Metapercept

JFrog artifactory user guide

Table of Contents

[Chapter 1-Introduction 0](#_Toc175957743)

[What is Artifactory? 0](#_Toc175957744)

[What is JFrog Artifactory? 0](#_Toc175957745)

[Key Feature 1](#_Toc175957746)

[Different products 2](#_Toc175957747)

[Chapter 2-Setting up JFrog Account 0](#_Toc175957748)

[Create JFrog Trail account 0](#_Toc175957749)

[Set up JFrog environment 1](#_Toc175957750)

[User interface 2](#_Toc175957751)

[Work Flow 4](#_Toc175957752)

[Chapter 3-Build 5](#_Toc175957753)

[What is builds? 5](#_Toc175957754)

[Build and metadata 5](#_Toc175957755)

[Managing and monitoring build 6](#_Toc175957756)

[Build Dashboard 7](#_Toc175957757)

[Chapter4 -Artifacts 8](#_Toc175957758)

[What is artifacts? 8](#_Toc175957759)

[Repository Types 8](#_Toc175957760)

[Artifacts management 0](#_Toc175957761)

[Browsing artifactory 1](#_Toc175957762)

[Deploy artifacts 5](#_Toc175957763)

[View artifacts information 6](#_Toc175957764)

# Chapter 1-Introduction

JFrog Artifactory is a leading software product in the IT industry for DevOps operation; it helps in maintaining high-quality, rapid release cycles and streamlines the software release process. This user guide will take you through various tool and the JFrog platform.The JFrog

## What is Artifactory?

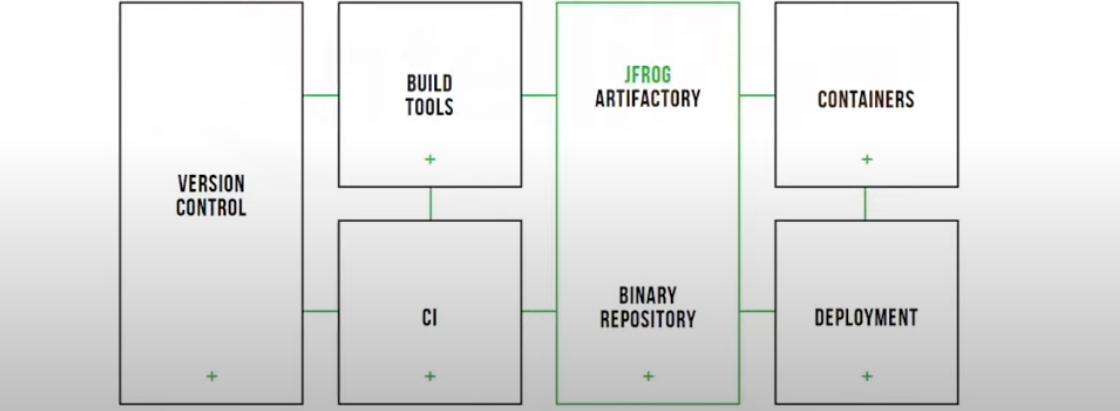
Artifactory is a global binary repository manager where you can store all of your dependencies and artifacts.

## What is JFrog Artifactory?

JFrog Artifactory Offers Divers Products For DevOps DevSecOps Operations.

**JFrog Artifactory**

JFrog is used in DevOps methodology to store the artifacts and sort then according to the need.JFrog Artifactory acts as Google Drive for Artifact. We can store all the artifacts and pull whichever is required to perform the task.



Jfrog can be integrated with the code pipeline. The version control will pass the main source code to build tools.Then build tool will create a build. process it and create artifacts’ all this artifactory will be stored in Jfrog articatory.then Jfrog will send whatever request is pulled for a deployment.

**JFrog Artifactory.Responsibilities**

The Jfrog Artifactroy responsibility is to store all the artifacts which can then go ahead for deployment the second responsibility of Jfrog is to help manage and download dependencies.

