

Cours de Programmation Python

Enseignant:

Ndombasi Diakusala Joao André, M.Eng. AI

<njadnissi@gmail.com>

Syllabus de Fondamentaux de la programmation en Python en 30 heures:

Ce cours est conçu pour des développeurs ayant des bases en Python et souhaitant développer ses compétences dans l'IoT.

1. **Heure 1-2 : Introduction à l'IoT et Python (2h)**
2. Qu'est-ce que l'Internet des Objets (IoT) ? Concepts fondamentaux
3. Rôle de Python dans l'IoT : avantages et cas d'utilisation
4. Architectures IoT : capteurs, actionneurs, passerelles, cloud
5. Écosystème matériel pour l'IoT (Raspberry Pi, ESP32/ESP8266, Arduino)
6. Mise en place de l'environnement de développement Python pour l'IoT (VS Code, outils spécifiques)
7. **Heure 3-5 : Fondamentaux de l'Électronique et des Capteurs (3h)**
8. Concepts électriques de base : tension, courant, résistance
9. Composants électroniques simples : LED, résistances, boutons
10. Présentation des capteurs courants (température, humidité, lumière, mouvement)
11. Principes de fonctionnement des capteurs analogiques et numériques
12. Lecture de fiches techniques de composants
13. **Heure 6-9 : Python sur Microcontrôleurs et Micro-ordinateurs (4h)**
14. Raspberry Pi avec Python :
15. Installation du système d'exploitation (Raspberry Pi OS)
16. Accès SSH et VNC
17. Gestion des GPIO (General Purpose Input/Output) avec RPi.GPIO
18. Contrôle de LED, boutons, buzzers
19. Lectures analogiques (avec convertisseur A/N externe si nécessaire)
20. MicroPython sur ESP32/ESP8266 (Introduction) :
21. Flashing MicroPython sur les cartes
22. Utilisation de l'IDE Thonny
23. Communication série et bases de script
24. **Heure 10-13 : Communication et Protocoles IoT (4h)**
25. Communication série (UART) :
26. Interfaçage avec des modules série (ex: GPS, capteurs)
27. Utilisation du module pyserial
28. Protocoles sans fil pour l'IoT :
    * 1. Wi-Fi : connexion, modes (client, AP)
      2. Bluetooth Low Energy (BLE) : concepts de base
      3. Introduction à Zigbee et LoRa (concepts, pas d'implémentation profonde)
      4. Protocoles de Messagerie IoT : Protocole MQTT, HTTP/REST dans l'IoT
29. **Heure 14-17 : Collecte et Traitement des Données de Capteurs (4h)**
30. Lecture de données de capteurs spécifiques (DHT11/22, BME280, PIR)
31. Traitement des données brutes : filtrage, moyennage
32. Formatage des données (JSON, CSV)
33. Stockage local des données (fichiers texte, SQLite sur Raspberry Pi)
34. Visualisation des données en temps réel (Matplotlib simple ou librairies web)
35. **Heure 18-21 : Plateformes Cloud IoT et Intégration (4h)**
36. Introduction aux plateformes Cloud IoT (ex: Thingspeak, Adafruit IO, AWS IoT Core, Google Cloud IoT)
37. Création de comptes et configuration de tableaux de bord
38. Envoi de données de capteurs vers le cloud via MQTT/HTTP
39. Réception de commandes du cloud pour contrôler des actionneurs
40. Visualisation et analyse des données sur la plateforme Cloud
41. **Heure 22-24 : Actionneurs et Contrôle à Distance (3h)**
42. Contrôle de relais et de moteurs (servos, moteurs DC)
43. Réception de commandes à distance via MQTT/HTTP pour l'activation d'actionneurs
44. Implémentation de logiques de contrôle simples (ex: allumer une lumière si la pièce est sombre)
45. Création d'une API REST simple sur le Raspberry Pi pour le contrôle local
46. **Heure 25-27 : Sécurité de Base et Bonnes Pratiques IoT (3h)**
47. Considérations de sécurité dans l'IoT : authentification, chiffrement
48. Gestion des identifiants et clés API
49. Mises à jour logicielles et matérielles
50. Bonnes pratiques de codage pour l'IoT (gestion de l'énergie, robustesse)
51. Surveillance et débogage d'applications IoT
52. **Heure 28-30 : Projet Pratique Intégré (3h)**
53. Choix d'un mini projet IoT
54. Conception et implémentation du projet en utilisant les connaissances acquises
55. Débogage et tests
56. Présentation des projets et discussion sur les défis rencontrés
57. Ressources et perspectives d'avenir dans l'IoT avec Python