

EXPERIMENT – 4

AIM - To understand Continuous Integration, install and configure Jenkins with Maven/Ant/Gradle to setup a build Job

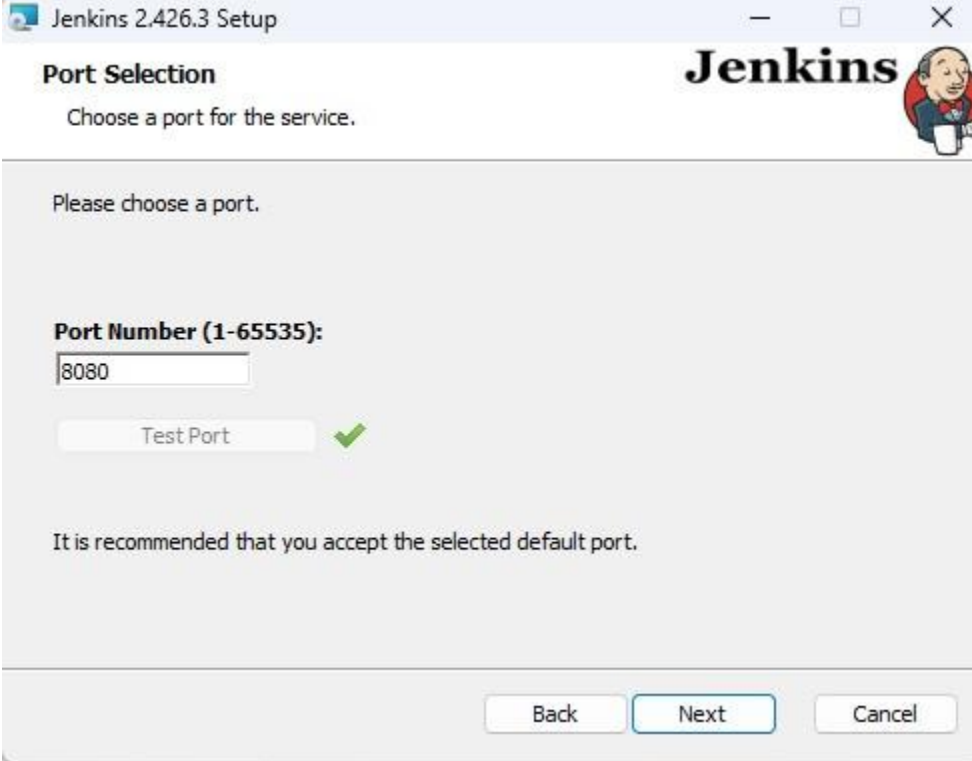
THEORY:

Jenkins is an open-source automation tool written in Java with plugins built for Continuous Integration purpose. Jenkins is used to build and test your software projects continuously making it easier for developers to integrate changes to the project and making it easier for users to obtain a fresh build. It also allows you to continuously deliver your software by integrating with many testing and deployment technologies.

Step 1: Install Jenkins from website



Step 2: Select the desired options



The screenshot shows the 'Port Selection' window of the Jenkins 2.426.3 Setup. The window has a title bar 'Jenkins 2.426.3 Setup' and a Jenkins logo. The main content area says 'Choose a port for the service.' and 'Please choose a port.' Below this, there is a label 'Port Number (1-65535):' followed by a text input field containing '8080'. A 'Test Port' button is next to the input field, with a green checkmark to its right. At the bottom, there is a message: 'It is recommended that you accept the selected default port.' and three buttons: 'Back', 'Next', and 'Cancel'.

Jenkins 2.426.3 Setup

Port Selection

Choose a port for the service.

Please choose a port.

Port Number (1-65535):


8080

Test Port ✓

It is recommended that you accept the selected default port.

Back Next Cancel

Step 3: Select the files where jdk 21 is installed



The screenshot shows the 'Select Java home directory (JDK or JRE)' window of the Jenkins 2.426.3 Setup. The window has a title bar 'Jenkins 2.426.3 Setup' and a Jenkins logo. The main content area says 'Select Java home directory (JDK or JRE)' and 'Please select the path of a Java Development Kit or Java Runtime Environment. Only Java 11, 17 and 21 are supported by Jenkins.' Below this, there is a text input field containing 'C:\Program Files\Java\jdk-21\'. A 'Change...' button is next to the input field. At the bottom, there are three buttons: 'Back', 'Next', and 'Cancel'.

