

Python Foundations

Durée: 3 Jours **Réf de cours: GK821587** **Méthodes d'apprentissage: Intra-entreprise & sur-mesure**

Résumé:

Exclusivité – Apprenez à écrire des programmes en Python grâce à cette formation complète et très pratique de 3 jours. Se déroulant dans un environnement d'apprentissage stimulant et axé sur la pratique, sous la houlette d'un formateur expert, cette formation de 3 jours s'adresse aux participants de tous horizons qui souhaitent s'initier à la programmation en Python, acquérir les concepts fondamentaux de ce langage et écrire du code Python efficace en utilisant les dernières fonctionnalités et techniques. Il convient également aux programmeurs et développeurs confirmés s'intéressant au développement en RPA/IA, ou au développement lié à la science des données avec Python comme langage de script principal. Parmi les autres profils professionnels concernés, on peut citer : Les administrateurs système, les administrateurs et ingénieurs réseau, les ingénieurs DevOps et les développeurs back-end.

Public visé:

Ce cours constitue une introduction adaptée aux participants de tous horizons qui souhaitent se familiariser avec les concepts fondamentaux de Python et acquérir les bases de la programmation. Les programmeurs et développeurs qui souhaitent réorienter leur carrière vers le développement en RPA/IA ou vers des projets liés à la science des données utilisant principalement Python comme langage de script

Objectifs pédagogiques:

- Dans un environnement d'apprentissage stimulant et pratique, et sous la houlette d'un formateur expert, les étudiants apprendront à écrire du code Python efficace en utilisant les dernières fonctionnalités et techniques, notamment :
 - Travailler avec les dates, les heures et les calendriers
 - Lire et écrire des fichiers contenant à la fois des données texte et binaires
- Créer des scripts Python fonctionnels en respectant les bonnes pratiques
 - Se familiariser avec la bibliothèque standard et ses modules facilitant le travail
- Explorer les types de données Python simples et complexes et les utiliser à bon escient
 - Comprendre les objets en Python et comment créer des classes
- Utiliser les fonctions intégrées et créer des fonctions personnalisées avec des paramètres et des types de retour
- Comprendre les fonctionnalités propres à Python telles que les compréhensions et les itérateurs

Pré-requis:

Une connaissance de base d'un langage de programmation ou de script est utile, mais pas indispensable.

Contenu:

JOUR 1 : INITIATION À LA PROGRAMMATION EN PYTHON

1- Variables et types de données en Python

- Introduction à Python avec un aperçu rapide de la syntaxe et des IDE
- Rappel sur la manière de déclarer et d'initialiser des variables de différents types de données en Python : nombres, chaînes de caractères, booléens, séquences
- Conventions et règles de nommage
- Fonctions et rôle de la fonction main()
- Arguments
- Paramètres de ligne de commande
- Règles de base relatives aux espaces blancs
- Les types en tant qu'objets
- Conversion de types
- Objets mutables vs immuables

2- Manipulation des chaînes de caractères en Python

- Littéraux de chaîne – Séquences d'échappement
- Mise en forme de l'affichage des chaînes
- Affichage et mise en forme des chaînes
- Script avec input()
- À propos de raw_input()

3- Manipulation des dates et des heures

- Manipulation des dates et des heures
- Conversion des horodatages
- Analyse des dates à partir de texte
- Mise en forme des dates
- Données calendaires

4- Utilisation des instructions conditionnelles et des boucles

- Instructions « if else »
- Instructions « if elif »
- Instructions « if » imbriquées
- Utilisation de « while »
- Boucles contrôlées par un compteur et par un événement
- « continue » et « break »
- Exemples d'itération avec « for »

5- Objets

- Présentation des objets
- Création et définition d'objets
- Utilisation des objets

6- Gestion de base des erreurs

- Différents types d'erreurs
- Gestion des erreurs
- Bonnes pratiques

JOUR 2 : TYPES DE DONNÉES COMPLEXES EN PYTHON

1- Listes et tuples

- Listes
- Listes mixtes
- Méthodes courantes des listes
- Autres opérations sur les listes – Opérations CRUD
- Utilisation des tuples
- Tuples immuables
- Indexation et découpage
- Copies superficielles et profondes des listes et des tuples
- Listes vs tuples