Here jfrog acts as a proxy for downloading and managing the required dependencies. So if the build tool requests for dependency jfrog will get the dependancy from a remote repository, we will store it in a local server used by the build tool.to build the source code.

## JFrog Artifactroy Installation

Pre-requisite:

* OpenJDK
* 2 GB RAM

## Key Feature

* + It is a universal binary repository manager.
  + Integration with CI/CD tools such as GitLab, Jenkins
  + Massively scalable
  + Security access control and traceability.

# Chapter 2-Setting up JFrog Account

This chapter helps users create and set up a JFrog account.

## Create JFrog Trail account

This section helps users create JFrog trail Account.

**Step 1**: Go to the web browser and search JFrog Artifactory.

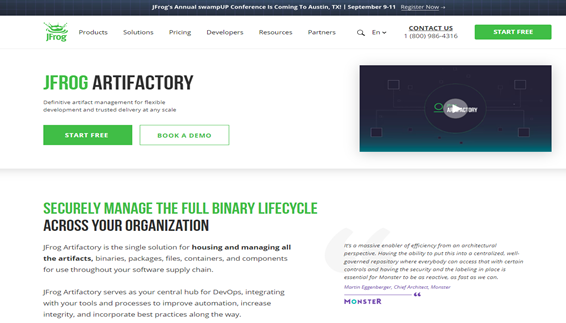
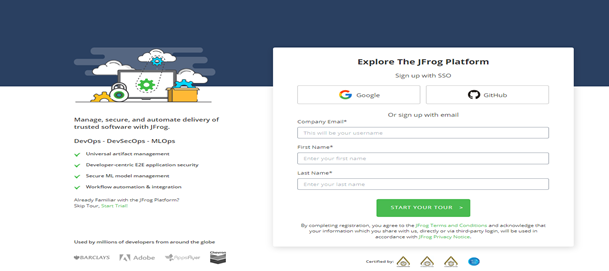


Figure 0‑1: JFrog Artifactory web application

**Step 2**: Click **Start Free** on the top right corner.

**Step output**: You will be re-directed to the sign-up window.



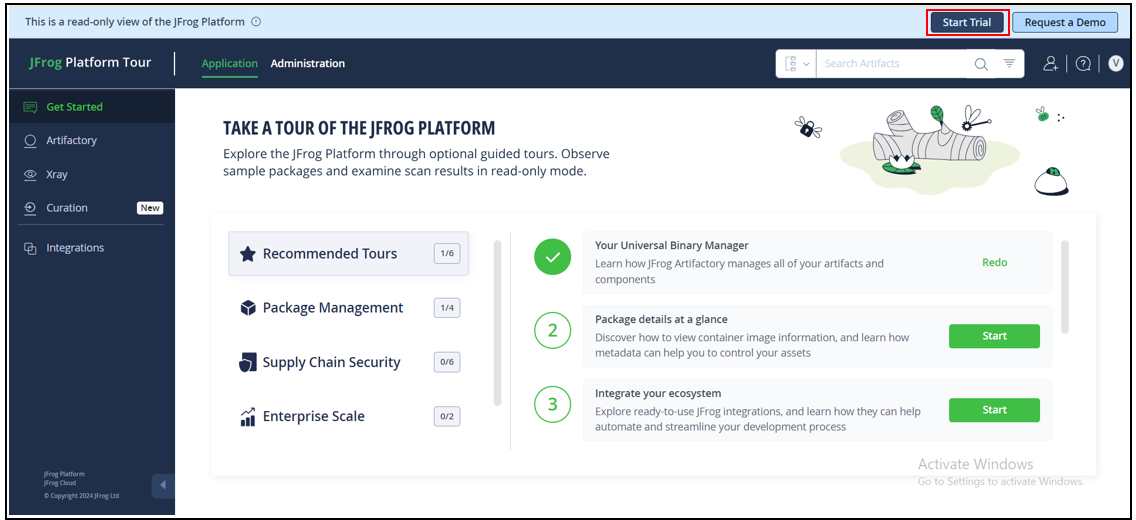
**Step 3**: Enter your name and email address. Else, you can sign up on Google or GitHub.

