

Durée : 3 jours soit 21 heures

Référence : IF-ANS

Public visé :

- Administrateurs Linux, DevOps

Pré-requis :

- Posséder des connaissances sur le shell et la ligne de commande " Linux"

Objectifs pédagogiques :

- Comprendre l'intérêt et le fonctionnement des outils d'automatisation de gestion de l'infrastructure
- Pouvoir écrire des *playbooks* pour gérer son infrastructure et déployer ses applications
- Savoir créer des Rôles réutilisables

Modalités pédagogiques :

Session dispensée en présentiel ou téléprésentiel, selon la modalité inter-entreprises ou intra-entreprises sur mesure.

La formation est animée par un(e) formateur(trice) durant toute la durée de la session et présentant une suite de modules théoriques clôturés par des ateliers pratiques validant l'acquisition des connaissances. Les ateliers peuvent être accompagnés de Quizz.

L'animateur(trice) présente la partie théorique à l'aide de support de présentation, d'animation réalisée sur un environnement de démonstration.

En présentiel comme en téléprésentiel, l'animateur(trice) accompagne les participants durant la réalisation des ateliers.

Moyens et supports pédagogiques :

**Cadre présentiel**

Salles de formation équipées et accessibles aux personnes à mobilité réduite.

- Un poste de travail par participant
- Un support de cours numérique ou papier (au choix)
- Un bloc-notes + stylo
- Vidéoprojection sur tableau blanc
- Connexion Internet
- Accès extranet pour partage de documents et émargement électronique

**Cadre téléprésentiel**

Session dispensée via notre solution iClassroom s'appuyant sur Microsoft Teams.

- Un compte Office 365 par participant
- Un poste virtuel par participant
- Un support numérique (PDF ou Web)
- Accès extranet pour partage de documents et émargement électronique

Modalités d'évaluation et suivi :

**Avant**

Afin de valider le choix d'un programme de formation, une évaluation des prérequis est réalisée à l'aide d'un questionnaire en ligne ou lors d'un échange avec le formateur(trice) qui validera la base de connaissances nécessaires.

**Pendant**

Après chaque module théorique, un ou des ateliers pratiques permettent la validation de l'acquisition des connaissances. Un Quizz peut accompagner l'atelier pratique.

**Après**

Un examen de certification si le programme de formation le prévoit dans les conditions de l'éditeur ou du centre de test (TOSA, Pearson Vue, ENI, PeopleCert)

**Enfin**

Un questionnaire de satisfaction permet au participant d'évaluer la qualité de la prestation.

**Description / Contenu**

**Module 1 : L' "infrastructure as code"**

- Définition, principes
- Concepts de base et premières manipulations d'Ansible
- Idempotence
- Installation
- Gestion des versions

**Module 2 : Premier déploiement, premières bonnes pratiques**

- Syntaxe YAML
- Les modules et les tâches
- Introduction à *Ansible Galaxy*
- Collections et *namespaces*
- Les *playbooks* et les *plays*
- Lancement d'un *playbook*
- Problème de la connexion distante

**Module 3 : Déploiement dynamique**

- Les inventaires
- Les variables dans Ansible
- Variables prédéfinies
- Précédence des variables
- Proposition de règles de nommage
- Les *templates* et filtres *Jinja2*

**Module 4 : Fonctions avancées d'Ansible**

- Les captures de résultats
- Les conditions
- L'idempotence "avancée"
- Les boucles
- Les *plugins* de *lookup*



- Intégration avec les plateformes de virtualisation
- Délégation de tâches

**Module 5 : Modularisation du code**

- Les "Handlers" et les "Notifications"
- Inclusion de fichiers
- Les rôles
- Les tags
- Bonnes pratiques

**Module 6 : Sécurité et bonnes pratiques**

- Les vaults
- La gestion des erreurs
- Les blocs de tâches

**Module 7 : mise en production**

- Du développement à la production : organisation des fichiers
- Tests qualité avec *ansible-lint*
- Explication de ZDD (*Zero Downtime Deployment*)
- Prise en compte du *Rolling Update*

**Module 8 : Pour aller plus loin...**

- Développer des filtres
- Développer des modules & plugins
- Intégration des *Containers*
- Intégration des équipements réseau
- Tests unitaires avec *molecule*
- Centralisation avec AWX
- Les Environnements d'Exécution et *ansible-builder*
- *ansible-pull*
- *ansible-navigator*