PROGRAMME DE FORMATION EN PROGRAMMATION PYTHON

Description

Se former au développement Python grâce à notre formation Python.

Le python est un langage de programmation devenu incontournable dans le développement de logiciels et de programmes. Le langage Python est particulièrement adapté à la programmation orientée objet. Avec cette formation pour développeur Python, vous aborderez les fonctionnalités de ce langage, ses capacités et son intérêt. Vous verrez les bases du langage de programmation Python, les différents modules et packages. Vous serez enfin formé à la programmation orientée objet ainsi qu'initié à Anaconda, à Spyder et à Jupyter. Bientôt, vous pourrez suivre cette formation Python à distance ou en présentiel.

Cette formation vous offrira une base solide en programmation « **Python** » qui de loin, est l'un des langages de programmation les plus recherchés actuellement sur le marché du travail et le langage d'excellence pour le Machine Learning, le Deep Learning et la Data Science.

Objectifs :

- Connaître l'intérêt de Python, son contexte ;
- Connaître les différents types de données et d'opérateurs ;
- Utiliser les fonctions Python ;
- Installer des modules et des packages en Python ;
- Utiliser Python pour la programmation orientée objet ;
- Manipuler quelques concepts avancés du langage pour une introduction au « Machine Learning » ;

Cible de la formation : Toute personne désireuse de s'initier au Langage Python (**Elève, Etudiant et Professionnels, développeurs**).

Prérequis : Initiation à l'algorithmique, Connaissances basiques en programmation.

Débouchés de la formation :

Analystes programmeurs, Ingénieurs programmeurs, Développeurs Web, Mobiles, Développeurs logiciels, Spécialistes IA, Spécialistes Big Data, Spécialistes Cyber-sécurité et Informatiques, etc.

Formateurs : - **Dr SANYA Max Fréjus Owolabi** / +229 61 33 26 52

- **Ir MAGNIDET Fifamè Merci-Ange** / +229 66 06 26 11

Nombre total d'heures : 70h

Intitulé	**Contenu**	**Masse horaire**
Conseils pour débutants	Quelques indications avant de commencer ce module Background préliminaire : Utilisation de l'outil informatique / Initiation à l'algorithmique	01h
Introduction au langage de programmation Python	C'est quoi Python ? Conseils pour installer Anaconda Notations utilisées Introduction à l'utilisation de Spyder Premier contact avec Python Premier programme	5h

	Commentaires Notion de bloc d'instructions et d'indentation Autres ressources Lancement de Jupyter et création d'un notebook Format Markdown Graphiques dans les notebooks Magic commands JupyterLab	
Variables et Affichage	Définition Types de variables Nommage Écriture scientifique Opérations Fonction type() Conversion de types Note sur la division de deux nombres entiers Note sur le vocabulaire et la syntaxe Minimum et maximum Applications Fonction print() Écriture formatée Écriture scientifique Ancienne méthode de formatage des chaînes de Note sur le vocabulaire et la syntaxe Applications	5h

Chaînes de caractères	Préambule Chaînes de caractères et listes Caractères spéciaux Préfixe de chaîne de caractères Méthodes associées aux chaînes de caractères Extraction de valeurs numériques d'une chaîne de caractères Conversion d'une liste de chaînes de caractères en une chaîne de caractères Exercices	4h
Listes	Définition Utilisation Opération sur les listes Indiçage Tranches Fonctions len(), range() et list() Listes de listes Minimum, maximum et somme d'une liste Applications Méthodes associées aux listes Construction d'une liste par itération Test d'appartenance Copie de listes Liste de compréhension Applications	5h
Containers, dictionnaires, tuples et	Containers Dictionnaires	5h

sets	Tuples Sets et frozensets Récapitulation des propriétés des containers Dictionnaires et sets de compréhension Module collections Applications	
Boucles, Comparaisons et Tests	Boucles for Comparaisons Boucles while Applications Définition de Tests Tests à plusieurs Importance de l'indentation Tests multiples Instructions break et continue Tests de valeur sur des Applications	5h
Fonctions	Principe et généralités Définition Passage d'arguments Renvoi de résultats Arguments positionnels et arguments par mot-clé Variables locales et variables Principe Applications Appel d'une fonction dans une fonction	5h

	Fonctions récursives Portée des variables Portée des listes Règle LGI Recommandations Applications	
Modules	Définition Importation de modules Obtenir de l'aide sur les modules importés Quelques modules courants Module random : génération de nombres aléatoires Module sys : passage d'arguments Module os : interaction avec le système d'exploitation Module NumPy : calculs sur des vecteurs ou des matrices Module matplotlib : génération de graphiques depuis Python Module scipy : Module pandas : analyse de données Applications	6h
Création de modules personnels	Pourquoi créer ses propres modules ? Création d'un module Utilisation de son propre module Docstrings Visibilité des fonctions dans un module Module ou script ? Application	5h

Fichiers	Lecture dans un fichier Écriture dans un fichier Ouvrir deux fichiers avec l'instruction with Note sur les retours à la ligne sous Unix et sous Windows Importance des conversions de types avec les fichiers Respect des formats de données et de fichiers Applications	5h
Programmation Orientée Objet	Construction d'une classe Espace de noms Polymorphisme Héritage Accès et modifications des attributs depuis l'extérieur Bonnes pratiques pour construire et manipuler ses classes Applications	5h
Bonus : Introduction au Machine Learning	Module Sklearn Apprentissage supervisé Apprentissage non supervisé	9h

<u>Moyens et méthodes pédagogiques</u> :

- La formation alterne entre présentations des concepts théoriques et mises en application à travers d'ateliers et exercices pratiques ;
- Les participants bénéficient des retours d'expérience terrains ou professionnels du formateur ou de la formatrice ;
- Un support de cours numérique est fourni aux participants.

Modalités d'évaluation :

- **En amont de la session de formation**, un questionnaire d'auto-positionnement est remis aux participants, afin qu'ils situent leurs connaissances et compétences déjà acquises par rapport au thème de la formation ;
- **En cours de formation**, l'évaluation se fait sous forme d'ateliers, exercices et travaux pratiques de validation, de retour d'observation et/ou de partage d'expérience, en cohérence avec les objectifs visés par la formation ;
- **En fin de session**, le formateur évalue les compétences et connaissances acquises par les apprenants grâce à un questionnaire reprenant les mêmes éléments que l'auto-positionnement, permettant ainsi une analyse détaillée de leur progression.

Critères de validation du Certificat :

70% de l'examen final et au moins 90% de suivi des cours et ateliers organisés.

Références bibliographiques :

- Hetland, M. L. (2006). Beginning Python: From Novice to Professional. http://202.159.8.148:8001/~digilib/index.php?p=show_detail&id=12395
- Pilgrim, M. (2009). Dive into Python 3. Dans Apress eBooks. https://doi.org/10.1007/978-1-4302-2416-7