CERTIF ACADEMY

Jenkins

Mettre en place la CI/CD

CERTIF ACADEMY

Compétence demandée :

Savoir écrire un Jenkinsfile

CERTIF ACADEMY

1. Organisation de Jenkins
2. Declarative pipeline (vs scripted pipeline)
3. Codons le pipeline
4. Jenkinsfile

CERTIF ACADEMY

Organisation Jenkins

CERTIF ACADEMY

ATELIER

Qu‘est-ce qu‘un agent / node / slave dans Jenkins ?

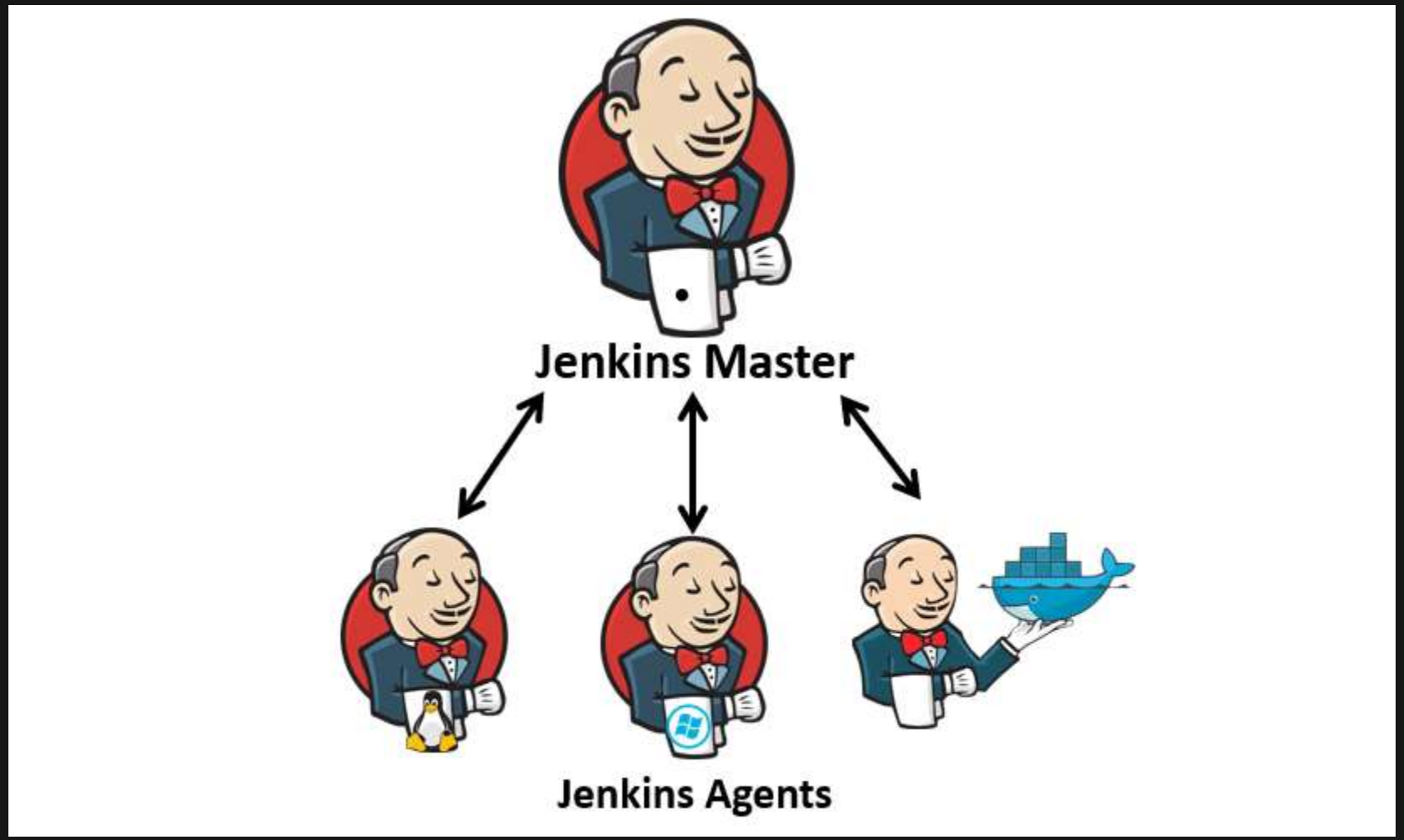
CERTIF ACADEMY

Un agent est un logiciel qui peut être utilisé par Jenkins pour faire tourner les jobs.

Ainsi Jenkins peut être le master/controller et peut utiliser 30 agents (sur linux, windows, mac ou même un conteneur) pour lancer les jobs.

C’est plutôt utile pour lancer des builds sur des plateformes spécifiques : utile pour les applications mobiles.

CERTIF ACADEMY



CERTIF ACADEMY

1. Organisation de Jenkins
2. Declarative pipeline (vs scripted pipeline)
3. Codons le pipeline
4. Jenkinsfile

CERTIF ACADEMY

Declarative pipeline

CERTIF ACADEMY

CREER UN PIPELINE

CERTIF ACADEMY

Saisissez un nom

erphrense-improved

» *Champ obligatoire*

Construire un projet free-style

Ceci est la fonction principale de Jenkins qui sert à builder (construire) votre projet. Vous pouvez intégrer tous les outils de gestion de version avec tous les systèmes de build. Il est même possible d'utiliser Jenkins pour tout autre chose qu'un build logiciel.

