Nexus Repository Manager

Agenda

01 What is an Artifact?

02 What is Nexus Repository?

03 Use Cases of Nexus Repository

04 Difference Between JFrog and Nexus

05 Nexus Repository Installation

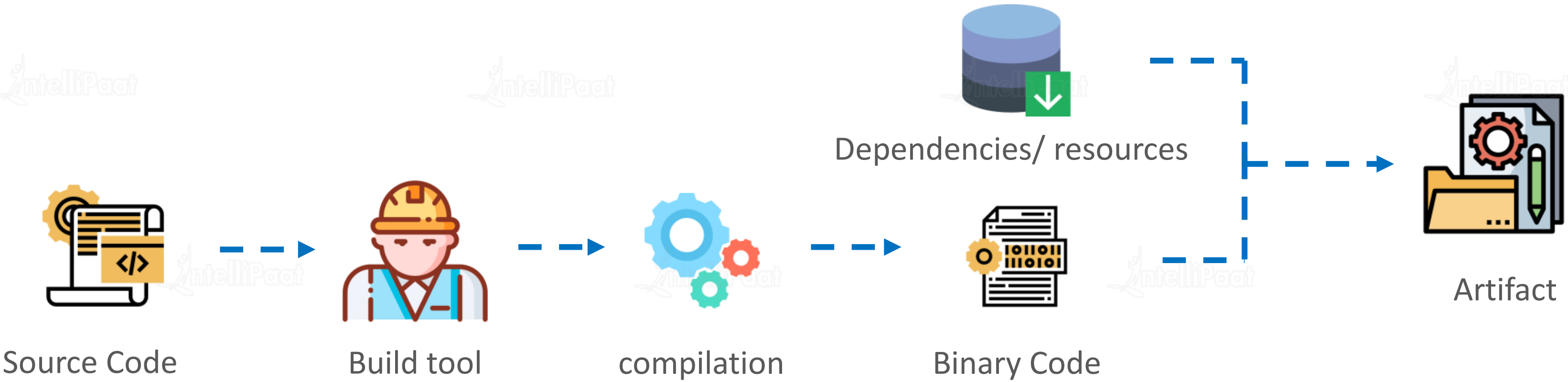
06 Nexus Repository Hands-on

What is an Artifact?

What is an artifact?

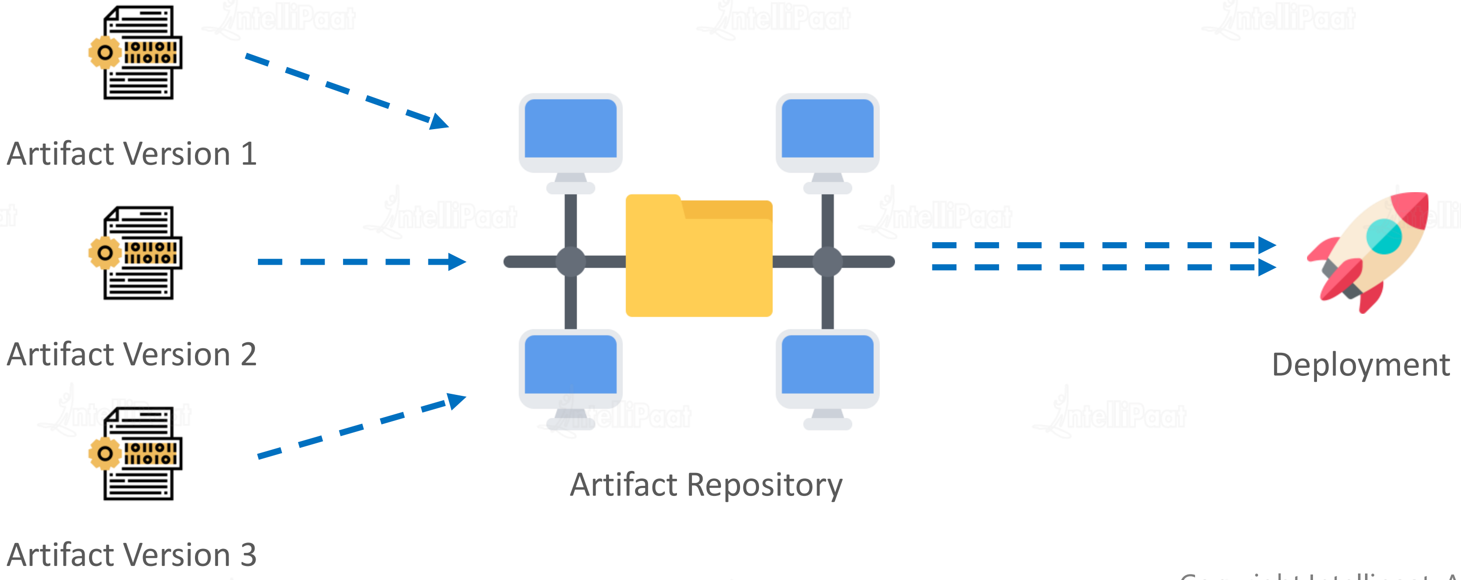
The files that contain both the compiled code and the resources that are used to compile them are known as artifacts. They are readily deployable files.

