

Jenkins for SDET: Final Checklist & Topic Summary

Core Jenkins Skills for SDET (You Have Covered)

1. Running Automation Test Suites

- Execute test frameworks (Playwright, Pytest, Selenium) using Jenkins.
- Trigger tests on code push or schedule.
- Example: Running Playwright tests inside a Docker container using Jenkins.

2. Docker as Jenkins Agent

- Using a Docker image or Dockerfile as the agent inside Jenkins pipeline.
- Benefit: Environment consistency, fewer setup steps.
- Options:
 - Use a public image like `python:3.10`.
 - Use your custom image pushed to Docker Hub.
 - Use `docker file` block to build locally.

3. Archive Test Reports

- Use `archiveArtifacts` to save HTML, XML, or JSON reports.
- Reports can be accessed via Jenkins UI.
- Example:

```
archiveArtifacts artifacts: 'reports/**', allowEmptyArchive: true
```

4. Email Notifications

- Notify test result via email.
- Configure SMTP + use `emailext` plugin.
- Example in `post` block:

```
post {
  always {
    emailext(
      subject: "Test Results",
      body: "Check the attached report.",
      to: "you@example.com"
    )
  }
}
```

5. Secure Git Credentials

- Use Jenkins credentials manager to store SSH keys or tokens.
- Clone private repositories securely.
- Example:

```
git credentialsId: 'my-git-creds', url: 'git@github.com:your/repo.git'
```

6. Declarative Pipeline Syntax

- Understand `pipeline`, `agent`, `stages`, `steps`, `environment`, `post` blocks.
- Example basic pipeline:

```
pipeline {
  agent any
  stages {
    stage('Test') {
      steps {
        sh 'pytest tests/'
      }
    }
  }
}
```

7. Parameterized Builds

- Choose test type, branch, or environment from Jenkins UI.
- Example:

```
parameters {
  string(name: 'BRANCH', defaultValue: 'main', description: 'Git
branch')
}
```

8. Parallel Test Execution (Optional)

- Run different test suites in parallel stages.
- Example:

```
stages {
  stage('Run Tests') {
    parallel {
      stage('UI Tests') { steps { sh 'pytest ui/' } }
      stage('API Tests') { steps { sh 'pytest api/' } }
    }
  }
}
```

9. Difference Between Agent and Slave

- **Agent:** Any node (machine or container) that runs your pipeline.
- **Slave:** Older term for a Jenkins agent node (remote Linux machine or VM).
- You can use Docker **inside** a slave.

10. Artifacts vs Volumes

- **archiveArtifacts:** Persists data within Jenkins UI.
 - **Volumes:** Mount external directory into container (not always needed if you archive reports).
-

Verdict

You are now equipped with:

- Docker-based Jenkins agents
- Declarative pipeline scripting
- Running Playwright/Pytest tests
- Storing artifacts
- Email notifications
- Secure Git integration

 That covers **everything essential for an SDET role using Jenkins!**

Let me know if you want this exported as a PDF or want to move on to Jenkinsfile best practices or advanced tips.