

Detailed Explanation of Java App Deployment & Nexus Integration

This document explains each command used during the deployment of a Java application and the

integration with Nexus Repository Manager. This covers server preparation, application build, and

artifact deployment.

1. Nexus Server Setup

Command: `sudo apt update`

Explanation: Updates the local package index to retrieve the latest list of available packages.

Command: `sudo apt install openjdk-11-jdk -y`

Explanation: Installs OpenJDK 11, which is required to run Nexus.

Command: `wget https://download.sonatype.com/nexus/3/nexus-unix-x86-64-3.79.0-09.tar.gz`

Explanation: Downloads the Nexus 3.79.0-09 tar.gz archive.

Command: `tar -xvzf nexus-unix-x86-64-3.79.0-09.tar.gz`

Explanation: Extracts the downloaded Nexus archive.

Command: `sudo mv nexus-3.79.0-09 /opt/nexus`

Explanation: Moves the extracted Nexus directory to /opt for standard software storage.

Command: `sudo adduser nexus`

Explanation: Creates a dedicated system user named 'nexus' for running the Nexus service.

Command: `sudo chown -R nexus:nexus /opt/nexus`

Explanation: Gives ownership of the Nexus files to the nexus user.

Command: `sudo chown -R nexus:nexus /opt/sonatype-work`

Explanation: Ensures the Nexus working directory is owned by the nexus user.

Command: `sudo vi /etc/systemd/system/nexus.service`

Explanation: Creates a systemd service unit file for managing Nexus as a service.**Command:** `sudo systemctl enable nexus`

Explanation: Enables the Nexus service to start on system boot.

Command: `sudo systemctl start nexus`

Explanation: Starts the Nexus service.

Command: `sudo systemctl status nexus`

Explanation: Displays the current status of the Nexus service.

Command: `http://54.160.252.15:8081/`

Explanation: Access the Nexus web interface via a browser.**2. Build Server Setup**

Command: `git clone https://github.com/daniyel7devops/docker-Java-kubernetes-project.git`

Explanation: Clones the Java project from GitHub.

Command: `cd docker-Java-kubernetes-project/`

Explanation: Navigates into the cloned project directory.

Command: `cd productcatalogue/`

Explanation: Navigates into the productcatalogue module.

Command: `mvn clean`

Explanation: Cleans the Maven target directory to prepare for a fresh build.

Command: `sudo apt install maven`

Explanation: Installs Maven, the build tool required to compile the project.

Command: `mvn`

Explanation: Checks if Maven is installed correctly by displaying help info.

Command: `vi pom.xml`

Explanation: Opens the Maven build file for editing.

Command: `<distributionManagement>...</distributionManagement>`

Explanation: Specifies where Maven should deploy built artifacts (Nexus repository URL).

Command: `vim /etc/maven/settings.xml`

Explanation: Opens Maven settings file to configure authentication details.

Command: `<server>...</server>`

Explanation: Adds Nexus credentials for repository authentication.

Command: `mvn package`

Explanation: Packages the Java project into a JAR/WAR file.

Command: `mvn clean install -DskipTests` Explanation: Builds the project without running tests and installs it into the local Maven repo.

Command: `mvn clean deploy -DskipTests`

Explanation: Builds and deploys the artifact to the remote Nexus repository.
3. Artifact Storage in Nexus

The command `mvn clean deploy` uploads the built JAR file to the Nexus repository. If the version is

a `-SNAPSHOT`, it's pushed to the `maven-snapshots` repository.

Otherwise, it should go to `maven-releases`. Ensure that the version in your `pom.xml` matches the

configured repository policy in Nexus.

Example:

`http://54.160.252.15:8081/repository/maven-snapshots/uk/co/danielbryant/djshopping/productcatalogue/0.0.1-SNAPSHOT/`

4. Conclusion

This deployment pipeline ensures proper artifact management through Nexus. Each command plays

a key role in preparing the environment,

building the Java application, and deploying artifacts for versioned storage and future CI/CD processes.

← → ↻ ⚠ Not secure 54.160.252.15:8081/#browse/browse:maven-snapshots:uk%2Fco%2Fdanielbryant%2Fdjshopping%2Fproductcatalogue%2Fmaven-metadata.xml.md5 ☆ ⓘ 🔔

Sonatype Nexus Repository
COMMUNITY 3.79.0-09

Browse / **maven-snapshots**

HTML View [Advanced search...](#)

- uk
 - co
 - danielbryant
 - djshopping
 - productcatalogue
 - 0.0.1-SNAPSHOT
 - maven-metadata.xml
 - maven-metadata.xml.md5**
 - maven-metadata.xml.sha1

[Delete asset](#)

Summary

Repository	maven-snapshots
Format	maven2
Path	uk/co/danielbryant/djshopping/productcatalogue/maven-metadata.xml.md5
Content type	application/octet-stream
File size	32 bytes
Blob created	Thu Apr 17 2025 12:59:28 GMT+0530 (India Standard Time)
Blob updated	Thu Apr 17 2025 12:59:28 GMT+0530 (India Standard Time)

← → ↻ ⚠ Not secure 54.160.252.15:8081/#browse/browse:maven-snapshots:uk%2Fco%2Fdanielbryant%2Fdjshopping%2Fproductcatalogue%2F0.0.1-SNAPSHOT ☆ ⓘ 🔔

Sonatype Nexus Repository
COMMUNITY 3.79.0-09

Browse / **maven-snapshots**

HTML View [Advanced search...](#)

- uk
 - co
 - danielbryant
 - djshopping
 - productcatalogue
 - 0.0.1-SNAPSHOT**
 - 0.0.1-20250417.072921-1
 - productcatalogue-0.0.1-20250417.072921-1.jar
 - productcatalogue-0.0.1-20250417.072921-1.jar.md5
 - productcatalogue-0.0.1-20250417.072921-1.jar.sha1
 - productcatalogue-0.0.1-20250417.072921-1.pom
 - productcatalogue-0.0.1-20250417.072921-1.pom.md5
 - productcatalogue-0.0.1-20250417.072921-1.pom.sha1
 - maven-metadata.xml
 - maven-metadata.xml.md5
 - maven-metadata.xml.sha1
 - maven-metadata.xml
 - maven-metadata.xml.md5
 - maven-metadata.xml.sha1

[Delete folder](#)

uk/co/danielbryant/djshopping/productcatalogue/0.0.1-SNAPSHOT