Pipeline

Organise des activités de longue durée qui peuvent sétendre sur plusieurs agents de construction. Adapté pour la création des pipelines (anciennement connues comme workflows) et/ou pour organiser des activités complexes qui ne sadaptent pas facilement à des tâches de type libre.

Construire un projet multi-configuration

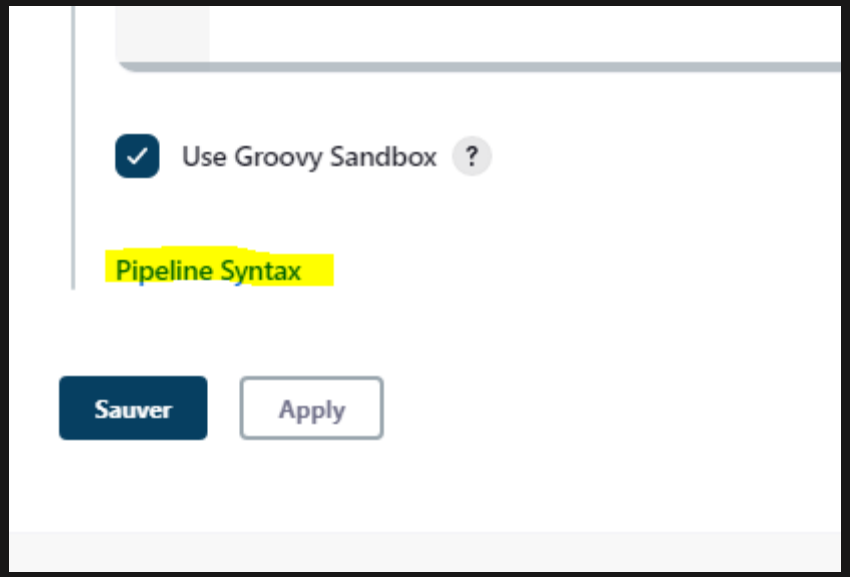
Adapté aux projets qui nécessitent un grand nombre de configurations différentes, comme des environnements de test multiples, des binaires spécifiques à une plateforme, etc.

Dossier

Crée un conteneur qui stocke des objets imbriqués. Utile pour grouper ensemble des éléments. Contrairement à une vue qui n'est qu'un filtre, un dossier crée un espace de nommage distinct, de sorte que vous pouvez avoir plusieurs éléments du même nom tant qu'ils se trouvent dans des dossiers différents.

OK

CERTIF ACADEMY



CERTIF ACADEMY

ATELIER

A votre avis, à quoi sert cette page ?

CERTIF ACADEMY

La page Pipeline Syntax est très utile pour savoir comment retranscrire les étapes de votre build !

CERTIF ACADEMY

Au lieu d'utiliser l'interface graphique de Jenkins pour définir les jobs, il est possible de le coder ! C'est un langage particulier, mais utile.

En effet, comme habituellement l'intégration et le déploiement se fait par projet, on pourrait compléter le projet par son code source et le code du CI/CD !

CERTIF ACADEMY

Nous allons donc apprendre à coder le pipeline réalisé lors du cours précédent ! ☺

CERTIF ACADEMY

1. Organisation de Jenkins
2. Declarative pipeline (vs scripted pipeline)
3. Codons le pipeline
4. Jenkinsfile

CERTIF ACADEMY

Codons le pipeline

CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

ATELIER

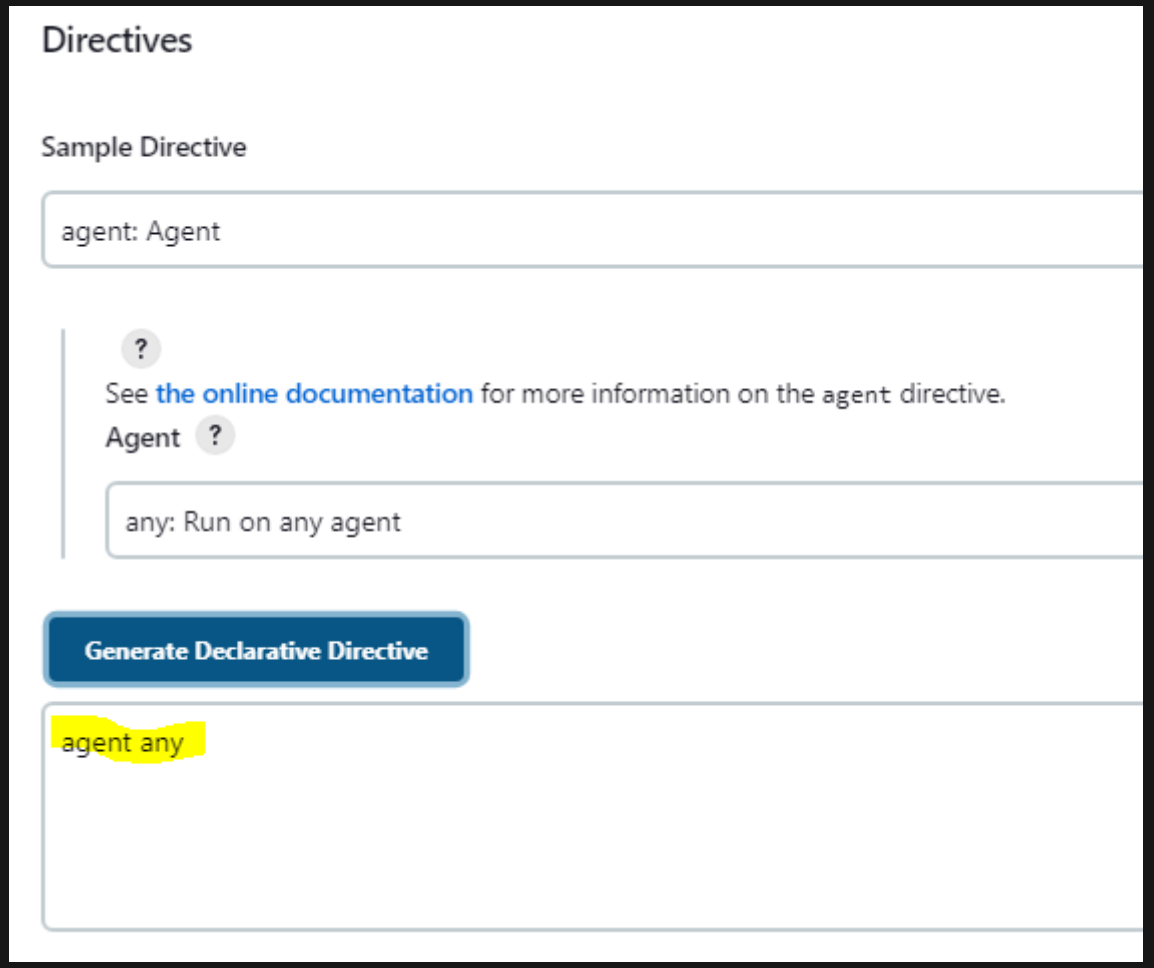
Comment traduire ces étapes en lignes de code ?

CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY



CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

Directives

Sample Directive

triggers: Triggers

?

See the online documentation for more information on the `triggers` directive.

Scrutation de l'outil de gestion de version ?

Planning ?

H/5 * * * *

Aurait été lancé à jeudi 24 novembre 2022 à 04:48:43 heure normale d'Europe centrale

Ignore post-commit hooks ?

Ajouter

Generate Declarative Directive

```
triggers {
    pollSCM 'H/5 * * * *'
}
```

CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

Steps

Sample Step

git: Git

git ?

Repository URL ?

https://github.com/xonatis-academy/epsi-dev709-ci.git

Branch ?

main

Credentials ?

- aucun -

+ Ajouter

Include in polling? ?

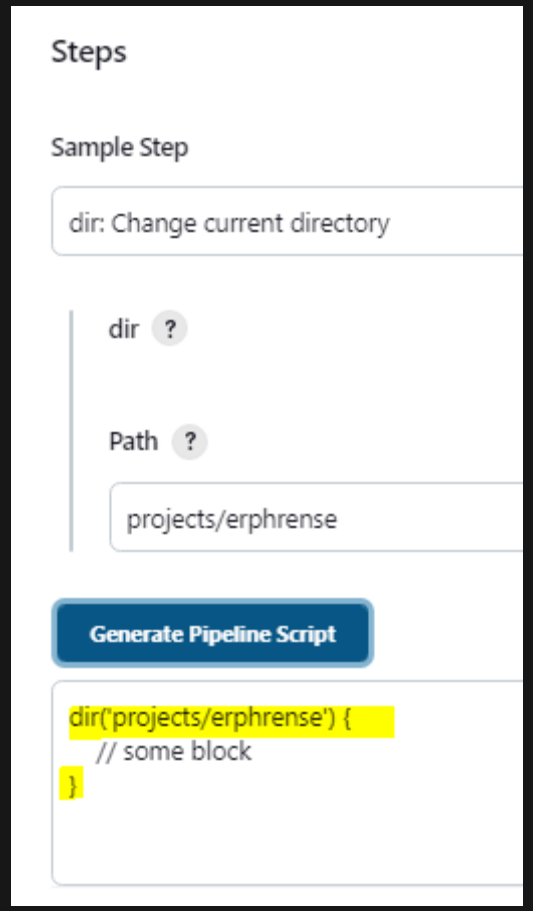
Include in changelog? ?

CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

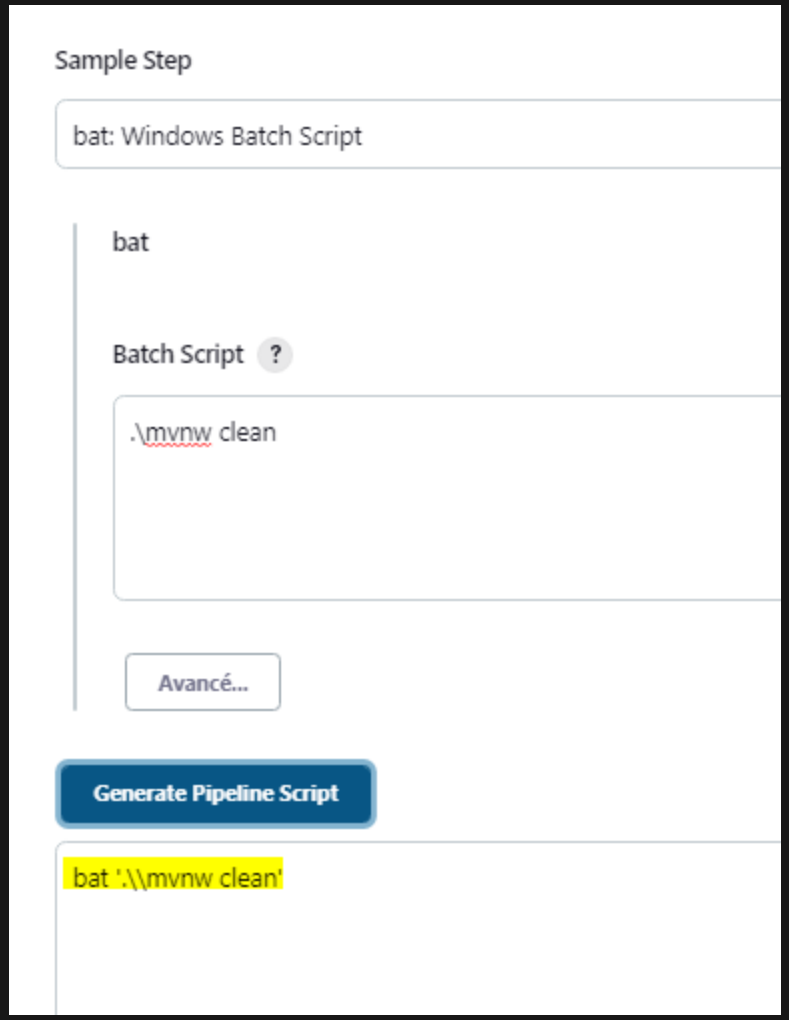


CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

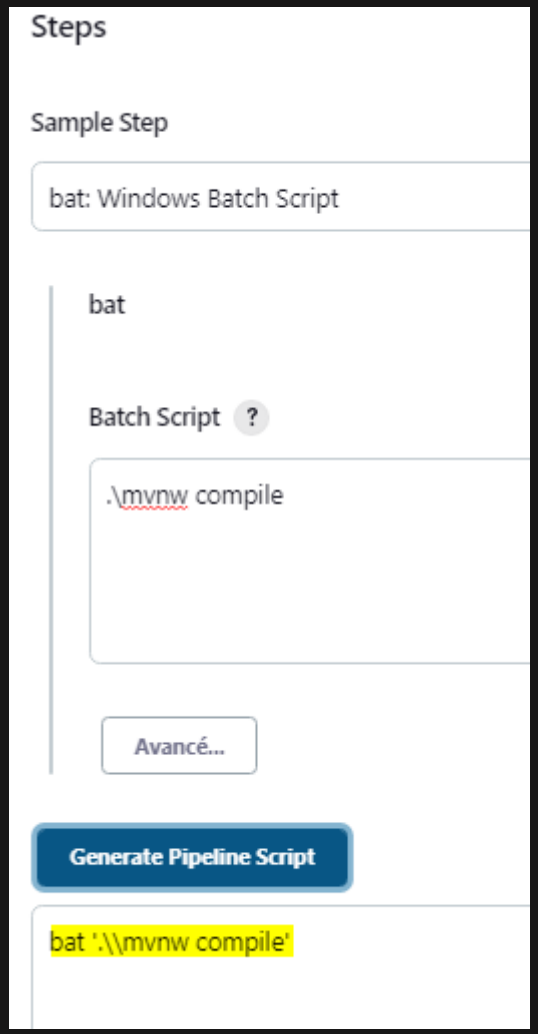


CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY



CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

Steps

Sample Step

bat: Windows Batch Script

bat

Batch Script ?

```
rename target\erphrense-0.0.1-SNAPSHOT.jar erphrense-%BUILD_NUMBER%.jar
```

Avancé...

Generate Pipeline Script

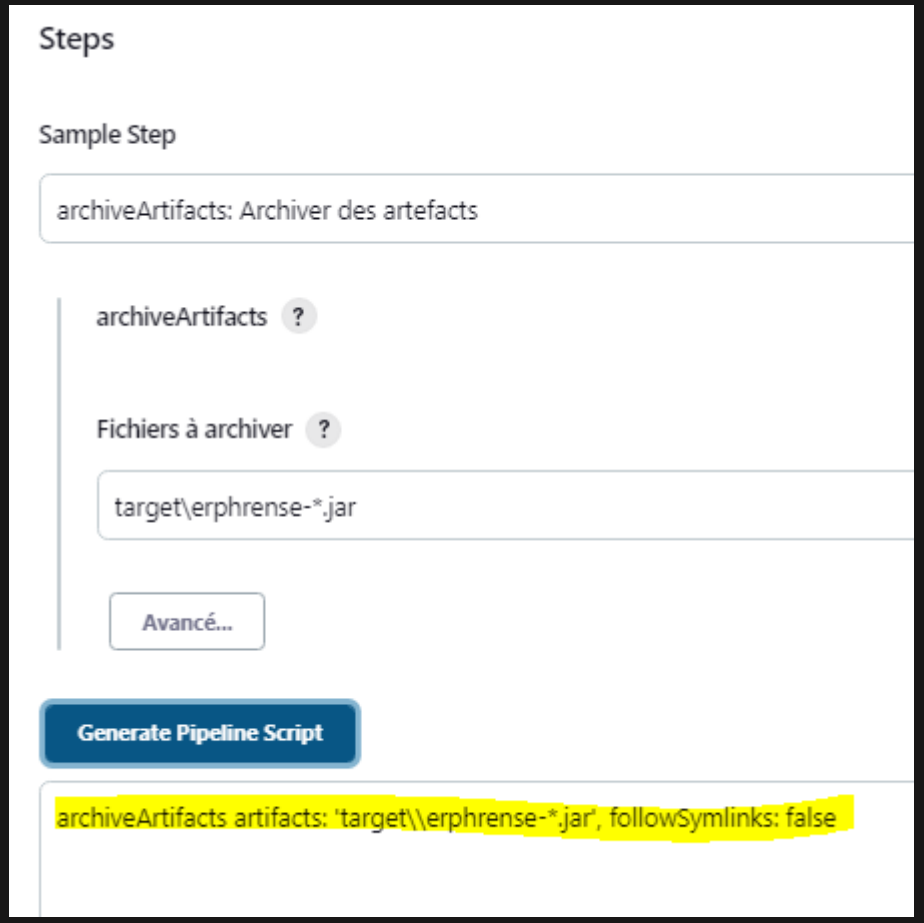
```
bat 'rename target\\erphrense-0.0.1-SNAPSHOT.jar erphrense-%BUILD_NUMBER%.jar'
```