**Step 4**: Click **start your guide**

## Set up JFrog environment

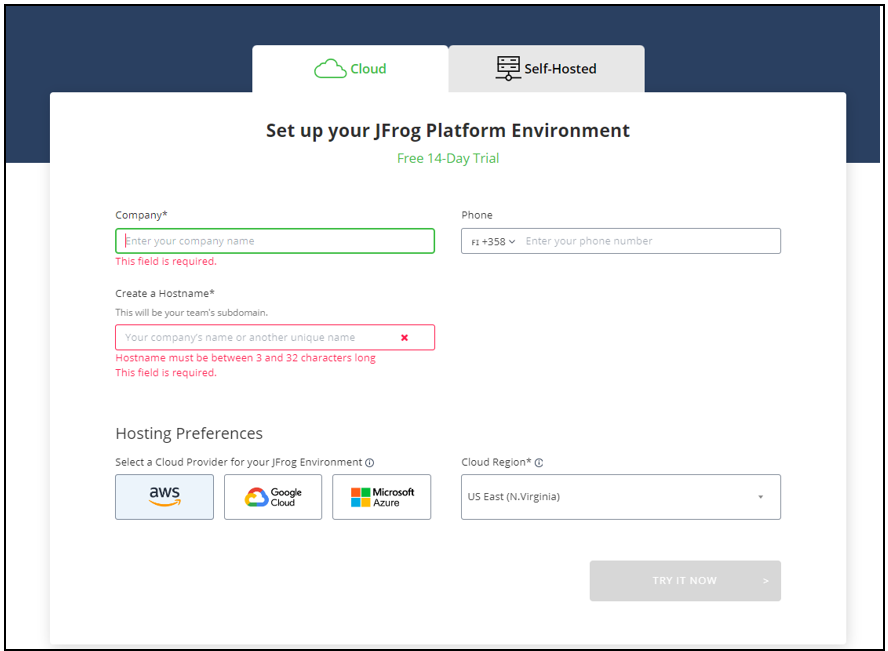
This section helps users set up JFrog environment.

**Step 1**: Click Start Trail on the top right corner.



The JFrog platform Self-hosted and cloud environment is displayed.

**Step 2**: Select a cloud or self-hosted environment.



**Step 3**: Enter Company, phone number, and hostname.

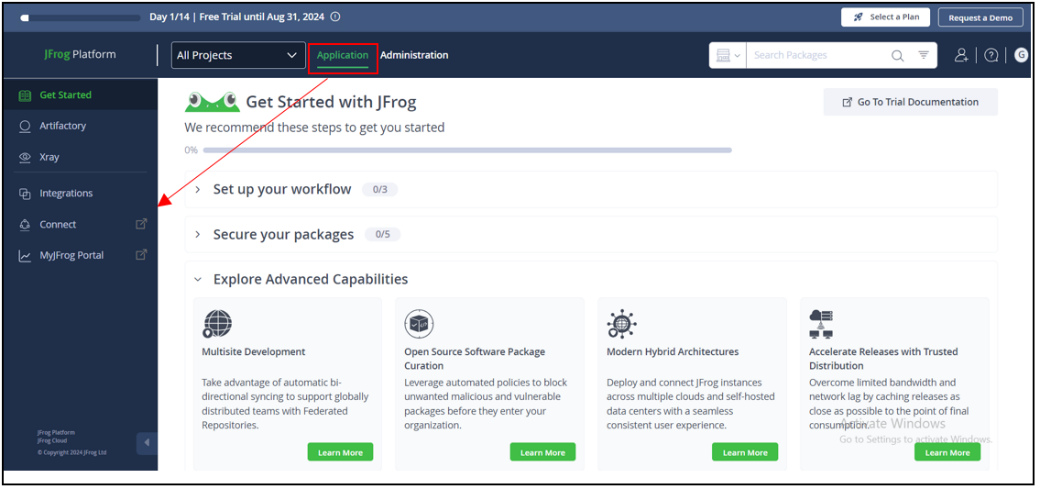
**Step 4**: Select cloud provider. (AWS, Google Cloud, Azure)

