



Smartfuture

Business | Education | Training solutions

Together for
a BRIGHTER
FUTURE

Formation sur

Intermediate Python



Durée
5 jours



<https://urlz.fr/mcmth>



www.smartfuture.tn

Pourquoi suivre la formation sur Intermédiaire Python

La formation en langage Python présente de nombreux avantages, notamment :

- **Facilité d'apprentissage** : Python est réputé pour sa syntaxe simple et intuitive, ce qui en fait un langage très accessible, même pour les débutants en programmation. Sa clarté et sa lisibilité rendent l'apprentissage plus rapide et plus facile par rapport à d'autres langages.
- **Polyvalence** : Python est un langage polyvalent qui peut être utilisé dans divers domaines tels que le développement web, l'analyse de données, l'intelligence artificielle, l'automatisation des tâches, le développement de jeux, etc. Il dispose d'une vaste bibliothèque standard et de nombreuses bibliothèques tierces qui offrent des fonctionnalités étendues.
- **Large communauté et support** : Python bénéficie d'une communauté de développeurs très active et engagée. Il existe de nombreux forums, groupes de discussion et tutoriels en ligne qui offrent un soutien et des ressources supplémentaires aux apprenants. La documentation officielle de Python est également très complète et régulièrement mise à jour.
- **Productivité accrue** : La syntaxe simple de Python permet de développer des programmes plus rapidement et avec moins de code que dans d'autres langages. Python favorise également la réutilisation de code grâce à l'utilisation de modules et de bibliothèques, ce qui permet d'accélérer le processus de développement.
- **Grande disponibilité d'outils et de frameworks** : Python dispose d'une multitude d'outils et de frameworks puissants qui simplifient le développement d'applications complexes. Des frameworks populaires tels que Django et Flask sont largement utilisés pour le développement web, tandis que des bibliothèques comme NumPy et Pandas sont utilisées pour l'analyse de données.
- **Intégration aisée** : Python peut être facilement intégré avec d'autres langages tels que C, C++, Java, etc. Il est souvent utilisé comme langage de script dans des applications plus vastes et peut également être utilisé pour automatiser des tâches dans des systèmes existants.
- **Portabilité** : Les programmes Python sont généralement portables, ce qui signifie qu'ils peuvent être exécutés sur différents systèmes d'exploitation tels que Windows, macOS et Linux sans nécessiter de modifications majeures. Cela facilite le déploiement des applications développées en Python.

Plan du cours

Introduction

- Présentation du langage et comparaison par rapport aux langages existants.
- Présentation de l'environnement.
- Premiers pas en python...

Syntaxe du langage Python

- Identifiants et références.
- Types de données disponibles.
- Les entrées / sorties standards (simples et formatées).
- Les conteneurs natifs (tuples, listes, dictionnaires).
- Les opérateurs et les expressions.
- Manipulation de chaînes de caractères.
- Structures de contrôle et structures itératives.
- Conventions de codage et règles de nommage.
- Les premiers scripts.

Programmation structurée

- Le modèle mémoire de python.
- Les sous-programmes et passage des paramètres.
- Les lambda fonctions
- Structuration du code en modules / package.
- La portée des variables.

Programmation Objet en Python

- Particularités du modèle objet de Python.
- Écriture de classes.
- Héritage et composition.
- Redéfinition surcharge.
- Les opérateurs.
- Bonnes pratiques et modèles de conception courants.
- Utilisation du mécanisme d'exception pour la gestion des erreurs.

Les exceptions

Les fichiers

Les tests unitaires

Programmation parallèle (Thread et processus)

L'introspection et la méta-programmation

- Les classes nouveau type.
- Le modèle objet python du point de vue de l'introspection.
- Les mécanismes de l'introspection.
- Les metaclasses dans python

Les design patterns dans python

- Les décorateurs de fonctions.
- Les décorateurs de classes.
- Implémentation de quelques design patterns (visiteur, observateur, décorateur)

Conclusion

- Analyse critique.
- Ouverture sur les technologies basées sur python