CERTIF ACADEMY

Algorithme CI/CD :

1. Utiliser n'importe quel agent disponible
2. Ajouter un trigger pour exécuter le polling toutes les 5 minutes
3. Clone la branche principale du remote
4. Changer le répertoire courant pour aller dans le projet
5. Faire un clean
6. Faire un compile
7. Lancer les tests
8. Packager l'appli
9. Renommer l'artefact
10. Archiver l'artefact

CERTIF ACADEMY

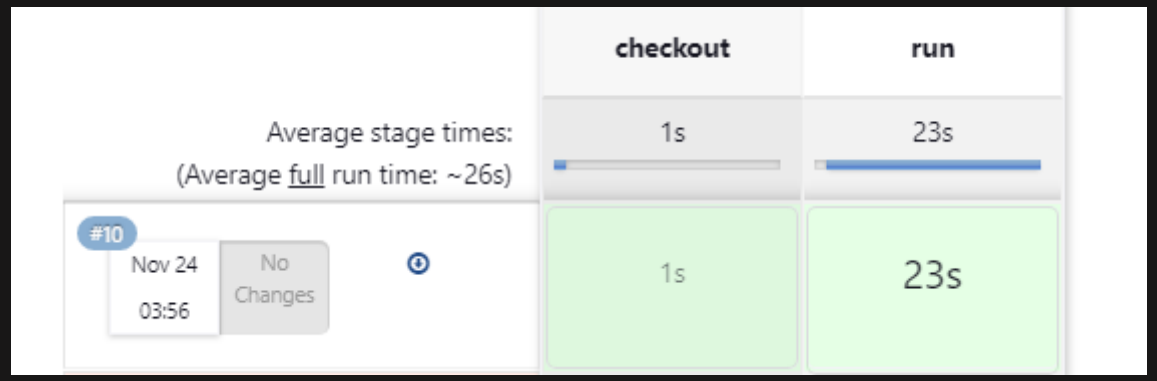


CERTIF ACADEMY

Script ?

```
pipeline {
    agent any

    triggers {
      pollSCM 'H/5 * * * *'
    }

  stages {
    stage('checkout') {
      steps {
        git branch: 'main', url: 'https://github.com/xonatis-academy/epsi-dev709-ci.git'
      }
    }

    stage('run') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw clean'
            bat './mvnw compile'
            bat './mvnw test'
            bat './mvnw package'
            bat 'rename target\\erphrense-0.0.1-SNAPSHOT.jar erphrense-%BUILD_NUMBER%.jar'
            archiveArtifacts artifacts: 'target\\erphrense-*.jar', followSymlinks: false
          }

      }
    }
  }
}
```

