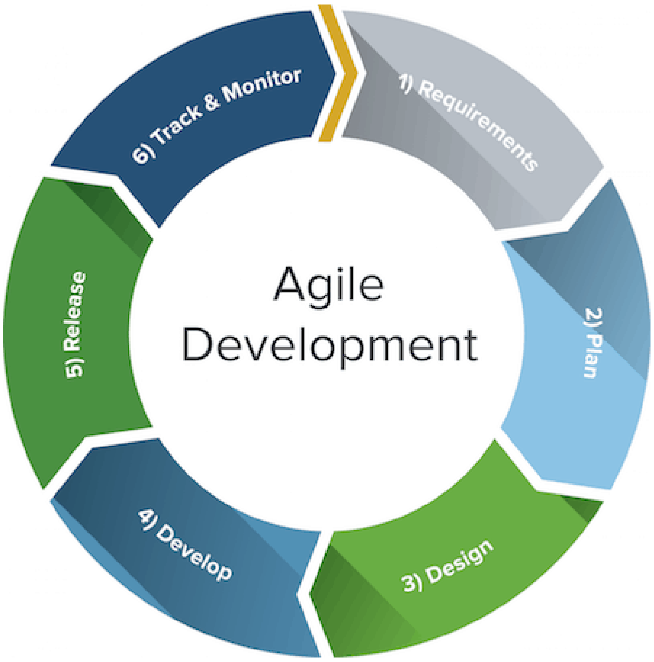
Full Stack Developers Certified Professional (FSDCP)

About
DevOpsSchool

DevOpsSchool is a unit of "Cotocus PVT ltd" and a leading platform which helps IT organizations and professionals to learn all the emerging technologies and trend which helps them to learn and embrace all the skills, intelligence, innovation and transformation which requires to achieve the end result, quickly and efficiently. We provide over 40 specialized programs on DevOps, Cloud, Containers, Security, AI, ML and on Big data that are focused on industry requirement and each curriculum is developed and delivered by leading experts in each domain and aligned with the industry standards.

What is Agile QA?

Agile is a modern approach of software development process which overcomes the challenges of earlier working models. The agile QA process begins at the inception of the software development life cycle, from the initial design meeting, through the development phase, to final testing and hardening of the application. In this program, you will get an in-depth knowledgeof the Agile QA.



Benefits of Agile QA:-

- Improved Flexibility to react to competitors.
- Increased frequency of new feature deliveries.
- It is flexible and highly adaptable to changes.
- Improved customer experience by reducing likelihood of encountering a defect.
- Improved ability to test granular level functions without impacting on the product.

