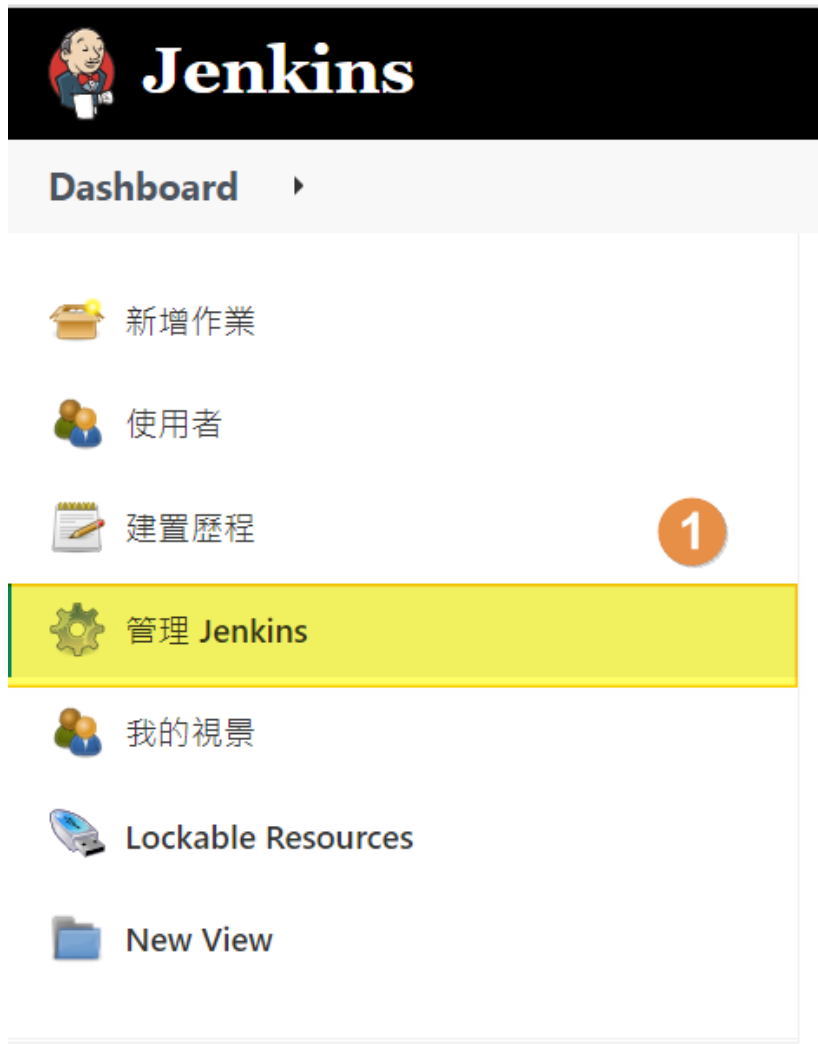
Apply



設定使用者認證



# 權限設定: 登入Jenkins --> 管理jenkins --> 設定全域安全性





## 設定全域安全性

### Authentication

---

☐ Disable remember me

### 安全性領域

---

☒ Jenkins 內建使用者資料庫 ?

☒ 允許使用者註冊 ?

**⚠ 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.**

☐ LDAP ?

☐ Unix user/group database ?

☐ 委派給 Servlet Container ?

☐ None

# 設定完後可以以其他使用者登入看看



## Welcome to Jenkins!

如果您還沒有帳號，[建一個吧](#)。

**Sign in**

☐ Keep me signed in

 **Jenkins**

Jenkins

新增作業

使用者

建置歷程

**管理 Jenkins** 1

我的視景

Lockable Resources

New View

建置佇列 ^  
佇列中沒有建置作業。

建置執行程式狀態 ^  
1 閒置  
2 閒置

### 管理 Jenkins

新版的 Jenkins (2.260) 已經可以[下載](#) ([改版記錄](#))。

Warnings have been published for the following currently installed components:

- [Validating String Parameter Plugin 2.4:](#)  
[Stored XSS vulnerability](#)
- [Script Security Plugin 1.74:](#)  
[Sandbox bypass vulnerability](#)
- [Mailer Plugin 1.32:](#)  
[Missing hostname validation](#)
- [Email Extension Plugin 2.74:](#)  
[Missing hostname validation](#)
- [Lockable Resources plugin 2.8:](#)  
[CSRF vulnerability](#)