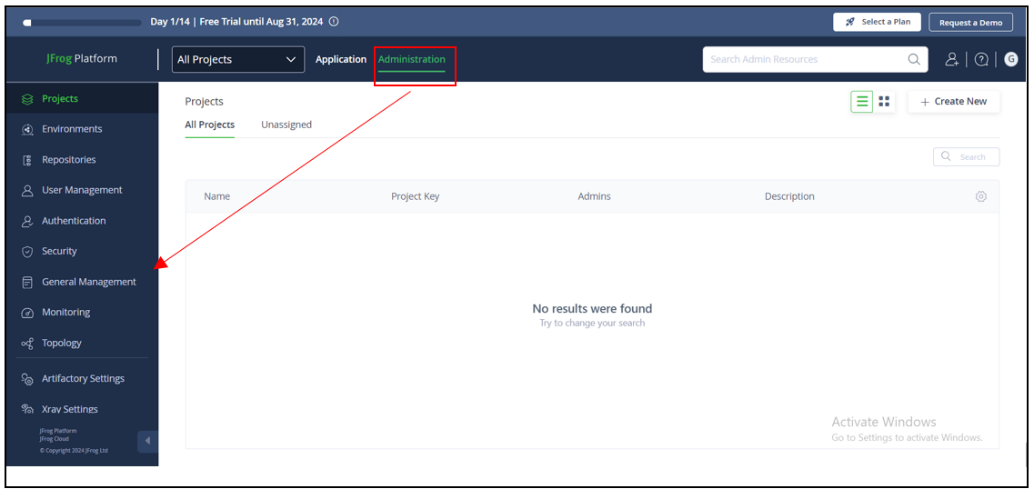
**Step 5**: Select cloud region.

**Step 6**: Click TRY IT NOW.

## User interface

This section provides a detailed overview of the JFrog platform user interface.





**Note:** Click the toggle selector on the top bar, and the updated side bar allows the user to quickly switch between application and administration functionality.

|  |  |
| --- | --- |
| **Field** | **Description** |
| Application (Sidebar ) | This application functionality allows users to   * Create and manage repository. * Universal package manager for your builds. * Manage all artifacts in one place. * Integrate with CI/CD tools such as Jenkins Git lab and Git hub. * All of the artifacts kept in JFrog Artifactory are scanned by JFrog X-ray. |
| Administration (Sidebar) | This administrative functionality allows users to   * Create new projects and assign responsibility. * Create environments for repositories * Set restrictions to different environments. * Monitor tab tracks progress of the environment. |
| Search panel | This panel enables users to perform quick searches for builds, packages, and artifacts. |
| Invite teammates | This panel allows users to collaborate with teammates. |
| Get help & learn | This panel allows users to get technical support and access to documentation guides of the JFrog platform. |
| All projects | This panel allows users to browse different projects and create new projects. |

## Work Flow

A screenshot of a computer diagram

Description automatically generated

# Chapter 3-Build

This chapter helps users understand Build, and various tools that help in managing and monitoring Build.

## What is Build?

## Build is a cycle of converting source code into standalone executable programs that run on a computer. Build is a key step in the Continuous Integration/Continuous Deployment pipeline, enabling automation of the software delivery.

## Metadata

It is information generated during software build, this includes details about artifacts deployed, dependencies, and information about the build environment.

## Managing and monitoring build

In the build panel, users can manage and monitor build created during continuous integration/ continuous delivery process.

