Orchard Training

DevOps

Course Curriculum

Document Version / Details: Ver. 2.1/ 02-Jan-2023

Topic & Skill	Content Coverage
DevOps Linux Essentials	o OS Introduction o Linux File structure o Basic Operational Commands 1. Files and directories 2. documentation - man pages 3. VI/nano Editor 4. Shell Scripting - Variables Loops and conditions 5. Controlling Processes 6. Utility commands like find, grep, sort, head, tail, tn, ftp, tar. 7. Conditions and Loops 8. Package management - Ubuntu, RHEL & Amazon Linux 9. linux based OS - RHEL, Ubuntu, Debian etc *#All essential linux commands required for the 30 day training program*
Introduction to DevOps	o Basic Programming + Compiling + Packaging + Release o Define DevOps o What is DevOps o SDLC models, Lean, ITIL, Agile o Why DevOps? o DevOps Goals & Benefits o DevOps CI/CD Process o DevOps Tools o Source Code Management o Purpose of Build & Release process o Configuration management o Monitoring – Significance and how to o What is DevSecOps o What is SRE o Difference between SRE & DevOps o SRE - Key terminologies
GIT : Version Control Maven Build Tool	o Introduction o What is Git o About Version Control System and Types o Difference between CVCS and DVCS o GIT Basics o GIT Command Line o Branching & Merging Strategies o Different type of GIT based tools Git Hands On: o Creating repository o Cloning, check-in and committing o Fetch pull and remote o Branching o Creating the Branches, switching the branches, merging the branches. Introduction to Maven POM file structure Dependency management Build jar/war files using Maven build

Topic & Skill	Content Coverage
Jenkins CI/CD Orchestration	o Understanding Continuous Integration Vs Continuous Delivery Vs Continuous Deployment o Jenkins installation - Windows, Linux, CLoud Platforms o Introduction about Jenkins o Jenkins Architecture o Build Cycle - Java o Jenkins – Plugins o Pipeline as Code – Declarative(default) Vs Scripted
Jenkins CI/CD Orchestration	Jenkins Hands On: o Jenkins hands on o master slave configuration o Creating Jobs - Declarative pipeline o Parameterized Jobs o Running the Jobs o Setting up the global environments for Jobs o Adding and updating Plugins o Disabling and deleting jobs
Introduction to AWS and DevOps Principles	o Introduction to AWS Services - Overview of Key AWS services used in DevOps, including compute, storage, databases, networking and more. o DevOps Culture and Principles - Understanding the culturalshift, Collaboration, automation, measurement and sharing (CAMS) principles of DevOps. o Version Control and Collaboration - Using Git for version control and colloboration on code Lab exercices to cover basic ussecases of EC2, App Hosting, S3, IAM, SNS, SQS, Route 53, VPC, Cloud Formation, CloudFront, AMI Snapshots, Static IP, Elastic Block storage etc.
AWS - Continous Integration and Continous Deployment	o Building CI/CD Pipelines - Setting up end to end CI/CD pipelines using AWS CodePipeline, AWS COdeBuild and AWS CodeDeploy. o Automated Testing - Integrating testing frameworks into the CI/CD pipeline for ensuring code quality - Sonar and JUNIT o Blue-Green Deployments - Implementing Blue Green deployment stratergies for minimising downtime during releases. Lab exercices to cover basic use cases for AWS CodePipeline, AWS COdeBuild and AWS CodeDeploy.
AWS - Scalability, Security and Monitoring	o Scalable strategies - Designing scalable architectures using AWS autoscaling, Elastic Load Balancing and serverless services. o Security Best Practices - Managing security through AWS IAM, security groups, encryption. o Monitoring and Observability - Implementing monitoring using Amazon CloudWatch Optional - other observability tools like prometheus and grafana. Lab exercices to cover basic usecases on ELB, ASG, IAM, CloudWatch

Topic & Skill	Content Coverage
AWS - Infrastructure as Code and Automation	o Infrastructure as Code (IaC) Fundamentals - In-depth look at IaC concepts using AWS CloudFormation o IaC Best Practices - Designing reusable and modular infrastructure components using templates o Automation with AWS Services - Implementing automated workflows using AWS Lambda, Step functions and other automation services. Lab exercices to cover basic usecases on CloudFormation, AWS Lambda, Step Functions
AWS -Advanced topics and Best Practices	o Advanced containerization - In-depth exploration of Docker, Kubernetes, Amazon ECS and Amazon EKS. (Moved to week three) o Serverless Architecture - Detailed look at serverless concepts using AWS Lambda, API Gateway and event-driven architectures - SNS SQS o Cost Optimization Stratergies - Understanding Cost allocation, budgeting and Optimizing resource usage using AWS Services. o DevOps Tools and Ecosystem - Exploring integration of DevOps Tools - Jenkins, Git, maven in AWS. Lab exercices to cover basic usecases on Docker, Kubernetes, ECS, EKS, AWS Lambda, API Gateway and event driven architectures.
Docker– Containers	o Introduction o What is a Docker & Containerisation o Understanding the Docker components o Platforms for Docker o installation o Creating containers o Docker compose hands-on o Create different environments with Docker o Build Containerized pipeline using Docker for Sample Web application in Jenkins
Kubernetes - Container Orchestration Platform Fundamentals	o Introduction to Kubernetes o Container Orchesrtration concepts o Kubernetes architecture o Setting up kubernetes cluster o Kubernetes Objects o Deploying Applications o Scaling and Autoscaling
Kubernetes - Container Orchestration Platform Best Practices and Operations	o Persistent Storage o ConfigMaps and secrets o Ingress Controllers o Network Policies o Deployments o Statefulsets o Custom resource Definition
Kubernetes - Container Orchestration Platform Best Practices and Operations	o Kubernetes Best Practices o Monitoring and Logging o Kubernetes Security o Cluster Backup and Disaster Recovery - Conceptual Knowledge o CI/CD Integration o Kubernetes Troubleshooting o Statefulsets o Helm o Amazon ECS and Amazon EKS. (moved from week 2)