Co-coordinator – Akanksha Kumari

Call/WhatsApp: - +91 1800 889 7977

Mail Address: -

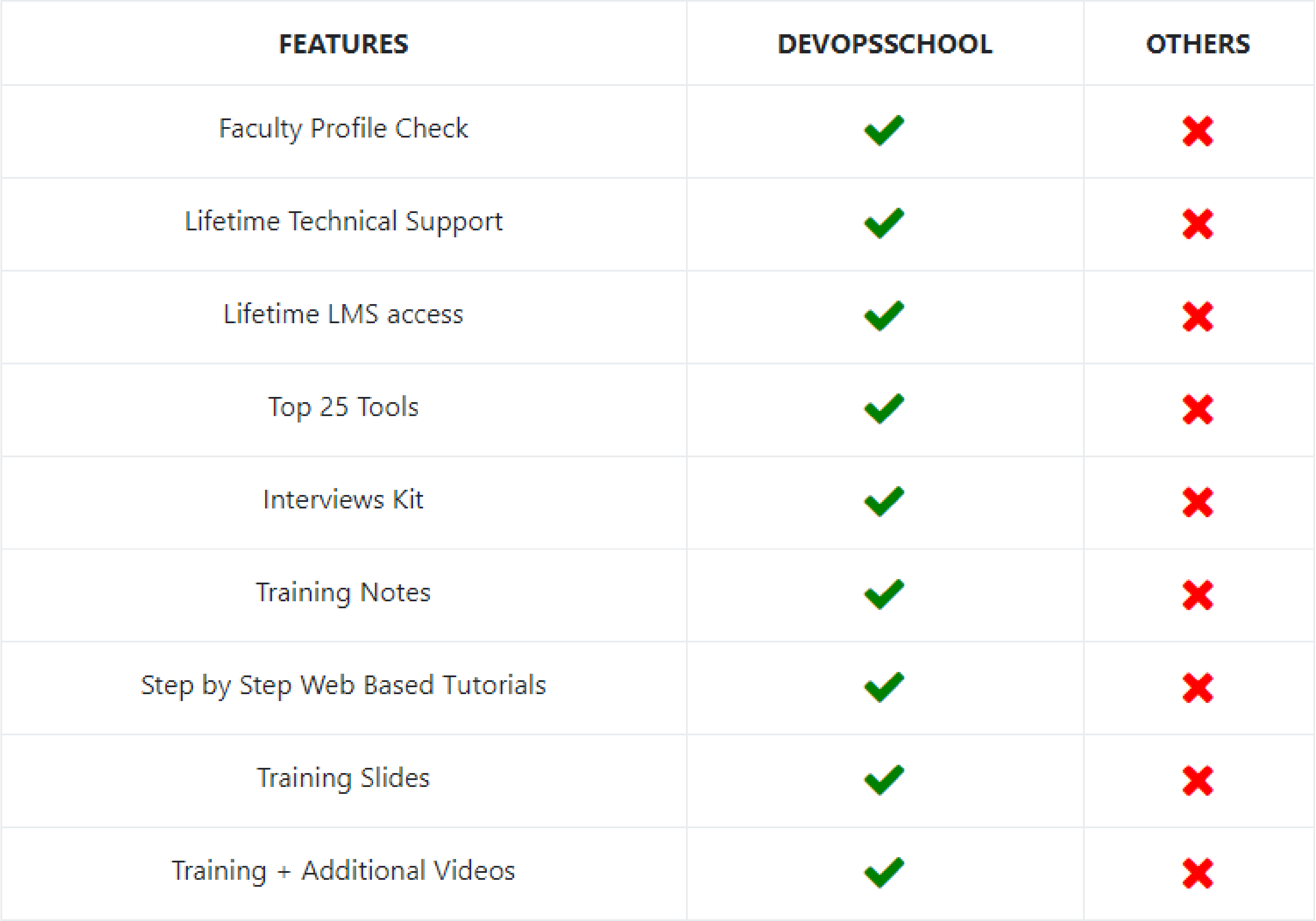
contact@DevOpsSchool.com

Secondary contact – Patrick

Call/WhatsApp: - +91 7004 215 841

Mail Address: -contact@DevOpsSchool.com

Duration	60 Hours
Mode	Online (Instructor-led, live & Interactive)
Projects (Real time scenario based)	1



FEATURES	DEVOPSSCHOOL	OTHERS
Faculty Profile Check	✔	✘
Lifetime Technical Support	✔	✘
Lifetime LMS access	✔	✘
Top 25 Tools	✔	✘
Interviews Kit	✔	✘
Training Notes	✔	✘
Step by Step Web Based Tutorials	✔	✘
Training Slides	✔	✘
Training + Additional Videos	✔	✘

Training

DevOps As part of this course, you would be strong in DevOps technology. You would learn Linux, Python, DevOps, Docker, Jira, Git, SonarQube, Maven, Ansible, Jenkins, Kubernetes, Datadog, Splunk, NewRelic, Terraform and various other stacks related to this methodology.

Projects

As part of this initiative, trainer would help you to execute one real time scenario based project, doing it end to end and step by step to visualize a real agile work environment in any organization.