Jenkins 2.426.3 Setup

Select Java home directory (JDK or JRE)

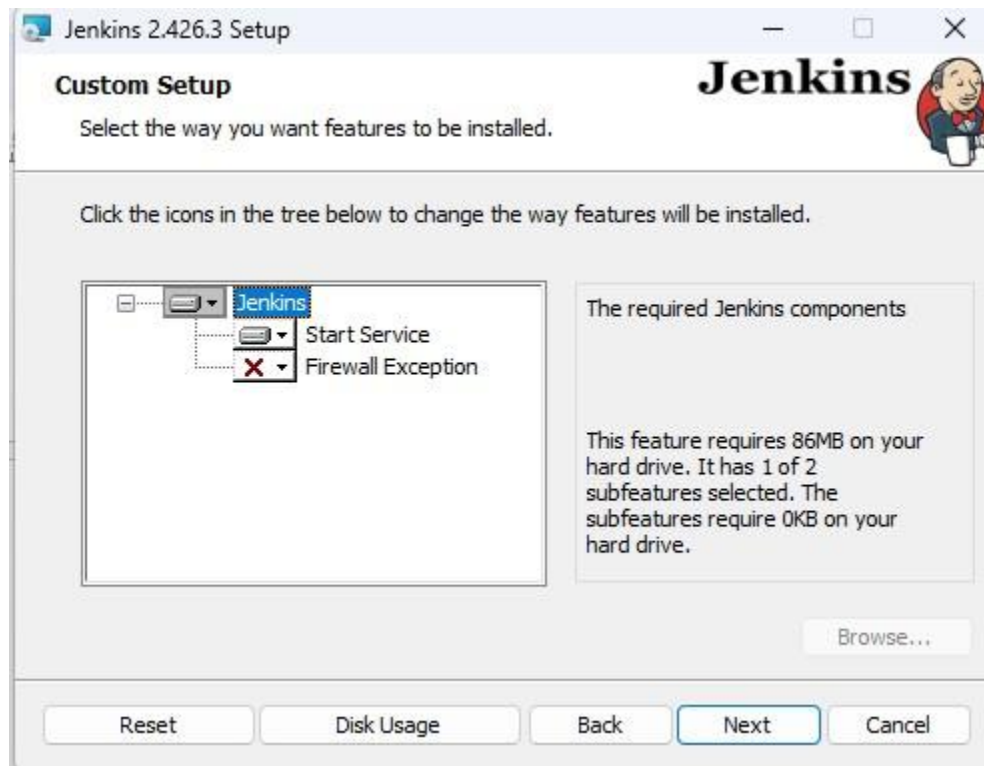
Please select the path of a Java Development Kit or Java Runtime Environment.
Only Java 11, 17 and 21 are supported by Jenkins.

C:\Program Files\Java\jdk-21\

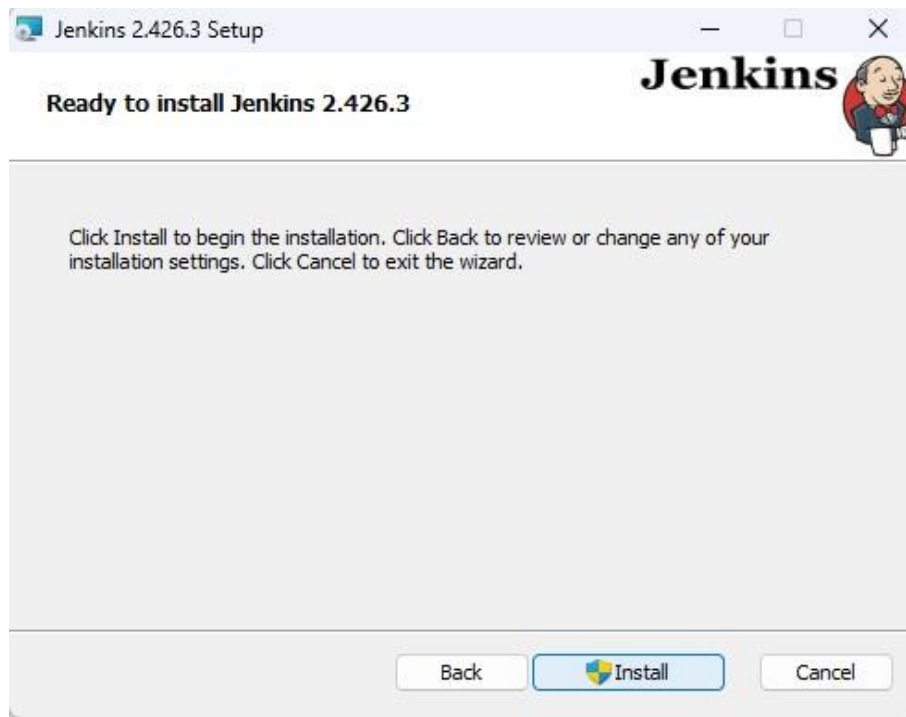
Change...

