



Développeur Python

Python - Perfectionnement

3 jours (21h00) | ★★★★★ 4,2/5 | PYT-PE | Code RS ou RNCP : RS6962 | Certification TOSA Programmer et automatiser des tâches avec Python (non incluse) | Évaluation qualitative de fin de stage | Formation délivrée en présentiel ou distanciel

Tarif : 2 063 € ht

Formations Informatique > Langages et développement > Développeur Python

Document mis à jour le 06/07/2026

Objectifs de formation

A l'issue de cette formation, vous serez capable de :

- Décrire les subtilités du langage Python et en tirer parti pour écrire des programmes bien structurés, robustes et efficaces
- Gérer le développement en langage Python, de façon approfondie.

Compétences attestées par la certification

Domaine 1 : Langage et syntaxe

- Maîtriser la syntaxe de base et les structures de contrôle : maîtriser la syntaxe de base de Python, incluant les boucles, les conditionnels et la gestion des erreurs, pour développer des programmes robustes et efficaces
- Définir et utiliser des fonctions : créer des fonctions en Python, comprendre la portée des variables, gérer les arguments et retours, et utiliser les fonctions lambda pour simplifier le code et améliorer sa lisibilité
- Appliquer des concepts avancés de programmation : utiliser les décorateurs, générateurs, compréhensions de listes et techniques asynchrones pour écrire un code Python plus performant et concis

Domaine 2 : Structures de données et objets

- Manipuler des types de données primitifs et composés : utiliser les listes, tuples, dictionnaires et ensembles, ainsi que les opérations courantes sur ces structures pour organiser et manipuler efficacement les données en Python
- Utiliser la programmation orientée objet : créer des classes, utiliser l'héritage et le polymorphisme, et appliquer des principes de conception avancée tels que les classes abstraites et les interfaces pour structurer des programmes orientés objet, facilitant la réutilisation et la maintenance du code
- Gérer et optimiser les données : manipuler les données de manière avancée avec des bibliothèques comme pandas et optimiser les structures de données pour améliorer les performances des programmes Python

Domaine 3 : Modules et packages

- Utilisation et création de modules : importer des modules existants, créer de nouveaux modules et structurer le code en modules réutilisables pour favoriser la modularité et la maintenabilité en Python
- Développer et distribuer des packages : créer et configurer des packages avec setuptools pour assurer leur distribution efficace

- Gérer les environnements et les dépendances : utiliser des environnements virtuels, gérer les dépendances avec pip et automatiser les configurations via Docker pour garantir des environnements de développement cohérents et reproductibles

Domaine 4 : Optimisation de code

- Analyser et profiler le code : utiliser des outils de mesure pour analyser la performance du code, identifier les goulots d'étranglement, et appliquer des méthodologies afin d'optimiser l'efficacité du programme
- Améliorer l'efficacité du code : appliquer des techniques pour réduire la complexité du code, optimiser les boucles et utiliser efficacement les ressources afin d'améliorer la performance globale des programmes
- Tester et déboguer : écrire des tests unitaires et d'intégration, utiliser des frameworks comme 'pytest' et appliquer des stratégies avancées de débogage pour garantir la qualité et la fiabilité du code.

Lien pour visualiser le détail de la certification enregistrée au RS :

<https://www.francecompetences.fr/recherche/rs/6962/>

Modalités, méthodes et moyens pédagogiques

Formation délivrée en présentiel ou distanciel* (blended-learning, e-learning, classe virtuelle, présentiel à distance).

Le formateur alterne entre méthode** démonstrative, interrogative et active (via des travaux pratiques et/ou des mises en situation).

Variables suivant les formations, les moyens pédagogiques mis en oeuvre sont :

- Ordinateurs Mac ou PC (sauf pour certains cours de l'offre Management), connexion internet fibre, tableau blanc ou paperboard, vidéoprojecteur ou écran tactile interactif (pour le distanciel)
- Environnements de formation installés sur les postes de travail ou en ligne
- Supports de cours et exercices

En cas de formation intra sur site externe à M2I, le client s'assure et s'engage également à avoir toutes les ressources matérielles pédagogiques nécessaires (équipements informatiques...) au bon déroulement de l'action de formation visée conformément aux prérequis indiqués dans le programme de formation communiqué.

* nous consulter pour la faisabilité en distanciel

** ratio variable selon le cours suivi

Prérequis

Avoir suivi le cours PYT "Python - Par la pratique" ou avoir les compétences équivalentes.

Public concerné

Développeurs, administrateurs et architectes.

Cette formation :

- Est animée par un consultant-formateur dont les compétences techniques, professionnelles et pédagogiques ont été validées par des diplômes et/ou testées et approuvées par l'éditeur et/ou par M2I Formation
- Bénéficie d'un suivi de son exécution par une feuille de présence émarginée par demi-journée par les stagiaires et le formateur.

