

**Experiment 4 : To understand Continuous Integration, install and configure Jenkins with Maven/Ant/Gradle to set up a build job.**

**Aim** : To understand the concept of **Continuous Integration (CI)** and implement it by installing and configuring **Jenkins** with **Maven, Ant, or Gradle** to automate the build process. This study aims to explore how Jenkins helps in setting up a CI pipeline, executing automated builds, and improving software development efficiency.

**Theory** :

**Theory of Continuous Integration Using Jenkins with Maven, Ant, or Gradle**

**Introduction to Continuous Integration (CI)**

Continuous Integration (CI) is a **software development practice** where developers frequently integrate their code changes into a shared repository. Each integration is verified using **automated builds and tests**, ensuring that issues are detected early. CI helps streamline the development process, reduces manual errors, and improves software quality.

**Key principles of CI:**

1. **Frequent Code Integration** – Developers merge changes multiple times a day.
2. **Automated Build Process** – Code is compiled, built, and tested automatically.
3. **Immediate Feedback** – Issues are detected early and fixed promptly.
4. **Consistent Environment** – CI ensures that software builds are reproducible across different environments.

To implement Continuous Integration, organizations use **CI tools like Jenkins**, which automates the build, test, and deployment process.

---

## **Jenkins: A CI/CD Automation Tool**

**Jenkins** is an open-source **automation server** that enables developers to **automate software builds, tests, and deployments**. It supports integration with version control systems (Git, SVN) and build tools like **Maven, Ant, and Gradle**.

### **Key Features of Jenkins**

- **Automated Builds:** Supports scheduled or triggered builds based on repository changes.

- **Build Pipelines:** Allows chaining multiple jobs for end-to-end automation.
  - **Plugin Support:** Offers 1,500+ plugins for integration with tools like Docker, Kubernetes, and Slack.
  - **Scalability:** Can distribute builds across multiple nodes for faster execution.
- 

## Build Tools: Maven, Ant, and Gradle

Build tools are essential in CI to **compile source code, resolve dependencies, and generate deployable artifacts.**

### 1. Apache Maven

- A widely used **Java-based build automation tool.**
- Uses **POM.xml (Project Object Model)** to define project dependencies, build lifecycle, and plugins.
- Supports phases like **clean, compile, test, package, install, and deploy.**
- Command to build a project:
  - mvn clean install

### 2. Apache Ant

- **Older than Maven**, but still used for Java builds.

- Uses an **XML-based build script (build.xml)** to define tasks.
- More flexible but requires explicit configurations.
- Command to execute a build:
- `ant build`

### 3. Gradle

- **Newer build tool**, used for **Java, Kotlin, and Android development**.
- Uses a **Groovy or Kotlin-based build script** instead of XML.
- Faster than Maven due to its **incremental build mechanism**.
- Command to build a project:
- `gradle build`

---

### Jenkins Integration with Maven, Ant, and Gradle

Jenkins can be configured to **automate builds** using these tools. The integration process involves:

1. **Installing Jenkins** and setting up build tools.
2. **Creating a job in Jenkins** that fetches source code from Git.

3. **Configuring build steps** to invoke Maven, Ant, or Gradle commands.
  4. **Executing automated builds** and monitoring results.
- 

### **Advantages of Using Jenkins for CI**

- **Faster Development Cycle** – Automated builds and testing reduce manual effort.
- **Early Bug Detection** – Continuous integration ensures quick issue identification.
- **Improved Collaboration** – Developers work on the latest stable codebase.
- **Efficient Deployment** – Jenkins supports integration with **Docker, Kubernetes, and cloud platforms**.

# Implementation :

The screenshot shows the Jenkins website's 'Download and deploy' page. The page is divided into two main sections: 'Stable (LTS)' and 'Weekly releases'. The 'Stable (LTS)' section provides information about Long-Term Support (LTS) release baselines, which are chosen every 12 weeks from the stream of regular releases. It mentions that every 4 weeks, stable releases are released, including bug and security fix backports. The 'Weekly releases' section states that this release line delivers bug fixes and new features rapidly to users and plugin developers who need them, generally delivered on a weekly cadence. Below these sections, there are buttons for 'Changelog', 'Upgrade Guide', and 'Past Releases'. The 'Downloading Jenkins' section provides instructions on how to download Jenkins, including reviewing the 'Hardware and Software requirements' section of the User Handbook, selecting a package, and verifying the download. It also provides links to 'Learn more about verifying Jenkins downloads'. The 'Download Jenkins 2.492.1 LTS for:' section shows a 'Generic Java package (.war)' with a SHA-256 hash. The 'Download Jenkins 2.497 for:' section also shows a 'Generic Java package (.war)' with a SHA-256 hash. Below this, there is a 'Thank you for downloading Windows installer' page, which provides instructions on how to change boot configuration, start/stop the service, and inherit existing Jenkins installation. The 'Thank you for downloading Windows installer' page includes a link to 'Click this link' if the download hasn't started. It also provides instructions on how to change boot configuration, start/stop the service, and inherit existing Jenkins installation. The 'See Also' section lists links to 'Running Jenkins behind Internet Information Server (IIS)', 'Running Jenkins behind nginx', and 'Running Jenkins behind Apache'. The footer of the page includes links to 'Resources', 'Project', 'Community', and 'Other'.