### System Configuration

 **設定系統**  
設定全域設定及路徑。

 **Global Tool Configuration**  
Configure tools, their locations and automatic installers.

### Security

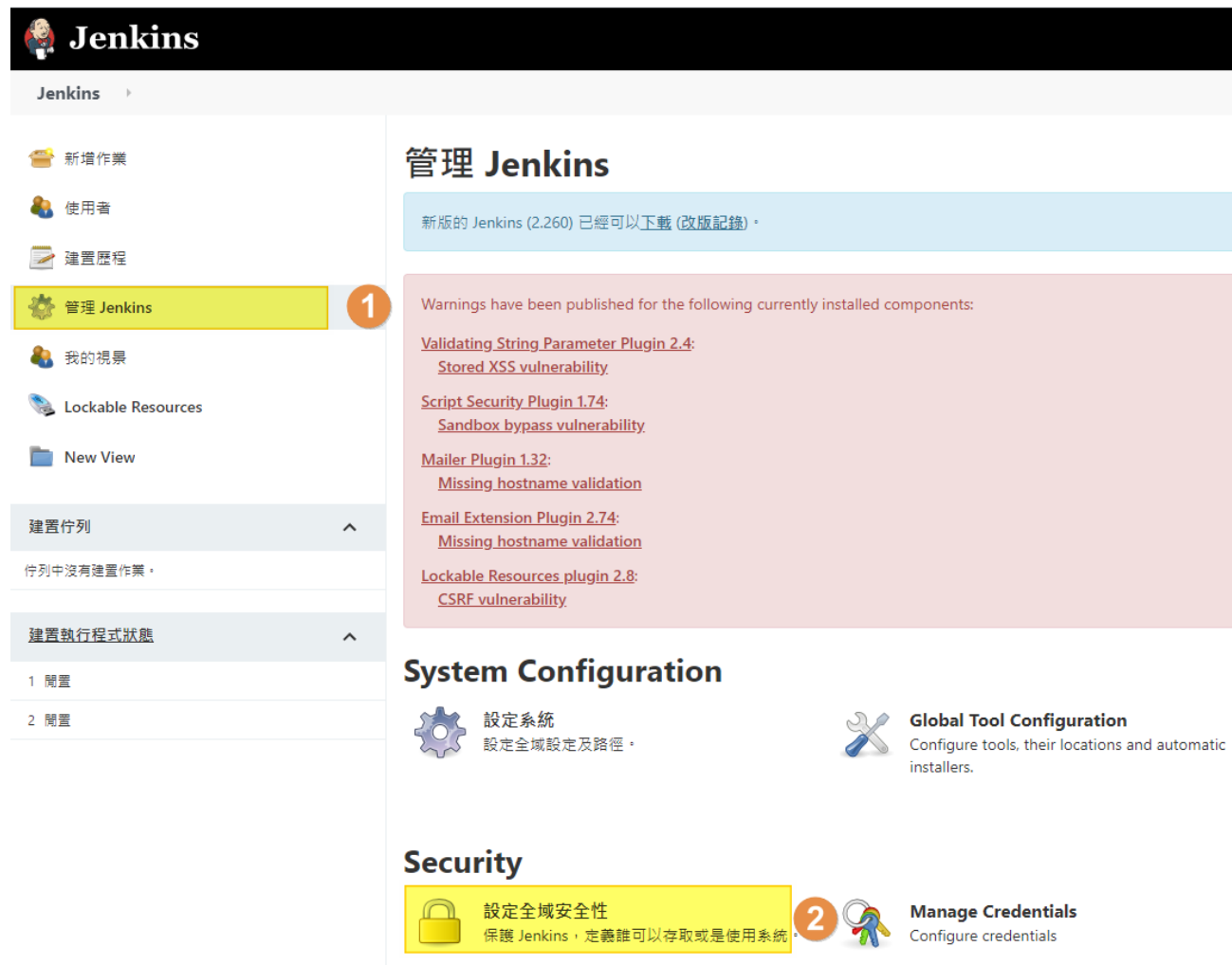
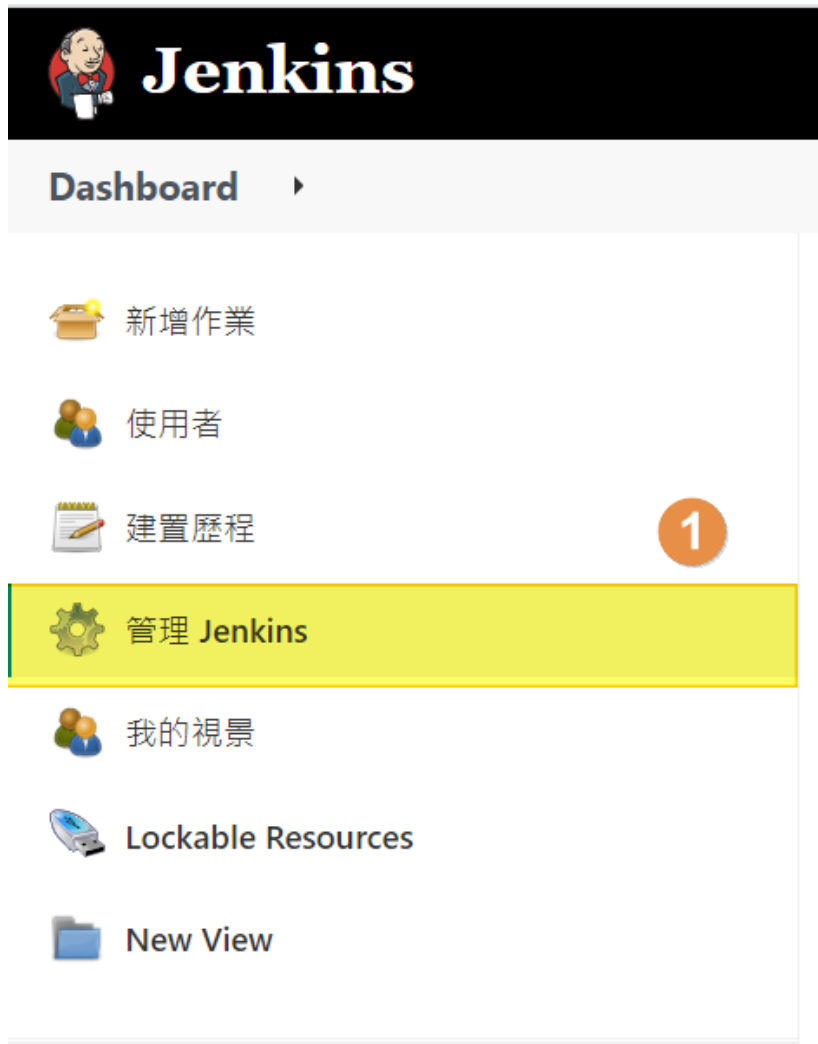
 **設定全域安全性**  
保護 Jenkins，定義誰可以存取或是使用系統。

2  **Manage Credentials**  
Configure credentials



權限設定

# 權限設定: 登入Jenkins --> 管理jenkins --> 設定全域安全性



# 設定全域安全性 --> 授權

## 授權

- ☒ Role-Based Strategy ?
- ☐ 大家都可以做任何事 ?
- ☐ 專案型矩陣授權策略 ?
- ☐ 登入成功的使用者可以做任何事 ?
- ☐ 矩陣型安全性 ?


