



Développeur Python

Python - Par la pratique

4 jours (28h00) | ★★★★★ 4,8/5 | PYT | Code RS ou RNCP : RS6202 | Certification TOSA Python (non incluse) | Évaluation qualitative de fin de stage | Formation délivrée en présentiel ou distanciel

Formations Informatique > Langages et développement > Développeur Python

Document mis à jour le 16/06/2024

Objectifs de formation

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

- Identifier les usages courants du langage
- Mettre en pratique le scripting en Python
- Structurer votre code en fonction, classes et modules
- Utiliser des modules existants
- Décrire la programmation réseau avec Python
- Expérimenter la programmation objet en Python.

Compétences attestées par la certification

- Les compétences attestées sont variables en fonction du score obtenu (de 551 à 1000)
- Ce score déterminera un niveau (opérationnel, avancé ou expert) selon lequel vos compétences vous seront attribuées

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

<https://www.francecompetences.fr/recherche/rs/6202/>

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
m2iformation.fr | client@m2iformation.fr | 01 44 53 36 00 (Prix d'un appel local)

Prérequis

Connaître un langage de programmation.

Public concerné

Développeurs.

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 élargée par demi-journée par les stagiaires et le formateur.

Programme

Jour 1

Introduction

- Historique
- Environnement de travail
- Champs d'application
- Scripts, fonctions et classes
- Versions architectures et plateformes pour Python
- Editeurs et environnements de développement (IDE)
- Installation et configuration de Python
- Conformité du codage à PEP8

Exemples de travaux pratiques (à titre indicatif)

- *Acquisition de l'environnement de développement*
- *Configuration de l'installation*

Programme type

- Programme minimal et point d'entrée
- Extension basée sur les packages : import
- Lancement de programmes et passage d'arguments
- Versions compilées de Python
 - .py
 - .pyc
- Console Python
- Encoding : utf-8

Exemples de travaux pratiques (à titre indicatif)

- *Ecriture d'un premier programme de base Python*
- *Echange avec la console*

Types de bases

- Typage faible
- Booléens, numériques
- Les chaînes de caractères

- Conversions de types, casting
- Types agrégés : Tuples, List, Set et dictionnaires

Exemples de travaux pratiques (à titre indicatif)

- Sur le programme Python démarré, travail sur les types de variable en Python
- Ajout de tableaux et travail sur ceux-ci

Jour 2

Les instructions de base

- Commentaires sur une ligne # ou plus """
- Notion de blocs et indentations
- Les opérateurs
 - Affectation
 - Arithmétique
 - Logiques
 - Relationnels
- Saisie et affichage : input, print et formatage
- Structure conditionnelle : la directive "Match"
- Les structures de boucle : While, For, Range()
- Break et continue
- La gestion des exceptions
 - Groupes d'exceptions
 - Notes d'exceptions
- L'import de modules

Exemples de travaux pratiques (à titre indicatif)

- Suite du travail sur le programme Python
- Utilisation des structures de contrôle
- Ajout de la gestion des exceptions et compréhension des modules

Procédures et fonctions

- def(), arguments et valeur de retour
- Variables globales et l'instruction globale
- Arguments par défaut, *args et **kwargs -
- Fonctions lambda

Exemples de travaux pratiques (à titre indicatif)

- Définition et exploitation de nouvelles fonctions avec passage de paramètres et types de retour
- Ecriture de fonctions lambda, compréhension de l'avantage d'utiliser celles-ci

Gestion des fichiers

- Gestion de fichiers et répertoires
- Accès séquentiel, aléatoire
- Le module os, os.path, shutil, zlib

Exemples de travaux pratiques (à titre indicatif)

- Création d'un nouveau programme qui accède aux fichiers et répertoires, différents types d'accès étudiés
- Travail sur les modules standards

Jour 3

Les modules

- Définition de modules et instruction import

- Modules en tant que fichier py et en tant que répertoire
- Résolution des modules
- Module et programme : `__main__`
- Installation de modules : `pip`, `easy_install`

Exemples de travaux pratiques (à titre indicatif)

- *Utilisation de modules systèmes*
- *Création de modules regroupant plusieurs fonctions écrites en cours*

Les classes

- Approche objets
- Classes et instances / objets : `self`
- Constructeur : `__init__`
- Données et méthodes membres
- Héritage multiple

Exemples de travaux pratiques (à titre indicatif)

- *Dans un nouveau programme, création de classes simples, instanciation de celles-ci pour récupérer des objets*
- *Ajout de méthodes et de propriétés*
- *Ajout de l'héritage et travail sur ce modèle*

Jour 4

Les librairies Python

- Accès aux bases de données relationnelles
- Programmation graphique UI avec Tkinter

Exemples de travaux pratiques (à titre indicatif)

- *Création d'un programme d'interface utilisateur utilisant la librairie Tkinter*
- *Ajout dans ce programme d'un accès à une base de données relationnelle puis exploitation de cet accès*

Mise au point de programme

- Débogage : exécution pas à pas

Python et le Web

- Scripts Python en tant que Common Gateway

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 s'effectuera en ligne et durera en moyenne 1h30
- 551 points minimum seront nécessaires pour l'obtention de la certification (le score obtenu attestera d'un niveau de compétence)

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. Les détails de l'accueil des personnes en situation de handicap sont consultables sur la page Accueil et 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.