**Download and deploy**

The Jenkins project produces two release lines: Stable (LTS) and weekly. Depending on your organization's needs, one may be preferred over the other. See the links below for more information and recommendations about the release lines.

**Stable (LTS)**

Long-Term Support (LTS) release baselines are chosen every 12 weeks from the stream of regular releases. Every 4 weeks we release stable releases which include bug and security fix backports. [Learn more...](#)

[Changelog](#) [Upgrade Guide](#) [Past Releases](#)

**Weekly releases**

This release line delivers bug fixes and new features rapidly to users and plugin developers who need them. It is generally delivered on a weekly cadence. [Learn more...](#)

[Changelog](#) [Past Releases](#)

**Downloading Jenkins**

Jenkins is distributed as WAR files, native packages, installers, and Docker images. Follow these installation steps:

- Before downloading, please take a moment to review the [Hardware and Software requirements](#) section of the User Handbook.
- Select one of the packages below and follow the download instructions.
- Once a Jenkins package has been downloaded, proceed to the [Installing Jenkins](#) section of the User Handbook.
- You may also want to verify the package you downloaded. [Learn more about verifying Jenkins downloads.](#)

**Download Jenkins 2.492.1 LTS for:**

Generic Java package (.war)

SHA-256: c053479a190273a4054ac30310b1b0895497b327a08055509b3a63a999e3

**Download Jenkins 2.497 for:**

Generic Java package (.war)

SHA-256: 9b0a673977186c7b10910560a1a7460649516dc3446851605c93641679668

**Thank you for downloading Windows installer**

Download hasn't started? [Click this link](#)

**Changing boot configuration**

By default, your Jenkins runs at <https://localhost:8080/>. This can be changed by editing `jenkins.xml`, which is located in your installation directory. This file is also the place to change other boot configuration parameters, such as JVM options, HTTPS setup, etc.

**Starting/stopping the service**

Jenkins is installed as a Windows service, and it is configured to start automatically upon boot. To start/stop them manually, use the service manager from the control panel, or the `sc` command line tool.

**Inheriting your existing Jenkins installation**

If you'd like your new installation to take over your existing Jenkins data, copy your old data directory into the new `JENKINS_HOME` directory.

**See Also**

- [Running Jenkins behind Internet Information Server \(IIS\)](#)
- [Running Jenkins behind nginx](#)
- [Running Jenkins behind Apache](#)

[Improve this page](#) [Report page issue](#)

**Resources**  
Downloads

**Project**  
Structure and governance

**Community**  
Forum

**Other**  
Code of Conduct

The image shows two screenshots from a Windows 11 desktop. The top screenshot is the Jenkins download page (jenkins.io/download/). It displays download links for Jenkins 2.492.1 LTS and 2.497. The page lists various operating systems and package types, including Generic Java package (.war), Docker, Kubernetes, Ubuntu/Debian, Red Hat Enterprise Linux and derivatives, Fedora, Windows, openSUSE, Arch Linux, FreeBSD, Gentoo, macOS, OpenBSD, and OpenIndiana Hipster. The bottom screenshot shows the 'Thank you for downloading Windows installer' page, which provides instructions on how to install Jenkins. A 'Jenkins 2.497 Setup' window is overlaid on the page, showing a 'Welcome to the Jenkins 2.497 Setup Wizard' dialog box with a 'Next' button.

