Jenkins with Docker Swarm #2

I. Prerequisites:

1. A Docker Swarm cluster is already set up.
2. Jenkins is installed and running.
3. Jenkins has Docker installed or Jenkins is set up with Docker agents that can int0065ract with the Docker Swarm cluster.
4. A Jenkins pipeline file (Jenkinsfile) is configured for CI/CD.

II. Installation

1. Setup Docker Swarm Cluster

```
$docker swarm init
```

*You can add worker nodes using the join command provided by Docker Swarm after initializing the manager node.

2. Install Docker Plugin on Jenkins

In Jenkins, go to Manage Jenkins > Manage Plugins, then search for and install:

- Docker Pipeline
- Docker Commons

*These plugins allow Jenkins to work with Docker commands and Swarm.

3. Create a Simple HTML Application

Place your HTML files in a directory (e.g., html-app/) and create a Dockerfile in the same directory:

```
# Dockerfile

FROM nginx:alpine

COPY . /usr/share/nginx/html
```

4. Push Your Application to a Git Repository

Ensure your HTML application, along with the Dockerfile, is in a version control repository like GitHub or GitLab.

5. Create a Jenkins Pipeline (Jenkinsfile)

Create a Jenkinsfile in your repository to define the Jenkins pipeline. Here's an example pipeline to build and deploy your HTML app to Docker Swarm:

```
pipeline {
    agent any

    environment {
        DOCKER_IMAGE = "your-dockerhub-username/html-app"  // Replace with your image
        DOCKER_TAG = "latest"  // You can use versioning if needed
    }

    stages {
        stage('Clone Repository') {
            steps {
                git branch: 'main', url: 'https://github.com/your-repo/html-app.git'  // Replace with your repository
            }
        }

        stage('Build Docker Image') {
            steps {
                script {
                    docker.build("${DOCKER_IMAGE}:${DOCKER_TAG}")
                }
            }
        }

        stage('Push Docker Image') {
            steps {
                script {
                    docker.withRegistry('https://registry.hub.docker.com', 'dockerhub-credentials') {
                        docker.image("${DOCKER_IMAGE}:${DOCKER_TAG}").push()
                    }
                }
            }
        }

        stage('Deploy to Docker Swarm') {
            steps {
                script {
                    // Deploy to Docker Swarm
                    sh """
                    docker service update --image ${DOCKER_IMAGE}:${DOCKER_TAG} my-html-app || docker service create --name my-html-app --publish 80:80 ${DOCKER_IMAGE}:${DOCKER_TAG}
                """
              }
            }
          }
        }

        post {
          always {
            cleanWs()
          }
        }
      }
```

6. Configure Jenkins Credentials for Docker Hub

In Jenkins, go to Manage Jenkins > Manage Credentials.

Add credentials for Docker Hub, with the ID dockerhub-credentials.

7. Run the Jenkins Pipeline

Go to your Jenkins dashboard, create a new pipeline job, and point it to the repository containing the Jenkinsfile.

Trigger the pipeline to build, push, and deploy your HTML app.

8. Access Your Application

Once the pipeline completes, you can access your HTML app by visiting any node in your Docker Swarm cluster on port 80.