

Django : Initiation + Approfondissement

INFORMATIONS GÉNÉRALES

Type de formation : Formation continue

Éligible au CPF : Non

Domaine : Développement

Action collective : Non

Filière : Open Source - LAMP : Linux Apache PHP

Rubrique : Langages : Python, Scala, GO,...

Code de formation : LEDN206

PRÉSENTATION

Objectifs & compétences

A l'issue de la formation, le stagiaire sera capable :

- Développer une application web robuste avec le framework Django

Public visé

Développeurs Python

Pré-requis

Maîtrise de la programmation objet en Python

€ Tarifs

Prix public : 2995 €

Tarif & financement :

Nous vous accompagnons pour trouver la meilleure solution de financement parmi les suivantes :

Le plan de développement des compétences de votre entreprise : rapprochez-vous de votre service RH.

Le dispositif FNE-Formation.

L'OPCO (opérateurs de compétences) de votre entreprise.

France Travail: sous réserve de l'acceptation de votre dossier par votre conseiller Pôle Emploi.

CPF -MonCompteFormation

Contactez nous pour plus d'information : contact@aston-institut.com

📍 Lieux & Horaires

Durée : 35 heures

Délai d'accès : Jusqu'à 8 jours avant le début de la formation, sous condition d'un dossier d'inscription complet

PROGRAMME

Découvrir Django

Introduction au Framework Django : versions, documentation
Django et le Design Pattern MVC/MVT : principes et intérêt technique
Écosystème des frameworks web pour Python (Pyramid, Flask, etc.)

Environnements de développement intégrés spécialisés

Installer Django et ses dépendances pour travailler sereinement

Découverte du serveur web de développement

Atelier : Installation de Django (environnement, dépendances)

Comprendre la structure d'un site Django

Notions de projet et d'application

Structure d'un projet Django

Structure d'une application Django

Paramétrer son site web

Ajouter des URLs et afficher du contenu

Découvrir le routeur d'URLs de Django

Créer des modèles d'URLs simples

Créer des modèles d'URLs avec des expressions régulières

Nommer ses modèles d'URLs pour simplifier leur utilisation

Découvrir les bases des vues avec Django

Créer des vues en écrivant des fonctions

Comprendre les requêtes et les réponses HTTP

Atelier : Créer ses propres vues et renvoyer du contenu simple

Afficher du contenu dynamique avec les gabarits (templates)

Templating avec Django : principe

Comprendre le moteur de gabarits de Django

Créer et utiliser des gabarits (templates), utiliser les contextes de gabarit

Découvrir les filtres et balises de gabarits

Organiser ses templates avec l'héritage et les blocs de gabarits

Atelier : Construction d'un template et affichage dans une vue

📅 Prochaines sessions

Consultez-nous pour les prochaines sessions.

Interagir avec l'utilisateur via les formulaires

Définir des formulaires avec les classes fournies par Django

Afficher un formulaire dans une page

Gérer la saisie dans les formulaires

Gérer le téléversement de fichiers

Validation des saisies

Atelier : Manipuler un formulaire simple avec Django

Gérer des données avec Django

L'accès aux données avec Django

Comprendre les bases de l'ORM de Django : configuration et usage

Définir et modifier des modèles pour interagir avec une base de données

Appréhender le système de migrations de Django

Utiliser les modèles et l'ORM pour requêter la base de données

Relier des tables entre elles en définissant des relations et clés

Gérer les requêtes s'exécutant en concurrence (objets F)

Héritage de classes de modèle

Atelier : implémentation d'une couche d'accès aux données avec Django

Créer automatiquement des formulaires via les modèles

Créer facilement des formulaires depuis des modèles de base de données

Personnaliser le formulaire associé à un modèle

Gérer la saisie de données avec un formulaire associé à un modèle

Mettre à jour des instances de modèle avec un formulaire

Gérer facilement le téléversement de fichiers

L'interface d'administration de Django

Découvrir l'interface d'administration de Django

Personnaliser l'interface pour ses propres modèles

Le système d'authentification et d'utilisateurs de Django

L'application d'authentification de Django

S'authentifier programmatically dans une vue (connexion, déconnexion)

S'authentifier avec un formulaire

Tester l'authentification dans des gabarits

Atelier : Créer un utilisateur et écrire des vues de connexion/déconnexion

Écrire des sites web internationaux

Comprendre la langue de base d'un site web

Traduction de texte dans le code Python

Rendre du texte traduisible dans les templates

Outils de traduction du texte (gettext)

Générer des catalogues de traduction

Tests automatisables d'un site web Django

Intérêt des tests unitaires

Écrire des tests unitaires

Utiliser le lanceur de tests Django

Comprendre le client de test Django

Atelier : Écrire un test d'une vue

Concepts avancés divers

Envoyer des courriers électroniques

Exporter des données (CSV ou autres)

S'authentifier via des comptes de réseaux sociaux (protocole OAuth etc.)

Comprendre le déploiement d'un site web Django en production

Atelier : envoyer un courrier

MODALITÉS**Modalités**

Modalités : en présentiel, distanciel ou mixte . Toutes les formations sont en présentiel par défaut mais les salles sont équipées pour faire de l'hybride. – Horaires de 9H à 12H30 et de 14H à 17H30 soit 7H – Intra et Inter entreprise.

Pédagogie : essentiellement participative et ludique, centrée sur l'expérience, l'immersion et la mise en pratique. Alternance d'apports théoriques et d'outils pratiques.

Ressources techniques et pédagogiques : Support de formation au format PDF ou PPT Ordinateur, vidéoprojecteur, Tableau blanc, Visioconférence : Cisco Webex / Teams / Zoom.

Pendant la formation : mises en situation, autodiagnostic, travail individuel ou en

sous-groupe sur des cas réels.

Méthode

Fin de formation : entretien individuel.

Satisfaction des participants : questionnaire de satisfaction réalisé en fin de formation.

Assiduité : certificat de réalisation.

Validations des acquis : grille d'évaluation des acquis établie par le formateur en fin de formation.