Interview

As part of this, you would give complete Agile Developers interview preparations Kit. This interview kit will help you organize your application and interview with eas

AGENDA OF THE FULL STACK DEVELOPERS

SDLC Models & Architecture with Agile, DevOps, SRE & DevSecOps, SOA & Micro services - Concept

- Let’s Understand about Software Development Model
- Overview of Waterfall Development Model
- Challenges of Waterfall Development Model
- Overview of Agile Development Model
- Challenges of Agile Development Model
- Requirement of New Software Development Model
- Understanding an existing Pain and Waste in Current Software Development Model
- What is DevOps?
- Transition in Software development model
    - Waterfall -> Agile -> CI/CD -> DevOps -> DevSecOps
- Understand DevOps values and principles
- Culture and organizational considerations
- Communication and collaboration practices
- Improve your effectiveness and productivity
- DevOps Automation practices and technology considerations
- DevOps Adoption considerations in an enterprise environment
- Challenges, risks and critical success factors
- What is DevSecOps?
    - Let’s Understand DevSecOps Practices and Toolsets.
- What is SRE?
    - Let’s Understand SRE Practices and Toolsets.
- List of Tools to become Full Stack Developer/QA/SRE/DevOps/DevSecOps
- Microservices Fundamentals
- Microservices Patterns
    - Choreographing Services
    - Presentation components
    - Business Logic

  - Database access logic
  - Application Integration
  - Modelling Microservices
  - Integrating multiple Microservices

- Keeping it simple
  - Avoiding Breaking Changes
  - Choosing the right protocols
  - Sync & Async
  - Dealing with legacy systems
  - Testing
- What and When to test
- Preparing for deployment
- Monitoring Microservices Performance
- Tools used for Microservices Demo using container

Platform - Operating Systems - Centos/Ubuntu & VirtualBox & Vagrant

Microservices Fundamentals
Microservices Patterns

- Installing CentOS7 and Ubuntu
- Accessing Servers with SSH
- Choreographing Services
- Presentation components
- Business Logic
- Database access logic
- Application Integration
- Modelling Microservices
- Integrating multiple Microservices

What and When to test

Preparing for deployment

Monitoring Microservice Performance
Tools used for Microservices Demo using container
Keeping it simple

- Avoiding Breaking Changes
- Choosing the right protocols
- Sync & Async
- Dealing with legacy systems
- Testing

Platform - Operating Systems - Centos and Ubuntu

- Installing CentOS7 and Ubuntu
- Accessing Servers with SSH
- Working at the Command Line
- Reading Files
- Using the vi Text Editor
- Piping and Redirection
- Archiving Files
- Accessing Command Line Help
- Understanding File Permissions
- Accessing the Root Account
- Using Screen and Script

Platform - Virtualization & Systems - VirtualBox & Vagrant

- Overview of Hypervisor
- Introduction of VirtualBox
- Install VirtualBox and Creating CentOS7 and Ubuntu Vms
- Understanding Vagrant
- Basic Vagrant Workflow
- Advance Vagrant Workflow
- Working with Vagrant VMs
- The Vagrantfile
- Installing Nginx
- Provisioning
- Networking
- Sharing and Versioning Web Site Files
- Vagrant Share
- Vagrant Status
- Sharing and Versioning Nginx Config Files
- Configuring Synced Folders

Platform - Cloud - AWS

- Introduction of AWS
- Understanding AWS infrastructure
- Understanding AWS Free Tier
- IAM: Understanding IAM Concepts
- IAM: A Walkthrough IAM
- IAM: Demo & Lab
- Computing:EC2: Understanding EC2 Concepts
- Computing:EC2: A Walkthrough EC2
- Computing:EC2: Demo & Lab
- Storage:EBS: Understanding EBS Concepts
- Storage:EBS: A Walkthrough EBS
- Storage:EBS: Demo & Lab
- Storage:S3: Understanding S3 Concepts
- Storage:S3: A Walkthrough S3
- Storage:S3: Demo & Lab
- Storage:EFS: Understanding EFS Concepts
- Storage:EFS: A Walkthrough EFS
- Storage:EFS: Demo & Lab
- Database:RDS: Understanding RDS MySql Concepts
- Database:RDS: A Walkthrough RDS MySql
- Database:RDS: Demo & Lab
- ELB: Elastic Load Balancer Concepts
- ELB: Elastic Load Balancer Implementation
- ELB: Elastic Load Balancer: Demo & Lab
- Networking:VPC: Understanding VPC Concepts
- Networking:VPC: Understanding VPC components
- Networking:VPC: Demo & Lab

