

Step-by-Step Guide: Build and Deploy a Simple App using Jenkins

This guide will walk you through **building and deploying** a simple application using **Jenkins CI/CD pipeline**.

Before you start, make sure you have the following:

Jenkins installed (locally, on a VM, or on Docker)

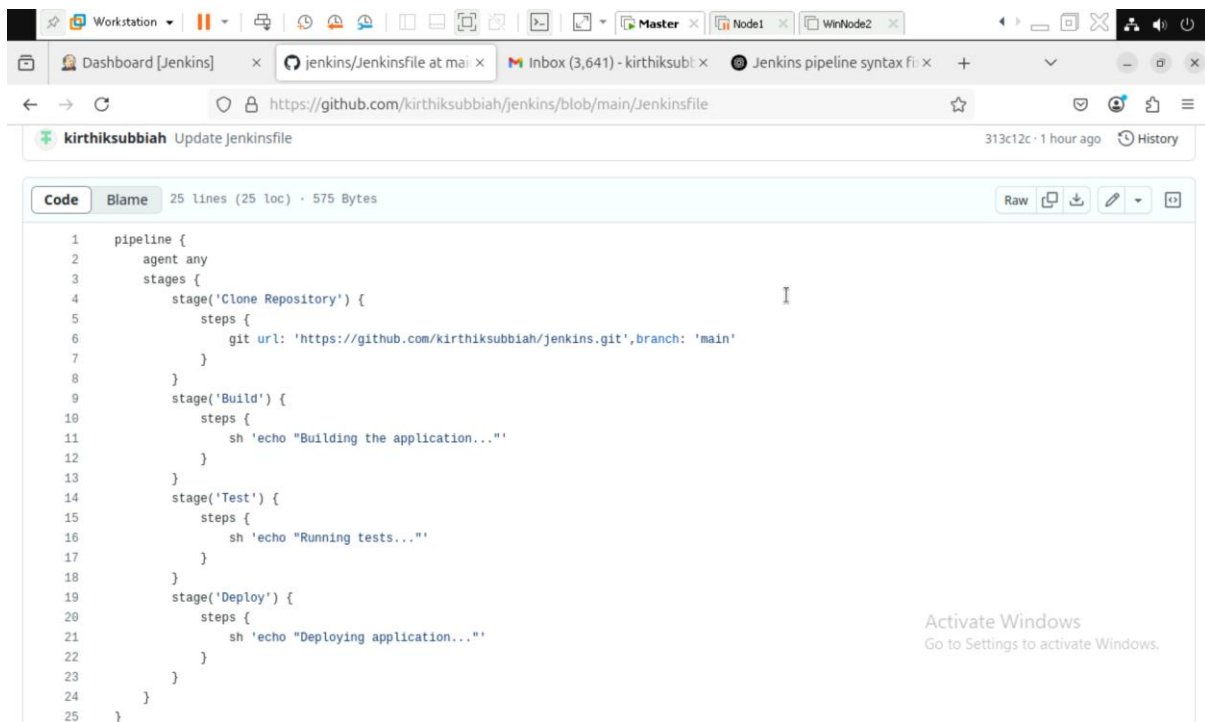
Git installed on Jenkins server

Docker installed if deploying with containers

A **GitHub repository** with a simple application (e.g., Python, Java, or Node.js)