Back Next Cancel

Step 4: Select the desired options



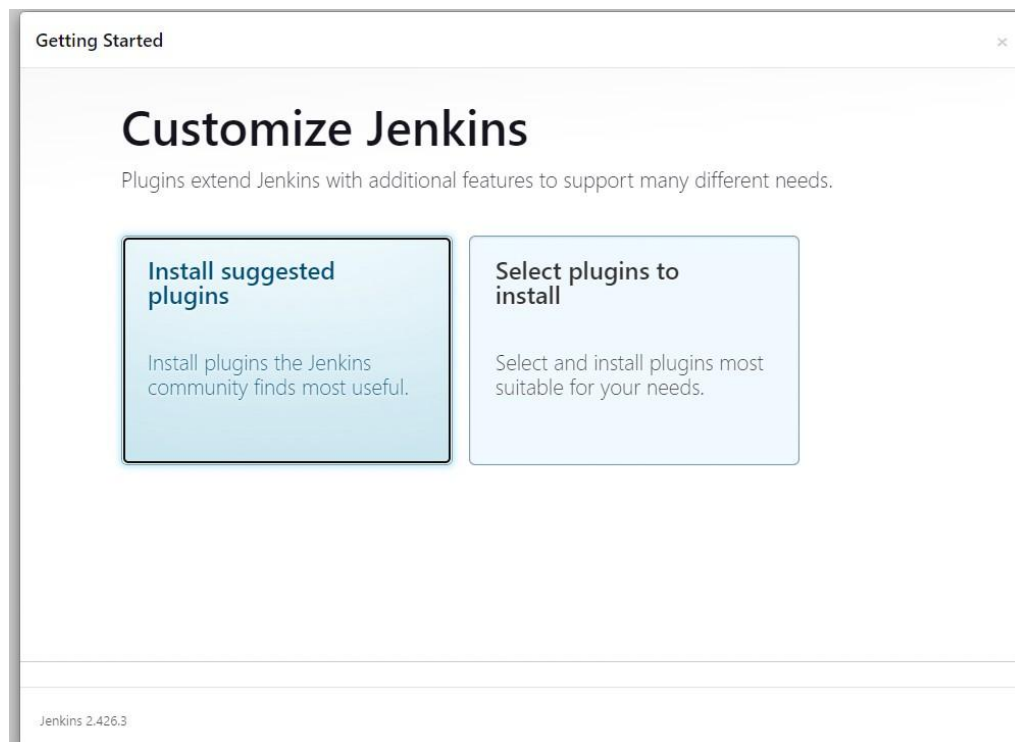
Step 5: Install the file



Step 6: Installation is finished



Step 7: Go on localhost:8080



Step 8: Select the options and wait for a few minutes

Getting Started

Getting Started

<input checked="" type="checkbox"/> Folders	<input checked="" type="checkbox"/> OWASP Markup Formatter	<input type="checkbox"/> Build Timeout	<input type="checkbox"/> Credentials Binding	** Icons API
<input type="checkbox"/> Timestamp	<input type="checkbox"/> Workspace Cleanup	<input type="checkbox"/> Ant	<input type="checkbox"/> Gradle	Folders
<input type="checkbox"/> Pipeline	<input type="checkbox"/> GitHub Branch Source	<input type="checkbox"/> Pipeline: GitHub Groovy Libraries	<input type="checkbox"/> Pipeline: Stage View	OWASP Markup Formatter
<input type="checkbox"/> Git	<input type="checkbox"/> SSH Build Agents	<input type="checkbox"/> Matrix Authorization Strategy	<input type="checkbox"/> PAM Authentication	
<input type="checkbox"/> LDAP	<input type="checkbox"/> Email Extension	<input type="checkbox"/> Mailer		