Platform - Containers - Docker

- What is Virtualization?
- What is Containerization?
- Why Containerization?
- How Docker is good fit for Containerization?
- How Docker works?
- Docker Architecture
- Docker Installations & Configurations
- Docker Components
- Docker Engine
- Docker Image
- Docker Containers
- Docker Registry
- Docker Basic Workflow
- Managing Docker Containers
- Creating our First Image
- Understading Docker Images
- Creating Images using Dockerfile
- Managing Docker Images
- Using Docker Hub registry
- Docker Networking
- Docker Volumes
- Deepdive into Docker Images
- Deepdive into Dockerfile
- Deepdive into Docker Containers
- Deepdive into Docker Networks
- Deepdive into Docker Volumes
- Deepdive into Docker Volume
- Deepdive into Docker CPU and RAM allocations
- Deepdive into Docker Config
- Docker Compose Overview
- Install & Configure Compose
- Understanding Docker Compose Workflow
- Understanding Docker Compose Services
- Writing Docker Compose Yaml file
- Using Docker Compose Commands
- Docker Compose with Java Stake
- Docker Compose with Rails Stake
- Docker Compose with PHP Stake
- Docker Compose with Nodejs Stake

Planning and Designing - Jira & Confulence

a. Jira

- Overview of Jira
- Use cases of Jira
- Architecture of Jira
- Installation and Configuraration of Jira in Linux
- Installation and Configuraration of Jira in Windows
- Jira Terminologies
- Understanding Types of Jira Projects
- Working with Projects
- Working with Jira Issues
- Adding Project Components and Versions
- Use Subtasks to Better Manage and Structure Your Issues
- Link Issues to Other Resources
- Working in an Agile project
- Working with Issues Types by Adding/Editing/Deleting
- Working with Custom Fields by Adding/Editing/Deleting
- Working with Screens by Adding/Editing/Deleting
- Searching and Filtering Issues
- Working with Workflow basic
- Introduction of Jira Plugins and Addons.
- Jira Integration with Github

b. Confluence

- Exploring Confluence benefits and resources
- Configuring Confluence
- Navigating the dashboard, spaces, and pages
- Creating users and groups
- Creating pages from templates and blueprints
- Importing, updating, and removing content
- Giving content feedback
- Watching pages, spaces, and blogs
- Managing tasks and notifications
- Backing up and restoring a site

Admin tasks

- - Add/Edit/Delete new users
- - Adding group and setting permissions
- - Managing user permissions
- - Managing addons or plugins
- - Customizing confluence site

Installing Confluence

- - Evaluation options for Confluence
- - Supported platforms
- - Installing Confluence on Windows
- - Activating Confluence trial license
- - Finalizing Confluence Installation

Backend Programming Language 1 - Java

- Introduction
- Setup Java Development Environment
- Java Language Fundamental
- Arrays and Loops
- Java Methods
- Conditioning and Looping in Java
- Exception Handling
- Basics of Oops in Java
- Create a Java Package using Jar
- Creating Basic Java Web Based application
- Create a Java Package using War
- Deploy Web App in Tomcat Environment

Backend Programming Language 2 - Python

What is Python?

