

Programme de formation Python

• Objectifs

Notre formation vous permettra d'acquérir les connaissances et compétences de base afin de pouvoir créer des programmes en Python. Sur 3 jours, vous aborderez les principes et fonctionnalités essentiels à l'utilisation du langage, de la syntaxe à la programmation orientée objet, en partant d'une base informative et théorique. Grâce à cette formation Python, vous pourrez finalement faire vos premiers pas avec le langage et bénéficier d'une base solide, première étape pour ensuite passer sur des utilisations plus complexes (Big Data, Data Science, etc.).

• Pré requis

Connaissance des bases de la programmation.

• Durée

3 jours

• Public

Developpeurs

• Plan de formation

Introduction au langage Python

Tour d'horizon du langage Python (historique, version actuelle, champs d'application, popularité, ...)

Les environnements de travail et de développement

Les particularités de la syntaxe et les guides associés (PEP8, Pyguide)

Comparaison des versions 2 et 3

Installer et configurer Python

Exemple de cas pratique : installation et/ou configuration de Python.

valeurs, créer des fonctions, utiliser des boucles.

Les modules et packages

Introduction aux modules, aux packages et au PythonPath

Tour d'horizon des modules standards

Installer des modules avec pip

Importer et utiliser des modules, des packages

Exemples de cas pratiques : création et utilisation de modules/packages, importation, installation.

Les bases de Python

Les types de données (strings, nombres, listes, dictionnaires, booléens)

Utiliser des tableaux statiques (tuple) et dynamiques (list)

Convertir les types de données

Les opérateurs logiques et de comparaison

Les boucles et conditions (if, elif, else, for, while, range())

Les fonctions en Python, paramètres, portée des variables

Les fonctions spéciales (map, lambda, filter)

Formuler et utiliser les expressions régulières en Python

Exemples de cas pratiques : faire afficher des

La programmation orientée objet en Python

Introduction à la programmation orientée objet

Définition d'une classe

Méthodes et propriétés d'une classe

Les différents types d'objets en Python

(numériques, itérables, autres)

Comprendre les notions d'objets mutables et immutables

Instancier des objets

L'héritage simple, l'héritage multiple et le polymorphisme

Exemples de cas pratiques : utiliser des classes et des objets dans un programme.

Améliorer son code en Python



Recommandations et bonnes pratiques
(nommage, organisation)
Gérer les exceptions (try, except, raise, finally)
Les assertions
Présentation de la programmation orientée par
les tests (Test Driven Development)
Initiation à Doctest et Unittest
Améliorer la performance d'un programme
Exemples de cas pratiques : identification et
correction d'erreurs, mise en place d'une
structure de contrôle (exception).

Aller plus loin

Le multithreading avec le module threading
Initiation aux interfaces graphiques avec
Tkinter
Introduction au framework Django pour le web

En complément

Les principaux modules
Tour d'horizon des bibliothèques et
frameworks
Présentation d'outils et de ressources utiles