* Setup and Running Jenkins
* Jenkins Overview (Lab: Installing a Jenkins Environment on Red Hat / Windows)
* Installing and Configuring a Jenkins Cluster (Lab: Installing the Jenkins Environment)
* Command Line Interface
* Configuring a Node
* Running Jenkins as Stand-Alone/ Apache Server
* Backup Jenkins data
* Manage Jenkins (People access control , Authentication )
* Configuration of Jenkins Server
* Configure Dashboard
* Configure System Environment
* Global Properties
* Configure Build Tools
* Configure Proxy
* Continuous Integration
* Install Git and Jenkins GitHub Plugin
* Configure Jenkins to Work with Java, Git and Maven
* Create and Configure a job
* Parameterized jobs setup
* Run a job manually
* Triggering a Build
* Scheduled Build job
* Manual Build job
* Polling SCM
* Maven and ANT Build Step
* Execute a Shell
* Post-Build Actions
* Archiving Build Results
* Email and other notifications
* Continuous Inspection with Jenkins
* Automating Unit and Integration Tests
* Configuring Test Reports in Jenkins
* Code Coverage Metrics Report
* Activating more PMD/CPD Rulesets, Creating Custom PMD/CPD Rules
* Finding Bugs with FindBugs
* Finding Security Defects with FindBugs
* Quality check reports
* Enforcing Minimum Quality Scores
* Integrating with Sonar (Sonar reports on code coverage, rule compliance, and documentation.)
* Continuous Delivery with Jenkins
* Archive Build Artifacts
* Deploy to Staging Environment
* Trouble Shooting: Deploy to Staging
* Release Management with Nexus
* Release Management with AWS S3
* Release Management with Apache Archiva
* Deployment using Jenkins + Nexus/S3 +Software configuration management tool like SALT and Ansible
* Automated Deployment and Continuous Delivery
* Jenkins Build Pipeline
* Parallel Jenkins Build
* Deploy to Production
* Trouble Shooting: Deploy to Production
* Deploying Scripting-based Applications Like PHP and Python on staging and production
* Calling web services after deployment (Like flushing CDN Data)
* Working with Automate Testing
* Automated Functional and Acceptance Tests
* Automated Performance Tests with JMeter
* Create Pipeline Project
* Pipeline as Code
* Automate existing Jenkins Pipeline
* Trouble Shooting: Automate existing Jenkins Pipeline
* Fully Automated Jenkins Pipeline
* Jenkins's Integration with Docker
* Jenkins pipelines with Docker
* Build, test, and run everything in Docker containers
* Advanced Jenkins
* Monitoring External jobs
* Distributed builds
* File fingerprint tracking
* Creating Templates
* Plugin Management
* Security best practices and Notification
* Best Practices on Jenkins