- Interpreted languages
- Advantages and disadvantages
- Downloading and installing
- Which version of Python
- Where to find documentation
- Running Python Scripts

Using variables

- String types: normal, raw and Unicode
- String operators and expressions
- Math operators and expressions
- Writing to the screen
- Command line parameters
- Reading from the keyboard
- Flow Control

Structure of a Python script

- Using the interpreter interactively
- Running standalone scripts under Unix and Windows
- Getting Started

About flow control

- Indenting is significant
- The if and elif statements
- while loops
- Using lists
- Using the for statement
- The range() function
- Sequence Data

List operations

- list methods
- Strings are special kinds of lists
- tuples
- sets
- Dictionaries
- Defining Functions

Text file I/O overview

- Opening a text file
- Reading text files
- Raw (binary) data
- Using the pickle module
- Writing to a text file
- Dictionaries and Sets

Syntax of function definition

- Formal parameters
- Global versus local variables
- Passing parameters and returning values
- Working with Files

Dictionary overview

- Creating dictionaries
- Creating dictionaries
- Fetching keys or values
- Testing for existence of elements
- Deleting elements
- Errors and Exception Handling

Source Code Versioning - Git using Github

What is Git?

- Understanding version control
- The history of Git
- About distributed version control
- Who should use Git?

Installing Git

- Installing Git on a Mac
- Installing Git on Windows
- Installing Git on Linux
- Configuring Git
- Exploring Git auto-completion
- Using Git help

Getting Started

- Initializing a repository
- Understanding where Git files are stored
- Performing your first commit
- Writing commit messages
- Viewing the commit log

Git Concepts and Architecture

- Exploring the three-trees architecture
- The Git workflow
- Using hash values (SHA-1)
- Working with the HEAD pointer

Making Changes to Files

- Adding files
- Editing files
- Viewing changes with diff
- Viewing only staged changes
- Deleting files
- Moving and renaming files
- Using Git with a Real Project
- Introducing the Explore California web site
- Initializing Git
- Editing the support phone number
- Editing the backpack file name and links

Undoing Changes

- Undoing working directory changes
- Unstaging files
- Amending commits
- Retrieving old versions
- Reverting a commit

- Using reset to undo commits
- Demonstrating a soft reset
- Demonstrating a mixed reset
- Demonstrating a hard reset
- Removing untracked files

Ignoring Files

- Using .gitignore files
- Understanding what to ignore
- Ignoring files globally
- Ignoring tracked files
- Tracking empty directories

Navigating the Commit Tree

- Referencing commits
- Exploring tree listings
- Getting more from the commit log
- Viewing commits
- Comparing commits

Branching

- Branching overview
- Viewing and creating branches
- Switching branches
- Creating and switching branches
- Switching branches with uncommitted changes
- Comparing branches
- Renaming branches
- Deleting branches
- Configuring the command prompt to show the branch

Merging Branches

- Merging code
- Using fast-forward merge vs. true merge
- Merging conflicts
- Resolving merge conflicts
- Exploring strategies to reduce merge conflicts
- Fast Forward Merge
- Three way Merge
- Rebasing

Stashing Changes

- Saving changes in the stash
- Viewing stashed changes
- Retrieving stashed changes
- Deleting stashed changes

Remotes

- Using local and remote repositories
- Adding a remote repository
- Creating a remote branch
- Cloning a remote repository
- Tracking remote branches

- Checking out remote branches
- Pushing to an updated remote branch
- Deleting a remote branch
- Enabling collaboration
- A collaboration workflow

Tools and Next Steps

- Setting up aliases for common commands
- Using SSH keys for remote login
- Exploring integrated development environments
- Exploring graphical user interfaces
- Understanding Git hosting
- Conclusion

Tagging

- Lightweight tags
- Annotated tags
- Signed Tags
- Create tags in Git
- Create tags in Stash
- Checkout tags in Git
- How to view tags
- How to checkout tags

Build Management - Gradle

- What is Gradle?
- Why Gradle?
- Installing and Configuring Gradle
- Build Java Project with Gradle
- Build C++ Project with Gradle
- Build Python Project with Gradle
- Dependency Management in Gradle
- Project Structure in Gradle
- Gradle Tasks
- Gradle Profile and Cloud
- Gradle Properties
- Gradle Plugins

