



format this document into a PDF

I can't directly generate PDF files, but here's your complete CI/CD document in **perfect Markdown format** ready for instant PDF conversion using any of these free online tools:

Recommended converters:

- <https://www.pdfforge.org/online/en/markdown-to-pdf>^[1]
- <https://codeshack.io/markdown-to-pdf-converter/>^[2]
- <https://markdownlivepreview.dev/tools/markdown-to-pdf>^[3]

CI/CD Pipeline: Jenkins + Docker + Nginx Web Server Deployment

Overview

Automated pipeline that builds Docker images from GitHub code and deploys Nginx web containers to AWS EC2.

Step 1 – Jenkins Server Setup (CI)

1. Install Jenkins
2. Install Docker + Git on Jenkins server
3. Add jenkins user to docker group: `sudo usermod -aG docker jenkins`
4. Install plugins: Git, Docker, Pipeline, SSH Agent
5. Configure credentials:
 - DockerHub: 'dockerhub-user-pass'
 - Deployment server SSH: 'deploy-server-ssh'
6. Restart Jenkins
7. Create Pipeline job pointing to GitHub repo

Step 2 – Deployment Server Setup (CD)

1. Install Docker on Ubuntu EC2 (54.226.161.122)
2. Add ubuntu user to docker group: `sudo usermod -aG docker ubuntu`
3. Ensure SSH access from Jenkins server
4. Open port 80 for web traffic

Step 3 – GitHub Repository Setup

1. Create repo: farhaz1449/simple-webpage-in-Docker-Container
2. Add: index.html + Dockerfile
3. Push to main branch
4. Add Jenkinsfile (below)
5. Create GitHub webhook → Jenkins

Jenkinsfile (Complete Pipeline)

```
pipeline {  
    agent any  
  
    environment {  
        DOCKERHUB_CREDENTIALS = credentials('dockerhub-user-pass')  
        IMAGE_NAME = "farhaz1449/nginx-web"  
        USER = "ubuntu"  
        PORT = "22"  
        SERVER_IP = "54.226.161.122"  
        APP_NAME = "nginx-web"  
        VERSION = "${BUILD_NUMBER}"  
    }  
  
    stages {  
        stage('Clone Code') {  
            steps {  
                git branch: 'main',  
                    url: 'https://github.com/farhaz1449/simple-webpage-in-Docker-Container'  
            }  
        }  
        stage('Build Docker Image') {  
            steps {  
                script {  
                    sh "docker build -t ${IMAGE_NAME}:${VERSION} ."  
                }  
            }  
        }  
        stage('Push to DockerHub') {  
            steps {  
                script {  
                    sh 'echo $DOCKERHUB_CREDENTIALS_PSW | docker login -u $DOCKERHUB_CRED...'   
                    sh "docker push ${IMAGE_NAME}:${VERSION}"  
                }  
            }  
        }  
        stage('Deploy') {  
            steps {  
                sshagent(credentials: ['deploy-server-ssh']) {  
                    sh '''  
                        ssh -o StrictHostKeyChecking=no $USER@$SERVER_IP \  
"docker pull ${IMAGE_NAME}:${VERSION} && \  
docker rm -f ${APP_NAME} || true && \  
                }  
            }  
        }  
    }  
}
```

```

        docker run -d --name $APP_NAME -p 80:80 $IMAGE_NAME:${VERSION}
    ...
}

}

}

}

post {
    always { echo "Pipeline completed. Build #${BUILD_NUMBER}" }
    failure { echo 'Deployment failed - check logs above.' }
    success { echo "Deployment successful! Check http://${SERVER_IP}" }
}
}

```

Pipeline Flow

Git Push → Webhook → Jenkins → Build → DockerHub → SSH → Deploy EC2:80

Live URL after success: <http://54.226.161.122>

Copy this entire Markdown → paste into any converter above → Download PDF instantly. [\[1\]](#)
[\[2\]](#)

**

1. <https://www.pdfforge.org/online/en/markdown-to-pdf>
2. <https://codeshack.io/markdown-to-pdf-converter/>
3. <https://markdownlivepreview.dev/tools/markdown-to-pdf>
4. <https://dillinger.io>
5. <https://apitemplate.io/pdf-tools/convert-markdown-to-pdf/>
6. <https://linangdata.com/pdf-generator/>
7. https://play.google.com/store/apps/details?id=come.shotdoor.markdown.pdf.fast.converter&hl=en_US
8. https://www.reddit.com/r/Markdown/comments/1hrssk7/markdown_to_pdf_converter/
9. <https://www.markdowntopdf.com>
10. <https://cloudconvert.com/md-to-pdf>