▶ DevOps Shack

Errors & Troubleshooting in Jenkins

[Click Here To Enrol To Batch-5 | DevOps & Cloud DevOps](#)

1. Jenkins Installation Issues

Error: `java.io.IOException: Cannot run program "git"`
Solution: Ensure Git is installed and added to the system PATH.

```
# Install Git
sudo apt-get install git

# Verify Git installation
git --version

# Add Git to PATH (if needed)
export PATH=$PATH:/usr/local/git/bin
```

Error: `HTTP ERROR 503 Service Unavailable`
Solution: This could be due to Jenkins not starting properly. Check the Jenkins logs and restart the Jenkins service.

```
# Check Jenkins logs
tail -f /var/log/jenkins/jenkins.log

# Restart Jenkins service
sudo systemctl restart jenkins
```

2. Jenkins Configuration Issues

Error: `No valid crumb was included in the request`
Solution: Disable CSRF Protection or configure it correctly.

```
# Disable CSRF Protection
Manage Jenkins > Configure Global Security > CSRF Protection > Uncheck
"Prevent Cross Site Request Forgery exploits"
```

Error: `Error: invalid Jenkins URL configured`

Solution: Ensure the Jenkins URL is correctly set in the configuration.

```
# Set Jenkins URL
Manage Jenkins > Configure System > Jenkins Location > Jenkins URL
```

3. Jenkins Pipeline Issues

Error: `No such DSL method 'pipeline' found`

Solution: Ensure the Pipeline plugin is installed and updated.

```
# Install Pipeline plugin
Manage Jenkins > Manage Plugins > Available > Pipeline

# Update Pipeline plugin
Manage Jenkins > Manage Plugins > Updates > Pipeline
```

Error: `Scripts not permitted to use method`

Solution: Approve the script methods in the Script Approval section.

```
# Approve script methods
Manage Jenkins > In-process Script Approval > Approve
```

4. Jenkins Build Issues

Error: `Failed to connect to repository`

Solution: Verify the repository URL and credentials.

```
# Check repository URL
https://github.com/user/repo.git

# Verify credentials
Manage Jenkins > Manage Credentials > Add Credentials
```

Error: `BUILD FAILURE: Could not resolve dependencies`

Solution: Check the repository configuration and ensure all dependencies are available.

```
# Update repository configuration
pom.xml or build.gradle

# Check dependencies
mvn clean install
```

5. Jenkins Plugin Issues

Error: `Plugin not installed properly`

Solution: Reinstall or update the plugin.

```
# Reinstall plugin
```

```
Manage Jenkins > Manage Plugins > Installed > Select Plugin > Uninstall >
Install

# Update plugin
Manage Jenkins > Manage Plugins > Updates
```

Error: `Plugin compatibility issues`

Solution: Check for compatibility issues and update plugins or Jenkins.

```
# Check plugin compatibility
https://plugins.jenkins.io/

# Update Jenkins
Manage Jenkins > Manage Plugins > Updates > Upgrade Jenkins
```

6. Jenkins Slave/Agent Issues

Error: `Agent not able to connect to Jenkins master`

Solution: Ensure the agent configuration is correct and the connection is allowed.

```
# Verify agent configuration
Manage Jenkins > Manage Nodes and Clouds > Configure

# Allow connections
Manage Jenkins > Configure Global Security > TCP port for JNLP agents >
Fixed
```

Error: `Agent is offline`

Solution: Restart the agent and ensure it's properly configured.

```
# Restart agent
ssh into the agent and restart the Jenkins agent service

# Check configuration
Manage Jenkins > Manage Nodes and Clouds > Configure
```

7. Jenkins Security Issues

Error: `Unauthorized access`

Solution: Set up proper user permissions and authentication methods.

```
# Configure user permissions
Manage Jenkins > Manage and Assign Roles > Assign Roles

# Set up authentication
Manage Jenkins > Configure Global Security > Security Realm > LDAP or
Jenkins' own user database
```

Error: `Security vulnerabilities`

Solution: Regularly update Jenkins and plugins, and follow security best practices.

```
# Update Jenkins and plugins
Manage Jenkins > Manage Plugins > Updates
```

```
# Follow security best practices
https://www.jenkins.io/doc/book/security/
```

