

# KAÏNA-COM

## CATALOGUE DE FORMATION

### Python avancé

---

**Langage de programmation de premier plan qui permet le développement d'outils et d'applications à diverses fins**



**Nos locaux**  
KAÏNA-COM France  
LE CARRÉ HAUSSMANN II  
6 Allée de la Connaissance  
77 127 Lieusaint



**Contact**  
+33(0)9 50 20 91 64



**E-mail**  
info@kaina-com.fr



**Site Internet**  
www.kaina-com.fr

## KPYT001 – Python avancé

---

**Référence** KPYT001

---

**Niveau**

- Débutant
- Intermédiaire
- Expert

---

**Nombre de jours** Programme de formation :

- 32 heures (4 heures/jour)

---

**Lieu de la formation**

- I: i-learning, Formation individuelle (Formation en ligne)
- V: v-learning, classe virtuelle
- C: c-learning, cours présentiel

**KAINA-COM**

LE CARRÉ HAUSSMANN II,  
6 Allée de la Connaissance  
77127 Lieusaint - France

---

**Prérequis** Connaissance de base de Python  
Un niveau d'anglais business moyen est requise car la formation sera dispensée en anglais.

---

**Public** Développeur Python.

---

*Ce sujet continue à la page suivante*



## KPYT001 – Python avancé, Suite

---

### Objectifs

Cette formation vous permettra d'enrichir toutes les connaissances applicables en programmation en utilisant Python. Le cours comprendra des connaissances théoriques et beaucoup de pratique dans tous les domaines, les participants acquerront la capacité de programmation via des scripts et maîtriser les principales fonctionnalités.

Le cours comprendra les sujets suivants.

Les caractéristiques uniques du langage, y compris les règles de syntaxe, les avantages et les inconvénients du langage par rapport à d'autres langages de programmation.

Installation d'un environnement de travail complet et création d'entrées et de sorties. Les participants auront à pratiquer, création, utilisation et développement dans un environnement Python, y compris l'utilisation de l'éditeur et du débogueur. Utilisation de variables et transtypage de variables (int, float, complex, boolean). Utilisation des instructions «if» - if, else, elif. Opérateurs booléens et opérateurs numériques. Utilisation de variables de chaînes; utilisation des fonctions de chaîne, y compris le découpage. Types de boucles et commandes associées telles que: break, continue, pass, else. Utilisation de modules, y compris - création et importation de modules. Types de données avancés: List, Tuple, Sets, Dictionary - création, manipulations sur les variables.

Lecture et écriture de fichiers: création, ouverture, fermeture, lecture, écriture, ajout, etc.

Types de fichiers - aléatoire et séquence

Fonctions - définition, utilisation, capacités, valeurs de retour.

Gestion des exceptions

Modules Python - Introduction, création, importation

Module OS - Utilisation des fonctionnalités du système d'exploitation

Expressions régulières - définition, règles, expressions d'écriture

OOP (Object Oriented Python)

Journalisation: journalisation dans des fichiers, définition de la gravité et du formatage des messages

Module Datetime

Conseils de performance Python

Tous les sujets pratiques qui seront appris tout au long du cours seront exercés pour acquérir une expertise pratique en combinant divers sujets.

---

*Ce sujet continue à la page suivante*



## KPYT001 – Python avancé, Suite

**Contenu du cours**      Contenu du cours :

Table 1: KPYT001 - Contenu du cours

Chapter	Description
<b>Day 1</b>	<ul style="list-style-type: none"> <li>• Introduction – Programming languages, uniqueness of Python as a programming language, interpreter</li> <li>• Installing of full working environment, familiarization of working environment</li> <li>• Use of comments</li> <li>• Class practice</li> <li>• Presentation of Python variables: int, float, Complex, String, Boolean Strings: Learning of functions and abilities for string manipulations String multiplications, String slicing, Casting, Class practice</li> <li>• Flow control – conditions               <ul style="list-style-type: none"> <li>– Use of: if, elif, else</li> <li>– Boolean operators</li> <li>– Numeric operators including power and modulo</li> <li>– Class practice</li> </ul> </li> <li>• Presentation of the debugger and practical use               <ul style="list-style-type: none"> <li>– Class practice</li> <li>– writing scripts and use of the debugger for troubleshooting</li> </ul> </li> <li>• Loops – for and while (Syntax and Differences)               <ul style="list-style-type: none"> <li>– Additional commands for control (break, continue, pass, else)</li> <li>– Infinite loops / Class practice</li> </ul> </li> <li>• Advanced data types: List – creation, assignment, access               <ul style="list-style-type: none"> <li>– List functions and operators, Slicing, Sort and reverse sort</li> <li>– Class practice</li> </ul> </li> </ul>

*Ce sujet continue à la page suivante*



## KPYT001 – Python avancé, Suite

### Contenu du cours, Suite

Chapter	Description
<b>Day 2</b>	<ul style="list-style-type: none"><li>• More advanced data types: Tuple, Sets, Dictionary – creation, assignment, access</li><li>• Class practice</li><li>• Reading and writing files:<ul style="list-style-type: none"><li>• Creation, open, close</li><li>• Read, write, append</li></ul></li><li>• File types – Random and Sequence</li><li>• Class practice<ul style="list-style-type: none"><li>– Functions</li><li>– definition, usage, capabilities, return values, documentation</li><li>– Class practice</li></ul></li><li>• Lambda functions<ul style="list-style-type: none"><li>– Definition</li><li>– Rules</li><li>– Class practice</li><li>– Exception Handling</li><li>– Introduction</li><li>– When is exception handling actually required? Examples</li><li>– Class practice</li></ul></li><li>• Summary</li></ul>

*Ce sujet continue à la page suivante*



## KPYT001 – Python avancé, Suite

---

### Contenu du cours, Suite

Chapter	Description
<b>Day 3</b>	<ul style="list-style-type: none"><li>• Python modules</li><li>• Introduction</li><li>• Using modules, modules creation, import modules</li><li>• Class practice</li><li>• OS Module – Using operating system functionality</li><li>• Class practice</li><li>• Regular Expressions</li><li>• definition</li><li>• rules</li><li>• write expressions</li><li>• Class practice</li><li>• OOP (Object Oriented Python)</li><li>• Logging to files</li><li>• IDE Output<ul style="list-style-type: none"><li>– Files</li><li>– Setting severity</li><li>– Messages formatting</li><li>– Class practice</li><li>– Datetime Module</li><li>– Class practice</li></ul></li></ul>

---

*Ce sujet continue à la page suivante*



## KPYT001 – Python avancé, Suite

---

### Contenu du cours, Suite

Chapter	Description
<b>Day 4</b>	<ul style="list-style-type: none"><li>• Multiprocessing</li><li>• Class practice</li><li>• Multithreading</li><li>• Class practice</li><li>• Python Performance Tips</li><li>• Class practice</li><li>• Mini Project</li></ul>
<b>The End</b>	<ul style="list-style-type: none"><li>• Summary</li><li>• Q&amp;A</li><li>• Evaluation</li></ul>

---