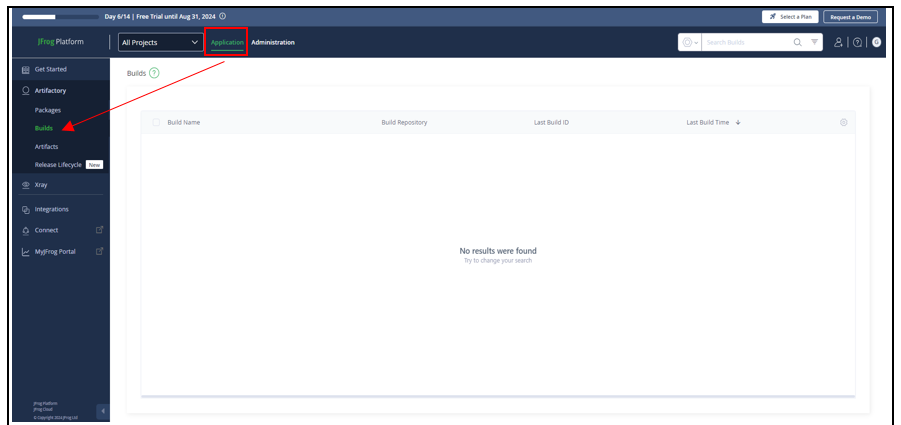
Here is step by step process on how to navigate build tab.

**Step1**: Log in to the JFrog Artifactory.

**Step2**: Click the Artifactory tab on the sidebar of the application functionality.

**Step outcome**: A drop-down list appears.

**Step3**: Click the Build form drop-down list.



## Build Dashboard

|  |  |
| --- | --- |
| Field | Description |
| Build column | * This column displays a specific build information, such as the build name, build ID from the build repository, and the last build time. * The user can customize the column. |
| Search field | The search field helps the user quickly find a specific build with a specific range date. |
| Get help and learn | This tab on the top left-hand side allows the user to access the JFrog documentation for the build section. |

# Chapter4 -Artifacts

This chapter helps users understand artifacts and various tools that help in managing and monitoring these.

## What is Artifact?

The file that contains both the binary code and also the resources used to compile that code is known as an artifact. For example in java .jar files are the artifacts used to compile and run code.

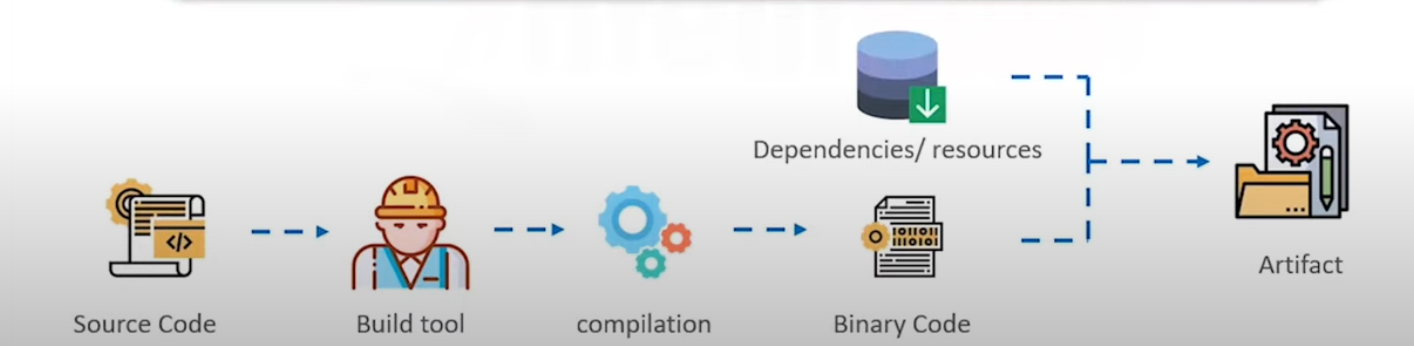


Figure 2Artifacts

## Artifact Repository

The repository that stores multiple versions of artifacts in it is known as an artifact repository. Each time a new artifact is created it is stored into the server allocated to the repository.

A computer network diagram with arrows and a folder

Description automatically generated

Figure 3 Artifact Repository

As shown in the above figure it is very easy to make changes if we are following the Agile module for the code deployment as it is easy to go back and deploy the data from the Artifact version 2 and then 3.

## Repository Types

Artifacts are stored in repositories. There are various kinds of repositories with distinct functions.

**Local repository:**

Artifacts are stored internally. And can be shared with teams that are located in remote locations.