In java an artifact would be a .jar, .war, .ear file
In NPM the artifact file would be a .tar.gz file
In .NET the artifact file would be a .dll file



What is an Artifact Repository?

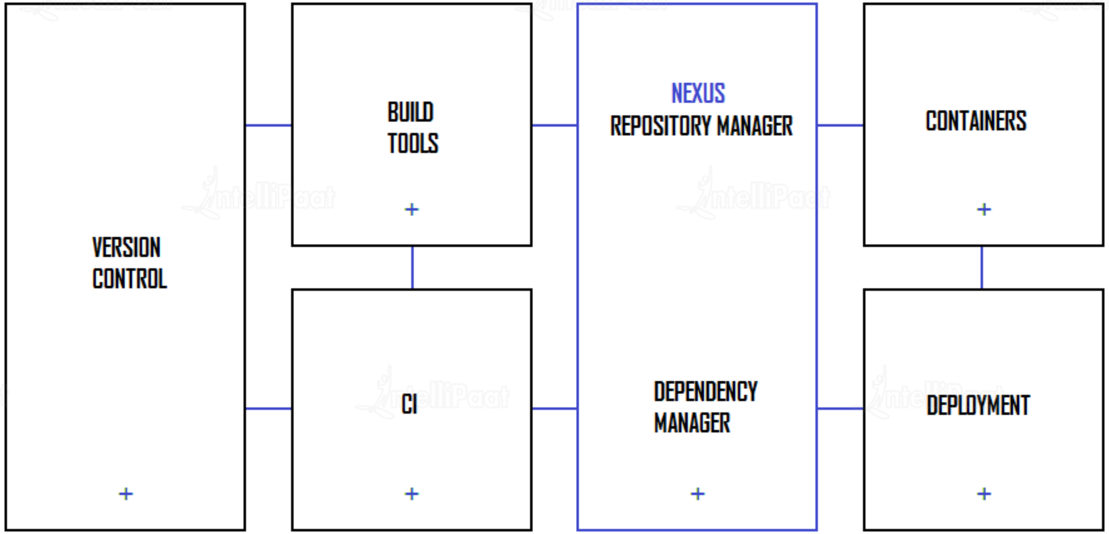
An artifact repository is a repository that can store multiple versions of artifacts. Each time a .war file or .tar.gz file is created, it is stored in a server dedicated for the artifacts



What is Nexus Repository?