2- Dictionnaires et ensembles

- Présentation et construction des dictionnaires
- Clés et valeurs
- Méthodes des dictionnaires
- À propos des ensembles
- Création d'ensembles
- Utilisation des ensembles
- Dictionnaires vs listes vs tuples vs ensembles

3- Comprendre les itérateurs

- La fonction range()
- Utilisation de range() sur len()
- Objets itératifs
- La fonction iter()
- Itération sur une séquence
- Séquences imbriquées
- Fonctions, mots-clés et opérateurs de séquence
- Boucles sur des ensembles de données
- Itération sur des listes
- Boucles sur des listes de listes
- Boucles avec des dictionnaires
- Boucles sur des listes de dictionnaires

4- Tri

- La fonction sorted()
- Clés alternatives
- Tri de collections
- Utilisation de operator.itemgetter()
- Tri inversé

5- Opérations de base sur les fichiers en Python

- Ouverture de fichiers
- Lecture de données à partir de fichiers
- Écriture de données dans des fichiers

Lecture et écriture de données brutes (binaires)

JOUR 3 : PROGRAMMATION MODULAIRE EN PYTHON

1- Fonctions Python

- Définition des fonctions
- Paramètres
- Valeurs de retour
- Fonctions imbriquées
- Masquage des variables
- Empêcher la modification des variables
- Méthodes de correspondance des arguments
- Compréhensions de listes de base
- Portée des variables : locale, globale, imbriquée

2- Modules et paquets

- Notions de base sur les modules
- Paquets
- Pip et pip3
- Environnements virtuels
- Modules définis
- Importation de modules
- Instructions « from import »
- Quelques modules utiles à connaître
- Alias de fonctions et de modules

3- La bibliothèque standard

- Le module sys
- Lancement de programmes externes
- Fonctions mathématiques
- Nombres aléatoires
- Lecture de données CSV

4- Les classes en Python

- À propos de la POO
- Définition d'une classe
- Constructeur
- Instanciation des classes
- Variables et méthodes de classe
- Variables d'instance
- Propriétés – getters et setters
- Sous-classes (héritage)
- Héritage multiple
- Méthodes statiques

PLAN DES TRAVAUX PRATIQUES

Ce cours comprend des activités pratiques visant à écrire et à exécuter des scripts de programmation en Python.

Méthodes pédagogiques :

Support de cours remis aux participants

Autres moyens pédagogiques et de suivi:

- Compétence du formateur : Les experts qui animent la formation sont des spécialistes des matières abordées et ont au minimum cinq ans d'expérience d'animation. Nos équipes ont validé à la fois leurs connaissances techniques (certifications le cas échéant) ainsi que leur compétence pédagogique.
- Suivi d'exécution : Une feuille d'émargement par demi-journée de présence est signée par tous les participants et le formateur.
- En fin de formation, le participant est invité à s'auto-évaluer sur l'atteinte des objectifs énoncés, et à répondre à un questionnaire de satisfaction qui sera ensuite étudié par nos équipes pédagogiques en vue de maintenir et d'améliorer la qualité de nos prestations.

Délais d'inscription :

- Vous pouvez vous inscrire sur l'une de nos sessions planifiées en inter-entreprises jusqu'à 5 jours ouvrés avant le début de la formation sous réserve de disponibilité de places et de labs le cas échéant.
- Votre place sera confirmée à la réception d'un devis ou ""booking form"" signé. Vous recevrez ensuite la convocation et les modalités d'accès en présentiel ou distanciel.
- Attention, si cette formation est éligible au Compte Personnel de Formation, vous devrez respecter un délai minimum et non négociable fixé à 11 jours ouvrés avant le début de la session pour vous inscrire via moncompteformation.gouv.fr.

Accueil des bénéficiaires :

- En cas de handicap : plus d'info sur globalknowledge.fr/handicap
- Le Règlement intérieur est disponible sur globalknowledge.fr/reglement