

Python, perfectionnement

4 j (28 heures)

Ref : PYT-PE

Formation éligible au CPF

Public

Ingénieurs et développeurs

Pré-requis

Niveau : Disposer de bonnes connaissances en développement Python

Techniques (formations en classe virtuelle) : Vous devez disposer d'un ordinateur connecté à internet, d'un micro et d'une caméra

Moyens pédagogiques

Modalité : Formation présentielle ou Formation distancielle (classe virtuelle) - Inter / Intra - Groupes de 4 à 12 stagiaires

Méthodes : Présentation des concepts, discussion technique, démonstrations, exercices et TP

Matériel :

Présentiel : Un poste informatique par stagiaire connecté à internet, à une imprimante en réseau et au réseau informatique,

Les salles sont équipées d'un tableau interactif ou d'un vidéoprojecteur et d'un paperboard

Distanciel : Aelion met à disposition de chaque stagiaire

- Un PC équipé des outils et logiciels nécessaires à la formation auquel vous accédez via un outil de prise en main à distance

- Un accès à un outil de classe virtuelle (Meet)

Support de formation : Un support de formation sera remis à chaque stagiaire en fin de formation : plateforme collaborative intégrant le code source des exercices réalisés en formation, webographie, mémos

Modalités de suivi et d'évaluation

Questionnaire d'évaluation des pré-requis, suivi des connaissances tout au long de la formation, Evaluation des acquis en fin de formation

Questionnaire d'évaluation de la satisfaction en fin de stage, feuille de présence émargée par demi-journée par les stagiaires et le formateur, Attestation de fin de formation

Le langage Python s'est imposé comme un langage majeur du développement d'application. Il allie une syntaxe simple et performance. Il couvre tous les domaines du développement et sa communauté importante propose des bibliothèques qui font référence. De nombreux domaines spécialisés sont couverts comme le développement web, le calcul scientifique, les data sciences, l'intelligence artificielle ou les SIG. A l'issue de la formation, vous serez capable d'utiliser les fonctionnalités avancées de Python ainsi que les principaux outils associés au langage, pour pouvoir répondre aux exigences de programmation.

Action collective OPCO ATLAS - [Inscription CampusAtlas](#)

Objectifs

- Utiliser les techniques avancées du langage Python : Context Manager, métaclasses, closures, fonctions avancées
- Optimiser les performances de vos programmes à l'aide du monitoring et du parallélisme
- Packager et déployer ses artefacts Python
- Exploiter des bibliothèques contribuant au succès du langage : calcul scientifique, Intelligence Artificielle, XML, réseau

Programme détaillé

EXPLOITER DES BIBLIOTHÈQUES CONTRIBUANT AU SUCCÈS DU LANGAGE

- Historique de python
- Panorama des solutions Python
- Environnements de développement

UTILISER LES TECHNIQUES AVANCÉES DU LANGAGE PYTHON (1/3)

- Gestion avancée des paramètres de fonctions
- Collections simples et avancées
- Itérateurs et générateurs avancés
- Expression for, list et dict comprehension
- Map/reduce et évaluation paresseuse

UTILISER LES TECHNIQUES AVANCÉES DU LANGAGE PYTHON (2/3)

- Gestion et définitions des exceptions
- Utilisation d'un context manager
- Méthodes asynchrones
- Typage avec les hints

UTILISER LES TECHNIQUES AVANCÉES DU LANGAGE PYTHON (3/3)

- Programmation orientée objets avancée
- Les propriétés
- Implémentation d'opérateur, builtin fonction ou context manager
- Héritage multiple et algorithme MRO
- Les décorateurs
- Variable et méthode de classe
- Classes abstraites et interfaces
- Métaclasses
- Introspection

OPTIMISER LES PERFORMANCES DE VOS PROGRAMMES A L'AIDE DU MONITORING ET DU PARALLELISME

- Programmation parallèle et concurrente
- Multi threading
- Multi process
- Pool de threads ou process
- Objets futurs
- Synchronisation
- Paradigme de la concurrence

PACKAGER ET DEPLOYER SES ARTEFACTS PYTHON

- Installer des librairies tierces
- Établir la liste de dépendances de son projet
- Packager ses librairies
- Environnement virtuel
- Profiler son code
- Tests unitaires

EXPLOITER DES LIBRAIRIES CONTRIBUANT AU SUCCES DU LANGAGE (1/5)

- XML et WEB SCRAPING
- Packages xml, lxml, BeautifulSoup
- Lire un document XML ou HTML
- Valider un document XML avec un schema XSD
- Chemins XPath et sélecteurs CSS
- Transformation python ou XSL
- Enregistrement

EXPLOITER DES LIBRAIRIES CONTRIBUANT AU SUCCES DU LANGAGE (2/5)

- Calcul scientifique avec NUMPY ET SCIPY
- Les types numériques
- Les tableaux à N dimensions
- Les statistiques
- Graphiques avec Matplotlib

EXPLOITER DES LIBRAIRIES CONTRIBUANT AU SUCCES DU LANGAGE (3/5)

- Taitement des données avec Pandas
- Les Dataframes
- Import/export des données
- Nettoyer, filtrer, transformer les données
- Combiner les données
- Grouper pour faire des statistiques

EXPLOITER DES LIBRAIRIES CONTRIBUANT AU SUCCES DU LANGAGE (4/5)

FRAMEWORKS DE MACHINE LEARNING

TensorFlow, Keras et ScikitLearn

Réseau de neurones

Régressions

Apprentissage

Prédiction

EXPLOITER DES LIBRAIRIES CONTRIBUANT AU SUCCES DU LANGAGE (5/5)

FRAMEWORKS WEB/REST ET ORM

Django, Flask et Fast Api

ORM SQL Alchemy

API Rest Rest

Routage et méthodes HTTP

DTO typé pour valider les données

Conversion JSON