

L8 - Le Langage Python

Programmation en Python

Objectifs

- Maîtriser les bases du langage Python
 - Approche modulaire
 - Python Orienté Objet
 - Exceptions
- Comprendre les spécificités de l'interpréteur Python

Matériel

- Support de cours imprimé (en anglais).
- Un PC par binôme.
- Documentation et solutions des exercices sur CDROM.

Prérequis

- Pas de prérequis particuliers

Environnement du cours

- Cours théorique
 - Support de cours au format PDF (en anglais) et une version imprimée lors des sessions en présentiel
 - Cours dispensé via le système de visioconférence Teams (si à distance)
 - Le formateur répond aux questions des stagiaires en direct pendant la formation et fournit une assistance technique et pédagogique
- Au début de chaque demi-journée une période est réservée à une interaction avec les stagiaires pour s'assurer que le cours répond à leurs attentes et l'adapter si nécessaire

Audience visée

- Tout ingénieur ou technicien en systèmes embarqués possédant les prérequis ci-dessus.

Plan du cours

Premier Jour

Introduction

- Historique
- Installation de Python
- L'interpréteur Python
- La ligne de commande Python

Les bases de Python

- Commentaires et instructions
- Variables, données et affectations

- Identificateurs et mot-clefs
- Types de base
- Expressions et opérateurs
 - Les opérateurs arithmétiques
 - Les opérateurs relationnels
 - Les opérateurs de choix
- Simple d'entrée / sortie
- Les structures de données
 - Dictionnaires
 - Ensembles

La structure d'un programme Python

- Instructions complexes
 - Séquences d'instruction
 - Conditions et alternatives
 - Boucles et les itérateurs
- Fonctions et procédures
 - Paramètres
 - variables locales et globales
 - les valeurs des paramètres par défaut
 - Appels de fonctions
- Les espaces de noms
- Les modules
 - Création de modules
 - Importation de fonctions à partir de modules

Second Jour

Structures de données évoluées

- Les chaînes de caractères
 - Indexation et tranches
 - Concaténation et répétition
 - Chaînes Unicode
 - Conversion de chaînes
 - Mise en forme
 - Chaînes de caractères et chaînes d'octets
 - Listes
 - Slicing avancé
 - Extraction et insertion
 - Opérations sur listes
 - Copie de listes
- Tuples
- Dictionnaires
 - Création d'un dictionnaire
 - Opérations sur les dictionnaires
 - Clefs et données

Programmation orientée objet

- Introduction
 - Les classes et les instances
 - Les attributs et les opérations
 - Relations et liens
 - Héritage et polymorphisme
- Pourquoi les objets

- Diviser pour régner
- Le paradigme de l'encapsulation
- Modularité et sécurité
- Avantages
 - Augmentation de la sécurité
 - le développement incrémental
 - Réutilisation de code

Programmation objet en Python

- Le concept de classe en Python
 - Tout est objet
- Définition des classes
 - Attributs de classe et d'instance
 - Méthodes d'instance et méthodes statique
 - Constructeurs

Troisième Jour

Exceptions

- Exceptions et erreurs
 - Types d'erreurs
 - Types d'exceptions
 - Assertions
- Gestion des exceptions
 - Le bloc try
 - Les blocs catch
 - Informations sur l'exception
 - Le bloc finally
- Levée d'exceptions
- L'instruction with

Entrées-sorties en Python

- Interaction avec l'utilisateur
 - Ecriture sur le terminal
 - Lecture à partir du terminal
- Fichiers
- Objets persistants
 - Sérialisation explicite avec repr
 - Sérialisation implicite avec pickle

Interfaces graphiques en Python

- De nombreux toolkits graphiques
 - PyQt
 - PyGTK
 - wxPython
 - Tkinter
- Programmation objet graphique avec Tkinter
 - Programmation événementielle
 - Les widgets Tkinter
 - Mise en page
 - Dessiner sur un canevas
 - Création d'un widget personnalisé

Quatrième Jour

La bibliothèque standard de Python

- Les modules standard
 - sys
 - logging
 - urllib et json
- Programmation réseau
 - Supports
 - Programmes client-serveur
- Programmation multithread
 - Création de threads
 - Partage de données et exclusion mutuelle
 - Synchronisation et communication

Fonctionnalités avancées de Python

- Fonctions avancées
 - Renvoi de valeurs multiples
 - Récupération de paramètres liste ou tuple
- Les méthodes et attributs spéciaux (`__xxxx__`)
- Les Métaclasses
- Les objets invoquables
- Les Conteneurs
 - Création d'un conteneur
 - Indexation d'un conteneur
 - Parcours d'un conteneur
- Les types numériques évolués
- La programmation dynamique
 - Création dynamique de fonctions (lambda)
 - Exécution et évaluation de chaînes de caractères