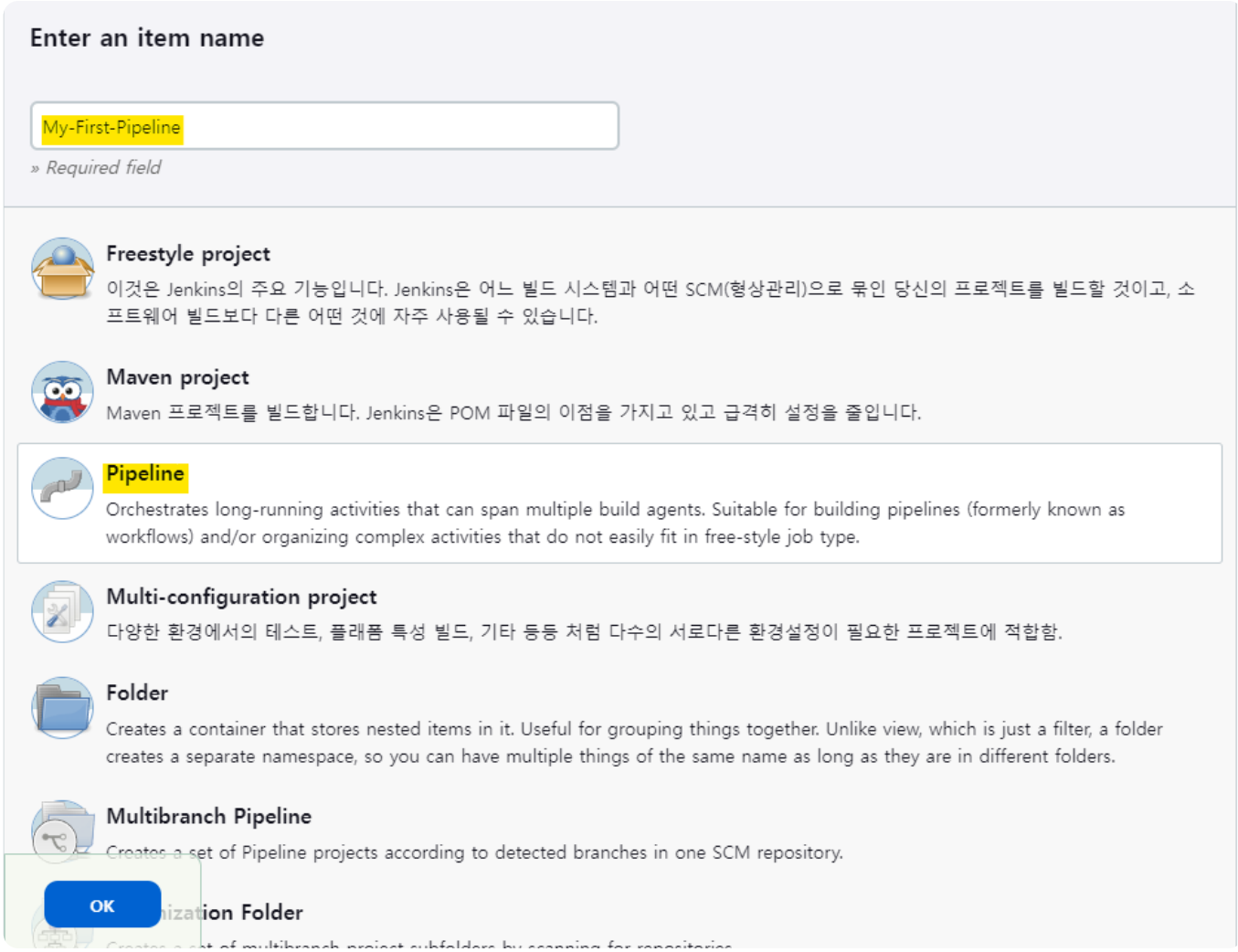
8. Pipeline 사용

1. item 생성



2. pipeline 생성 및 빌드 테스트

- 2-1)

```
pipeline {
    agent any
    stages {
        stage('Compile') {
```

```
            steps {
                echo "Compiled successfully!";
            }
        }

        stage('JUnit') {
            steps {
                echo "JUnit passed successfully!";
            }
        }

        stage('Code Analysis') {
            steps {
                echo "Code Analysis completed successfully!";
            }
        }

        stage('Deploy') {
            steps {
                echo "Deployed successfully!";
            }
        }
    }
}
```

- 빌드 테스트

- 2-2) post 단계 추가

```
pipeline {
    agent any
    stages {
        stage('Compile') {
            steps {
                echo "Compiled successfully!";
            }
        }

        stage('JUnit') {
            steps {
                echo "JUnit passed successfully!";
            }
        }

        stage('Code Analysis') {
            steps {
                echo "Code Analysis completed successfully!";
            }
        }

        stage('Deploy') {
            steps {
                echo "Deployed successfully!";
            }
        }
    }

    post {
      always {
        echo "This will always run"
      }
      success {
        echo "This will run when the run finished successfully"
      }
      failure {
        echo "This will run if failed"
      }
      unstable {
        echo "This will run when the run was marked as unstable"
      }
      changed {
        echo "This will run when the state of the pipeline has changed"
```

```
        }
    }
}
```