8. Jenkins Performance Issues

Error: `Jenkins is slow`
Solution: Optimize Jenkins performance by tuning JVM options and managing resources.

```
# Tune JVM options
JENKINS_JAVA_OPTIONS="-Djava.awt.headless=true -Xmx2g -XX:+UseG1GC"

# Manage resources
Manage Jenkins > Configure System > Monitor and manage system resources
```

Error: `Out of memory errors`
Solution: Increase the heap size and manage build logs.

```
# Increase heap size
JENKINS_JAVA_OPTIONS="-Xmx4g"

# Manage build logs
Manage Jenkins > Configure System > Build Record Root Directory
```

9. Jenkins Backup and Restore Issues

Error: `Backup failed`
Solution: Ensure backup scripts and configurations are correct.

```
# Verify backup script
rsync -av /var/lib/jenkins/ /backup/jenkins/

# Check configurations
Manage Jenkins > ThinBackup > Settings
```

Error: `Restore failed`
Solution: Ensure the backup files are intact and restore process is followed correctly.

```
# Verify backup files
ls /backup/jenkins/

# Follow restore process
Manage Jenkins > ThinBackup > Restore
```

10. Jenkins Notifications Issues

Error: `Email notifications not sent`
Solution: Verify email configurations and credentials.

```
# Configure email notifications
Manage Jenkins > Configure System > Email Notification
```

```
# Verify SMTP settings
smtp.example.com
```

Error: `Slack notifications not working`

Solution: Check Slack plugin configuration and integration settings.

```
# Configure Slack notifications
Manage Jenkins > Configure System > Slack

# Verify integration
Slack Webhook URL
```

11. Jenkins Job Scheduling Issues

Error: `Jobs not triggered as scheduled`

Solution: Verify the cron syntax and check the system clock.

```
# Verify cron syntax
H/5 * * * *

# Check system clock
date
```

Error: `Builds are not being triggered`

Solution: Check the build triggers configuration.

```
# Configure build triggers
Job Configuration > Build Triggers > Poll SCM or Build Periodically
```

12. Jenkins Docker Integration Issues

Error: `Cannot connect to Docker daemon`

Solution: Ensure Docker is running and the user has permission to run Docker commands.

```
# Start Docker service
sudo systemctl start docker

# Add user to Docker group
sudo usermod -aG docker $USER
```

Error: `Docker build failed`

Solution: Check the Dockerfile and build context.

```
# Verify Dockerfile
FROM openjdk:8
COPY . /app
WORKDIR /app
RUN ./mvnw package

# Specify build context
docker build -t my-app .
```

13. Jenkins SCM Integration Issues

Error: `Error checking out from SCM`
Solution: Verify SCM repository URL and credentials.

```
# Check SCM repository URL
https://github.com/user/repo.git

# Verify credentials
Manage Jenkins > Manage Credentials > Add Credentials
```

Error: `Branch not found`
Solution: Ensure the branch exists in the repository.

```
# List branches in repository
git branch -r

# Checkout the branch
git checkout branch-name
```

14. Jenkins Blue Ocean Issues

Error: `Blue Ocean plugin not displaying pipelines`
Solution: Ensure the Blue Ocean plugin is installed and updated.

```
# Install Blue Ocean plugin
Manage Jenkins > Manage Plugins > Available > Blue Ocean

# Update Blue Ocean plugin
Manage Jenkins > Manage Plugins > Updates > Blue Ocean
```

Error: `Blue Ocean UI not loading`
Solution: Check browser console for errors and Jenkins logs.

```
# Check browser console
F12 (Developer Tools) > Console

# Check Jenkins logs
tail -f /var/log/jenkins/jenkins.log
```

15. Jenkins Pipeline Syntax Issues

Error: `Expected a step @ line XX`
Solution: Review the syntax of the Jenkinsfile.

```
# Example Jenkinsfile
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                sh 'mvn clean install'
```

```
        }
      }
    }
}
```

Error: `Unknown parameter type`

Solution: Ensure the parameters are correctly defined in the Jenkinsfile.

```
# Example Jenkinsfile with parameters
pipeline {
    agent any
    parameters {
        string(name: 'ENV', defaultValue: 'dev', description:
'Environment')
    }
    stages {
        stage('Build') {
            steps {
                sh 'mvn clean install'
            }
        }
    }
}
```

16. Jenkins User Interface Issues

Error: `Jenkins UI not responding`

Solution: Check server resources and restart Jenkins.

```
# Check server resources
top

# Restart Jenkins
sudo systemctl restart jenkins
```

