LAB3

What are the different methods to trigger pipeline in Jenkins?
- Trigger builds remotely
- Build after other projects are built
- Build periodically
- GitHub hook trigger
- Poll SCM

What is the benefit of using master-slave architecture rather than building on master only?
- Produce backups
- Scale out the application
- Split your work without causing a mess
- Top-notch performance

What is the difference between authorization and authentication?

- Authentication verifies the identity of a user or service.
- Authorization determines their access rights.

What is the benefit of making an organization job in Jenkins?

Organization Folders enable Jenkins to monitor an entire GitHub Organization, Bitbucket Team/Project, GitLab organization, or Gitea organization and automatically create new Multibranch Pipelines for repositories which contain branches and pull requests containing a Jenkinsfile.

make jenkins-shared-library and make your jenkinsfile which you used in lab2
to point to this library

MarwanIbrahimKhamis Update check.groovy

..

check.groovy	Update check.g
deployment.groovy	up
test.groovy	up

Give feedback

36 lines (30 sloc) | 744 Bytes

```
1   @Library('lab3')_
2
3   pipeline {
4       agent any
5       parameters {
6           booleanParam(name:'project', defaultValue: true, description:'this paramater help you to know project name')
7           choice(name: 'namespace', choices:['dev','prod','stage'], description: '' )
8       }
9
10      stages {
11          stage('check') {
12              steps {
13                  check(namespace)
14
15              }
16          }
17
18          stage('test') {
19              when {
20                  expression{
21                      params.project == true
22                  }
23              }
24              steps {
25                  test()
26              }
27          }
28
29          stage('deployment') {
30              steps {
31                  deployment()
32              }
33          }
34      }
35
36  }
```

Name ?

lab3

Default version ?

main

Currently maps to revision: 6aa35eba2714a8b8fa6ece95ccf18d9d77c26c68

☐ Load implicitly ?
☑ Allow default version to be overridden ?
☑ Include @Library changes in job recent changes ?
☐ Cache fetched versions on controller for quick retrieval ?

Retrieval method

Modern SCM

Loads a library from an SCM plugin using newer interfaces optimized for this purpose. The recommended option when available.

Source Code Management

Git

Project Repository ?

https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git

Credentials ?

MarwanIbrahimKhamis/******

Jenkins

Search (CTRL+K) 🔔 2 🛡 2 👤 Marwan Ibrahim Khamis ElSayed Aly ⏻ log o

- Status
- Changes
- Console Output
  - View as plain text
- Edit Build Information
- Delete build '#12'
- Parameters
- Git Build Data
- Restart from Stage
- Replay
- Pipeline Steps
- Workspaces
- Previous Build

✅ **Console Output**

```
Started by user Marwan Ibrahim Khamis ElSayed Aly
Obtained Jenkinsfile from git
https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git
Loading library lab3@main
Attempting to resolve main from remote references...
 > git --version # timeout=10
 > git --version # 'git version 2.30.2'
using GIT_ASKPASS to set credentials
 > git ls-remote -h -- https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git #
timeout=10
Found match: refs/heads/main revision 6aa35eba2714a8b8fa6ece95ccf18d9d77c26c68
The recommended git tool is: git
using credential github-token
 > git rev-parse --resolve-git-dir
/var/jenkins_home/workspace/lab3@libs/4b7e456f87ac3f96c9e7c7331912f9f55a93e4e00947072d382aadfa9
6671a8d/.git # timeout=10
Fetching changes from the remote Git repository
 > git config remote.origin.url
https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git # timeout=10
Fetching without tags
Fetching upstream changes from
```

← Previous Build

Fetching upstream changes from
https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.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/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git
+refs/heads/*:refs/remotes/origin/* # timeout=10
Checking out Revision 6aa35eba2714a8b8fa6ece95ccf18d9d77c26c68 (main)
> git config core.sparsecheckout # timeout=10
> git checkout -f 6aa35eba2714a8b8fa6ece95ccf18d9d77c26c68 # timeout=10
Commit message: "Update Jenkinsfile"
> git rev-list --no-walk ff958b66426d768e445c24707d70289ba299b071 # timeout=10
[Pipeline] Start of Pipeline
[Pipeline] node
Running on Jenkins in /var/jenkins_home/workspace/lab3
[Pipeline] {
[Pipeline] stage
[Pipeline] { (Declarative: Checkout SCM)
[Pipeline] checkout
The recommended git tool is: git
using credential github-token
> git rev-parse --resolve-git-dir /var/jenkins_home/workspace/lab3/.git # timeout=10
Fetching changes from the remote Git repository
> git config remote.origin.url
https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git # timeout=10
Fetching upstream changes from
https://github.com/MarwanIbrahimKhamis/Marwan_Sprints_Bootcamp.git
> git --version # timeout=10

[Pipeline] {
[Pipeline] stage
[Pipeline] { (check)
[Pipeline] echo
checking your code
[Pipeline] echo
prod
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (test)
[Pipeline] echo
testing your app
[Pipeline] }
[Pipeline] // stage
[Pipeline] stage
[Pipeline] { (deployment)
[Pipeline] echo
your code is deployed right now
[Pipeline] echo
this build number 12
[Pipeline] }
[Pipeline] // stage
[Pipeline] }
[Pipeline] // withEnv
[Pipeline] }
[Pipeline] // node
[Pipeline] End of Pipeline
Finished: SUCCESS