**Remote repository:**

Artifacts are stored in a remote location it serves as a cache and proxy artifacts form external repositories.

**Virtual repository:**

It aggregates local, remotes and other repositories and lets you access them through a single URL.

## Artifacts management

In the artifacts panel, users can manage and monitor artifacts created during continuous integration/ continuous delivery process.

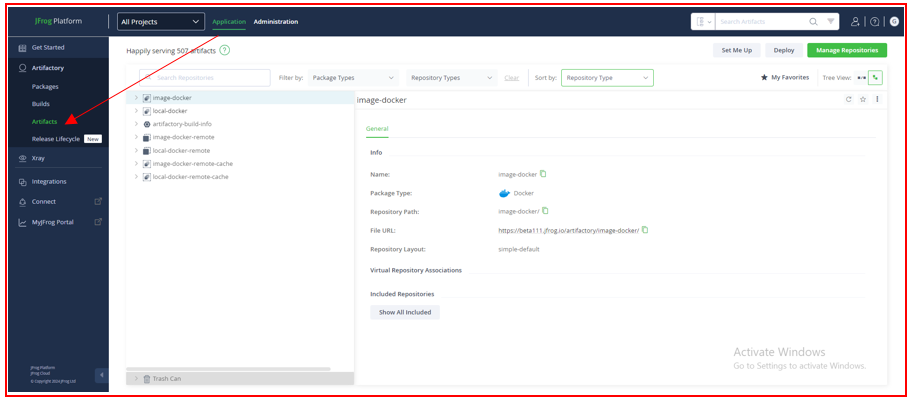
Here is step by step process on how to navigate artifacts tab.

**Step1**: Log in to the JFrog Artifactory.

**Step2**: Click the Artifactory tab on the sidebar of the application functionality.

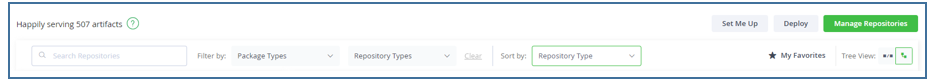
**Step outcome**: A drop-down list appears.

**Step3**: Click the Artifacts form drop-down lists.



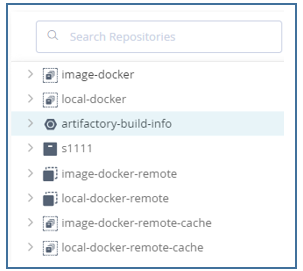
## Browsing artifactory

This tab helps users navigate the repository in JFrog Artifactory. This section takes you through how to browse artifacts and apply filters for quick searches.



**Simple browsing**

This search field allows the user to search a specific repository by name.



**Apply filter**

This section helps users apply filters to quickly navigate artifacts with specific criteria.

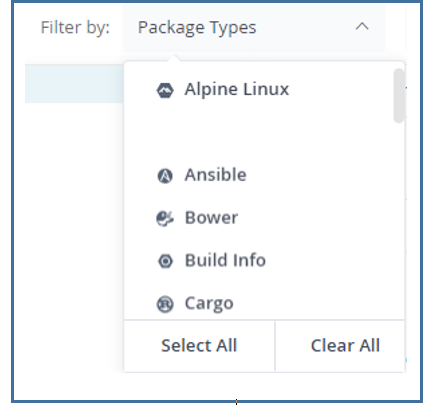
**Package type**

This tab allows users to filter artifacts-based on package types. The user can select one or multiple package types.

**Step1**: Click package type tab.

**Step outcome**: a drop-down list appears.

**Step2**: Select package type.



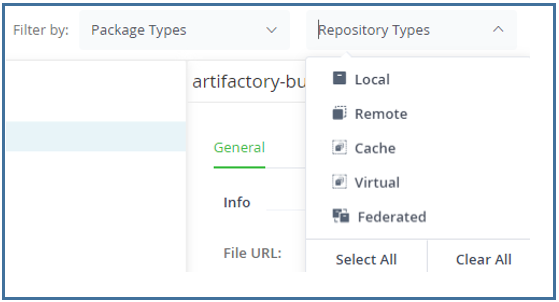
**Repository type**

This tab allows users to filter artifacts based on the repository type; user can select single or multiple repository types.

