

Audit « Python »

Client : ECOLE CENTRALE DE NANTES

Formateur : RINCE Romain

Contact audit : Olivia Thilleul - olivia.thilleul@ec-nantes.fr

Lieu de la formation : Dans les locaux de ECOLE CENTRALE DE NANTES à NANTES

Nombre de stagiaires : 3

Durée de la formation : 3 jours - le 01/02/2022 et du 03/02/2022 au 04/02/2022

Les connaissances préalables et les objectifs des stagiaires sont en adéquation avec le contenu et la durée de la formation : ☒ Oui ☐ Non

Pré-requis techniques, puissance machine : Ordinateurs capables de bureautique

Pré-requis d'installation logicielle : Python, version supérieure ou égale à 3.8

Version du produit : = ou > 3.8

Support formateur : ☒ Oui ☐ Non

Si non, support recommandé :

Profil des stagiaires : ☒ Débutant ☐ Intermédiaire ☐ Avancé

Attentes des stagiaires :

Découverte de Jupyter et Matplotlib

Au moins une introduction sur les modules Numpy et Panda, cela aiderait pour mieux se projeter dans l'utilité de Python appliqué à leur besoin métier

Numpy : 1D, rajoute des fonctionnalités sur les tableaux, créer des tableaux...

Panda : manipuler matrice, ajouter une colonne aux matrices, faire des moyennes...

Éventuellement Scipy : fonctions mathématiques, régressions... dépend des tableaux numpy

Complément d'information

A l'issue de la formation les apprenants souhaiteraient pouvoir progressivement remplacer leur utilisation de matlab par Python. Du moins, ils souhaiteraient disposer des compétences de base nécessaires pour comprendre l'environnement de Python et une introduction à ses différentes bibliothèques (quitte à chercher seuls ensuite les fonctions spécifiques qui répondront à leurs besoins précis).

Le but serait donc à l'issue de la formation de savoir (ou être en capacité de chercher les fonctions pour):

- charger des données de fichiers externes (types csv, txt, netcdf...) si possible accéder aux données qui sont en base de données mySQL
 - manipuler ces données (qui sont forcément des matrices 1D, 2D voir 3D) par des opérations simples (multiplier, diviser...), faire des moyennes
 - Sauvegarder les données créées par ces opérations, soit dans un nouveau tableau / Nouvelle variable, soit en rajoutant une colonne à un tableau existant
 - (si possible réaliser des corrélations entre ces données et calculer les indicateurs statistiques correspondant CORR, NRMSE, SI...)
 - réaliser des boucles for, if, etc...
 - Représenter les données sous formes graphiques, a minima tracer y en fonction de x, superposer deux courbes ou nuages de points, si possible graphiques 2D types Rose de vents
-

Objectif de la formation :

Cette formation vous permettra d'acquérir les connaissances et compétences de base afin de pouvoir créer des programmes en Python. 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, développement Web, etc).

Plan de formation : *(merci d'apporter toute modification utile au présent plan de cours en ajoutant ou barrant tout chapitre utile ou exclus du projet de formation. Vous pouvez également copier-coller votre propre plan de cours si vous jugez qu'il est plus en adéquation avec les attentes et les objectifs des stagiaires).*

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)

Bonnes pratiques de nommage et organisation du code

Comparaison des versions 2 et 3

Outils et ressources utiles

Installer et configurer Python

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

Les bases de Python

Les types de données (strings, nombres, booléens, etc.)

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

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

Gérer les exceptions (try, except, raise, finally)

Les assertions

Exemples de cas pratiques : faire afficher des valeurs, créer des fonctions, utiliser des boucles, travailler sur les listes et les chaînes de caractères, identification et correction

d'erreurs, mise en place d'une structure de contrôle (exception).

Les modules et packages

Introduction aux modules, aux packages et au PythonPath

Importer et utiliser des modules, des packages

Tour d'horizon des modules standards

Installer des modules avec pip

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

La programmation orientée objet en Python

Introduction à la programmation orientée objet

Définir les classes et les objets

Méthodes et propriétés d'une classe

Les différents types d'objets en Python (numériques, itérables, autres)

Comprendre les notions d'objets mutables et immutables

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

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

Aller plus loin

Le multithreading avec le module threading

Initiation aux interfaces graphiques avec Tkinter

Introduction au framework Django pour le web

Exemples de cas pratiques : premiers pas et première application avec Tkinter et Django.

LE PLAN DE COURS SUIVANT A-T-IL ETE MODIFIE ?

☒ Oui ☐ Non