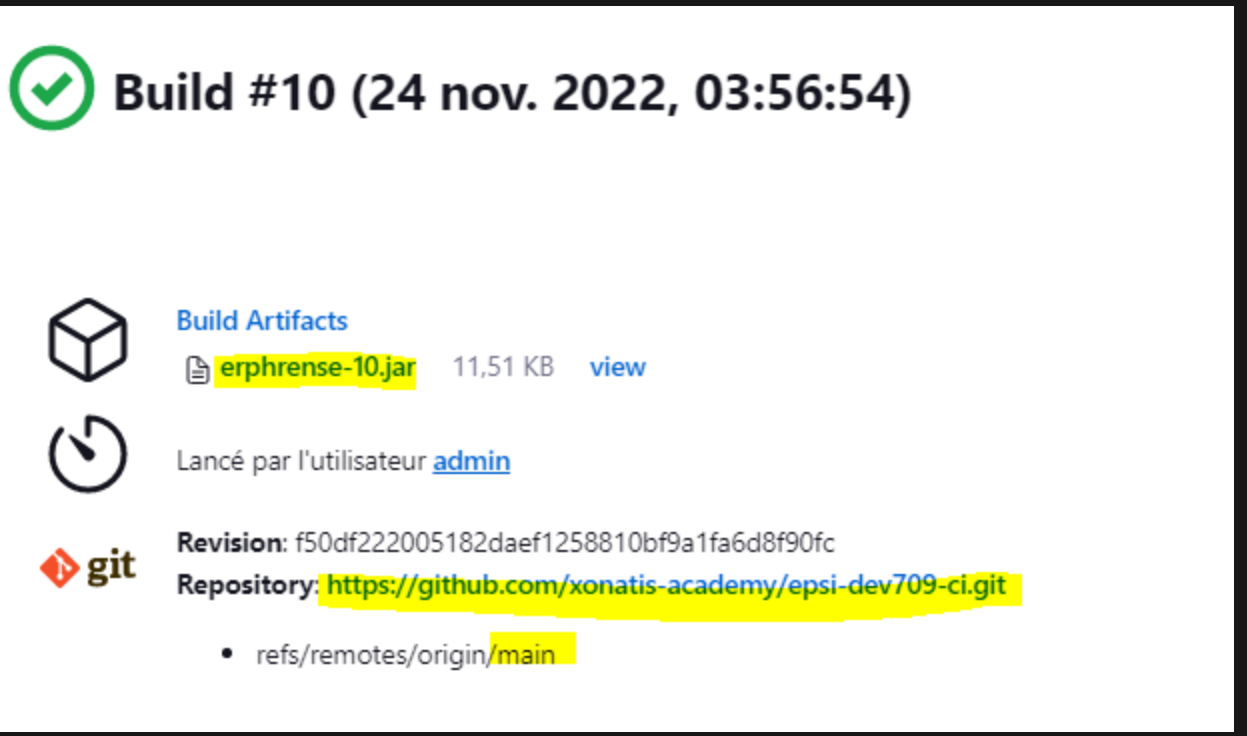
CERTIF ACADEMY



CERTIF ACADEMY

Comme vu avant, le job s’est bien déroulé !

CERTIF ACADEMY



CERTIF ACADEMY

Comme vu avant, on a construit un bon artefact !

CERTIF ACADEMY

Dernier Log du dernier accès à Git

```
Started on 24 nov. 2022, 04:08:00
Using strategy: Default
[poll] Last Built Revision: Revision f50df222005182daef1258810bf9a1fa6d8f90fc (refs/remotes/origin/main)
The recommended git tool is: NONE
No credentials specified
 > git.exe --version # timeout=10
 > git --version # 'git version 2.31.0.windows.1'
 > git.exe ls-remote -h -- https://github.com/xonatis-academy/epsi-dev709-ci.git # timeout=10
Found 1 remote heads on https://github.com/xonatis-academy/epsi-dev709-ci.git
[poll] Latest remote head revision on refs/heads/main is: f50df222005182daef1258810bf9a1fa6d8f90fc - already built by 10
Done. Took 0,6 s
No changes
```

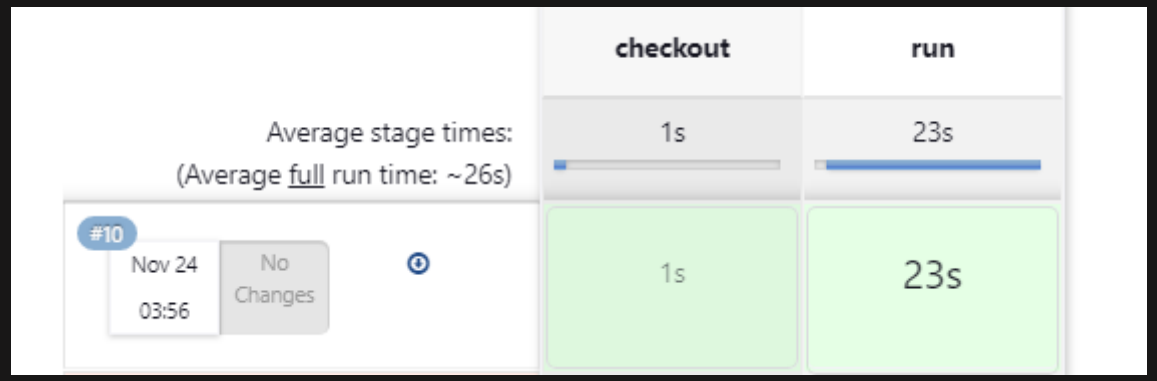
CERTIF ACADEMY

CERTIF ACADEMY

Comme vu avant, on a toujours le polling pour la détection des changements !

CERTIF
ACADEMY

CERTIF ACADEMY



CERTIF ACADEMY

ATELIER

Voyez-vous des choses à améliorer ?

CERTIF ACADEMY

Cela dit, un pipeline comme cela est trop monolithique, il serait préférable de séparer les différentes étapes en des stages différents

Script ?

```
pipeline {
    agent any

    triggers {
      pollSCM 'H/5 * * * *'
    }

  stages {
    stage('checkout') {
      steps {
        git branch: 'main', url: 'https://github.com/xonatis-academy/epsi-dev709-ci.git'
      }
    }

    stage('clean') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw clean'
          }

      }
    }

    stage('compile') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw compile'
          }

      }
    }

    stage('test') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw test'
          }

      }
    }
```

CERTIF ACADEMY

Stage View

	checkout	clean	compile	test	package	archive
Average stage times: (Average full run time: ~31s)	1s	4s	6s	8s	8s	659ms
#13 Nov 24 04:24 No Changes	1s	4s	6s	8s	8s	659ms

CERTIF ACADEMY

https://plugins.jenkins.io/git/

CERTIF ACADEMY

Checkout with defaults

Checkout from the git plugin source repository using https protocol, no credentials, and the master branch.

```
checkout([$class: 'GitSCM',
          branches: [[name: '*/master']],
          userRemoteConfigs: [[url: 'https://github.com/jenkinsci/git-plugin.git']]])
```

Checkout with a specific branch

Checkout from the git client plugin source repository using https protocol, no credentials, and a non-default branch.

```
checkout([$class: 'GitSCM',
           branches: [[name: 'stable-2.x']],
           userRemoteConfigs: [[url: 'https://github.com/jenkinsci/git-client-plugin.git']]])
```

Checkout with ssh and a private key credential

Checkout from the git client plugin source repository using ssh protocol, ssh private credentials, and a non-default branch. The git plugin supports private key credentials provided by the Jenkins credentials plugin.

```
checkout([$class: 'GitSCM',
           branches: [[name: 'stable-2.x']],
           userRemoteConfigs: [[credentialsId:  'my-ssh-private-key-id',
                                url: 'ssh://github.com/jenkinsci/git-plugin.git']]])
```

Checkout with https and a username/password credential

Checkout from the git client plugin source repository using https protocol, username/password credentials, and a non-default branch. The git plugin supports username/password credentials provided by the Jenkins credentials plugin.