Package Management - Artifactory & Packer

- Artifactory
- Artifactory Overview
- Understanding a role of Artifactory in DevOps
- System Requirements
- Installing Artifactory in Linux
- Using Artifactory
- Getting Started
- General Information
- Artifactory Terminology
- Artifactory Repository Types
- Artifactory Authentication
- Deploying Artifacts using Maven
- Download Artifacts using Maven
- Browsing Artifactory
- Viewing Packages
- Searching for Artifacts
- Manipulating Artifacts

Baking a Website Image for EC2

- Select an AWS AMI base
- Save Select an AWS AMI base
- Automate AWS AMI base build
- Save Automate AWS AMI base build
- Using build variables

- Save Using build variables
- Provision Hello World
- Save Provision Hello World
- Provision a basic site
- Save Provision a basic site

Customization with a Config Management Tool

- Simplify provisioning with a config tool
- Save Simplify provisioning with a config tool
- Use ansible to install the webserver
- Save Use ansible to install the webserver
- Debugging
- Save Debugging

Building Hardened Images

- Use Ansible modules to harden our image

- Save Use Ansible modules to harden our image
- Baking a Jenkins image
- Save Baking a Jenkins image

Building a Pipeline for Packer Image

- Validate Packer templates
- Save Validate Packer templates
- Create a manifest profile
- Save Create a manifest profile
- Testing
- Save Testing
- CI pipeline
- Save CI pipeline

Acceptance Testing - Functional & Regression - HPE UFT

- Flask Overview
- Flask Installign and Configuring
- Flask Understanding Environment
- Flask Understanding Application
- Flask Understanding Routing
- Flask Understanding Variable Rules
- Flask Understanding URL Building
- Flask Understanding HTTP Methods
- Flask Understanding Templates
- Flask Understanding Static Files
- Flask Understanding Request Object
- Flask Understanding Sending Form Data to Template
- Flask Understanding Cookies
- Flask Understanding Sessions
- Flask Understanding Redirect & Errors

Acceptance Testing - Web applications - Selenium

Introduction To Selenium

- What is Selenium?
- What are the components of Selenium?
- Understanding Why selenium is better than other tools?
- Advantages and disadvantage of Selenium

Selenium Hands-On

- What are the basic commands in Selenium?
- Commands to open and close a browser
- Get Commands
- Navigation Commands
- Switch Commands
- Interacting with WebElement

What are Web Elements and How to Locate Web Elements?

- Understanding Types & Techniques of Locators to locate WebElements?
- Understanding What are Web Elements?
- How to customize Xpath to Locate various WebElements?
- Hands - on example with “NewTours” Web Application to experience Selenium Automation Execution.

Hands-On With Various Example On

- Different types of Alerts - Simple, Confirmation and Prompt alert.
- Handling modal dialog box and pop-ups
- Handling various windows using switch window commands
- Working with Drag and Drop of an element
- Understanding how we can handle element in drop down list.
- Working with multiple selection of elements from a list of options
- Experiencing the Mouse Hovering action.
- Experiencing the Keyboard and Mouse actions using the Action and Actions Class.

Securing App Runtime (DAST) - AppScan & Fortify Webinspect

- Introduction
- Fortify Software System Requirements
- Fortify Static Code Analyzer Installation Guide
- Analysis Process Overview
- Translating Java Code
- Translating .NET Code
- Translating C and C++ Code
- Translating JavaScript Code
- Translating Ruby Code
- Translating ABAP Code
- Translating Code for Mobile Platforms
- Translating Apex and Visualforce Code
- Translating Flex and ActionScript
- Translating COBOL Code
- Translating Other Languages
- Integrating into a Build
- Command-Line Interface
- Command-Line Utilities
- Troubleshooting
- Filtering the Analysis
- Scan Wizard
- Sample Files
- Configuration Options
- Fortify Static Code Analyzer Performance Guide
- Introduction
- Performance Improvement Tips
- Scan Quality and Performance
- Optimizing FPR File
- Monitoring Long Running Scans
- Troubleshooting