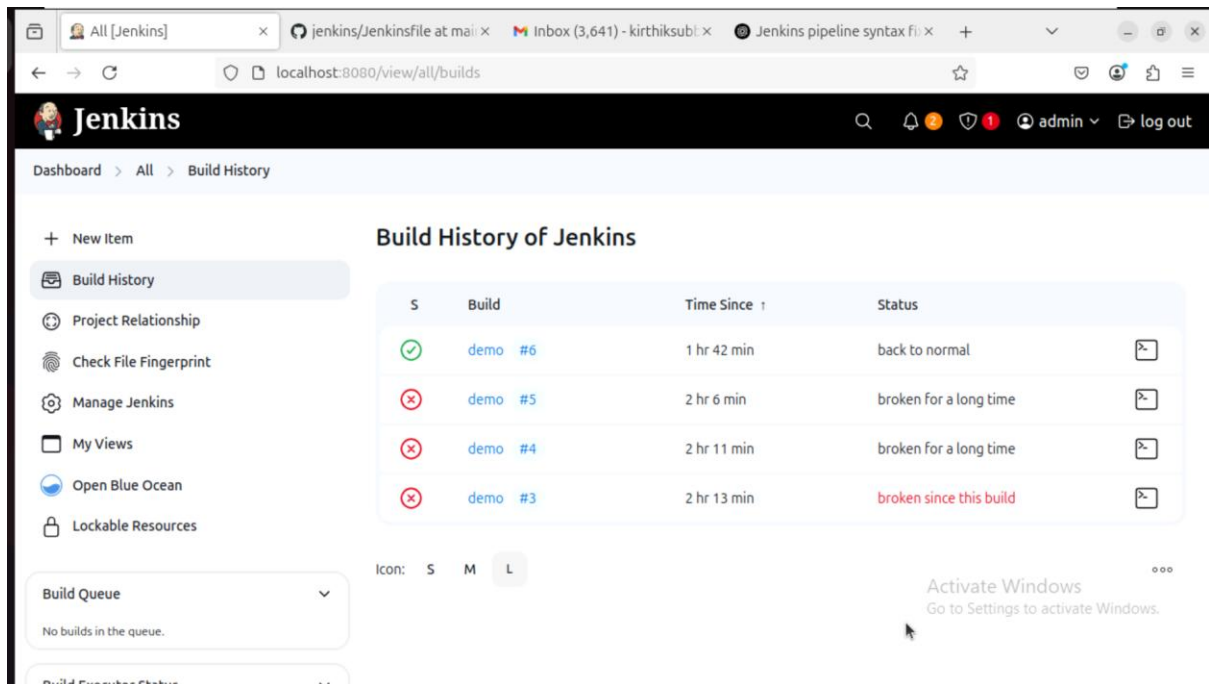
create in the next step).

CODE:

A screenshot of a web browser displaying a GitHub repository. The browser's address bar shows the URL 'https://github.com/kirthiksubbiah/jenkins/blob/main/Jenkinsfile'. The repository name 'kirthiksubbiah' is visible in the top left, and the file name 'Jenkinsfile' is in the top right. The file content is displayed in a code editor with line numbers from 1 to 25. The code is a Jenkins pipeline script. On the right side of the code editor, there are icons for 'Raw', 'Copy', 'Download', 'Edit', and 'Diff'. An 'Activate Windows' watermark is visible in the bottom right corner of the code editor area.

```
1 pipeline {
2   agent any
3   stages {
4     stage('Clone Repository') {
5       steps {
6         git url: 'https://github.com/kirthiksubbiah/jenkins.git',branch: 'main'
7       }
8     }
9     stage('Build') {
10      steps {
11        sh 'echo "Building the application..."'
12      }
13    }
14    stage('Test') {
15      steps {
16        sh 'echo "Running tests..."'
17      }
18    }
19    stage('Deploy') {
20      steps {
21        sh 'echo "Deploying application..."'
22      }
23    }
24  }
25 }
```

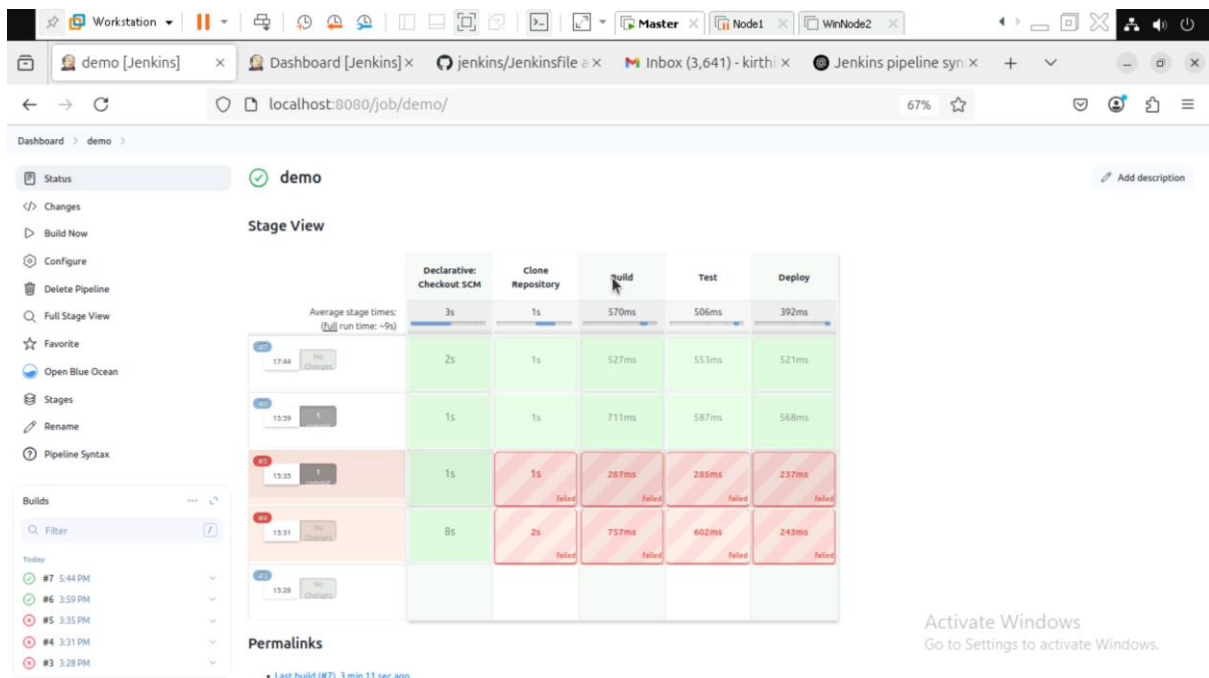
BUILD HISTORY OF JENKINS:



The screenshot shows the Jenkins web interface at localhost:8080/view/all/builds. The page title is "Build History of Jenkins". On the left, there is a sidebar with a "New Item" button and a list of links: "Build History" (selected), "Project Relationship", "Check File Fingerprint", "Manage Jenkins", "My Views", "Open Blue Ocean", and "Lockable Resources". Below the sidebar, there are sections for "Build Queue" (showing "No builds in the queue.") and "Build Executor Status". The main content area displays a table of build history. The table has columns: "S" (status icon), "Build" (name and number), "Time Since" (duration), and "Status" (description). The table shows four builds: #6 (green checkmark, "back to normal"), #5 (red X, "broken for a long time"), #4 (red X, "broken for a long time"), and #3 (red X, "broken since this build"). Below the table, there is a "Icon:" section with "S", "M", and "L" options. A watermark "Activate Windows" is visible in the bottom right corner.

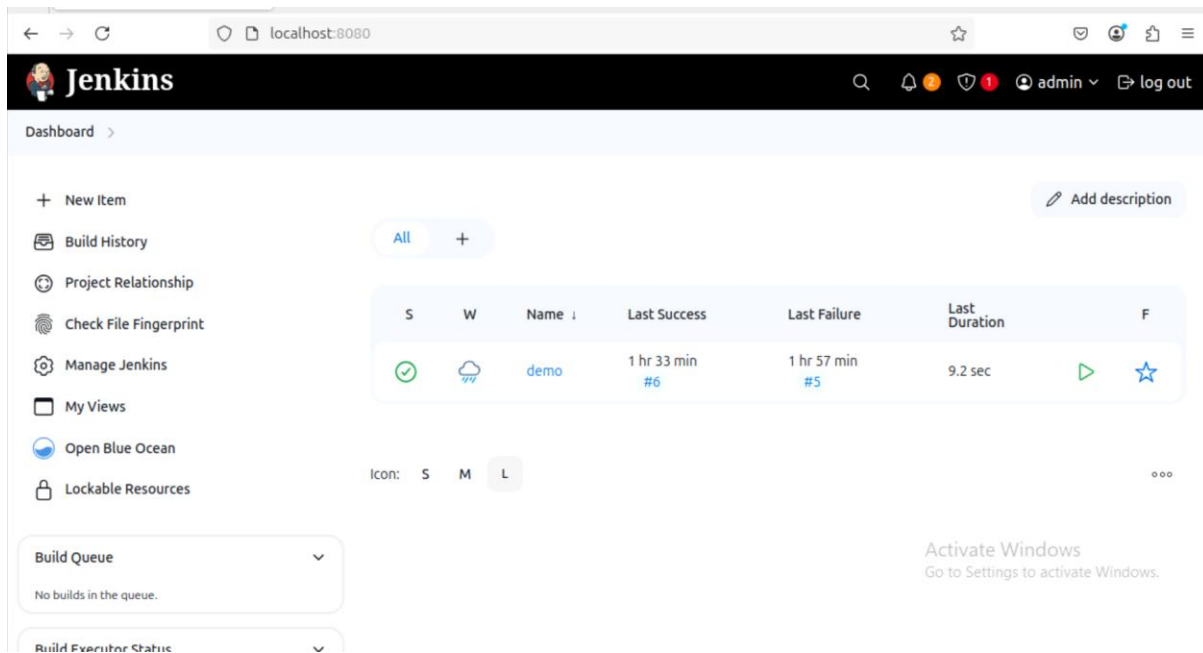
S	Build	Time Since	Status
✓	demo #6	1 hr 42 min	back to normal
✗	demo #5	2 hr 6 min	broken for a long time
✗	demo #4	2 hr 11 min	broken for a long time
✗	demo #3	2 hr 13 min	broken since this build

STAGE VIEW:

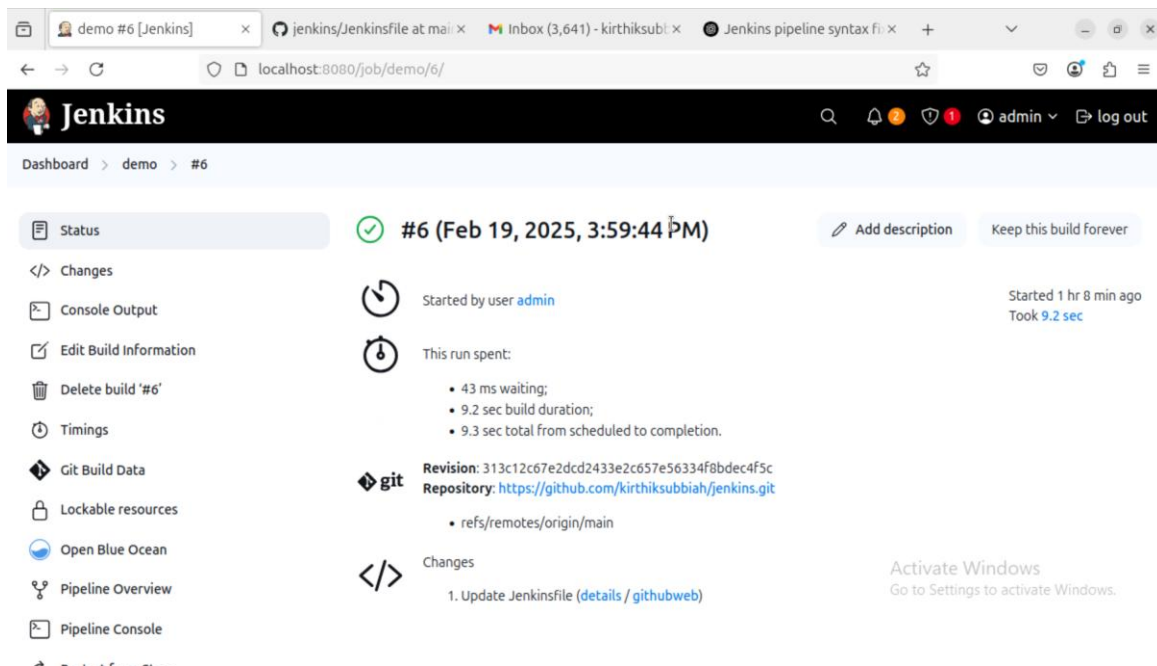


The screenshot shows the Jenkins web interface at localhost:8080/job/demo/. The page title is "demo". On the left, there is a sidebar with a "Status" button (selected) and a list of links: "Changes", "Build Now", "Configure", "Delete Pipeline", "Full Stage View", "Favorite", "Open Blue Ocean", "Stages", "Rename", and "Pipeline Syntax". Below the sidebar, there is a "Builds" section with a "Filter" button and a list of builds: #7 (green checkmark, "5:44 PM"), #6 (green checkmark, "3:59 PM"), #5 (red X, "3:35 PM"), #4 (red X, "3:31 PM"), and #3 (red X, "3:28 PM"). The main content area displays a "Stage View" table. The table has columns: "Declarative: Checkout SCM", "Clone Repository", "Build", "Test", and "Deploy". The table shows five builds: #7, #6, #5, #4, and #3. Build #7 and #6 are green, while #5, #4, and #3 are red. A watermark "Activate Windows" is visible in the bottom right corner.

	Declarative: Checkout SCM	Clone Repository	Build	Test	Deploy
Average stage times: (full run time: ~9s)	3s	1s	570ms	506ms	392ms
#7 17:44	2s	1s	527ms	553ms	521ms
#6 15:59	1s	1s	711ms	587ms	568ms
#5 15:35	1s	1s	287ms	285ms	237ms
#4 15:31	8s	2s	757ms	602ms	243ms
#3 15:28					



STATUS:



Create a Jenkinsfile (CI/CD Pipeline)

In your GitHub repository, create a new file named **Jenkinsfile** and add the following code:

```
groovy
CopyEdit
pipeline {
```

```
agent any
stages {
  stage('Clone Repository') {
    steps {
      git 'https://github.com/your-username/your-repo.git'
    }
  }
  stage('Build') {
    steps {
      sh 'echo "Building the application..."'
    }
  }
  stage('Test') {
    steps {
      sh 'echo "Running tests..."'
    }
  }
  stage('Deploy') {
    steps {
      sh 'echo "Deploying application..."'
    }
  }
}
}
```