

Durée : 3 jours soit 21 heures

Référence : IF-DJGO

Public visé :

- Développeurs Web
- Chefs de projet

Pré-requis :

- Connaissances de base Python - HTML/CSS

Objectifs pédagogiques :

- Mettre en oeuvre le Framework Web Django - comprendre les écritures Python avancées

Compétences acquises à l'issue de la formation :

- Mettre en oeuvre le Framework Web Django - comprendre les écritures Python avancées

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

Informations sur l'accessibilité :



Description / Contenu

Module 1 : Introduction

- 1.1.Installation et prérequis
- 1.2.Le modèle MVC
- 1.3.Architecture d'une application Web
- 1.4.Rappels Python

Module 2 : Prise en main - 1er projet

- 2.1.Commandes d'administration
- 2.2.Création du projet
- 2.3.Configuration
- 2.4.Interface d'administration
- 2.5.Création d'un modèle
- 2.6.Création de vues
- 2.7.Création d'un gabarit
- 2.8.Création d'un formulaire

Module 3 : Administration

- 3.1.Personnalisation de l'interface
- 3.2.Gestion évoluée des modèles

Module 4 : Les Templates (Gabarits)

- 4.1.Principe - choix du moteur
- 4.2.Le langage DTL
- 4.2.1.Les variables
- 4.2.2.Les filtres
- 4.2.3.Les tags
- 4.3.Création de filtres
- 4.4.Création de tags
- 4.5.Le moteur Jinja2

Module 5 : Les vues

- 5.1.Fonctions "raccourcis"
- 5.2.Réécriture des pages d'erreur
- 5.3.Les "Class Based Views"
- 5.4.Les Mixins
- 5.5.Les vues génériques
- 5.6.Les Context Processors

Module 6 : Les modèles

- 6.1.Concepts
- 6.2.Modèles et Champs
- 6.2.1.Relation 1-to-1
- 6.2.2.Relation n-to-1
- 6.2.3.Relation n-to-n
- 6.3.Requêtage
- 6.4.Agrégation
- 6.5.Transactions
- 6.6.Utilisation de l'ORM "Hors Django"

Module 7 : Les formulaires

- 7.1.La classe Form
- 7.2.Les champs de formulaires
- 7.3.Affichage des formulaires
- 7.4.Les Widgets
- 7.5.Les Assets
- 7.6.Les Formsets

- 7.7.Les ModelForm

Module 8 : Utilisateurs et droits

- 8.1.Composants mis en oeuvre
- 8.2.Choix des backends
- 8.3.Extensions disponibles
- 8.4.Les signaux
- 8.5.Ecrire son propre décorateur

Module 9 : Divers

- 9.1.Paramètres généraux - Settings
- 9.2.Flash Message
- 9.3.Gestion des Logs
- 9.4.Simplification des URL
- 9.5.Téléchargement de fichiers
- 9.6.Modèles avancés
- 9.7.Gestion des mails
- 9.8.Requêtes AJAX
- 9.9.Internationalisation
- 9.10.Optimisations
- 9.11.Caches

Module 10 : Sécurité

- 10.1.Injection SQL
- 10.2.Attaque XSS
- 10.3.Attaque CSRF
- 10.4.Clickjacking
- 10.5.Middleware de sécurisation

Module 11 : Tests

- 11.1.Automatiser les tests
- 11.2.Lancement des tests

Module 12 : Déploiement

- 12.1.Comment déboguer ?
- 12.2.Passage en production
- 12.3.Intégration Apache
- 12.4.Production & Développement
- 12.5.Checklist de déploiement