- 2-2) 실행 결과

Stage Logs (Declarative: Post Actions)

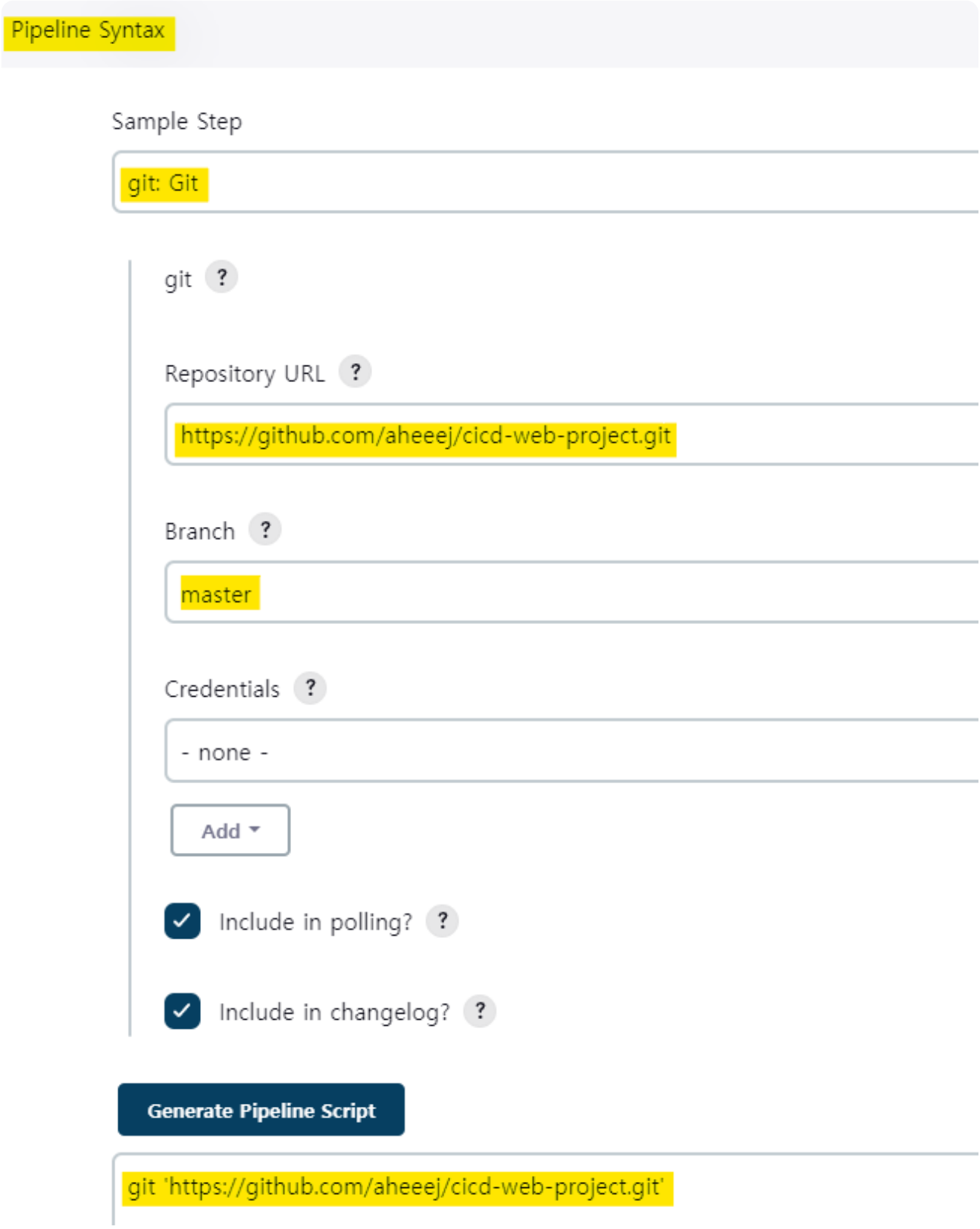
Print Message -- This will always run (self time 32ms)

Print Message -- This will run when the run finished successfully (self time 22ms)

3. maven build pipeline

- 새로운 item 생성

- pipeline syntax 사용하여 script 생성 (repo가 public 설정되어 있어서 계정 정보 X)



- pipeline 생성

```
pipeline {
    agent any
    tools {
```

```
        maven 'Maven3.9.4'
    }
    stages {
        stage('github clone') {
            steps {
                git 'https://github.com/aheeej/cicd-web-project.git'
            }
        }
        stage('build'){
            steps {
                sh '''
                echo build start
                    mvn clean compile package -DskipTests=true
                '''
            }

        }
    }
}
```

- 빌드 테스트 정상 성공

- docker에 war파일 ssh로 배포 단계 추가를 위해 pipeline syntax에서 스크립트 확인

sshPublisher ?

SSH Publishers

SSH Server
Name ?

docker-server

고급

Transfers

Transfer Set
Source files ?

target/*.war

Remove prefix ?

target

Remote directory ?

.

Exec command ?

docker build --tag aheeej/exam_01 -f Dockerfile .

All of the transfer fields (except for Exec timeout) support substitution of **Jenkins environment varia**

고급

- 파이프라인 추가