Error: `Error rendering UI elements`

Solution: Clear browser cache and check for plugin conflicts.

```
# Clear browser cache
Ctrl+Shift+R (hard refresh)

# Check for plugin conflicts
Manage Jenkins > Manage Plugins > Installed
```

17. Jenkins Node/Agent Configuration Issues

Error: `Agent disconnected`

Solution: Ensure network connectivity and proper configuration.

```
# Check network connectivity
ping agent-hostname

# Configure agent
Manage Jenkins > Manage Nodes and Clouds > Configure
```

Error: `Failed to launch agent`

Solution: Verify agent launch method and permissions.

```
# Verify agent launch method
Manage Jenkins > Manage Nodes and Clouds > Configure > Launch method

# Check permissions
chmod +x agent.jar
```

18. Jenkins Pipeline Execution Issues

Error: `Pipeline script returned exit code 1`

Solution: Check the script for errors and ensure all commands are correct.

```
# Example script with error handling
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                script {
                    try {
                        sh 'mvn clean install'
                    } catch (Exception e) {
                        currentBuild.result = 'FAILURE'
                        throw e
                    }
                }
            }
        }
    }
}
```

Error: `Timeout exceeded`

Solution: Increase the timeout value in the pipeline.

```
# Increase timeout
pipeline {
    agent any
    options {
        timeout(time: 60, unit: 'MINUTES')
    }
    stages {
        stage('Build') {
            steps {
                sh 'mvn clean install'
            }
        }
    }
}
```

19. Jenkins Artifacts Management Issues

Error: `Failed to archive artifacts`

Solution: Verify the artifact paths and storage space.

```
# Verify artifact paths
archiveArtifacts artifacts: '**/target/*.jar'

# Check storage space
df -h
```

Error: `Artifacts not found`

Solution: Ensure the build generates the artifacts in the specified path.

```
# Generate artifacts
mvn package

# Specify artifact path in Jenkinsfile
archiveArtifacts artifacts: '**/target/*.jar'
```

20. Jenkins Notifications Configuration Issues

Error: `Failed to send email`

Solution: Verify SMTP server settings and credentials.

```
# Configure SMTP server
Manage Jenkins > Configure System > Email Notification > SMTP server

# Verify credentials
smtp.example.com
```

Error: `Notification plugin not configured`

Solution: Install and configure the appropriate notification plugin.

```
# Install notification plugin
Manage Jenkins > Manage Plugins > Available > Email Extension Plugin

# Configure notification plugin
Job Configuration > Post-build Actions > Editable Email Notification
```

21. Jenkins Environment Variables Issues

Error: `Environment variable not found`

Solution: Ensure the environment variable is correctly defined and exported.

```
# Define and export environment variable
export VAR_NAME=value

# Access environment variable in Jenkinsfile
pipeline {
    agent any
    environment {
        VAR_NAME = 'value'
    }
    stages {
        stage('Build') {
            steps {
```

```
                sh 'echo $VAR_NAME'
            }
        }
    }
}
```

Error: `Variable substitution not working`

Solution: Check the syntax and ensure the variables are properly referenced.

```
# Correct variable substitution
pipeline {
    agent any
    environment {
        VAR_NAME = 'value'
    }
    stages {
        stage('Build') {
            steps {
                sh 'echo ${VAR_NAME}'
            }
        }
    }
}
```

22. Jenkins File Permission Issues

Error: `Permission denied`

Solution: Ensure the Jenkins user has the necessary permissions.

```
# Change file permissions
chmod +x script.sh

# Change file ownership
chown jenkins:jenkins script.sh
```

Error: `Cannot write to directory`

Solution: Verify the directory permissions and ensure Jenkins has write access.

```
# Change directory permissions
chmod -R 755 /path/to/directory

# Change directory ownership
chown -R jenkins:jenkins /path/to/directory
```

23. Jenkins Groovy Script Issues

Error: `Groovy script compilation error`

Solution: Check the syntax and ensure all required libraries are imported.

```
// Example Groovy script
import groovy.json.JsonSlurper

def json = new JsonSlurper().parseText('{"name": "Jenkins"}')
println json.name
```

Error: `Missing method exception`

Solution: Ensure the method is correctly defined and accessible.

```
// Define and use a method in Groovy script
def greet(name) {
    return "Hello, ${name}"
}

println greet('Jenkins')
```

