Makina Corpus Formation
49 grande rue Saint-Michel
31400 Toulouse
Email: formation@makina-corpus.com
Tel: 0661342710

Formation Développer en langage Python orienté objet

Mise à jour janvier 2024

Inter 2400€ HT/participant
Intra 6000€ HT* groupe de 6 à 8 participants (selon profils et niveaux)
(*hors frais de déplacement et personnalisation pour un programme sur mesure)

Au cours de cette formation, vous serez guidé à travers les différents aspects de Python orienté objet, depuis les fondamentaux jusqu'aux concepts avancés. Vous découvrirez l'historique et les différentes versions de Python, ainsi que les caractéristiques uniques de ce langage. Vous explorerez également la bibliothèque standard de Python, les modules d'extension et la gestion des dépendances avec pip.

Grâce à une approche pratique et orientée projet, vous développerez une compréhension approfondie des types de données non-modifiables et modifiables, des structures conditionnelles et répétitives, des fonctions, des modules et des paquets. Vous découvrirez comment exploiter la puissance de la programmation orientée objet en Python, en maîtrisant la création de classes, les instances, les attributs, les méthodes, l'héritage et la polymorphisme.

La formation abordera également des sujets tels que la manipulation de fichiers, la gestion des exceptions et l'utilisation des modules de la bibliothèque standard pour interagir avec le système d'exploitation et les différentes fonctionnalités du langage.

À la fin de cette formation, vous serez en mesure de travailler de manière autonome avec Python orienté objet et d'appliquer vos connaissances pour résoudre des problèmes complexes. Vous êtes prêt à réussir les épreuves de votre certification API Society et a obtenir votre diplôme officiel.

Durée: 28.00 heures (4.00 jours)

À QUI S'ADRESSE CETTE FORMATION ?

Profil du participant

- Développeurs
- Ingénieurs et chercheurs

Prérequis

- Connaissances de base en algorithmie

OBJECTIFS PÉDAGOGIQUES

1. Comprendre les caractéristiques du langage Python, son historique et les différentes versions.
2. Maîtriser les types de données non-modifiables tels que les booléens, les nombres et les chaînes de caractères, ainsi que leurs méthodes et opérations associées.
3. Manipuler des structures de données modifiables telles que les listes, les dictionnaires et les ensembles, en utilisant leurs méthodes et opérations spécifiques.
4. Appliquer les structures conditionnelles et répétitives (if, elif, else, while, for) pour la logique de contrôle du programme.
5. Comprendre les concepts de fonctions, de modules et de paquets, et les utiliser dans le développement Python.
6. Maîtriser la programmation orientée objet en Python, en comprenant les concepts fondamentaux tels que les classes, les objets, les attributs et les méthodes.
7. Manipuler des fichiers en utilisant les fonctions et les méthodes appropriées.
8. Gérer les exceptions en utilisant les instructions try, except, else et finally.
9. Utiliser les modules de la bibliothèque standard de Python pour interagir avec le système d'exploitation, le système de fichiers et les expressions rationnelles.
10. Comprendre les principes de base des tests unitaires en utilisant l'instruction assert et le module unittest.

Makina Corpus Formation
49 grande rue Saint-Michel
31400 Toulouse
Email: formation@makina-corpus.com
Tel: 0661342710

CONTENU (PROGRESSION PÉDAGOGIQUE)

Introduction

- Historique (auteur, date de la première version)
- Versions de Python (branches 2 et 3)
- Caractéristiques du langage (multi-paradigme, typage, dynamique fort, syntaxe claire)
- Panorama de la bibliothèque standard
- Modules d'extension et commande pip
- Principe de fonctionnement de l'interpréteur (bytecode PYC)
- Interpréteur officiel Cpython et autres interpréteurs (micropython, brython, pypy, numba)
- Ressources (site internet python.org, accès aux documentations)
- Fonction help et chaînes documentaires
- Principe de l'indentation pour délimiter les blocs d'instruction
- Commentaire
- Mots-clés réservés
- Conventions de nommage
- Interpréteur interactif
- Programme autonome
- Fonctions intégrées élémentaires : print(), type(), input(), len()

Types de données non-modifiables

- Utilité des types non-modifiables (optimisation mémoire), fonctions id() et hash(), opérateur is
- Principes des séquences ordonnées (str, tuple et list) et collections (dict, set)
- Booléen (bool), objet True et False
- Nombre (int, float, complex), constructeurs, opérateurs >>, <<, |, &, // et **
- Notations binaire, octale et hexadécimale, fonctions hex(), oct(), bin()
- Chaîne de caractères unicode (str), définition avec simple et double guillemets, chaines multilignes avec triple simple ou double guillemets, constructeur
- Indiçage positif et négatif, tranche de valeurs (slice), opérateurs + et *
- Méthodes incontournables de str : split(), replace(), upper(), lower(), upper(), strip(), join()
- Chaîne de caractères formatées (%s, %d, %f) et méthode format()
- Tableau d'octets (bytes), constructeur
- Tuple (tuple), constructeur, opérateurs + et *, méthodes count() et index()
- L'objet None et la fonction repr()