**Jenkins** cd - Blog Success Stories Contributor Spotlight Documentation Plugins Community Subprojects Security About Download Search

Download Jenkins 2.492.1 LTS for:

- Generic Java package (.war)
- Docker
- Kubernetes
- Ubuntu/Debian
- Red Hat Enterprise Linux and derivatives
- Fedora
- Windows
- openSUSE
- FreeBSD
- Gentoo
- macOS
- OpenBSD
- OpenIndiana Hipster

Download Jenkins 2.497 for:

- Generic Java package (.war)
- Docker
- Ubuntu/Debian
- Red Hat Enterprise Linux and derivatives
- Fedora
- Windows
- openSUSE
- Arch Linux
- FreeBSD
- Gentoo
- macOS
- OpenBSD
- OpenIndiana Hipster

Packages marked third party may not be updated as frequently as packages supported by the Jenkins project directly.

**Thank you for downloading Windows installer**

Download hasn't started? [Click this link](#)

**Changing boot configuration**

By default, your Jenkins runs at `https://localhost:8080/`. This can be changed to other ports, such as `https://localhost:8081/`, or to other protocols, such as `https://localhost:8080/https`. This file is also the place to change other boot configuration parameters.

**Starting/stopping the service**

Jenkins is installed as a Windows service, and it is configured to start automatically. You can start or stop the service from the control panel, or the `sc` command line tool.

**Inheriting your existing Jenkins installation**

If you'd like your new installation to take over your existing Jenkins data, copy the `jenkins` folder from the old installation to the new one.

**See Also**

- Running Jenkins behind Internet Information Server (IIS)
- Running Jenkins behind nginx
- Running Jenkins behind Apache

**Jenkins 2.497 Setup**

Welcome to the Jenkins 2.497 Setup Wizard

The Setup Wizard will install Jenkins 2.497 on your computer. Click Next to continue or Cancel to exit the Setup Wizard.

