

Apprendre à programmer avec le langage Python

| |
|--|
| Date et durée |
| Code formation : PYT01FR Durée : 4 jours Nombre d'heures : 28 heures |
| Description |
| <p>Python est un langage de programmation interprété très populaire. Il est utilisé principalement pour le développement web, la création de logiciels, la data science et le scripting. Python offre donc de nombreuses possibilités et fait partie des langages les plus faciles à apprendre.</p> <p>Grâce à notre formation « apprendre à programmer avec le langage Python », vous découvrirez les bases, les modules de la bibliothèque, la gestion des données, etc. Tout le programme est accompagné de travaux pratiques qui permettent de réaliser différents exercices de programmation avec Python et ses outils. Développeur débutant ou confirmé, savoir programmer avec Python est un véritable atout.</p> |
| Objectifs |
| <ul style="list-style-type: none">• Comprendre les fondamentaux du langage Python ;• Connaître les bonnes pratiques de programmation ;• Comprendre et utiliser Python pour la programmation orientée objet ;• Savoir utiliser les différents modules et outils ;• Écrire et exécuter des scripts Python ;• Gérer et exploiter les données ;• Manipuler et parser des fichiers XML ;• Maîtriser et optimiser son environnement de développement. |
| Pré-requis |
| <ul style="list-style-type: none">• Connaître un ou plusieurs langages de programmation ;• Avoir des notions en programmation orientée objet (facultatif). |
| Cette formation s'adresse aux profils suivants |
| <u>Développeur</u> |
| Programme |
| <h3>Le socle de Python</h3> <ul style="list-style-type: none">• Pourquoi programmer avec le langage Python ;• Présentation d'exemple de projet écrit en Python ; |

- Documentation sur les versions de Python ;
- Installation du logiciel Python (dernière version) ;
- Premier pas avec l'interpréteur de commande ;
- Découverte des avantages de l'interpréteur en mode interactif ;
- Création de votre premier programme.

Les fondamentaux du langage Python

- Les méthodes d'organisation du code ;
- Les bonnes pratiques de nommages ;
- Les types et les variables du langage ;
- Les erreurs et les exceptions ;
- Les instructions conditionnelles ;
- Les principes sur l'appel de fonction ;
- La syntaxe Python dans l'interpréteur.

La programmation orientée objet

- Les bases d'instanciation d'une classe ;
- Les techniques et les propriétés des classes ;
- Les modèles d'héritage ;
- Les erreurs et les exceptions ;
- Listing sur les inévitables classes.
- -Les types de données et les variables
- Les meilleures structures de données ;
- Les tuples et les séquences ;
- La programmation fonctionnelle ;
- La structure des piles et des files ;
- L'utilisation des dictionnaires ;
- Les listes en compréhension.

Les annexes Python

- Les itérateurs et les générateurs ;
- Rendu des scripts Python en exécutable ;
- L'association des opérateurs aux listes ;
- Succession de test pour les listes ;
- Comparaison des séquences.

Les modules et les packages Python

- Le principe des modules avec Python ;
- Vue d'ensemble des principaux modules (module standard RE, module standard OS, module standard CSV) ;
- Trouver les modules et paquets Python ;
- Installation de modules tiers ;
- Le dépôt tiers officiel (Python Package Index) ;
- Importation des modules dans Python ;
- Présentation de Virtualenv.

Gérer une base de données avec Python

- Les objets fichier en Python ;
- Documentation sur les fichiers SGBD ;
- Spécification de l'API de base de données ;
- Connexion à une BDD MySQL ;

- Le langage SQL et Python ;
- Introduction sur l'utilisation du toolkit SQLAlchemy.

Le traitement des XML avec Python

- Les bases pour manipuler des fichiers XML ;
- Les méthodes pour parser des données XML ;
- La bibliothèque ElementTree ;
- Le traitement des données avec l'API SAX et DOM.

Maîtriser l'environnement de développement (IDE)

- Exploiter le plugin de développement Eclipse ;
- Utilisation de PyChecker pour les bugs ;
- Contrôle qualité du code avec Pylint ;
- Vérification du style de codage avec le guide PEP 8 ;
- Génération de documentation ;
- Assimilation des avantages à utiliser les outils de test.

Le développement piloté par les tests

- Introduction au test driven development ;
- Présentation du module Doctest et Unittest.

Optimisation et débogage

- Débogage et profilage ;
- Programmation parallèle avec le multi threading ;
- Optimisation de la mémoire cache ;
- Analyse sur l'utilisation de la mémoire.