CERTIF ACADEMY

1. Organisation de Jenkins
2. Declarative pipeline (vs scripted pipeline)
3. Codons le pipeline
4. Jenkinsfile

CERTIF ACADEMY

Le Jenkinsfile

CERTIF ACADEMY

Il est possible de déporter le code du pipeline directement dans le SCM (git par exemple) afin de centraliser le code du projet :

- Code source de l'application
- Code du CI/CD du pipeline

CERTIF ACADEMY

Pour ce faire, il faut créer une fichier nommé « Jenkinsfile » (par convention)

CERTIF ACADEMY

Jenkinsfile U

Jenkinsfile

```
pipeline {
    agent any

    triggers {
      pollSCM 'H/5 * * * *'
    }

  stages {

    stage('checkout') {
      steps {
        checkout
      }
    }

    stage('clean') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw clean'
          }

      }
    }

    stage('compile') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw compile'
          }

      }
    }

    stage('test') {
      steps {
          dir('projects/erphrense') {
            bat './mvnw test'
          }

```

CERTIF ACADEMY

	Declarative: Checkout SCM	checkout	clean	compile	test	package	archive
Average stage times: (Average full run time: ~31s)	1s	1s	3s	6s	7s	7s	573ms
#16 Nov 24 05:09 2 commits	1s	1s	3s	6s	7s	7s	573ms

CERTIF ACADEMY

Gestion des identifiants

CERTIF ACADEMY

Afin de gérer des identifiants dans Jenkins (n'importe lesquels), l'idée est la chose suivante :

1. Enregistrer les identifiants à l'intérieur de Jenkins
2. Lier ces identifiants à des variables dans votre job
3. Utiliser ces variables dans votre job

CERTIF ACADEMY

Afin de gérer des identifiants dans Jenkins (n'importe lesquels), l'idée est la chose suivante :

1. Enregistrer les identifiants à l'intérieur de Jenkins
2. Lier ces identifiants à des variables dans votre job
3. Utiliser ces variables dans votre job

CERTIF ACADEMY

Security

Configurer la sécurité globale
Sécuriser Jenkins; définir qui est autorisé à accéder au système.

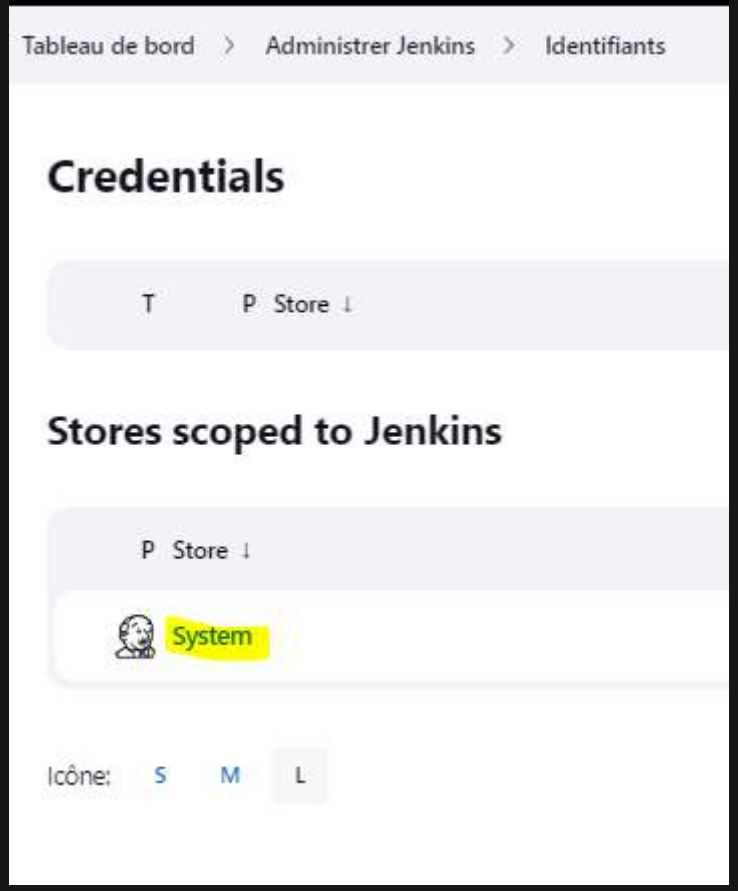
Manage Credentials
Configure credentials

Configure Credential Providers
Configure the credential providers and types

Gérer les utilisateurs
Créer/supprimer/modifier les utilisateurs qui peuvent se logger sur ce serveur Jenkins