24. Jenkins Job Configuration Issues

Error: `Invalid job configuration`

Solution: Review the job configuration and correct any errors.

```
# Export job configuration for review
Manage Jenkins > Manage Old Data > Export Configuration

# Validate configuration XML
xmllint --noout /var/lib/jenkins/jobs/jobname/config.xml
```

Error: `Cannot save job configuration`

Solution: Ensure Jenkins has write access to the job configuration files.

```
# Change configuration file permissions
chmod 644 /var/lib/jenkins/jobs/jobname/config.xml

# Change configuration file ownership
chown jenkins:jenkins /var/lib/jenkins/jobs/jobname/config.xml
```

25. Jenkins Plugin Dependency Issues

Error: `Plugin dependency missing`

Solution: Install the required dependency plugin.

```
# Install dependency plugin
Manage Jenkins > Manage Plugins > Available > Select Dependency Plugin >
Install

# Verify plugin dependencies
Manage Jenkins > Manage Plugins > Installed
```

Error: `Plugin version conflict`

Solution: Resolve the conflict by updating or downgrading the plugins.

```
# Update conflicting plugin
Manage Jenkins > Manage Plugins > Updates > Select Plugin > Update

# Downgrade conflicting plugin (if needed)
Manage Jenkins > Manage Plugins > Installed > Select Plugin > Downgrade
```

26. Jenkins Build Output Issues

Error: `Build output not visible`

Solution: Check the build log configuration and ensure it is enabled.

```
# Enable build log
Job Configuration > Build > Add build step > Shell > Command

# Access build log
Build History > Select Build > Console Output
```

Error: `Build log too large`

Solution: Configure log rotation and limit the log size.

```
# Configure log rotation
Manage Jenkins > Configure System > Log Rotation

# Limit log size
Job Configuration > Build Environment > Discard Old Builds > Max # of
builds to keep
```

27. Jenkins Master-Slave Communication Issues

Error: `Master cannot connect to slave`

Solution: Ensure network connectivity and proper slave configuration.

```
# Check network connectivity
ping slave-hostname

# Configure slave node
Manage Jenkins > Manage Nodes and Clouds > New Node > Configure
```

Error: `Slave node offline`

Solution: Restart the slave agent and verify the configuration.

```
# Restart slave agent
ssh into the slave and restart the Jenkins agent service

# Verify slave configuration
Manage Jenkins > Manage Nodes and Clouds > Select Node > Configure
```

28. Jenkins Credential Management Issues

Error: `Invalid credentials`

Solution: Verify the credentials and update them if necessary.

```
# Update credentials
Manage Jenkins > Manage Credentials > Select Credentials > Update

# Verify credential usage in job
Job Configuration > Source Code Management > Credentials
```

Error: `Credentials not found`

Solution: Ensure the credentials are correctly defined and accessible.

```
# Define credentials
Manage Jenkins > Manage Credentials > Add Credentials

# Use credentials in Jenkinsfile
pipeline {
    agent any
    stages {
        stage('Checkout') {
            steps {
                git credentialsId: 'my-credentials-id', url:
'https://github.com/user/repo.git'
            }
        }
    }
}
```

29. Jenkins Webhook Issues

Error: `Webhook not triggering jobs`
Solution: Verify the webhook configuration and ensure Jenkins is accessible.

```
# Verify webhook configuration
GitHub/Bitbucket Repository > Settings > Webhooks

# Ensure Jenkins is accessible
http://jenkins-url/github-webhook/
```

Error: `Invalid webhook payload`
Solution: Check the payload format and ensure it matches the expected structure.

```
# Example webhook payload
{
  "ref": "refs/heads/main",
  "before": "old_commit_id",
  "after": "new_commit_id"
}

# Verify Jenkins webhook configuration
Manage Jenkins > Configure System > GitHub/Bitbucket Webhook
```

30. Jenkins Backup and Restore Issues

Error: `Backup not created`
Solution: Ensure the backup directory is writable and the backup plugin is configured correctly.

```
# Verify backup directory permissions
chmod 755 /backup/jenkins

# Configure backup plugin
Manage Jenkins > ThinBackup > Settings
```

Error: `Restore process failed`
Solution: Ensure the backup files are intact and follow the correct restore process.

```
# Verify backup files
ls /backup/jenkins

# Restore Jenkins from backup
Manage Jenkins > ThinBackup > Restore > Select Backup > Restore
```

