Jenkins pipeline can be integrated with Git, a popular version control system, to manage and track changes to your source code. Integrating Jenkins with Git allows you to automatically build and test your code every time you push changes to your Git repository. This helps to catch and fix issues early in the development process and ensures that your code is always in a releasable state.

Here's how you can integrate Jenkins with Git:

Set up a Git repository: Start by creating a Git repository for your source code and pushing your code to the repository.

Install the Git plugin in Jenkins: Go to the Jenkins plugin manager, search for the Git plugin, and install it.

Configure Jenkins for Git: In Jenkins, go to Manage Jenkins > Configure System and scroll down to the Git section. Here, you can configure the path to your Git executable and set up any global Git settings.

Create a Jenkins pipeline: In Jenkins, create a new pipeline job and select "Pipeline" as the project type. In the pipeline configuration, select "Git" as the source code management system and enter the repository URL.

Set up a webhook: To automatically trigger a build every time you push changes to your Git repository, set up a webhook in your Git repository that sends a notification to Jenkins.

Start building: Every time you push changes to your Git repository, Jenkins will automatically build and test your code, providing real-time feedback on the status of your pipeline.