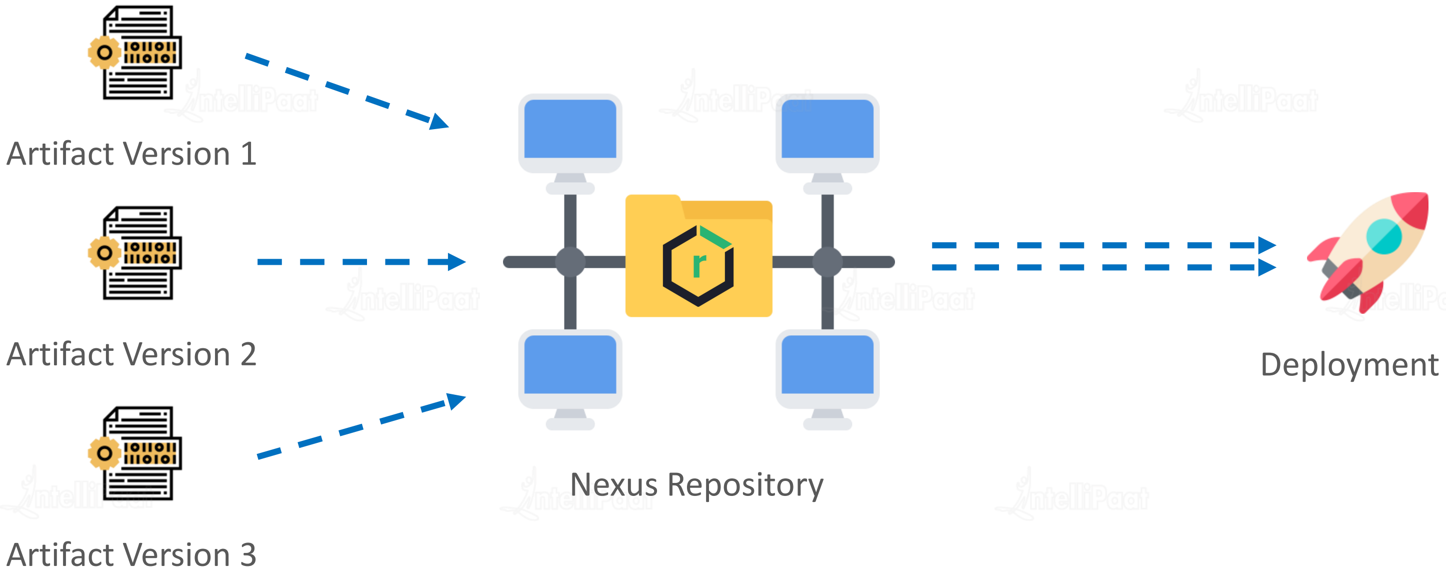
What is Nexus Repository?

Nexus Repository is a tool used in DevOps methodology for multiple purposes. One of its main purposes is to store artifacts (readily deployable code) that have been created in the code pipeline. Another purpose is to act as a buffer for downloading dependencies for the build tools and languages