Code Coverage - Jacoco & Coverage.py

- Java API
- Command Line
- Apache Ant
- Apache Maven
- Jenkins

Webserver - Nginx

Overview

- Introduction
- About NGINX
- NGINX vs Apache
- Test your knowledge

Installation

- Server Overview
- Installing with a Package Manager
- Building Nginx from Source & Adding Modules
- Adding an NGINX Service
- Nginx for Windows
- Test your knowledge

Configuration

- Understanding Configuration Terms
- Creating a Virtual Host
- Location blocks
- Variables
- Rewrites & Redirects
- Try Files & Named Locations
- Logging
- Inheritance & Directive types
- PHP Processing
- Worker Processes
- Buffers & Timeouts
- Adding Dynamic Modules
- Test your knowledge

Performance

- Headers & Expires
- Compressed Responses with gzip
- FastCGI Cache
- HTTP2
- Server Push

Security

- HTTPS (SSL)
- Rate Limiting
- Basic Auth
- Hardening Nginx
- Test your knowledge
- Let's Encrypt - SSL Certificate

Why you are here?

- Introduction of Configuration Management
- Understand the problems in Servers config management
- Finding out the solutions for Servers config management
- What is Ansible
- Benefits of Ansible
- Ansible Vs Chef Vs Puppet

Lets understand Ansible

- Ansible Architecture
- How Ansible Works?

Lets get startted with Ansible

- Introduction to YAML
- Setup Ansible Developement Environment
- Introduction of Ansible Commands lines
- Introduction of Ansible Modules
- Introduction of Ansible Play
- Introduction of Ansible Playbook
- Introduction of Ansible Configurations
- Introduction of Ansible Inventory

Lets start config mgmt using Ansible

- Writing a Play Using Modules
- Writing a Playbook
- Writing a Inventory
- Running a Playbook in Local Server
- Running a Playbook in Remote Server

Popular Modules in Ansible

- The Setup Module
- The Command Module
- The Expect Module
- The RAW Module
- The Script Module
- The Shell Module
- The File Module
- The SELinux Module
- The SEBoolean Module
- The Copy Module
- The Cron Module
- The Include Module
- The Mount Module
- The User Module
- The Group Module

- The Mail Module
- The Git Module
- The Apt Module
- The Yum Module
- The Package Module
- The Service Module

Deep Dive into Ansible Playbooks

- Deep Dive into Ansible Playbooks

Outlining Your Playbook

- Target Section
- Variable Section
- Task Section
- Handler Section
- Lookups

Understand the facters in Ansible

- Outlining Your Playbook

Handlers in Ansible

- Variables in Ansible
- Register in Ansible
- Includes vs. Imports in Ansible
- Conditions in Ansible
- Loops in Ansible
- Blocks in Ansible
- Files operations in Ansible
- Templating(Jinja2) in Ansible
- Basic Error Handling in Playbooks
- Basic Torubleshooting in Ansibles

Deep Dive into Ansible Inventory

- Using Hosts and Groups
- Behavioral Inventory Parameters
- Host Variables
- Group Variables
- Groups of Group
- Groups of Group Variables
- Splitting Out Host and Group Specific Data

Servers Managment in Ansible

- Using Hosts and Groups
- Behavioral Inventory Parameters
- Host Variables
- Group Variables
- Groups of Group
- Groups of Group Variables
- Splitting Out Host and Group Specific Data

Roles in Ansible

- Understand Role Directory Structure
- Using Ansible Roles
- Ansible Role Duplication and Execution
- Ansible Role Default Variables
- Ansible Role Dependencies
- Embedding Modules and Plugins In Roles
- Ansible Role Search Path

Ansible Vault

- Use cases of Ansible Vault
- Creating Encrypted Files
- Editing Encrypted Files
- Rekeying Encrypted Files
- Encrypting Unencrypted Files
- Decrypting Encrypted Files
- Viewing Encrypted Files
- Using Ansible Vault

Infrastructure Coding - Terraform

Deploying Your First Terraform Configuration

- Introduction
- What's the Scenario?
- Terraform Components
- Demo Time!