Back Next Cancel

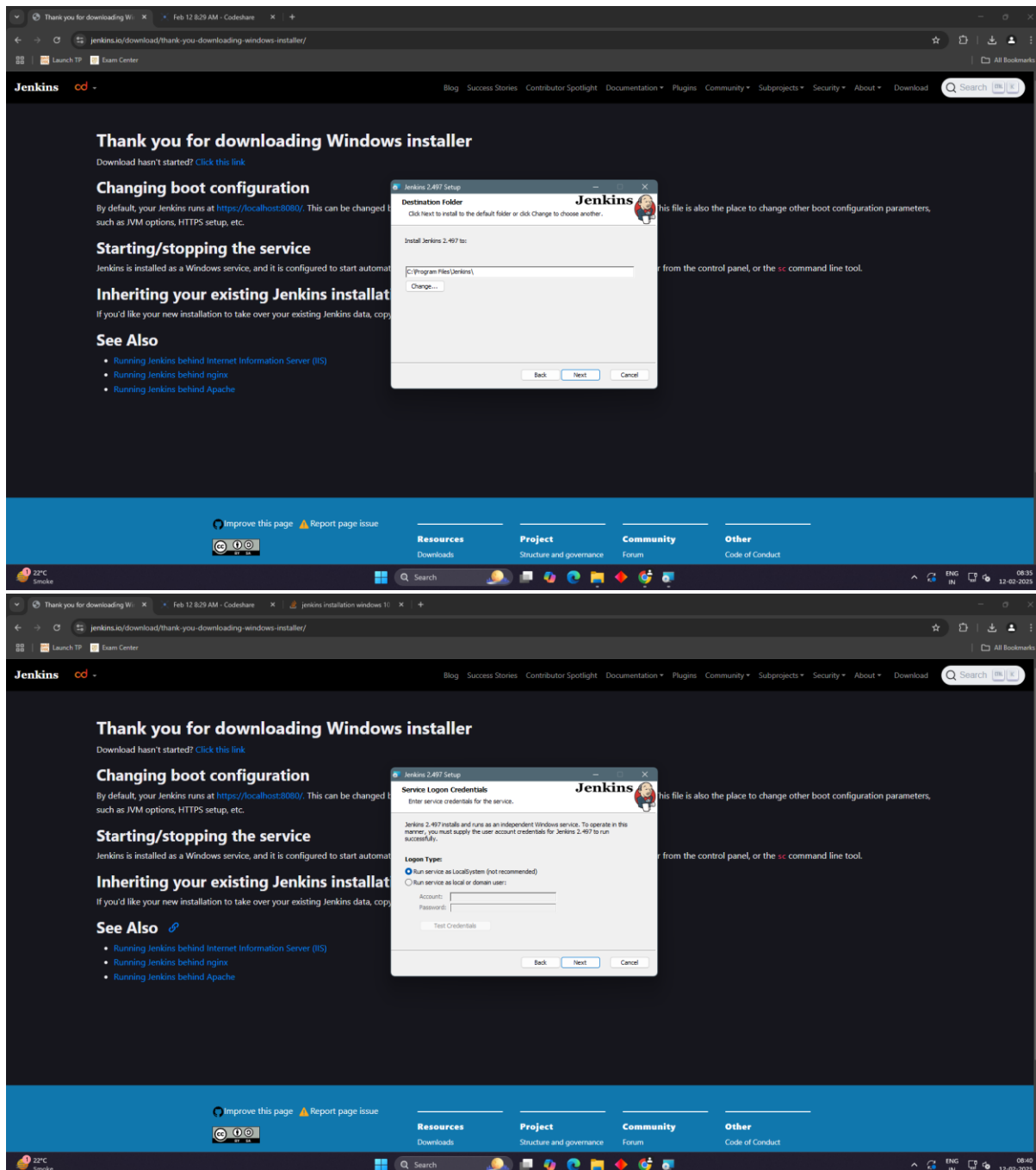
Improve this page Report page issue

**Resources**  
Downloads

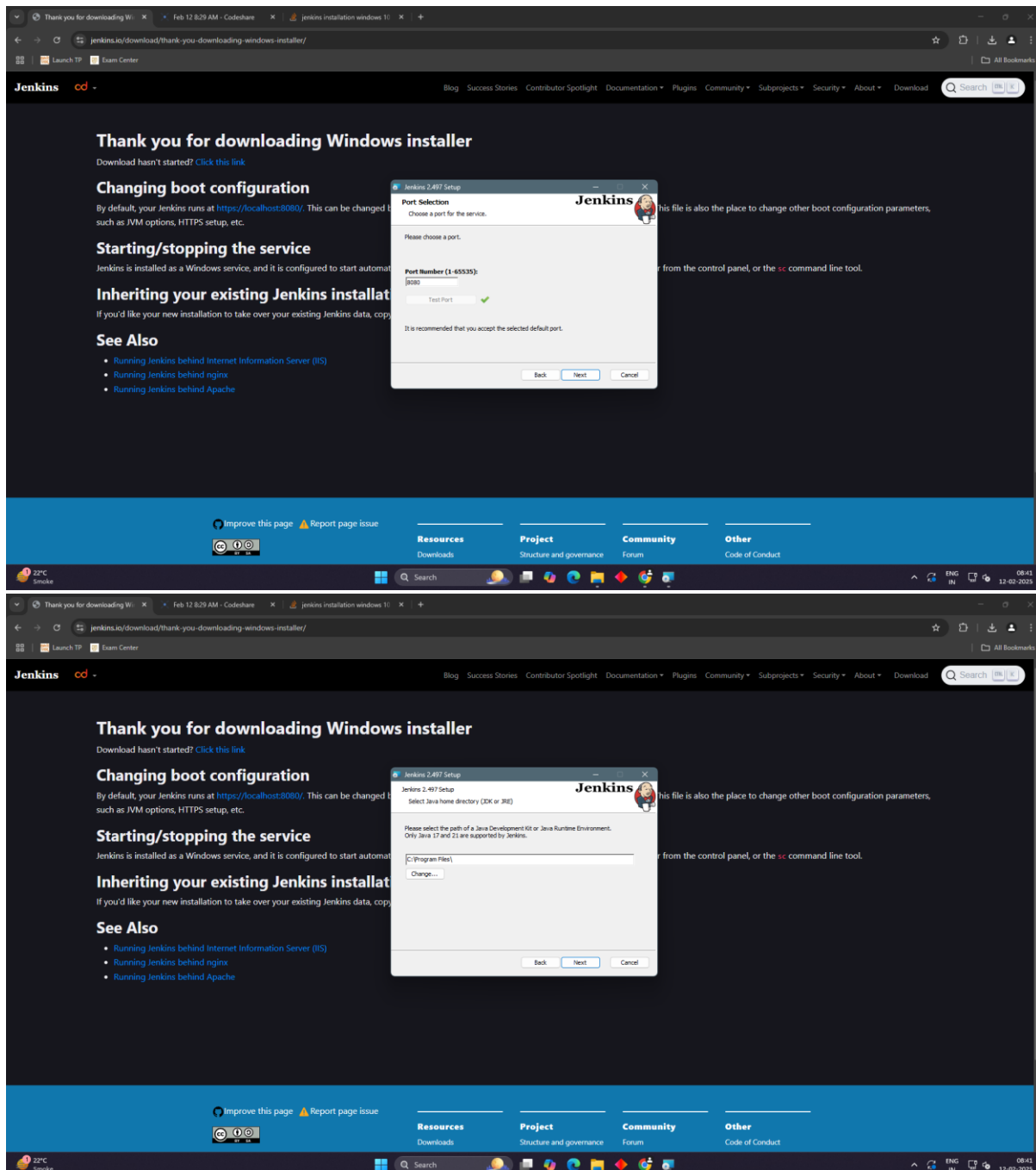
**Project**  
Structure and governance

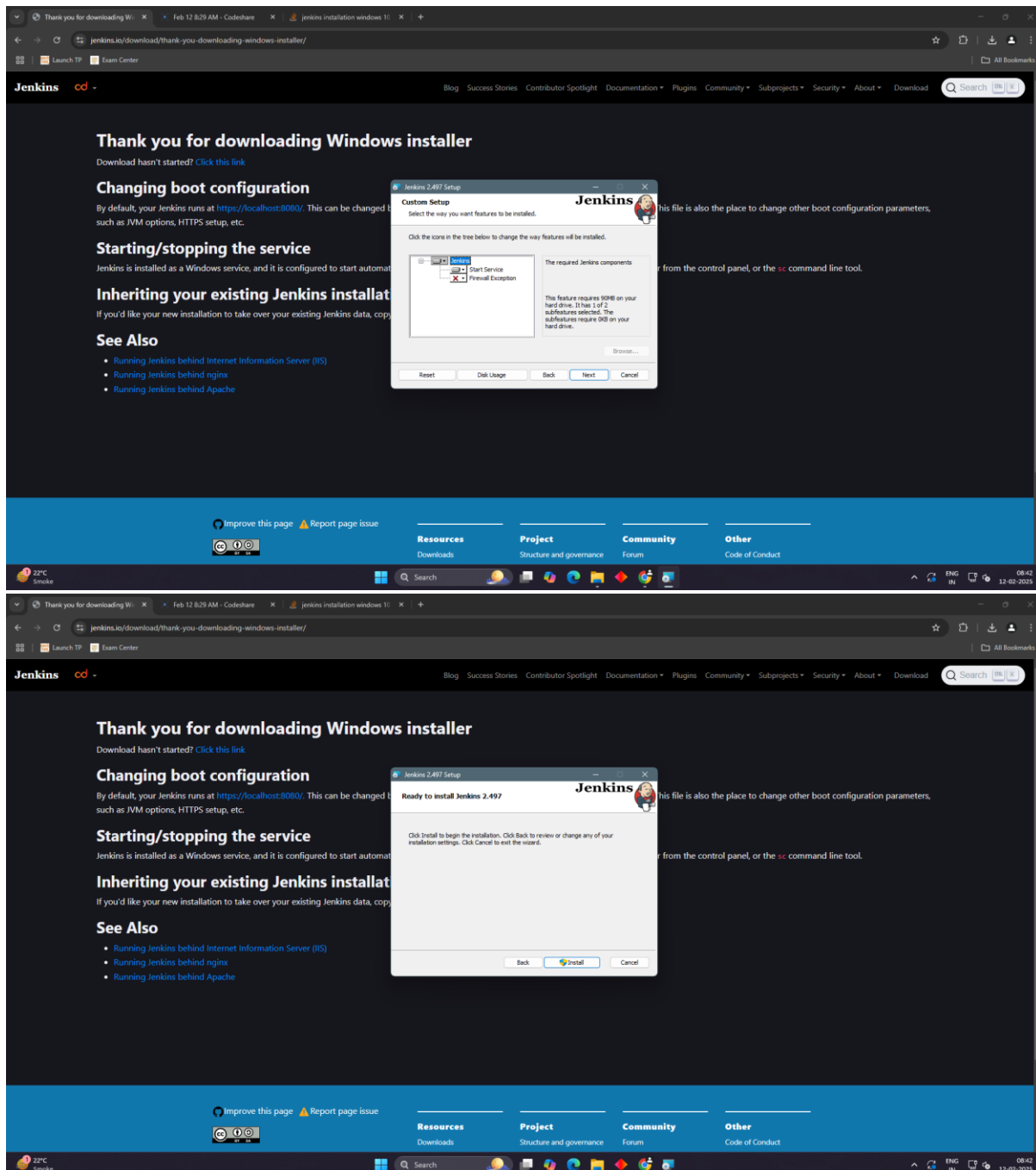
**Community**  
Forum

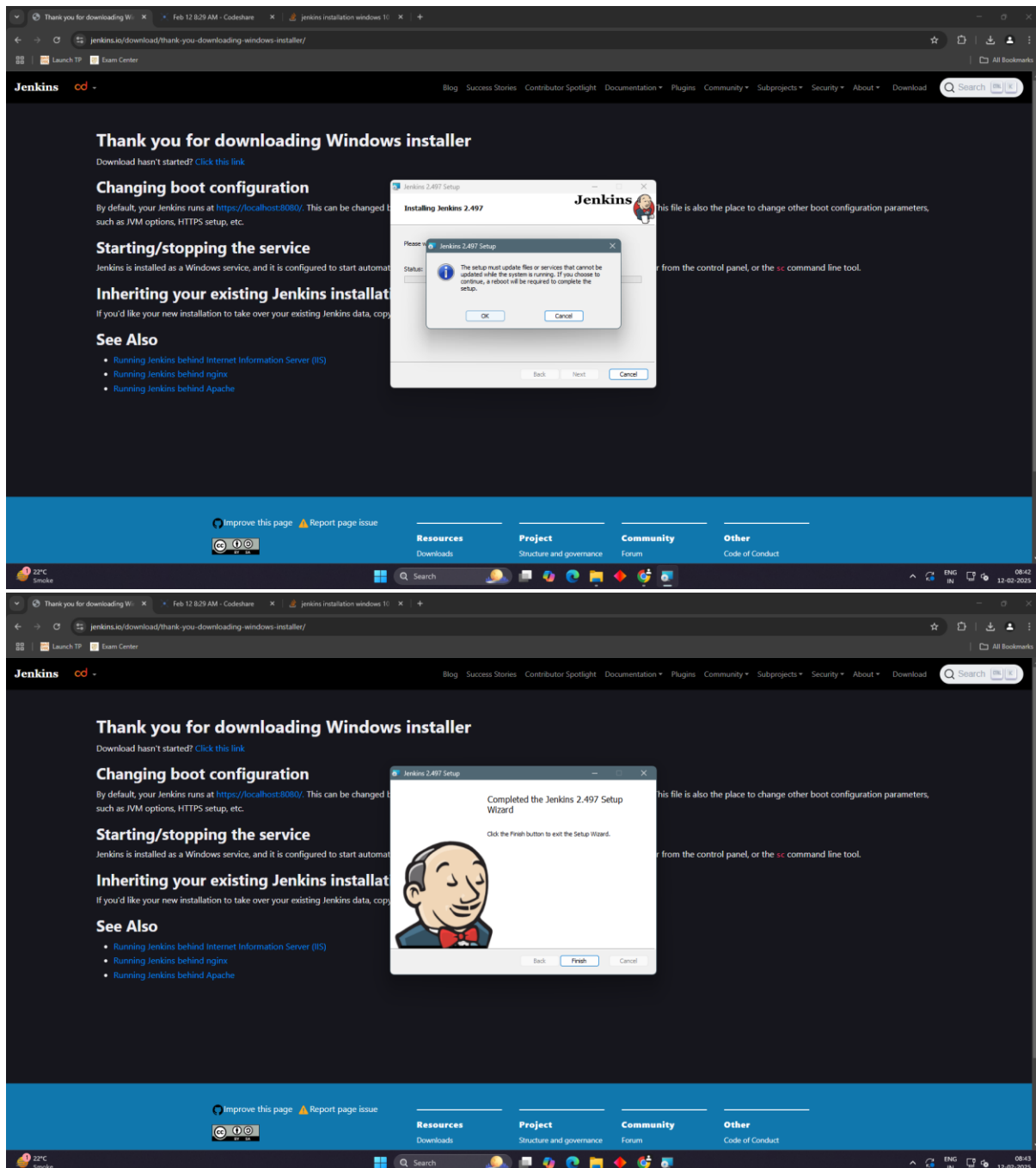
**Other**  
Code of Conduct

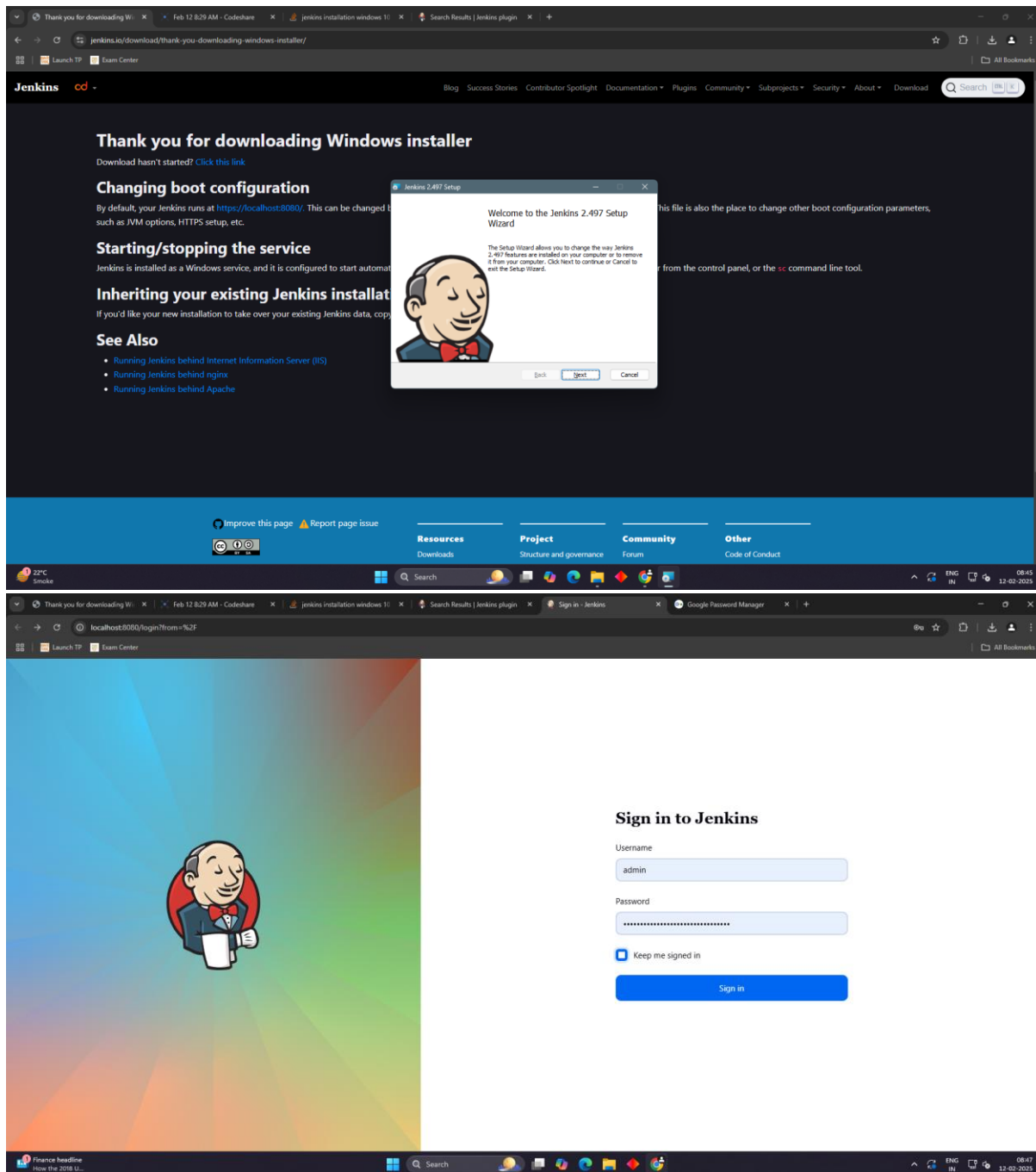












The image shows two screenshots of the Jenkins installation and login process. The top screenshot displays the Jenkins website's 'Thank you for downloading Windows installer' page. It includes links for downloading the installer, changing boot configuration, starting/stopping the service, and inheriting existing Jenkins installations. A 'Jenkins 2.497 Setup Wizard' window is overlaid, showing a welcome message and a 'Next' button. The bottom screenshot shows the Jenkins login page at 'localhost:8080/login?from=%2F'. It features a large Jenkins logo on the left and a login form on the right with fields for 'Username' (admin) and 'Password', a 'Keep me signed in' checkbox, and a 'Sign in' button. The Windows taskbar at the bottom of both screenshots shows the date as 12-02-2023 and the time as 08:47.

**Thank you for downloading Windows installer**  
Download hasn't started? [Click this link](#)

**Changing boot configuration**  
By default, your Jenkins runs at <https://localhost:8080/>. This can be changed to other ports, such as JVM options, HTTPS setup, etc.

**Starting/stopping the service**  
Jenkins is installed as a Windows service, and it is configured to start automatically. You can start or stop the service from the control panel, or the `sc` command line tool.