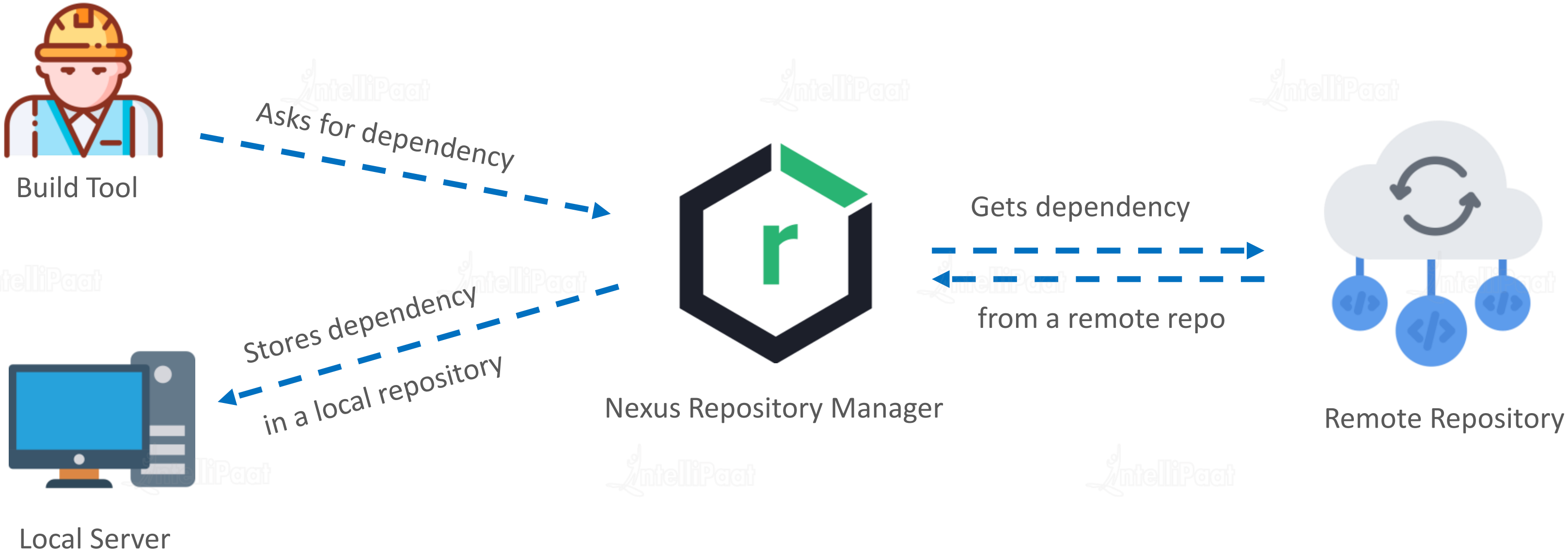
Nexus Repository: Responsibilities

1. Helps in storing readily deployable code



Nexus Repository: Responsibilities

2. Helps in downloading and managing dependencies



Features of Nexus Repository

Features of Nexus Repository

- Atlassian Crowd Support
- Staging and Build Promotion
- Tagging
- User Token Support

Features of Nexus Repository

Atlassian Crowd Support

Nexus provides support for Atlassian Crowd, which is a security software used to manage and track user accounts and application accesses

It comes in-built with Nexus Repository Management Pro and can be easily configured according to an organization's needs

Crowd

Features of Nexus Repository

Staging and Build Promotion

Nexus provides a staging feature that can easily be integrated with an organization’s software development life cycle

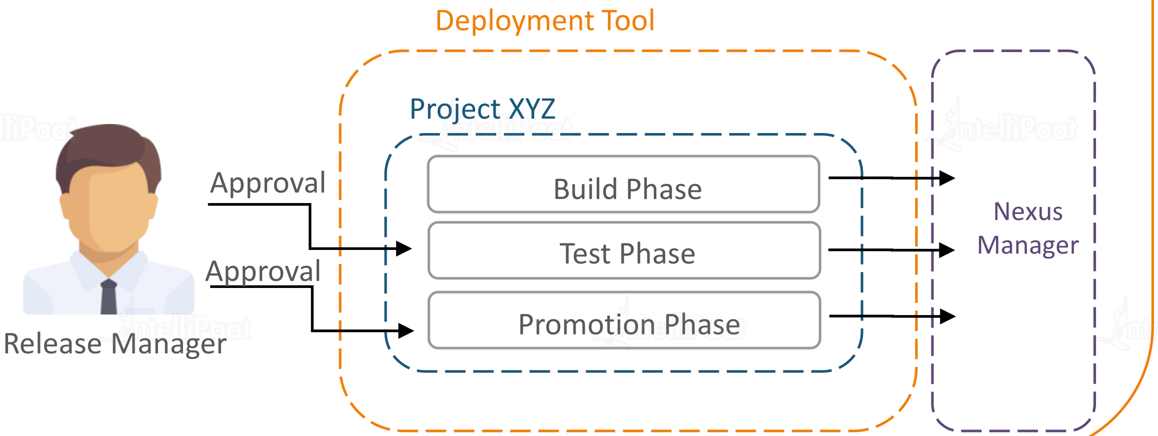
It helps in staging all the release candidates (components of the software) that are tested meticulously before being isolated for release onto the production stage