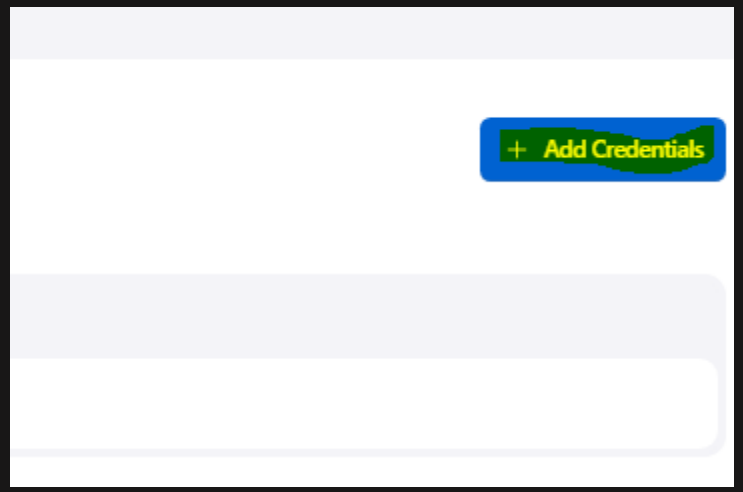
In-process Script Approval
Allows a Jenkins administrator to review proposed scripts (written e.g. in Groovy) which run inside the Jenkins process and so could bypass security restrictions. **1 scripts pending approval.**

CERTIF ACADEMY

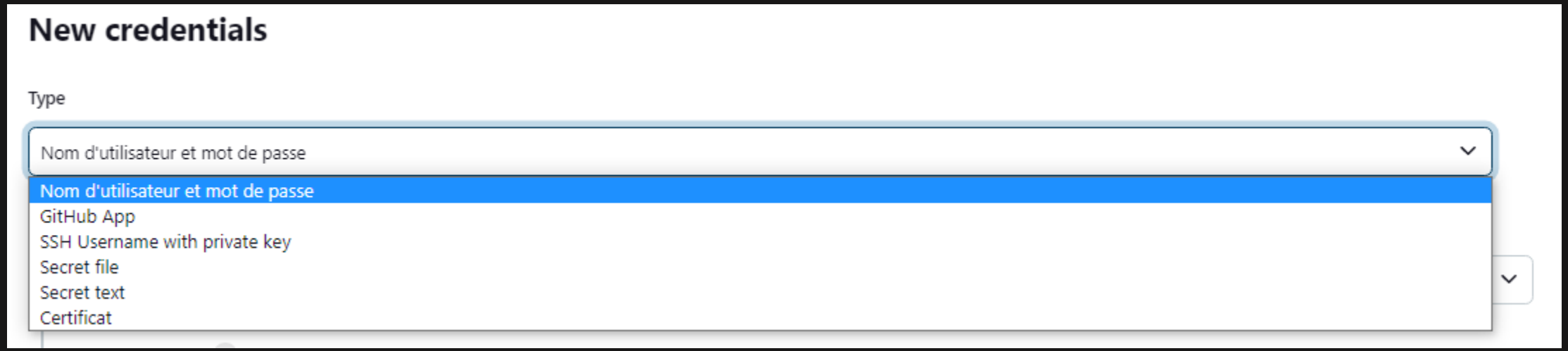


CERTIF ACADEMY

CERTIF ACADEMY



CERTIF ACADEMY

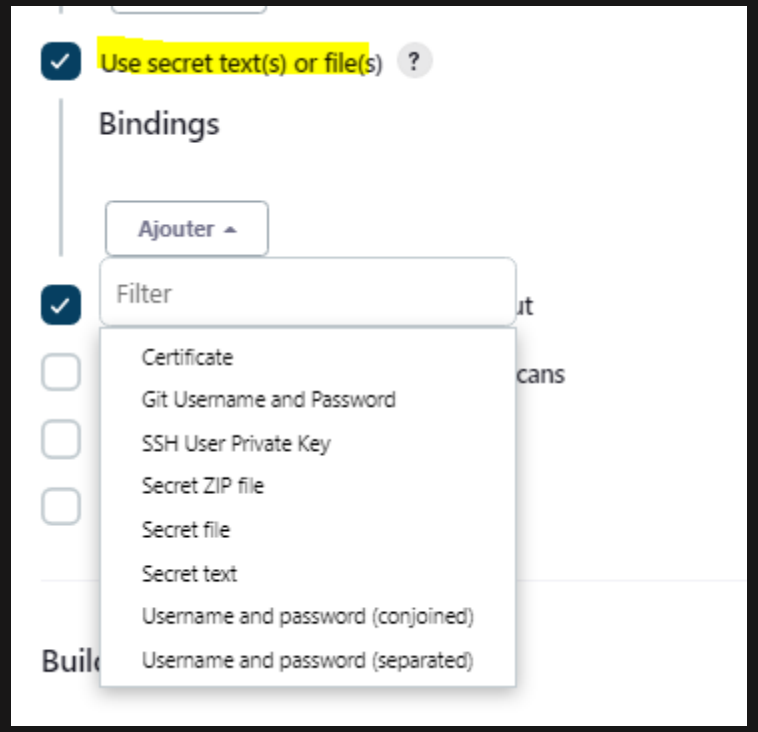


CERTIF ACADEMY

Afin de gérer des identifiants dans Jenkins (n'importe lesquels), l'idée est la chose suivante :

1. Enregistrer les identifiants à l'intérieur de Jenkins
2. Lier ces identifiants à des variables dans votre job
3. Utiliser ces variables dans votre job

CERTIF ACADEMY



CERTIF
ACADEMY