31. Jenkins Job Execution Issues

Error: `Job hangs during execution`

Solution: Identify the step causing the hang and increase verbosity or timeout.

```
# Increase verbosity
sh 'command --verbose'

# Set timeout in Jenkinsfile
pipeline {
    agent any
    options {
        timeout(time: 30, unit: 'MINUTES')
    }
    stages {
        stage('Build') {
            steps {
                sh 'command'
            }
        }
    }
}
```

Error: `Job aborted`

Solution: Check for manual intervention or system limits.

```
# Check for manual abort
Build History > Select Build > Console Output

# Review system limits
ulimit -a
```

32. Jenkins Artifact Storage Issues

Error: `Unable to store artifacts`

Solution: Verify the storage path and ensure there is sufficient space.

```
# Verify storage path
Manage Jenkins > Configure System > Artifact Storage

# Check disk space
df -h
```

Error: `Artifacts corrupted`

Solution: Ensure the artifacts are correctly generated and stored.

```
# Generate artifacts
mvn package
```

```
# Archive artifacts in Jenkinsfile
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                sh 'mvn package'
                archiveArtifacts artifacts: '**/target/*.jar'
            }
        }
    }
}
```

33. Jenkins Plugin Compatibility Issues

Error: `Plugin incompatible with Jenkins version`
Solution: Update Jenkins and plugins to compatible versions.

```
# Update Jenkins
Manage Jenkins > Manage Plugins > Updates > Upgrade Jenkins

# Update incompatible plugins
Manage Jenkins > Manage Plugins > Updates
```

Error: `Plugin dependencies missing`
Solution: Install all required dependencies for the plugin.

```
# Install dependency plugins
Manage Jenkins > Manage Plugins > Available > Select Dependency Plugins >
Install

# Verify installed plugins
Manage Jenkins > Manage Plugins > Installed
```

34. Jenkins Slave Node Issues

Error: `Slave node not starting`
Solution: Check the slave node configuration and ensure the necessary permissions.

```
# Verify slave node configuration
Manage Jenkins > Manage Nodes and Clouds > Configure

# Set appropriate permissions
chmod +x /path/to/slave.jar
chown jenkins:jenkins /path/to/slave.jar
```

Error: `Slave node disconnects frequently`
Solution: Check network stability and adjust the connection settings.

```
# Check network stability
ping -c 10 slave-hostname

# Adjust connection settings
Manage Jenkins > Manage Nodes and Clouds > Configure > Advanced > Retention
Strategy > Keep agent online
```

35. Jenkins Build Trigger Issues

Error: `Build triggers not working`
Solution: Verify the trigger configuration and ensure the triggering events are correctly set.

```
# Configure build triggers
Job Configuration > Build Triggers > Poll SCM or Build Periodically

# Ensure triggering events are set
GitHub/Bitbucket Repository > Settings > Webhooks > Configure
```

Error: `SCM polling not working`
Solution: Ensure the SCM repository is accessible and the polling schedule is correct.

```
# Verify SCM repository URL
https://github.com/user/repo.git

# Set polling schedule
Job Configuration > Build Triggers > Poll SCM > Schedule
```

36. Jenkins Security Configuration Issues

Error: `Unauthorized access detected`
Solution: Review and update security settings.

```
# Configure security settings
Manage Jenkins > Configure Global Security > Security Realm

# Set proper access control
Manage Jenkins > Manage and Assign Roles > Assign Roles
```

Error: `CSRF protection issues`
Solution: Configure CSRF protection correctly or disable it if necessary.

```
# Configure CSRF protection
Manage Jenkins > Configure Global Security > CSRF Protection

# Disable CSRF protection (if needed)
Manage Jenkins > Configure Global Security > Uncheck "Prevent Cross Site
Request Forgery exploits"
```

37. Jenkins Logging Issues

Error: `Insufficient logging information`
Solution: Increase log verbosity and configure log settings.

```
# Increase log verbosity
Manage Jenkins > System Log > Add new log recorder

# Configure log settings
Manage Jenkins > Configure System > Logging
```

Error: `Logs not rotating`

Solution: Configure log rotation and ensure sufficient storage.

```
# Set up log rotation
Manage Jenkins > Configure System > Log Rotation

# Check storage space
df -h
```