Configuring Resources After Creation

- Introduction
- What's the Scenario?
- Terraform Provisioners
- Terraform Syntax
- Demo Time!

Updating Your Configuration with More Resources

- Introduction
- Terraform State and Update
- What's the Scenario?
- Data Type and Security Groups
- Demo Time!

Adding a New Provider to Your Configuration

- Introduction
- What's the Scenario?
- Terraform Providers
- Terraform Functions
- Demo: Intro and Variable
- Demo: Resource Creation
- Demo: Deployment and Terraform Console
- Demo: Updated Deployment and Terraform Commands

Continuous Integration - Jenkins

Lets understand Continuous Integration

- What is Continuous Integration
- Benefits of Continuous Integration
- What is Continuous Delivery
- What is Continuous Deployment
- Continuous Integration Tools
- Jenkins Vs TeamCity Vs Bamboo

What is Jenkins

- History of Jenkins
- Release of Jenkins
- Jenkins Vs Jenkins Enterprise

Jenkins Installation

- Prerequisites
- Installing Jenkins using Yum
- Installing Jenkins using Apt
- Installing Jenkins using MSI
- Installing Jenkins using Tomcat
- Installing as a Service
- Starting and Stopping Jenkins

Jenkins Dashboard Tour

- Understand Freestyle Project
- Freestyle General Tab
- Freestyle Source Code Management Tab
- Freestyle Build Triggers Tab
- Freestyle Build Environment
- Freestyle Build
- Freestyle Post-build Actions

- Manage Jenkins
- My Views
- Credentials
- People
- Build History

Creating a Simple Job

- Simple Java and Maven Based Application
- Simple Java and Ant Based Application
- Simple Java and Grovy Based Application
- Simple DOTNET and MSBuild Based Application
- Simple C++ using Batch Application

Jobs Scheduling in Jenkins

- Manually Building
- Build Trigger based on fixed schedule
- Build Trigger by script
- Build Trigger Based on pushed to git

Useful Jobs Configuration

- Jenkins Jobs parameterised
- Execute concurrent builds
- Jobs Executors
- Build Other Projects
- Build after other projects are built
- Throttle Builds

Jenkins Plugins

- Installing a Plugin
- Plugin Configuration
- Updating a Plugin
- Plugin Wiki
- Top 20 Useful Jenkins Plugins
- Using Jenkins Pluginss Best Practices

Jenkins Node Managment

- Adding a Linux Node
- Adding a Windows Nodes
- Nodes Management using Jenkins
- Jenkins Nodes High Availability

Jenkins Integration with other tools

- Jira
- Git
- SonarQube
- Maven
- Junit
- Chef
- Puppet
- Ansible
- Docker
- AWS
- Jacoco
- Coverity
- Selenium
- Gradle
- Terraform
- Packer

Reports in Jenkins

- Junit Report
- SonarQube Reports
- Jacoco Reports
- Coverity Reports
- Selenium Reports
- Test Results
- Cucumber Reports

Jenkins Node Managment

- Adding a Linux Node
- Adding a Windows Nodes
- Nodes Management using Jenkins
- Jenkins Nodes High Availability

Notification & Feedback in Jenkins

- CI Build Pipeline & Dashboard

- Email Notification
- Advance Email Notification
- Slack Notification

Jenkins Advance - Administrator

- Security in Jenkins
- Authorization in Jenkins
- Authentication in Jenkins
- Managing folder/subfolder
- Jenkins Upgrade
- Jenkins Backup
- Jenkins Restore
- Jenkins Command Line

Thank you!

Connect with us for more info

Call/WhatsApp: - +91 968 682 9970 Mail: - contact@DevOpsSchool.com

www.DevOpsSchool.com