在這種配置下，您可以透過一張大表格，設定每個人可以做的每件事。  
每一欄都表示一項權限。將滑鼠游標移到權限名稱上，可以看到權限代表的意義說明。

一行就是一個使用者或群組 (依據安全性領域不同，一般也叫做「角色」)。包括 "anonymous" 特殊使用者，代表沒有通過驗證的人; 另外也有 "authenticated"，代表所有驗證通過的人 (換句話說，就是除了匿名使用者以外的所有人)。透過表格下方的文字方塊可以新增使用者、群組、角色進來，按一下 [x] 圖示可以把它由表格中移掉。

儲存

Apply


# 權限設定: 管理jenkins --> Manage and Assign Roles


 **Jenkins**


搜尋


ciduser 登出


Jenkins


 新增作業


 使用者

 建置歷程

 **管理 Jenkins**

 我的視景

 Lockable Resources

 New View

建置佇列

佇列中沒有建置作業。

建置執行程式狀態

1 開置

2 開置

## 管理 Jenkins

新版的 Jenkins (2.260) 已經可以下載 (改版記錄)。

Warnings have been published for the following currently installed components:

[Validating String Parameter Plugin 2.4:](#)  
[Stored XSS vulnerability](#)

[Script Security Plugin 1.74:](#)  
[Sandbox bypass vulnerability](#)

[Mailer Plugin 1.32:](#)  
[Missing hostname validation](#)


[Email Extension Plugin 2.74:](#)  
[Missing hostname validation](#)


[Lockable Resources plugin 2.8:](#)  
[CSRF vulnerability](#)


[Go to plugin manager](#)


[Configure which of these warnings are shown](#)

## System Configuration


 **設定系統**  
設定全域設定及路徑。


 **Global Tool Configuration**  
Configure tools, their locations and automatic installers.


 **管理外掛程式**  
外掛程式是能擴充 Jenkins 功能的程式，您可以在這裡新增、移除、停用或是啟用它們。  
有更新


 **管理節點**  
新增、移除、控制或監控 Jenkins 執行作業的各個節點。

## Security


 **設定全域安全性**  
保護 Jenkins，定義誰可以存取或是使用系統。

 **Manage Credentials**  
Configure credentials


 **Configure Credential Providers**  
Configure the credential providers and types


 **Manage and Assign Roles**  
Handle permissions by creating roles and assigning them to users/groups


# Manage and Assign Roles


 **Jenkins**


[Jenkins](#) > [Manage and Assign Roles](#)


 新增作業


 使用者

 建置歷程

 管理 Jenkins

 我的視景

 Lockable Resources

 New View

建置佇列

佇列中沒有建置作業。

建置執行程式狀態

1 閒置

2 閒置



## Manage and Assign Roles

 [Manage Roles](#)  
Manage Roles

 [Assign Roles](#)  
Assign Roles

 [Role Strategy Macros](#)  
Provides info about macro usage and available macros



# Manage Roles

roles綁定它可以做的事情，或者item roles 綁定他可以用的專案，在這個專案可以做甚麼 (pattern那邊可以用正規表示法)

Jenkins Manage and Assign Roles

## Manage Roles

### Global roles

Role	整體			Credentials				Agent									作業										執行			視景				SCM	Lockable	Unlocked
	Administer	Read	SystemRead	Create	Delete	ManageDomains	Update	View	Build	Configure	Connect	Create	Delete	Disconnect	ExtendedRead	Provision	Build	Cancel	Configure	Create	Delete	Discover	ExtendedRead	Move	Read	Workspace	Delete	Replay	Update	Configure	Create	Delete	Read	Tag	Reserve	
Developer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
Job Builder	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
ally	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		
cicduer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>		
deployer	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	

Role to add

### Item roles

Role	Pattern	Credentials				作業										執行			SCM	Lockable Resources			
		Create	Delete	ManageDomains	Update	View	Build	Cancel	Configure	Create	Delete	Discover	ExtendedRead	Move	Read	Workspace	Delete	Replay	Update	Tag	Reserve	Unlock	View
group_builder	"co-jcic-connector-publish-image"	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
group_deployer	"Pipeline co-jcic-connector-notify-ocp"	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
group_developer	"co-jcic-connector-env"	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Role to add






# Assign Roles

user 綁 roles (Global roles) (因為只有role可以綁可以做的事情)

user綁item role(類似group感覺，給專案做設定)

## Assign Roles

### Global roles






User/group	Developer	Job Builder	admin	ally	cicduer	deployer
 Ally Lee	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 mark	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 paul	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
 sarah	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 Anonymous	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

User/group to add



Add

### Item roles

User/group	group_builder	group_deployer	group_developer
 Ally Lee	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
 mark	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
 paul	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
 sarah	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
 Anonymous	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

User/group to add



Add

實作:

新增4個使用者 User\_A, User\_B , User\_C ,User\_D

有個 test floder, 裡面有三個pipeline

test1,test2,test3

User\_A ,是test專案的領導,可看到專案底下所有pipeline

User\_B ,可以看到test專案, 並且可以使用test1 pipeline,

User\_C, 可以看到test專案, 並且可以使用test2 pipeline,

User\_D,是訪客只能看見

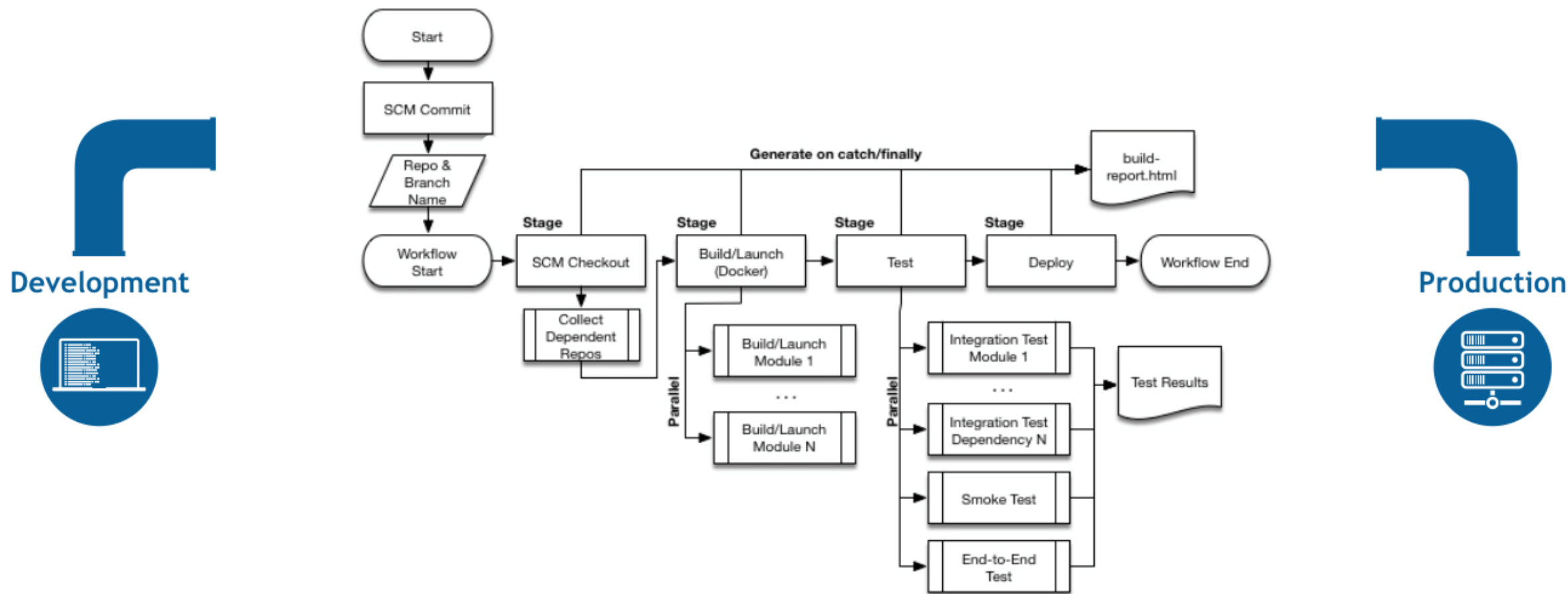


# Jenkins

Pipeline實作教學

# Jenkins pipeline 介紹

Jenkins Pipeline是一套插件，支持將持續交付管道實現和集成到 Jenkins 中。



# 準備pipeline 所需環境

1. 安裝所需的plugin

## 準備 workspace環境 **Global Tool**

1. 找到包含的image
2. Dockerfile自製image
3. 嘗試是否可以把 mnt進去，或進入container安裝

# pipeline 常用語法

```
pipeline {  
    /* insert Declarative Pipeline here */  
}
```

在聲明式流水線中有效的基本語句和表達式遵循與

- 流水線頂層必須是一個 `pipeline { }`
- 沒有分號作為語句分隔符，，每條語句都必須在自己的行上。
- 塊只能由 屬性引用語句被視為無參方法調用。例如, `input`被視為`input()`

# agent

指定你pipeline將會在Jenkins環境中執行的位置，這取決於 該部分必須在 agent 的設定

```
pipeline {  
    /* insert Declarative Pipeline here */  
    agent any  
}
```

## any

在任何可用的代理上執行流水線或階段。例如: `agent any`

## none

當在 比如: `pipeline stage agent agent none`



# environment

設定這個agent內部的环境變數，或者特定步驟中的环境變數

```
pipeline {  
    agent any  
  
    environment {  
        app_env = "uat"  
    }  
  
    stages {  
        stage('準備環境') {  
            steps {  
                sh 'printenv'  
            }  
        }  
    }  
}
```

# stages

在pipeline 用於設立多個pipeline階段關卡，包著多個的stage(關卡)

```
pipeline {  
  agent any  
  stages {  
    stage('Example') {  
      steps {  
        echo 'Hello World'  
      }  
    }  
  }  
}
```

# stage('Example')

設定階段性的關卡與關卡名稱

## steps

在關卡內要執行的步驟，包含一個或多個步驟

JavaScript ▾

```
pipeline {  
  agent any  
  stages {  
    stage('Example') {  
      steps {  
        echo 'Hello World'  
        sh 'printenv'  
      }  
    }  
  }  
}
```

## 建立第一個pipeline

```
pipeline {  
  agent any  
  stages {  
    stage('Example') {  
      steps {  
        echo 'Hello World'  
      }  
    }  
  }  
}
```

# Pipeline的排程設定

使用cron表達時間，如: 00 10 26 10 \*，此設定為10月26日早上10:00會進行建置（以jenkins內部時間為依據）

### Build Triggers

☐ Build after other projects are built

☒ 定期建置

排程

40 07 26 10 \*

**⚠ Spread load evenly by using 'H 07 26 10 \*' rather than '40 07 26 10 \*'**

Would last have run at Monday, October 26, 2020 7:40:16 AM UTC; would next run at Tuesday, October 26, 2021 7:40:16 AM UTC.