Since the feature allows for multiple candidates to be staged at the same time, based on the requirements, the candidates can be discarded or promoted

Features of Nexus Repository

Tagging

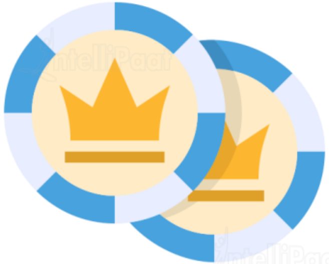
Nexus provides the ability to tag the components so that they can be associated with each other. The usage of this feature is entirely up to the user as it has no specific use case, but it's mostly used in life cycles to keep track of different builds or to differentiate between multiple projects during merges



Features of Nexus Repository

User Token Support

Nexus uses a two-part token system to overcome the limited security of the build tools and removes the need to store sensitive information in text



Features of Nexus Repository

High Availability

Repository Health Check (RHC)

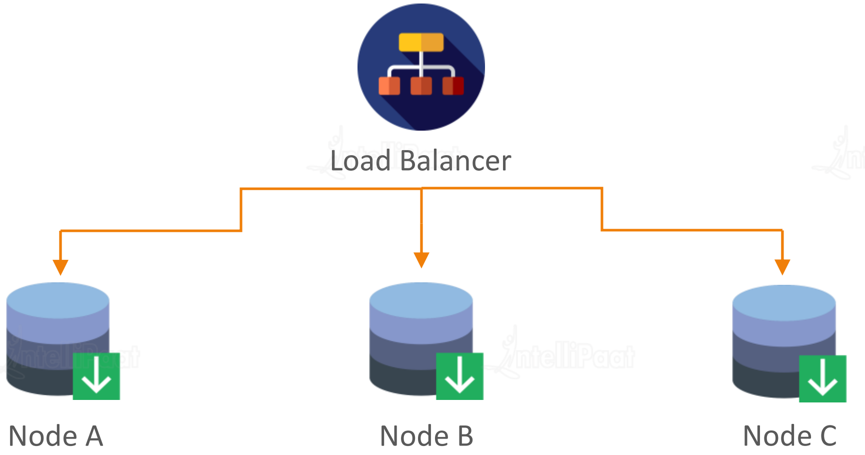
Enterprise Support

Group Blob Stores

Features of Nexus Repository

High Availability

Nexus has a feature called High Availability Clustering (HAC) that keeps multiple redundant nodes inside the data center to increase the availability of the artifacts and dependencies in case any of the nodes crashes or becomes unavailable due to some reason



Features of Nexus Repository

Repository Health Check (RHC)

Nexus can recognize security risks in the code at an early stage. The feature that it uses for that particular purpose does mainly three things:

- Provides guidance on what has to be upgraded and what has to be reworked or replaced
- Lists down the components of the software based on their susceptibility and also lists the number of times the components are downloaded
- Gives monthly summarized reports on the health of the code pipeline revealing any dangerous trends or vulnerabilities

Features of Nexus Repository

Enterprise Support

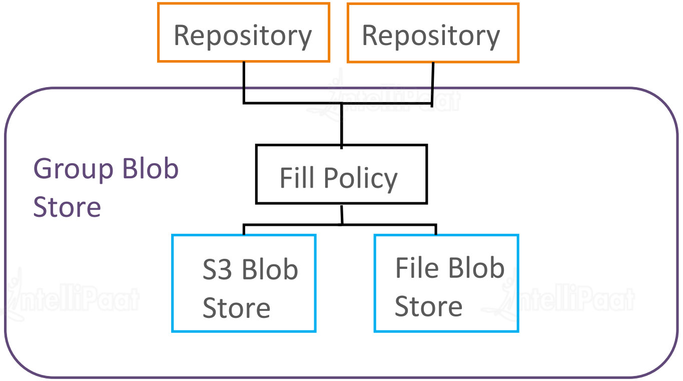
Nexus provides full-time support with a skilled team only focused on resolving issues