Programme

Jour 1

Langage

- Appel de fonctions aspects avancés : *args, **kwargs
- Lambda, filter et map
- Utilisation avancée des modèles de données : list, dic, stack, queue
- Utilisation avancée des fonctions : passages d'arguments
- Aspects avancés de la Programmation Orientée Objets (POO)

Exemples de travaux pratiques (à titre indicatif)

- Création d'un programme avec exploitation avancée des collections de Python
- Changement des types de passage d'argument

Programmation multithread

- Concepts de bases : programme, thread, synchronisation
- Gestion de threads : modules thread, threading
- Threads et la Programmation Orientée Objets
- Gestion des aspects concurrentiels : lock, mutex, sémaphores...
- Threads et échanges de données
- Notion de pool de threads

Exemples de travaux pratiques (à titre indicatif)

- Création d'un programme lançant plusieurs threads
- Synchronisation de ces threads pour obtenir un résultat dépendant de traitements parallèles
- Protection des données globales du programme écrit grâce aux mutex et lock
- Ajout d'un sémaphore pour l'accès à des ressources en nombre restreint
- Echange de données entre threads lancés
- Travail sur la notion de pool de threads disponibles

Programmation réseau avec les sockets

- Rappels sur le TCP/IP et concepts de base de l'API socket
- Utilisation du module socket
- Socket en mode connecté : TCP ou stream
- Socket en mode non connecté : UDP ou datagram
- Les sockets et la Programmation Orientée Objets
- Combinaison des sockets et des threads

Exemples de travaux pratiques (à titre indicatif)

- Création d'un programme serveur puis client échangeant des données via les sockets en TCP puis UDP
- Démonstration de l'avantage du multithreading dans ce cas pour les serveurs

Jour 2

Python et XML

- Concepts de base
 - DOM (Document Object Model)
 - SAX (Simple API for XML)
 - Parser...
- Gestion de fichiers XML selon SAX et selon DOM
- Requêtage Xpath et transformation avec XSL

Exemples de travaux pratiques (à titre indicatif)

- Création d'un programme de lecture d'un flux de données de taille importante via SAX
- Mise à jour d'une structure via DOM

Programmation graphique

- Concept de programmation événementielle
- Tkinter : présentation et mise en oeuvre
 - Les widgets standards
 - Mise en page avec grid et pack
- Présentation d'autres bibliothèques graphiques : PyGTK, PyQt

Exemples de travaux pratiques (à titre indicatif)

- Ecriture d'un programme utilisant Tkinter d'échange utilisateur avec l'interface graphique

Persistance de données

- Concepts de base : sérialisation / désérialisation
- Différents modèles de persistance : Pickle
- Persistance texte avec JSON et XML

Les bases de données

- Concepts de base : SQL, tables, curseur
- Gestion de la base de données SQLite et MySQL
- Présentaion du NoSQL

Exemples de travaux pratiques (à titre indicatif)

- Création d'un programme qui sérialise un flux JSON
- Ecriture d'un programme d'accès à une base de données MySQL
- Mise au point de requêtes de lecture, inertion, mise à jour depuis Python

Jour 3

Intégration Python/C et Python/Java

- Présentation générale et mise en oeuvre de SWIG
- Python/C et les packages
 - Natifs C
 - Natifs Java

Exemple de travaux pratiques (à titre indicatif)

- Création d'un programme interfaçant avec des API écrites en C et en Java

Mise au point de programme

- Débogage : exécution pas à pas
- Modes : verbose et trace
- Analyse des performances et profiling

Certification (en option, hors inscription via le CPF)

- Prévoir l'achat de la certification en supplément (ne concerne pas les inscriptions via le CPF pour lesquelles la certification est incluse)
- L'examen (en français) sera passé soit à la fin de la formation, soit ultérieurement dans nos centres de formation
- Il est composé de 35 questions, s'effectuera en ligne et durera en moyenne 1h30
- A l'issue de la certification, le candidat se verra attribuer un score (0 à 1000), correspondant à un niveau (Initial / Basique / Opérationnel / Avancé / Expert) :

- En deçà du score de 551 points, le candidat se verra attribuer une attestation de passage de la certification. Ces niveaux Initial et Basique ne valident cependant pas l'obtention de la certification
- Les niveaux Opérationnel, Avancé et Expert permettent, quant à eux, au titulaire de la certification de faire valoir ses compétences et d'attester de son opérationnalité selon son profil et l'emploi qu'il exerce

Le contenu de ce programme peut faire l'objet d'adaptation selon les niveaux, prérequis et besoins des apprenants.

Modalités d'évaluation des acquis

- En cours de formation, par des études de cas ou des travaux pratiques
- Et, en fin de formation, par un questionnaire d'auto-évaluation et/ou une certification (proposée en option)

Accessibilité de la formation

pagebreakavoidchecked="true";

Le groupe M2I s'engage pour faciliter l'accessibilité de ses formations. Toutes nos formations sont accessibles aux personnes en situation de handicap : les détails de l'accueil des personnes sont consultables sur la page Politique Handicap.

Modalités et délais d'accès à la formation

pagebreakavoidchecked="true";

Les formations M2I sont disponibles selon les modalités proposées sur la page programme. Les inscriptions sont possibles jusqu'à 48 heures ouvrées avant le début de la formation. Dans le cas d'une formation financée par le CPF, ce délai est porté à 11 jours ouvrés.