☐ Build when a change is pushed to GitLab. GitLab webhook URL: https://192.168.1.47:8443/project/co-jcic-connector-env

☐ GitHub hook trigger for GITScm polling

☐ 輪詢 SCM

☐ 停用Build

☐ 靜候時間

☐ 遠端觸發建置 (例如: 透過 Script 腳本)

### Advanced Project Options

儲存

Apply

進階...

五個欄位分別代表

1. 分 (MINUTE)：0–59 分
2. 時 (HOUR)：0–23 時
3. 日 (DOM)：1–31 日
4. 月 (MONTH)：1–12 月
5. 星期 (DOW)：星期 0–7（其中 0 與 7 都代表星期天）

# Jenkinsfile

Jenkinsfile 是Pipeline 的腳本，由Groovy語言構成。

- 方便code review 和迭代pipeline
- 方便追蹤pipeline的修改和紀錄每次變更
- 確保pipeline的來源正確性，方便專案成員的多人的追蹤和編輯

# Jenkinsfile - github建立Jenkinsfile










在github建立Jenkinsfile

1 / assertj-n-mockito Private

[Issues](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)

main 1 branch 0 tags

[Go to file](#) [Add file](#) [Code](#)

 allylin0711 Update Jenkinsfile	16a3443 3 minutes ago	 4 commits
 ide-config	project	4 days ago
 src	project	4 days ago
 .editorconfig	project	4 days ago
 .gitignore	project	4 days ago
 Jenkinsfile	Update Jenkinsfile	3 minutes ago
 README.md	project	4 days ago
 pom.xml	project	4 days ago

# Jenkinsfile - 建立ssh key

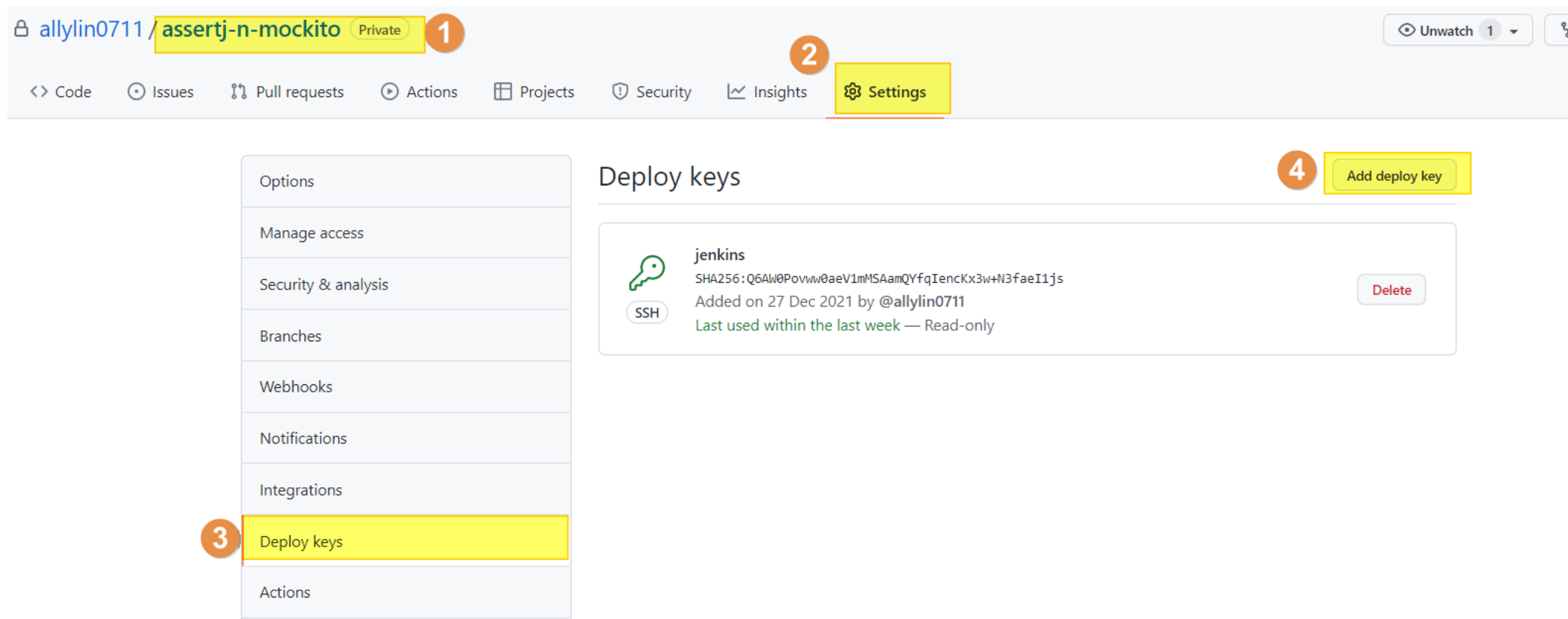
建立ssh key ,記得要把key放在jenkins\_home/.ssh內部且jenkins要有權限

```
jenkins@9a4cf2dcfbce:/opt$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/var/jenkins_home/.ssh/id_rsa): id_rsa
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in id_rsa
Your public key has been saved in id_rsa.pub
The key fingerprint is:
SHA256:Q6AW0Povww0aeVlmMSAamQYfqIencKx3w+N3faeIljs jenkins@9a4cf2dcfbce
The key's randomart image is:
+---[RSA 3072]---+
|.o+=o o.|
|..+++.+ .o|
|. +o.o .o|
|+ *. .+|
|. * + . +S|
|o + O . .|
|. B * o|
|. = + o.E..|
| + o. o+.o|
+----[SHA256]-----+
jenkins@9a4cf2dcfbce:/opt$ ls
id_rsa id_rsa.pub java jenkins-plugin-manager.jar maven
```