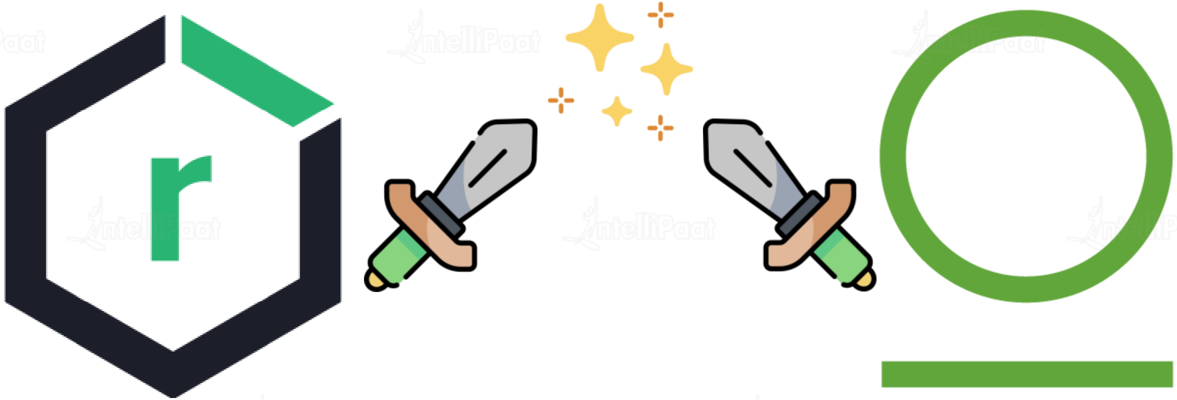
Features of Nexus Repository

Group Blob Stores

Nexus provides group blob stores: a combination of multiple blob stores acting as one

Fill Policies are used to select the blob store that is used, allowing users to choose and use their storage

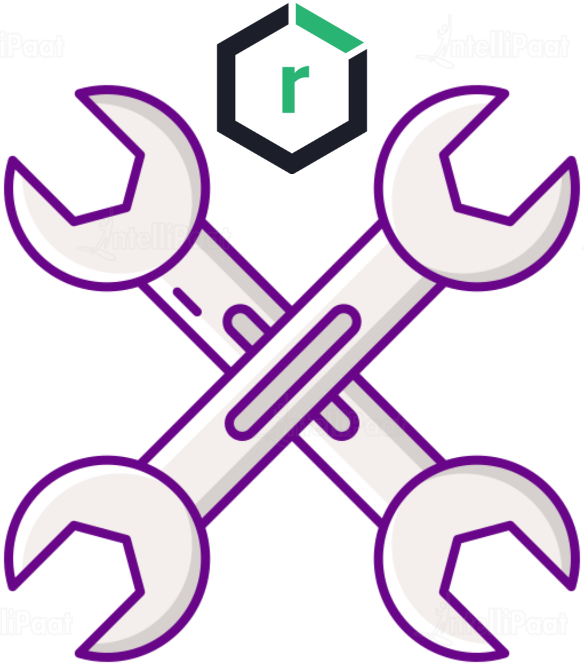




Nexus vs Artifactory

Nexus vs Artifactory

Nexus	Artifactory
Uses a file system for storage	Uses Derby DB for storage by default
Cannot generate and save settings.xml	Can generate and save settings.xml
Limited in replicating repositories	Easily replicates repositories to other artifactory instances
Limited build time	Faster build time

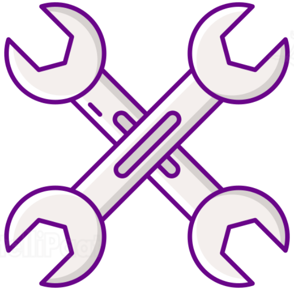


Nexus Repository Installation

Nexus Repository Installation: Prerequisite

Prerequisites:

1. Have OpenJDK
2. Have 2 GB RAM and 15 GB of hard drive space



Nexus Repository Installation

Installing OpenJDK

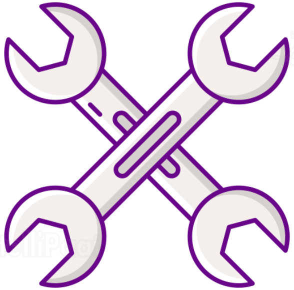
```
$ cd /opt
$ yum install java-1.8.0-openjdk-devel
```

Nexus Repository Installation

To download **Nexus Repository**:

- Go to https://www.sonatype.com/nexus-repository-oss/
- Click on **GET REPOSITORY OSS**
- Click on the UNIX version and as soon as it starts downloading, cancel it
- Then, go to Download history and copy the link address of the file

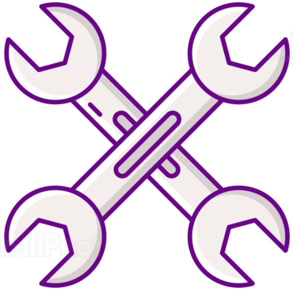
```
$ sudo wget https://sonatype-download.global.ssl.fastly.net/repository/repositoryManager/3/nexus-3.19.0-01-unix.tar.gz
```



Nexus Repository Installation

- To start Nexus Repository, go inside the untarred folder and type the below script:

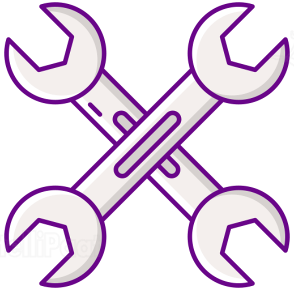
```
$ tar zxf nexus-3.19.0-01-unix.tar.gz
$cd nexus-3.19.0-01/bin/
$ ./nexus
$ ./nexus start
$ ./nexus status
```

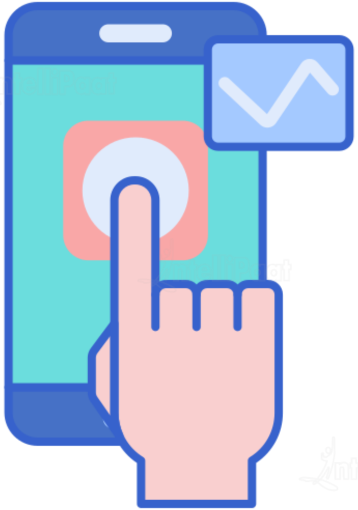


Nexus Repository Installation

- If it is running, head to the browser and open Nexus at the specified port number:

<Instance IP address>:8081





Nexus Repository Hands-on

Nexus Repository Hands-on

1. Run Nexus Repository and set-up the admin

2. Install Git to get the sample code

3. Install Maven to compile and package the sample code

4. Connect Maven to Nexus

5. Check how the artifact versioning works in Nexus

Nexus Repository Hands-on

1. Run Nexus Repository and set-up the admin

This will give you the administrator access

Nexus Repository Hands-on

2. Install Git to get the sample code

2.1 Install Git so that you can clone the sample code from GitHub and then compile it

Nexus Repository Hands-on

3. Install Maven to compile and package the sample code

3.1 Using Maven, compile the downloaded file

Nexus Repository Hands-on

4. Connect Maven to Nexus

4.1 Go to the home directory and create a folder **.m2** and also create **settings.xml**

4.2 Configure the **settings.xml** file

4.3 Run the compile command, and you will see all the downloaded dependencies

Nexus Repository Hands-on

5. Check how the artifact versioning works in Nexus

5.1 Run the deploy command and wait for the deployment

5.2 Edit the **pom.xml** file with a new version, simulating version change, and see the change reflecting after a new build on Nexus

5.3 Edit the **pom.xml** file again and use **-snapshot** to see its change reflecting on Nexus

IntelliPaat

India: +91-7847955955

US: 1-800-216-8930 (TOLL FREE)

sales@intellipaat.com

24/7 Chat with Our Course Advisor