Types de données modifiables

- Listes (list), constructeur, opérateur + et *, méthodes apprend(), insert(), sort(), revers(), remove(), extend(), pop(), clear(), get()
- Manipulation de pointeurs
- Copie superficielle via la méthode copy() ou l'indiçage [:]
- Copie en profondeur avec la fonction deepcopy() du module copy
- Différences entre les opérateurs == et is, opérateur + et *
- Fonctions del(), sorted(), reversed(), range()
- Principe de fonctionnement des objets itérables
- Dictionnaires (dict), constructeur, méthodes keys(), values(), items(), update(), get()
- Set (set), constructeur, opérateurs - | & et ^

Makina Corpus Formation
49 grande rue Saint-Michel
31400 Toulouse
Email: formation@makina-corpus.com
Tel: 0661342710

Structures conditionnelles et répétitives

- Structure conditionnelle if ... elif ... else
- Opérateur ternaire
- Structure répétitive while
- Structure répétitive for
- Instructions break et continue
- Fonction enumerate()
- Bloc else sur structure répétitive
- Liste en intension (comprehension list)

Fonctions, modules et paquets

- Définition et appel d'une fonction
- Espace de noms local, global, pré-défini (__builtins__) et fonction dir()
- Retourner des valeurs, instruction return
- Fonctions génériques (duck typing)
- Valeurs par défaut
- Passage par étiquette
- Nombre d'arguments arbitraire (*args, **kwargs)
- Fonctions anonymes (lambda)
- Fonctions eval(), exec(), map() et filter()
- Importation de modules
- Création d'un module
- Bloc if __name__ == ''__main__''
- Importation de paquet
- Création d'un paquet (__init__.py)
- Instruction yield

Programmation Orientée Objet

- Concepts fondamentaux de la POO (séparation du code, encapsulation, héritage)
- Notions de classe d'objet, d'objet (instance), d'attribut et de méthode
- Définition d'une classe d'objet
- Instanciation d'objets, fonction isinstance()
- Constructeur (__init__)
- Attributs et méthodes
- Mot-clé self
- Surcharge d'affichage (__str_)
- Surcharge d'opérateurs (__eq__, __add__, ...)
- Propriété (fonction spéciale property), accesseur et mutateur
- Espaces de noms global, de l'objet, de la classe
- Variable de classe
- Constructeur à nombre d'arguments arbitraire (*args, **kwargs)
- Héritage de classe (généralisation), fonctions issubclass(), super() et méthode mro()

Makina Corpus Formation
49 grande rue Saint-Michel
31400 Toulouse
Email: formation@makina-corpus.com
Tel: 0661342710

Manipulation des fichiers

- Fonction open() et méthode close()
- Méthodes readline() et readlines()
- Objet itérable
- Instruction with avec les fichiers
- Méthodes read() et write()
- Méthodes tell() et seek()
- Méthode writelines()
- Modules complémentaires : struct, csv, json, xml

Exceptions

- Principe de fonctionnement
- Exceptions pré-définies et arbre d'héritage
- Instructions try ... except ... else ... finally
- Propagation des exceptions
- Déclenchement d'exceptions
- Définition d'une exception

Modules de la bibliothèque standard

- Interaction avec l'interpréteur : module sys
- Interaction avec le système d'exploitation : module os
- Interaction avec le système de fichiers : module os.path
- Expressions rationnelles : module re
- Tests unitaires : instruction assert, module unittest
- Tour d'horizon d'autres modules intéressants de la bibliothèque standard : datetime, math, timeit, urllib, collections, csv, json, unittest, sqlite3

Makina Corpus Formation
49 grande rue Saint-Michel
31400 Toulouse
Email: formation@makina-corpus.com
Tel: 0661342710

ORGANISATION

Formateur
Formation assurée par un expert Python

Moyens pédagogiques et techniques
- Accueil des stagiaires dans une salle dédiée à la formation.
- Documents supports de formation projetés.
- Exposés théoriques
- Étude de cas concrets
- Quiz en salle
- Mise à disposition en ligne de documents supports à la suite de la formation.

Dispositif de suivi de l'exécution de d'évaluation des résultats de la formation
- Feuilles de présence dématérialisées.
- Questions orales ou écrites (QCM)
- Mises en situation.
- Formulaires d'évaluation de la formation.

Délais d'accès
La convocation est envoyée 7 jours avant le début de la formation après réception du bon de commande signé.

Coordonnées de l'équipe pédagogique
- Responsable formation, handicap et votre formateur : Cécile Chardonneau formation@makina-corpus.com
- Suivi facturation : Nathalie Carles Salmon administration@makina-corpus.com