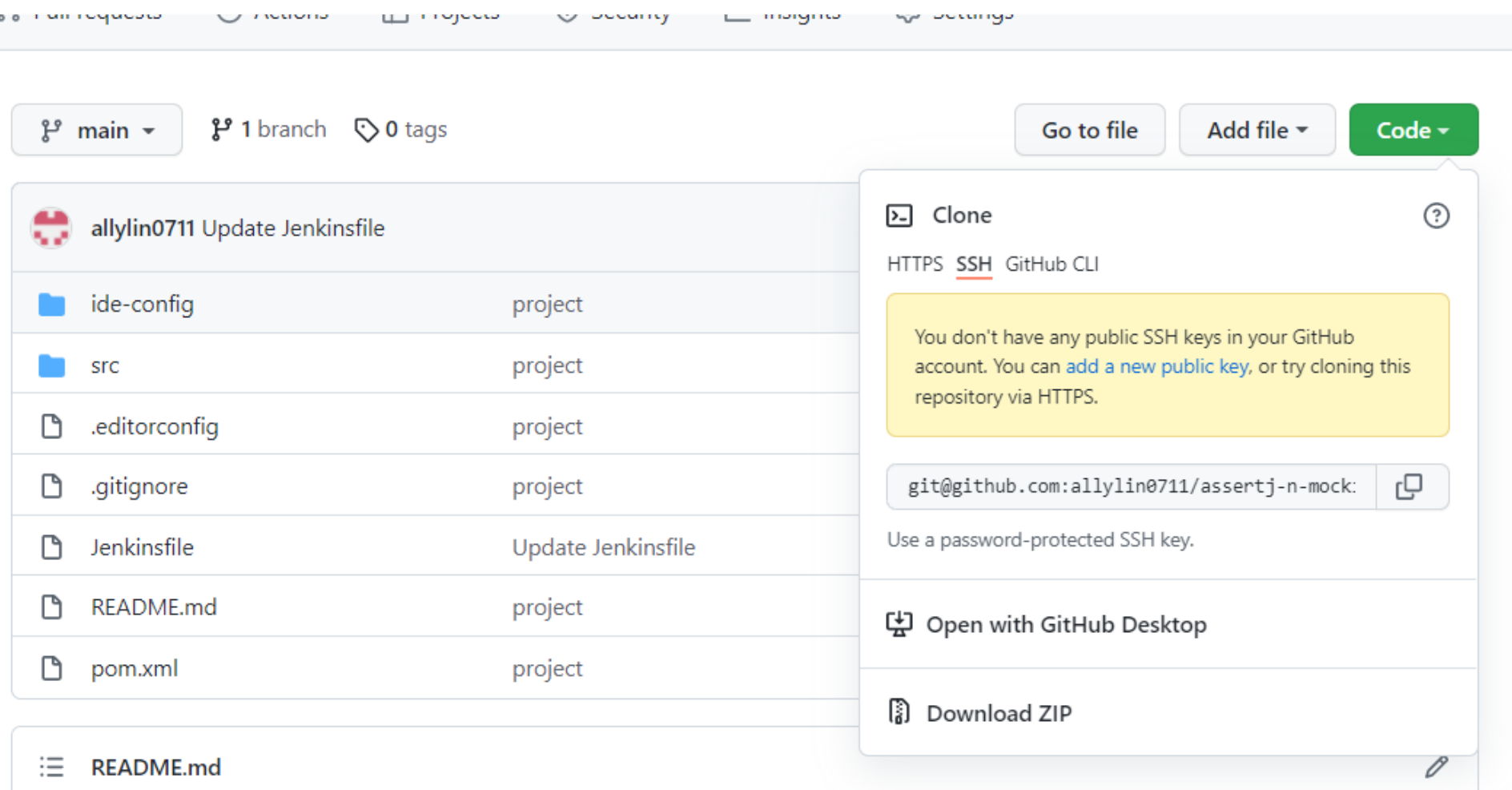
# Jenkinsfile - 在github建立deploy key

Id\_rsa.pub是公鑰放在github deploy keys內



The screenshot shows the GitHub interface for a repository named `allylin0711/assertj-n-mockito`. The repository is marked as `Private`. The navigation bar at the top includes links for `Code`, `Issues`, `Pull requests`, `Actions`, `Projects`, `Security`, `Insights`, and `Settings`. The `Settings` tab is selected, indicated by a yellow highlight and a red circle with the number 2. On the left sidebar, the `Deploy keys` option is highlighted in yellow, with a red circle and the number 3 next to it. The main content area is titled `Deploy keys` and features a yellow `Add deploy key` button with a red circle and the number 4. Below this, a single deployment key named `jenkins` is listed. It is an `SSH` key with the SHA256 hash `Q6AW0PovwW0aeV1mMSAamQYfqIencKx3w+N3faeI1js`. It was added on 27 Dec 2021 by `@allylin0711` and is marked as `Read-only` with the note `Last used within the last week`. A `Delete` button is visible next to the key.

# Jenkinsfile - 跟github的SSH連結



The screenshot shows the GitHub repository page for 'allylin0711 Update Jenkinsfile'. The repository has 1 branch and 0 tags. The file list includes 'ide-config' (project), 'src' (project), '.editorconfig' (project), '.gitignore' (project), 'Jenkinsfile' (Update Jenkinsfile), 'README.md' (project), and 'pom.xml' (project). The 'Code' dropdown menu is open, showing options: 'Clone', 'Open with GitHub Desktop', and 'Download ZIP'. The 'Clone' option is selected, and the 'SSH' method is highlighted. A yellow warning box states: 'You don't have any public SSH keys in your GitHub account. You can [add a new public key](#), or try cloning this repository via HTTPS.' The SSH URL is 'git@github.com:allylin0711/assertj-n-mock:'.

File/Folder	Type
ide-config	project
src	project
.editorconfig	project
.gitignore	project
Jenkinsfile	Update Jenkinsfile
README.md	project
pom.xml	project

# Jenkinsfile -建立pipeline

## Pipeline

Definition

Pipeline script from SCM

SCM ?