** - required dependency

Jenkins 2.426.3

Step 9: Fill the appropriate credentials

Getting Started

Create First Admin User

Username

Password

Confirm password

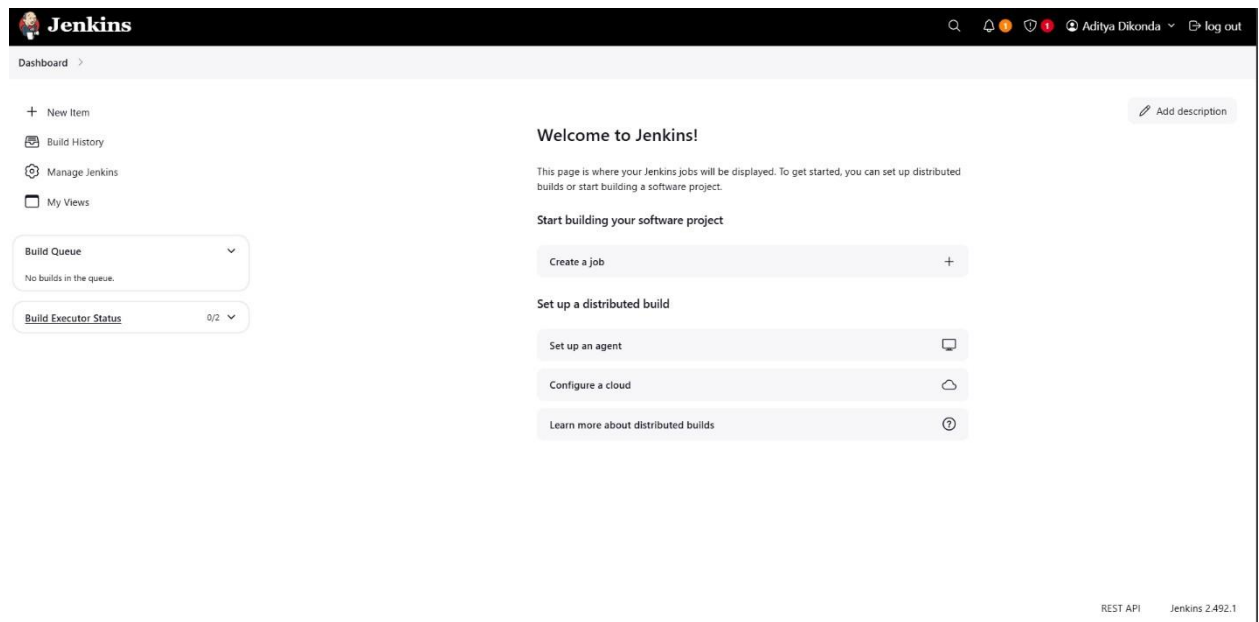
Full name

E-mail address

Jenkins 2.426.3

[Skip and continue as admin](#) [Save and Continue](#)

Step 10: Congratulations! You have created your Jenkins account



The screenshot shows the Jenkins Dashboard interface. At the top, there's a dark header with the Jenkins logo, a search icon, notification icons, and a user profile for 'Aditya Dikonda' with a 'log out' button. Below the header, the main content area is divided into a left sidebar and a main panel. The sidebar contains links for 'New Item', 'Build History', 'Manage Jenkins', and 'My Views'. Below these are two status boxes: 'Build Queue' showing 'No builds in the queue' and 'Build Executor Status' showing '0/2'. The main panel features a 'Welcome to Jenkins!' message, a brief explanation of the page's purpose, and a 'Start building your software project' section with a 'Create a job' button. Below that, a 'Set up a distributed build' section includes buttons for 'Set up an agent', 'Configure a cloud', and 'Learn more about distributed builds'. At the bottom right, the footer displays 'REST API' and 'Jenkins 2.492.1'.

CONCLUSION:

Jenkins with Maven simplifies Continuous Integration, automating builds and improving project efficiency for faster development cycles and higher software quality.