38. Jenkins Script Security Issues

Error: `Script approval required`

Solution: Approve the script in the Script Approval section.

```
# Approve scripts
Manage Jenkins > In-process Script Approval > Approve

# Example Jenkinsfile with scripts
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                script {
                    def output = sh(script: 'command', returnStdout: true)
                    println(output)
                }
            }
        }
    }
}
```

Error: `Security exceptions`

Solution: Ensure scripts comply with security policies and approve necessary signatures.

```
# Review and approve security exceptions
Manage Jenkins > In-process Script Approval

# Ensure script compliance
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                script {
                    try {
                        sh 'command'
                    } catch (Exception e) {
                        currentBuild.result = 'FAILURE'
                        throw e
                    }
                }
            }
        }
    }
```

```
}
```

39. Jenkins Performance Tuning Issues

Error: `High CPU usage`
Solution: Optimize Jenkins configurations and manage system resources.

```
# Optimize Jenkins configurations
Manage Jenkins > Configure System > Resource Management

# Manage system resources
top
```

Error: `Memory leaks`
Solution: Identify memory leaks and increase heap size.

```
# Increase heap size
JENKINS_JAVA_OPTIONS="-Xmx4g"

# Monitor for memory leaks
jmap -histo:live <Jenkins PID>
```

40. Jenkins Webhook Delivery Issues

Error: `Webhook payload delivery failed`
Solution: Check Jenkins accessibility and webhook configurations.

```
# Ensure Jenkins is accessible
http://jenkins-url/github-webhook/

# Verify webhook configurations
GitHub/Bitbucket Repository > Settings > Webhooks > Configure
```

Error: `Invalid webhook signature`
Solution: Verify the secret token and payload signature.

```
# Verify secret token
GitHub/Bitbucket Repository > Settings > Webhooks > Secret

# Check payload signature
Manage Jenkins > Configure System > GitHub/Bitbucket Webhook
```

41. Jenkins Authentication Issues

Error: `LDAP authentication failed`
Solution: Verify LDAP server configuration and credentials.

```
# Configure LDAP
Manage Jenkins > Configure Global Security > Security Realm > LDAP

# Verify LDAP server settings
Server: ldap://ldap.example.com
```

```
Root DN: dc=example,dc=com
```

Error: `Active Directory authentication failed`

Solution: Ensure the Active Directory configuration is correct.

```
# Configure Active Directory
Manage Jenkins > Configure Global Security > Security Realm > Active
Directory

# Verify Active Directory settings
Domain: example.com
Bind DN: CN=admin,OU=Users,DC=example,DC=com
```

42. Jenkins Pipeline Shared Library Issues

Error: `Shared library not found`

Solution: Ensure the library repository is accessible and correctly referenced.

```
# Configure shared library
Manage Jenkins > Configure System > Global Pipeline Libraries

# Reference shared library in Jenkinsfile
@Library('shared-library') _
```

Error: `Library loading error`

Solution: Check the library configuration and update the repository.

```
# Verify library configuration
Library Configuration > SCM > Git > Repository URL

# Update library repository
git pull origin main
```

43. Jenkins Artifact Deployment Issues

Error: `Deployment failed`

Solution: Check the deployment script and environment configuration.

```
# Verify deployment script
sh 'scp target/app.war user@server:/path/to/deploy/'

# Ensure environment configuration
export DEPLOY_PATH=/path/to/deploy/
```

Error: `Artifact not found for deployment`

Solution: Ensure the artifact is correctly generated and archived.

```
# Generate and archive artifacts
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                sh 'mvn package'
```

```
                archiveArtifacts artifacts: '**/target/*.jar'
            }
        }
    }
}
```

44. Jenkins Groovy Script Sandbox Issues

Error: `Scripts not permitted to use method`
Solution: Approve the necessary methods in the script approval section.

```
# Approve script methods
Manage Jenkins > In-process Script Approval > Approve

# Example of script requiring approval
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                script {
                    def output = sh(script: 'command', returnStdout: true)
                    println(output)
                }
            }
        }
    }
}
```

Error: `RejectedAccessException`
Solution: Adjust the script to comply with the sandbox restrictions or approve the script.

```
# Adjust script to comply with sandbox
@NonCPS
def myMethod() {
    // Non-sandboxed code
}

# Approve script in Jenkins
Manage Jenkins > In-process Script Approval > Approve
```

