Jenkins Beginner Assignments with Theory

1. Install Jenkins

Theory:

Jenkins is an open-source automation server used for continuous integration and delivery (CI/CD).

It automates software development processes like building, testing, and deploying code.

Topics Covered:

- What is Jenkins?
- CI/CD concepts
- Jenkins architecture
- Installing Jenkins

Practical Steps:

1. Download Jenkins from jenkins.io
2. Install Java
3. Start Jenkins via localhost:8080
4. Unlock with admin password
5. Install plugins
6. Create user

Goal:

Understand how Jenkins is set up and accessible.

2. Create Your First Freestyle Job

Theory:

Freestyle project is the simplest job type in Jenkins to perform tasks like running a script.

Topics Covered:

- Freestyle jobs
- Jenkins workspace
- Build steps (Shell/Bash)

Practical Steps:

1. New Item Freestyle Project Hello-World
2. Add shell build step: echo 'Hello from Jenkins'
3. Save and build
4. View console output

Jenkins Beginner Assignments with Theory

Goal:

Understand job execution and Jenkins UI.

3. Automate GitHub Code Pull

Theory:

Jenkins can pull source code from Git repositories and build them.

Topics Covered:

- Git integration
- SCM configuration
- GitHub connectivity

Practical Steps:

1. Push script to GitHub
2. Create Freestyle job
3. Use Git repo URL in SCM
4. Build and display code

Goal:

Learn how Jenkins fetches code from GitHub.

4. Build Triggers

Theory:

Jobs can be triggered manually or via polling, webhooks, or schedules.

Topics Covered:

- Jenkins cron syntax
- SCM polling
- Build automation

Practical Steps:

1. Enable Poll SCM
2. Use cron: H/2 * * * *
3. Commit to GitHub
4. Jenkins builds automatically

Jenkins Beginner Assignments with Theory

Goal:

Understand Jenkins build triggers.

5. Archive Build Artifacts

Theory:

Artifacts are build outputs that Jenkins can archive for future access.

Topics Covered:

- Artifact management
- Post-build actions
- File patterns

Practical Steps:

1. Script creates result.txt
2. Archive result.txt in Post-build actions
3. Build job and verify artifact

Goal:

Learn how to persist build outputs.

6. Create Your First Pipeline Job

Theory:

Pipeline jobs use scripts (Groovy DSL) to define multi-stage processes.

Topics Covered:

- Declarative pipeline syntax
- Stages and steps
- Visual pipeline

Practical Steps:

1. New Item Pipeline
2. Add pipeline script with stages: Build, Test, Deploy
3. Run and observe output

Goal:

Understand pipelines and CI stages.

Jenkins Beginner Assignments with Theory

7. Use Jenkinsfile from GitHub

Theory:

Pipelines can be defined in a Jenkinsfile in the source code repository.

Topics Covered:

- Jenkinsfile
- Pipeline as Code
- Pipeline from SCM

Practical Steps:

1. Push Jenkinsfile to GitHub
2. Create Pipeline Job
3. Choose Pipeline from SCM
4. Build and observe pipeline

Goal:

Use pipelines as code stored in version control.