Git

Repositories ?

Repository URL ?

git@github.com:allylin0711/assertj-n-mockito.git

Credentials ?

jenkins (qithub for jenkins ssh key) 

Add

進階...

Add Repository

Repository URL ?

Please enter Git repository.

Credentials ?


- none - 

Add

test

Jenkins

# Jenkinsfile -新增add credentials

 **Add Credentials**

**Domain**  
Global credentials (unrestricted)

**Kind**  
SSH Username with private key

**Scope ?**  
Global (Jenkins, nodes, items, all child items, etc)

**ID ?**  
github\_for\_jenkins

**Description ?**  
github for jenkins ssh key

**Username**  
jenkins

☐ Treat username as secret ?

**Private Key**  
☒ Enter directly  
**Key**  
Enter New Secret Below  
這裡放私鑰\_ id\_rsa

**Branches to build** ?

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

X

\*/main

Add Branch

**儲存庫瀏覽器** ?

(自動)

▼

**Additional Behaviours**

新增 ▼

**Script Path** ?

Jenkinsfile

☒ **Lightweight checkout** ?

[Pipeline Syntax](#)

# Jenkinsfile –可以看到每次的commit紀錄和差異

The screenshot displays the SoftLeader IDE interface with the following components:

- Top Bar:** Contains menu items (File, Edit, View, Repository, Actions, Tools, Help) and a toolbar with icons for Commit, Pull, Push, Fetch, Branch, Merge, Stash, Discard, and Tag. On the right, there are icons for Git-flow, Remote, Terminal, Explorer, and Settings.
- Repository Tabs:** Shows several open repositories: jasmine-express (58), fressia (16), tsl-bank (1), prudential-echange, and assertj-n-mockito.
- Left Sidebar:**
  - WORKSPACE:** Includes File Status, History (selected), and Search.
  - BRANCHES:** Shows the 'main' branch selected.
  - TAGS:** No tags are currently visible.
  - REMOTES:** No remotes are currently visible.
  - STASHES:** No stashes are currently visible.
- Main Panel:**
  - Commit History:** A table showing the commit history for the 'main' branch.

Graph	Description	Date	Author	Commit
	Update Jenkinsfile	27 十二月 2021 11	Ally <61672386+>	16a3443
	Update Jenkinsfile	27 十二月 2021 11:	Ally <61672386+>	6044127
	Create Jenkinsfile	27 十二月 2021 10:	Ally <61672386+>	b8acf71
	project	23 十二月 2021 11:	ally <ally.lee@sof>	99de51e
  - Commit Details:** For the selected commit (16a3443), it shows:
    - Commit:** 16a34433e689984319be8701e11c449f0d1ea6df [16a3443]
    - Parents:** 60441271e4
    - Author:** Ally <61672386+allylin0711@users.noreply.github.com>
    - Date:** 2021年12月27日 上午 11:37:09
    - Committer:** GitHub
  - File Explorer:** Shows the 'Jenkinsfile' file selected.
  - Diff View:** Displays the changes in the Jenkinsfile. The selected hunk (Hunk 1: Lines 21-32) shows the addition of a new stage and steps.

```

21 21 .....sh 'printenv'
22 22 .....}
23 23 .....}
24 24 .....+.....
25 25 .....+.....stage('Jenkinsfile-') {
26 26 .....+.....steps {
27 27 .....+.....echo 'jenkinsfile 與 github ssh 連線測試成功'
28 28 .....+.....}
29 29 .....+.....}
24 30 .....
25 31 .....
26 32 .....

```

# 環境準備 - 新增maven , java版本

```
version: '3.8'
services:
  jenkins:
    image: jenkins/jenkins:2.325-jdk8
    restart: always
    ports:
      - 8082:8080
      - 50002:50000
      - 8443:8443
    # environment:
    #   - "JENKINS_OPTS= --httpPort=8080 --httpsPort=8443 --httpsKeyStore=/var/ssl/server1
    volumes:
      #   - C:\Users\admin\Desktop\Dockerfile\47\jenkins\firtbank\compose\ssl:/var/ssl
      - C:\Users\admin\Desktop\fisrtbank_Workshop\jenkins\jenkins_home:/var/jenkins_home
      - C:\Users\admin\.m2:/var/jenkins_home/.m2
      - /var/run/docker.sock:/var/run/docker.sock
      - C:\Users\admin\Desktop\fisrtbank_Workshop\jenkins\opt:/opt
```

```
C:\Users\admin\Desktop\testpipeline\bak\mimosa-prudential-echange-master\console>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED        STATUS        PORTS
9a4cf2dcfbce   jenkins/jenkins:2.325-jdk8         "/sbin/tini -- /usr/..." 3 days ago    Up 3 days    0.0.0.0:8082->8080/tcp, :::8082->8080/tcp, 0.0.0.0:50002->50000/tcp, :::50002->50000/tcp
jenkins_jenkins_1

C:\Users\admin\Desktop\testpipeline\bak\mimosa-prudential-echange-master\console>docker exec -it 9a /bin/bash
jenkins@9a4cf2dcfbce:/$ cd /opt
jenkins@9a4cf2dcfbce:/opt$ ls
java jenkins-plugin-manager.jar maven
jenkins@9a4cf2dcfbce:/opt$ ls -l
total 6048
drwxrwxrwx 1 root    root      4096 Dec 24 02:57 java
-rw-r--r-- 1 jenkins jenkins 6192861 Dec 20 10:21 jenkins-plugin-manager.jar
drwxrwxrwx 1 root    root      4096 Dec 23 04:39 maven
jenkins@9a4cf2dcfbce:/opt$
```

