

Formation Python, Programmation (Avancé)

Présentation

Cette formation vous permettra de développer vos compétences fondamentales en programmation Python pour créer des applications fiables et stables. Vous améliorerez votre maîtrise de Python grâce aux meilleures pratiques en matière de programmation orientée objet, de test, de débogage, de mesure et d'amélioration des performances des applications et de développement de services RESTful.

Durée : 28,00 heures (4 jours)

Tarif INTRA : Nous consulter

Objectifs de la formation

- Utiliser des design patterns et les meilleures pratiques dans les applications Python
- Réaliser des tests unitaires, débogages et installations de programmes et modules Python
- Profiler l'exécution d'un programme et améliorer les performances
- Appliquer des fonctionnalités de programmation Python avancées pour des programmes efficaces, fiables et maintenables

Prérequis

Avoir suivi le cours "Développer avec Python" ou avoir au moins trois mois d'expérience en programmation Python.

Public

Développeurs Python

Programme de la formation

Rappels importants sur le langage

- Syntaxe de base et nouveautés récentes
- Rappels sur les fonctions



- Conditions ternaires, listes en compréhensions, one-liners

Fonctions avancées

- Accès aux attributs des fonctions
- Les générateurs
- Les décorateurs
- Les closures

Travaux pratiques : création de décorateurs de contrôle, mise en place d'itérables selon plusieurs techniques

Programmation orientée objet avancée

- Rappels sur la programmation orientée objet
- Ajout de propriétés à une classe
- Introspection
- Redéfinition des méthodes spéciales : itérateurs, context managers, ...
- Définition des classes de base abstraites
- Création de métaclasses

Travaux pratiques : exploration des différentes fonctionnalités avancées

Vérification du code et tests unitaires

- Tester les meilleures pratiques
- Développement et exécution de tests unitaires Python
- Simplifier les tests automatisés avec nose
- Vérification du comportement du code
- Utiliser des mocks
- Détection des erreurs et techniques de debugging
- Vérifier votre code pour des bugs potentiels avec Pylint
- Extraction des informations d'erreur depuis les exceptions
- Tracer l'exécution d'un programme avec l'IDE PyCharm

Travaux pratiques : test des différents programmes créés précédemment

Interfaçage avec les webservices et les clients de REST

- Présentation de Flask, Django, etc.
- Mise en place d'un service REST
- Générer des réponses JSON pour soutenir les clients Ajax
- Clients Python REST

- Envoi de demandes REST à partir d'un client Python
- Consommation de données de réponse JSON et XML

Travaux pratiques : consommation d'API et webscrapping

Mesure et amélioration des performances de l'application

- Chronométrage de l'exécution des fonctions
- Profilage de l'exécution d'un programme
- Manipuler un profil d'exécution de manière interactive
- Utilisation des fonctionnalités du langage Python pour les performances
- Appliquer efficacement les structures de données, notamment les listes, les dictionnaires et les tuples
- Remplacement de l'interpréteur Python standard par PyPy

Installation et distribution des modules

- Gestion des dépendances d'un projet avec pip
- Le fichier requirements.txt
- La gestion des versions
- Installation de modules du dépôt PyPi à l'aide de pip
- Portage du code entre les versions Python
- Création de paquets de modules et d'applications Python
- Mise en place d'environnements virtuels (virtualenv, conda, pipenv, poetry, ...)
- Construire un paquet de distribution avec setuptools
- Télécharger vos modules Python sur un dépôt local

Travaux pratiques : mise en place d'un projet créé avec environnement virtuel, packaging et déploiement sur PyPI

Exécution simultanée

- Création et gestion de plusieurs fils de contrôle avec la classe Thread
- Synchronisation des fils à l'aide de verrous
- Lancement des commandes du système d'exploitation en tant que sous-processus
- Synchronisation des processus avec les files d'attente
- Paralléliser l'exécution en utilisant des pools de processus et des exécuteurs

Travaux pratiques : exécution de traitements en parallèle et mesure des performances

Aller plus loin

- Présentation de Cython, CPython
- Le typage avec MyPy

Organisation

Formateur

Les formateurs de Docaposte Institute sont des experts de leur domaine, disposant d'une expérience terrain qu'ils enrichissent continuellement. Leurs connaissances techniques et pédagogiques sont rigoureusement validées en amont par nos référents internes.

Riches de leur expérience sur le sujet, ils sauront accompagner vos collaborateurs dans leur montée en compétence.

Moyens pédagogiques et techniques

- Alternance d'apports théoriques et de mises en pratique contextualisées.
- Progression pédagogique structurée, de la compréhension à la mise en œuvre opérationnelle.
- Pédagogie immersive favorisant une mise en application immédiate des compétences.
- Formation interactive et participative s'appuyant sur des ateliers pratiques, études de cas et échanges collectifs (+ 60 % du temps de formation.)
- Séquences pédagogiques rythmées et évaluations formatives régulières pour renforcer l'ancrage des apprentissages.
- Supports de cours, ressources documentaires et références mis à disposition par le formateur

Dispositif de suivi de l'exécution et de l'évaluation des résultats de la formation

En amont de la formation *(évaluation en ligne dans l'extranet apprenant)*

- Recueil des besoins et des attentes des participants.
- Auto-positionnement permettant d'évaluer le niveau initial des apprenants.

Tout au long de la formation

- Évaluations continues réalisées par le formateur à travers des questions orales, exercices, QCM, études de cas et mises en situation pratiques.

À l'issue de la formation (*évaluations en ligne dans l'extranet apprenant*)

- Auto-positionnement afin de mesurer l'acquisition des compétences visées.
- Validation de l'acquisition des compétences par le formateur
- Questionnaire de satisfaction « à chaud » à l'issue de la formation
- Questionnaire de satisfaction "à froid", 45 jours après la formation afin d'évaluer l'ancrage des acquis et leur mise en application en situation professionnelle.

Accessibilité

Nos formations peuvent être adaptées à certaines conditions de handicap. Nous contacter pour toute information et demande spécifique.