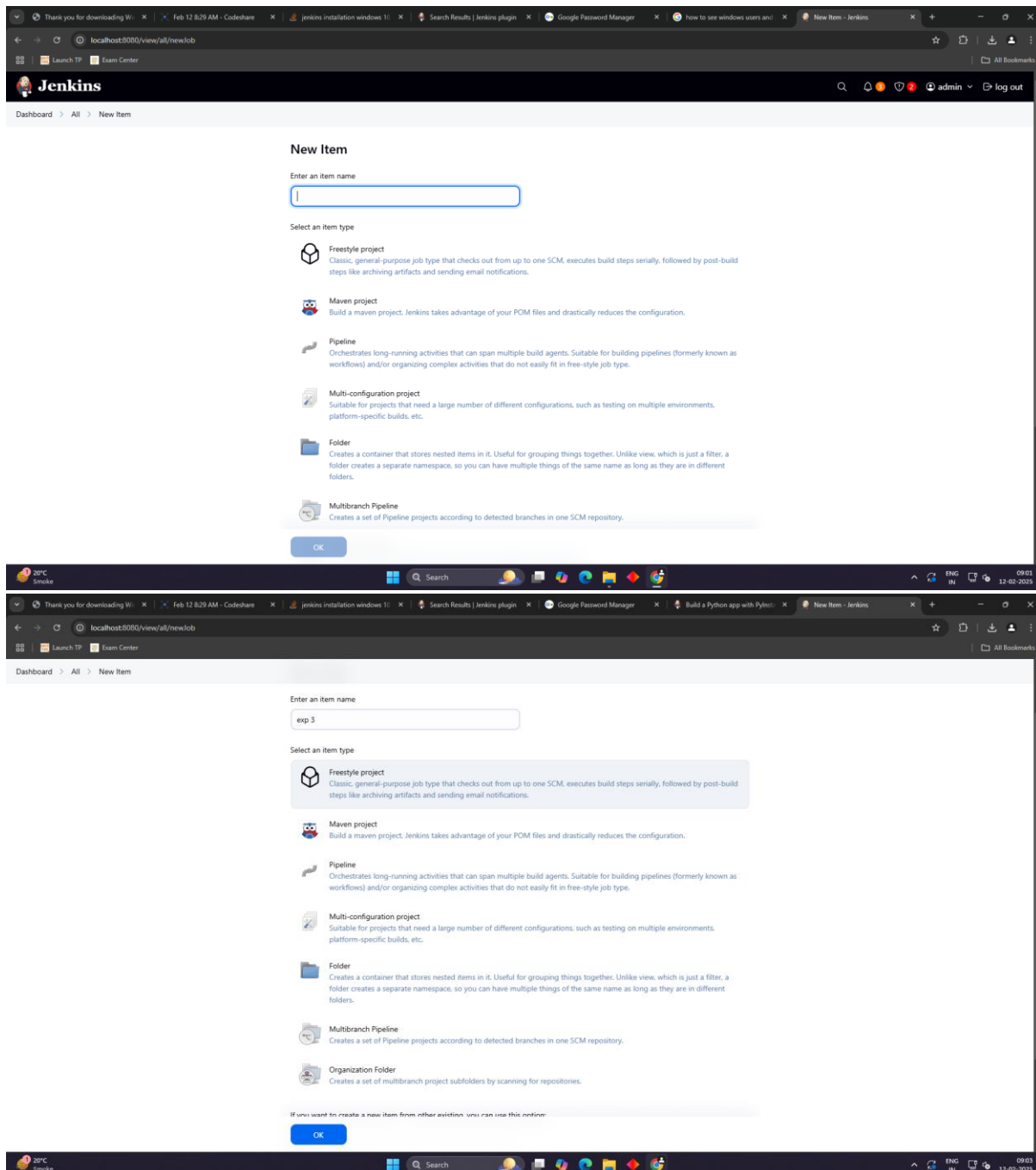
**Inheriting your existing Jenkins installation**  
If you'd like your new installation to take over your existing Jenkins data, copy the `jenkins` folder from the old installation to the new one.

**See Also**

- Running Jenkins behind Internet Information Server (IIS)
- Running Jenkins behind nginx
- Running Jenkins behind Apache

**Jenkins 2.497 Setup Wizard**  
Welcome to the Jenkins 2.497 Setup Wizard  
The Setup Wizard allows you to change the way Jenkins 2.497 features are installed on your computer or to remove it from your computer. Click Next to continue or Cancel to exit the Setup Wizard.

**Sign in to Jenkins**  
Username  
admin  
Password  
\*\*\*\*\*  
☒ Keep me signed in  
Sign in



**Conclusion :** Thus we have successfully installed and configured Jenkins.