Step-by-Step Guide: Jenkins Build Pipeline on Windows (with Git)

Step 1: Prepare Jenkins on Windows

- Jenkins installed and running as a Windows service.

- Git installed and added to the system PATH.

- Jenkins user has proper permissions.

- Make sure Jenkins has internet access for pulling Git repos.

Step 2: Create Freestyle Jobs with Git Integration

Job 1:

1. Dashboard -> New Item -> Name: Job1 -> Freestyle project -> OK

2. Under 'Source Code Management' tab:

   - Select 'Git'

   - Repository URL: https://github.com/your-user/your-repo.git

   - Branch: */main or */master

3. Build step -> Execute Windows batch command:

   git --version

   echo Cloning repo in Job 1...

   timeout /t 5

4. Post-build Actions -> Build other projects -> Job2

5. Save.

Job 2:

1. Dashboard -> New Item -> Job2 -> Freestyle project -> OK

2. Under 'Source Code Management' tab:

   - Select 'Git'

- Repository URL: https://github.com/your-user/your-repo.git

- Branch: */main

3. Build step -> Execute Windows batch command:

git status

echo Running Job 2...

timeout /t 5

4. Post-build Actions -> Build other projects -> Job3

5. Save.

Job 3:

1. Dashboard -> New Item -> Job3 -> Freestyle project -> OK

2. Under 'Source Code Management' tab:

- Select 'Git'

- Repository URL: https://github.com/your-user/your-repo.git

- Branch: */main

3. Build step -> Execute Windows batch command:

git log -1

echo Running Job 3...

timeout /t 5

4. Save.

Step 3: Install Build Pipeline Plugin

1. Manage Jenkins -> Manage Plugins

2. Available tab -> Search: Build Pipeline Plugin

3. Install and restart or install without restart.

Step 4: Create Build Pipeline View

1. Dashboard -> + New View

2. Name: MyPipelineView -> Select 'Build Pipeline View' -> OK

Step 5: Configure Pipeline View

1. Set Initial Job: Job1

2. Set:

- Builds to display: 5
- Refresh frequency: 5
- Enable manual trigger: checked

3. Apply -> OK

Step 6: Run the Pipeline

1. Go to MyPipelineView

2. Click 'Run' under Job1

3. Observe Job1 -> Job2 -> Job3

4. All jobs should turn green on success.

5. Git output should show in console logs of each job.

Final Notes:

- Ensure repo is public or provide credentials for private repos.

- Test each job individually before chaining.

- Use 'timeout /t 5' to delay so build status is visible.