Topic & Skill	Content Coverage
Kubernetes - Container Orchestration Platform Advanced Kubernetes concepts	o Kubernetes Advance Topics o Deployements using ArgoCD
Ansible - Config Management	o What is Ansible o Why Ansible o Basic Ansible Terminology o Infra as Code o Introduction to Playbooks o Playbook Structure o Introduction to Modules o Variables and Facts o Ansible Configuration Hierarchy o Ansible in the Cloud Hands-on: o Creating an Ansible Home Base o Setting up Test Environment o Creating Playbooks o Jenkins integration o Trigger playbooks form Jenkins
Project Milestone 2 - Create an end to end CI/CD pipeline in AWS platform using Jenkins as the orchestration tool, Github as the SCM, Maven as the Build tool, Deploy in a docker instance and create a Docker image, Store the docker	
Python Fundamentals	o Introduction to Python o Python Basics - Variables, Basic IO, Operators o Control Structures o Functions o List and Tuples o Dictionaries o File Handling o Exception Handling o Modules and Packages o Object Oriented Programming o Introduction to Libraries (Optional) o Practice Examples - Scripting exercises and examples throughout the day to reinforce concepts.
Azure Fundamentals	o Introduction to Microsoft Azure o Azure Services and Solutions o Azure Portal and Azure Resource Manager o Azure Virtual Machines o Azure Storage o VPC + Azure Networking (Optional) o Azure Identity and access Management o Azure App Services o Monitoring and Management (Optional) o Security and compliance (Optional) o Cost Management and Optimisation (Optional)
Azure DevOps	o Introduction to Azure DevOps o Version control with Azure Repos o Azure Pipelines CI/CD Fundamentals o Building CI Pipelines o Deploying applications with CD Pipelines Lab: o Building CI Pipelines o Deploying using CD Pipelines

Topic & Skill	Content Coverage
Terraform Basics	Understanding Infrastructure as Code (IaC) o Introduction to Infrastructure as Code (IaC) o Why Terraform? o Terraform vs. other IaC tools o Key concepts: Declarative vs. imperative, Desired State Configuration (DSC), Idempotency Getting Started with Terraform o Installing Terraform o Initializing a Terraform project o Terraform configuration language (HCL) o Terraform workflow: plan, apply, destroy o Managing state files Terraform Fundamentals o Providers and resources o Variables and data types o Outputs o Modules: organization and reusability o Terraform state management: remote state, backends HANDS On-Lab o Setting up a simple infrastructure with Terraform o Deploying and managing resources on a cloud provider (e.g., AWS, Azure, Google Cloud)
Terraform Advanced	Advanced Terraform Configuration o Dependency management o Workspaces o Remote execution with Terraform Cloud/Enterprise o Using provisioners and local-exec o Understanding count, for_each, and dynamic blocks Terraform in Production o Best practices for Terraform code organization o Security considerations o Continuous Integration/Continuous Deployment (CI/CD) pipelines with Terraform o Managing Terraform with version control (e.g., Git) Terraform Modules and Collaboration o Building and publishing modules o Using community modules o Collaborative Terraform development with Git and version control o Terraform Enterprise features for team collaboration Troubleshooting and Optimization o Debugging Terraform configurations o Terraform graph and plan inspection o Performance optimization techniques o Handling Terraform errors and common pitfalls
GitHub actions	o Introduction to Github actions o Getting started with Github actions o Basic workflows o Building and testing with Github actions o Continous Deployment with GitHub Actions

Topic & Skill	Content Coverage
Introduction to Site Reliability Engineering	o Introduction to Site Reliability Engineering o SRE VS DevOps o SLIs, SLOs and Error Budgets o Monitoring and alerting o Incident management and Post Mortems o Load Balancing and traffic Management o Capacity Planning and Auto Scaling o Disaster reecovery and Auto-Scaling
SRE Advanced	o Chaos Engineering o Automation and Infrastructure as Code o Distributed Monitoring and Observability o Performance Optimization and Efficiency o Managing Microservices and Containers o Continous improvement and SRE culture
Splunk Essentials	o Introduction to Splunk o Benefits of Splunk in DevOps o Splunk Architecture o Search Processing Language (SPL) o Splunk Search Interface o Data Visualization o Hands-on Labs
Lab Use Cases / POC	o Revison and practice of concepts through hands-on exercisies.

LTIMindtree

Let’s get to the future, faster. Together.