

Stage pratique de 5 jour(s)  
Réf : PYT

## Participants

Ce cours s'adresse aux ingénieurs qui souhaitent acquérir une formation à l'approche objet et aux spécificités du langage Python.

## Pré-requis

Connaissances de base en programmation.

Prix 2012 : 2315€ HT

## Dates des sessions

### Paris

26 mar. 2012, 21 mai 2012  
8 oct. 2012, 17 déc. 2012

# Python, programmation objet

## OBJECTIFS

Ce stage intensif a pour objectifs d'initier les participants aux méthodes et réflexes de la programmation orientée objet et de leur apporter la maîtrise opérationnelle du langage Python.

### 1) Syntaxe du langage Python

### 2) Approche Orientée Objet

### 3) Programmation Objet en Python

### 4) Utilisation StdLib

### 5) Outils QA

### 6) Création IHM TkInter

### 7) Interfaçage Python/C

### 8) Conclusion

## 1) Syntaxe du langage Python

- Identifiants et références.
- Types de données disponibles.
- Les conteneurs natifs (tuples, listes, dictionnaires).
- Ecriture de fonctions, structures de contrôle.
- Conventions de codage et règles de nommage.
- Structuration du code en modules.
- Manipulation de chaînes de caractères.
- Utilisation des fichiers.

## 2) Approche Orientée Objet

- Principes du paradigme Objet (classes, héritage, encapsulation, polymorphisme...).
- Présentation de quelques diagrammes UML.
- Notion de modèle de conception (design pattern).

## 3) Programmation Objet en Python

- Particularités du modèle objet de Python.
- Ecriture de classes.
- Nécessité du paramètre self.
- Dérivation et composition.
- Bonnes pratiques et modèles de conception courants.
- Utilisation du mécanisme d'exception pour la gestion des erreurs.

## 4) Utilisation StdLib

- Arguments passés sur la ligne de commande.
- Utilisation d'expressions rationnelles.
- Manipulation du système de fichiers.
- Présentation de quelques modules importants de la bibliothèque standard.
- Empaquetage et installation d'une bibliothèque Python.
- Accès aux bases de données relationnelles.

## 5) Outils QA

- Outils d'analyse statique de code (pylint, pychecker).
- Extraction automatique de documentation.
- Le débogueur de Python.
- Développement piloté par les tests.

## 6) Création IHM TkInter

- Principes de programmation des interfaces graphiques.
- Présentation de la bibliothèque TkInter.
- Présentation des widgets disponibles.

## 7) Interfaçage Python/C

- Utilisation du profileur de code.
- Réécriture d'une fonction Python en C.
- Utilisation d'une bibliothèque C existante avec Swig.

## 8) Conclusion

- Analyse critique de Python.
- Evolution du langage.
- Eléments de webographie et de bibliographie.