# Pipeline的 workspace位置

ns > jenkins\_home >



搜尋 jenkins\_home

名稱	修改日期	類型	大小
java	2021/12/23 上午 11:50	檔案資料夾	
.m2	2021/12/23 上午 11:50	檔案資料夾	
caches	2021/12/20 下午 05:19	檔案資料夾	
jobs	2021/12/24 下午 12:19	檔案資料夾	
logs	2021/12/20 下午 02:17	檔案資料夾	
mnt	2021/12/20 下午 06:21	檔案資料夾	
nodes	2021/12/20 下午 01:52	檔案資料夾	
plugins	2021/12/24 上午 11:03	檔案資料夾	
secrets	2021/12/20 下午 05:19	檔案資料夾	
tools	2021/12/27 上午 10:19	檔案資料夾	
updates	2021/12/23 上午 11:57	檔案資料夾	
userContent	2021/12/20 下午 01:52	檔案資料夾	
users	2021/12/21 上午 10:33	檔案資料夾	
war	2021/12/20 下午 01:52	檔案資料夾	
workflow-lib	2021/12/20 下午 01:56	檔案資料夾	
workspace	2021/12/27 上午 10:44	檔案資料夾	
.bash_history	2021/12/24 上午 11:15	BASH_HISTORY 檔案	2 KB
.lastStarted	2021/12/24 上午 11:04	LASTSTARTED 檔案	0 KB
.owner	2021/12/27 上午 10:35	OWNER 檔案	1 KB
atomic1284754155115259369tmp	2021/12/21 上午 10:52	檔案	0 KB



# .m2檔位置

```
- :: drwxr-xr-x. 3 1000 mygroup 24 8月 26 12:19 .m2
-rw-r--r--. 1 1000 mygroup 216 10月 26 2020 njnm.plugins.aws_batch.AwsBatchBuilder.xml
-rw-r--r--. 1 1000 mygroup 907 8月 23 17:43 nodeMonitors.xml
drwxrwxrwx. 2 1000 cicduser 6 9月 22 2020 nodes
-rw-r--r--. 1 1000 mygroup 275 10月 26 2020 org.jenkinsci.plugins.github_branch_source.GitHubConfigu
-rw-r--r--. 1 1000 mygroup 46 8月 30 11:36 org.jenkinsci.plugins.workflow.flow.FlowExecutionList.xml
-rw-r--r--. 1 1000 mygroup 153 10月 26 2020 org.jenkinsci.plugins.workflow.flow.GlobalDefaultFlowDur
-rw-r--r--. 1 1000 mygroup 219 10月 26 2020 org.jenkinsci.plugins.workflow.libs.GlobalLibraries.xml
-rw-r--r--. 1 1000 mygroup 236 10月 26 2020 org.jenkins.plugins.lockableresources.LockableResourcesM
-rw-r--r--. 1 1000 mygroup 42 9月 3 09:32 .owner
drwxrwxrwx. 87 1000 cicduser 12288 8月 30 11:14 plugins
-rw-r--r--. 1 1000 mygroup 131 8月 30 11:36 queue.xml
-rw-r--r--. 1 1000 mygroup 131 8月 23 17:31 queue.xml.bak
-rw-r--r--. 1 1000 mygroup 369 10月 26 2020 scriptApproval.xml
-rwxrwxrwx. 1 1000 cicduser 64 9月 22 2020 secret.key
-rwxrwxrwx. 1 1000 cicduser 0 9月 22 2020 secret.key.not-so-secret
drwxrwxrwx. 4 1000 cicduser 4096 10月 26 2020 secrets
drwxrwxrwx. 2 1000 cicduser 182 9月 2 17:43 updates
drwxrwxrwx. 2 1000 cicduser 24 9月 22 2020 userContent
drwxrwxrwx. 7 1000 cicduser 184 10月 8 2020 users
drwxrwxrwx. 11 1000 cicduser 4096 9月 22 2020 war
drwxrwxrwx. 2 1000 cicduser 6 9月 22 2020 workflow-libs
drwxr-xr-x. 12 1000 mygroup 208 8月 30 11:35 workspace
[root@localhost jenkins]# pwd
/data/jenkins/test_test/jenkins
[root@localhost jenkins]#
```

# 若執行pipeline 的git指令中遇到有ssl錯誤

若在git clone錯誤,  
可在pipeline中加入或者在Jenkins container 內加入:

```
git -c http.sslVerify=false clone
```

```
git config --global http.sslVerify false
```

# Jenkinsfile (Declarative Pipeline)

```
pipeline {  
    agent any  
    stages {  
        stage('source code') {  
            steps {  
                echo 'git source code..'   
            }  
        }  
  
        stage('Build') {  
            steps {  
                echo 'Building..'   
            }  
        }  
  
        stage('Test') {  
            steps {  
                echo 'Testing..'   
            }  
        }  
  
        stage('Deploy') {  
            steps {  
                echo 'Deploying....'   
            }  
        }  
    }  
}
```

撰寫自己的pipeline ~

寫pipeline將source code從建置包版

# 給個小提示~

```
stages {
  stage('準備環境') {
    steps {
      sh 'java -version'
      sh 'mvn -version'
      sh 'git --version'
      sh 'pwd'
      sh 'printenv'
    }
  }
  stage('srouce code') {
    steps {
      sh 'pwd'
      sh 'rm -Rf assertj-n-mockito'
      sh 'git clone git@github.com:allylin0711/assertj-n-mockito.git'
    }
  }
  stage('目前目錄，進入目錄') {
    steps {
      sh 'pwd'
      dir("assertj-n-mockito") {
        sh 'pwd'
      }
    }
  }
  stage('清理環境') {
    steps {
      dir("assertj-n-mockito") {
        sh "mvn clean -e "
      }
    }
  }
}
```