Creating a Declarative Pipeline in Jenkins: Step-by-Step Guide

Step 1: Open Jenkins

1. Launch Jenkins by navigating to http://localhost:8080 in your browser.
2. On the left-hand side menu, click **"New Item"**.

Step 2: Create a New Project

1. Provide a project name, e.g., "Pipeline Script."
2. Select **"Pipeline"** (instead of Freestyle Project for a declarative pipeline).
3. Click **"OK"** to create the project.

Step 3: Configure the Pipeline Script

1. In the newly created project, scroll down to the **Pipeline** section.
2. Select **Pipeline script** from the Definition dropdown.
3. Copy and paste the following declarative pipeline script into the Script section:

groovy

CopyEdit

```
pipeline {
    agent any

    tools {
        maven 'MAVEN-HOME' // Configure Maven in Jenkins
    }

    stages {
        stage('Git Repo & Clean') {
            steps {
                // Remove directory (Windows command)
                //bat "rmdir /s /q mavenjava"

                // Clone your GitHub repository
                bat "git clone <Your_GitHub_Repository_URL>"

                // Clean Maven project
        bat "mvn clean -f mavenjava"
      }
    }

    stage('Install') {
      steps {
        // Install Maven dependencies
        bat "mvn install -f mavenjava"
      }
    }

    stage('Test') {
      steps {
        // Run Maven tests
        bat "mvn test -f mavenjava"
      }
    }

    stage('Package') {
      steps {
        // Package the Maven project
        bat "mvn package -f mavenjava"
      }
    }
  }
}
```

- Replace <Your_GitHub_Repository_URL> with your Git repository link.
- Adjust mavenjava with the actual project directory name if needed.

Step 4: Save and Execute

1. Click **Save** and then **Apply**.
2. To run the pipeline, click **"Build Now"** in the project dashboard.

If you need further clarification or details on any step, feel free to ask!

4o