**Step1**: Click repository type tab.

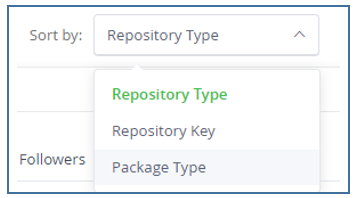
**Step outcome**: A drop-down list appears.

**Step2**: Select Repository Type.



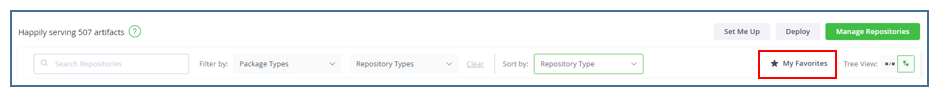
**Sort repository**

This tab allows users to sort repository by their type.



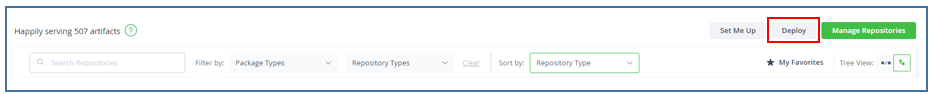
**Favorites repository**

This tab helps users to browse a repository that is marked as favorites.



## Deploy artifacts

This tab allows the users to deploy artifacts into a local repository. Artifacts can be deployed individual or in multiple.



**Step1**: Log in to the JFrog Artifactory.

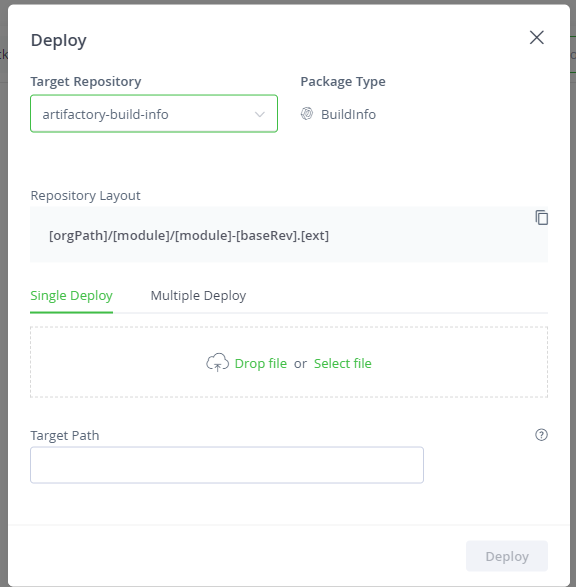
**Step2**: Click the Artifactory tab on the sidebar of the application functionality.

**Step outcome**: A drop-down list appears.

**Step3**: Click the Artifacts form drop-down list.

**Step4**: Click deploy on the right hand of the browsing tab.

**Step5**: Deploy window is appeared.

  
**Step6**: Select the target repository for storing artifacts.

**Step7**: Drop a file else you can select file form device.

**Step8**: Specify the target path for repositories.

**Step9**: Click Deploy.

## View artifacts information

This tab allows the user to monitor and manage information related to Artifacts.

|  |  |
| --- | --- |
| **Field** | **Description** |
| General tab | This tab shows comprehensive information about the artifacts, including the build name, when the builds were created, and the updated package type. |
| Effective permission tab | * This tab helps the user understand and manage the permission applied to specific artifacts or sets of artifacts. * It displays a list of permissions for a user or group. |
| Properties tab | * This tab allows users to manage metadata associated with artifacts. This metadata is organized into key-value pairs known as "properties." * The user can add a new property to an artifacts from this property tab. Enter property name and property value. |
| Follower tab | * This tab allows the user to monitor artifacts folders or repository * User can add or remove follower from this tab.   C:\Users\satyam\Documents\My Projects\JFrog user guide 2\Content\Resources\Images\image30.png |