45. Jenkins Email Notification Issues

Error: `Email not sent`
Solution: Verify email server configuration and credentials.

```
# Configure email notification
Manage Jenkins > Configure System > Email Notification

# Verify SMTP settings
SMTP server: smtp.example.com
```

Error: `Email format issues`

Solution: Ensure the email template is correctly defined.

```
# Configure email template
Manage Jenkins > Configure System > Extended E-mail Notification > Default
Content

# Example email template
Subject: Build ${BUILD_STATUS}: ${JOB_NAME} ${BUILD_NUMBER}
Body: Build ${BUILD_STATUS}: ${JOB_NAME} ${BUILD_NUMBER}\n\nCheck console
output at ${BUILD_URL}
```

46. Jenkins API Token Issues

Error: `API token invalid`

Solution: Regenerate the API token and update the references.

```
# Regenerate API token
Manage Jenkins > Manage Users > User > Configure > API Token > Generate

# Update references with new API token
```

Error: `API token not recognized`

Solution: Ensure the API token is correctly used in scripts or integrations.

```
# Use API token in script
curl -u user:api-token http://jenkins-url/job/job-name/build

# Verify API token in integration settings
```

47. Jenkins Configuration as Code (JCasC) Issues

Error: `JCasC YAML syntax error`

Solution: Validate the YAML configuration file.

```
# Validate YAML syntax
yamllint jenkins.yaml

# Example JCasC configuration
jenkins:
  systemMessage: "Welcome to Jenkins"
  securityRealm:
    local:
      allowsSignup: false
      users:
        - id: "admin"
          password: "${JENKINS_ADMIN_PASSWORD}"
```

Error: `JCasC configuration not applied`

Solution: Ensure the JCasC plugin is installed and correctly configured.

```
# Install JCasC plugin
Manage Jenkins > Manage Plugins > Available > Configuration as Code >
Install
```

```
# Apply JCasC configuration
java -jar jenkins.war --argumentsRealm.passwd.admin=admin --
argumentsRealm.roles.admin=admin --httpPort=8080 --config jenkins.yaml
```

48. Jenkins Database Integration Issues

Error: `Database connection failed`

Solution: Verify database configuration and credentials.

```
# Configure database connection
Manage Jenkins > Configure System > Database

# Verify database settings
Database URL: jdbc:mysql://localhost:3306/jenkins
Database Username: jenkins
```

Error: `Database query error`

Solution: Check the query syntax and ensure the database is accessible.

```
# Verify query syntax
SELECT * FROM jobs;

# Ensure database is accessible
mysql -u jenkins -p -h localhost -D jenkins
```

49. Jenkins Pipeline Parameters Issues

Error: `Parameter not found`

Solution: Ensure the parameters are correctly defined in the Jenkinsfile.

```
# Define parameters in Jenkinsfile
pipeline {
    agent any
    parameters {
        string(name: 'ENV', defaultValue: 'dev', description:
'Environment')
    }
    stages {
        stage('Build') {
            steps {
                sh 'echo Building for environment: ${ENV}'
            }
        }
    }
}
```

Error: `Parameter value not passed`

Solution: Ensure the parameter value is correctly passed during job execution.

```
# Pass parameter value during job execution
curl -X POST http://jenkins-url/job/job-name/buildWithParameters?ENV=prod

# Verify parameter usage in Jenkinsfile
pipeline {
    agent any
```

```
    parameters {
        string(name: 'ENV', defaultValue: 'dev', description:
'Environment')
    }
    stages {
        stage('Build') {
            steps {
                sh 'echo Building for environment: ${ENV}'
            }
        }
    }
}
```

50. Jenkins Job Log Issues

Error: `Log file not found`
Solution: Ensure the job configuration includes logging and check the log directory.

```
# Configure job logging
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                sh 'command'
                archiveArtifacts artifacts: 'logs/*.log'
            }
        }
    }
}
```

```
# Check log directory
ls /var/lib/jenkins/jobs/job-name/builds/lastSuccessfulBuild/log
```

Error: `Log file too large`
Solution: Configure log rotation and limit the log size.

```
# Configure log rotation
Manage Jenkins > Configure System > Log Rotation
```

```
# Limit log size in job configuration
pipeline {
    agent any
    stages {
        stage('Build') {
            steps {
                sh 'command'
                archiveArtifacts artifacts: 'logs/*.log'
                sh 'truncate -s 10M logs/*.log'
            }
        }
    }
}
```