PYTHON

Programmation Orientée Objet

Plan de cours N° : 1287

Durée : 5 jours (35h)

PARTICIPANTS / PRE-REQUIS

Développeurs, administrateurs systèmes, ingénieurs scientifiques désireux d'apprendre la programmation orientée objet en Python.

Avoir suivi la formation "Introduction au langage Python " ou disposer de soldes bases sur le langage.

OBJECTIFS PEDAGOGIQUES

Définir les bases et conventions Python. Expérimenter la programmation orientée objet en Python. Gérer les exceptions de manière structurée. Explorer les aspects avancés de POO en Python. Appliquer les bonnes pratiques de développement. Etablir la persistance d'objet en Python. Utiliser des design patterns en Python.

MOYENS PEDAGOGIQUES

Réflexion de groupe et apports théoriques du formateur
Travail d'échange avec les participants sous forme de réunion-discussion
Utilisation de cas concrets issus de l'expérience professionnelle
Validation des acquis par des exercices de synthèse
Alternance entre apports théoriques et exercices pratiques (en moyenne 30 et 70%)
Remise d'un support de cours.
Assistance post-formation d'une durée de 1 an sur le contenu de la formation via notre adresse mail dédiée formateurs@atp-formation.com

MOYENS PERMETTANT LE SUIVI DE L'EXECUTION ET DES RESULTATS

Feuille de présence signée en demi-journée,
Evaluation des acquis tout au long de la formation,
Questionnaire de satisfaction,
Attestation de stage à chaque apprenant,
Positionnement préalable oral ou écrit,
Evaluation formative tout au long de la formation,
Evaluation sommative faite par le formateur ou à l'aide des certifications disponibles.

MOYENS TECHNIQUES EN PRESENTIEL

Accueil des stagiaires dans une salle dédiée à la formation, équipée d'ordinateurs, d'un vidéo projecteur d'un tableau blanc

MOYENS TECHNIQUES DES CLASSES A DISTANCE

A l'aide d'un logiciel comme Teams, Zoom etc... un micro et éventuellement une caméra pour l'apprenant, suivez une formation en temps réel et entièrement à distance. Lors de la classe en ligne, les apprenants interagissent et communiquent entre eux et avec le formateur.
Les formations en distanciel sont organisées en Inter-Entreprise comme en Intra-Entreprise. L'accès à l'environnement d'apprentissage (support de cours, labs) ainsi qu'aux preuves de suivi et d'assiduité (émargement, évaluation) est assuré.
Les participants recevront une convocation avec lien de connexion
Pour toute question avant et pendant le parcours, une assistance technique et pédagogique est à disposition par mail et par téléphone auprès de notre équipe par téléphone au 04.76.41.14.20 ou par mail à contact@atp-formation.com

ORGANISATION

Les cours ont lieu de 9h00-12h30 13h30-17h00

PROFIL FORMATEUR

Nos formateurs sont des experts dans leurs domaines d'intervention
Leur expérience de terrain et leurs qualités pédagogiques constituent un gage de qualité.

ACCESSIBILITE

Les personnes atteintes de handicap souhaitant suivre cette formation sont invitées à nous contacter directement, afin d'étudier ensemble les possibilités de suivre la formation.

MISE A JOUR

28/12/2023

31, av. du Granier
38240 Meylan
Tél. 04 76 41 14 20
Fax 04 76 41 38 45

SARL au capital de 120 000€ - 390 032 050 RCS GRENOBLE
SIRET 390 032 050 00010 - APE 8559A - TVA Intracom. FR03390032050

PYTHON

Programmation Orientée Objet

Plan de cours N° : 1287

Durée : 5 jours (35h)

INTRODUCTION ET RAPPELS PYTHON

- Python, un langage polyvalent
- Python 2 vs Python 3
- La PEP 8 et les conventions de codage Python
- Les types de base
  - int
  - float
  - str
  - bool
- Les principales collections
  - list
  - tuple
  - set
  - dict

INTRODUCTION A LA PROGRAMMATION ORIENTEE OBJET EN PYTHON

- Pourquoi programmer de manière orientée objet ?
  - Concept d'encapsulation
  - Concept d'héritage
- Mise en œuvre de classes
  - Définition et instanciation d'une classe
  - Définition d'attributs de classe
  - Définition de méthodes de classe
  - Constructeurs et destructeurs
  - Mise en œuvre de propriétés
  - Redéfinition d'opérateurs
- Mise en œuvre de l'héritage
  - Surcharge vs redéfinition
  - Notion de polymorphisme
  - L'héritage multiple en Python

GESTION STRUCTUREE DES EXCEPTIONS

- Comment fonctionne la gestion des exceptions en Python ?
- La hiérarchie de classes d'exceptions
- L'instruction try/except/finally
- L'instruction raise
- Définir ses propres classes d'exception

ASPECT AVANCE DE PROGRAMMATION ORIENTEE OBJET EN PYTHON

- Coder une classe "itérable" en Python
  - Le pattern _iter_/_next_
  - Utilisation de vos objets via l'instruction for
  - Comparaison entre itérateurs et générateurs Python
- Définition d'un ContextManager
  - Le pattern _enter_/_exit_
  - Utilisation de vos objets via l'instruction with
  - ContextManager et gestion des exceptions
- Gestion des types abstraits via le module "abc"
  - Qu'est-ce qu'un type abstrait ?
  - Difficultés inhérentes au langage Python
- La réflexion en Python
  - Qu'est-ce que la réflexion en programmation Orientée Objet ?
  - Mise en œuvre de la réflexion
- Développement de décorateur Python
  - Qu'est-ce qu'un décorateur Python ?
  - Mise en œuvre de la réflexion

BONNES PRATIQUES DE DEVELOPPEMENT

- Documenter son code et ses classes avec les docstrings
- Mise en œuvre de tests DocTest
- Mise en œuvre de procédure de tests unitaires avec UnitTest
- Intégration de l'outil UnitTest avec votre IDE
- Le TDD (Test Driven Development)

PERSISTENCE D'OBJET EN PYTHON

- Persistance d'objets en JSON
  - Présentation d'un module JSON
  - Persistance de données Python en JSON
  - Persistance de l'état d'un objet
  - Rechargement d'objet à partir d'un flux JSON
- Persistance d'objets en base de données via SqlAlchemy
  - Qu'est-ce qu'un ORM (Object Relational Mapping) ?
  - Définir les données du mapping
  - Manipulation CRUD de vos données
  - Mapping des relations entre tables
  - Le chargement paresseux (Lazy Loading)

31, av. du Granier
38240 Meylan
Tél. 04 76 41 14 20
Fax 04 76 41 38 45

SARL au capital de 120 000€ - 390 032 050 RCS GRENOBLE
SIRET 390 032 050 00010 - APE 8559A - TVA Intracom. FR03390032050

PYTHON

Programmation Orientée Objet

Plan de cours N° : 1287

Durée : 5 jours (35h)

MISE EN ŒUVRE DE DESIGN PATTERNS (PATRONS DE CONCEPTION) EN PYTHON

- Qu'est-ce qu'un design pattern ?
- Quelles sont les spécificités Python quant à la mise en œuvre de design patterns ?
- Les différentes catégories de design patterns
- Quelques patterns de création (Creational Pattern)
  - Le design pattern Singleton
  - Le design pattern Factory Method
  - Le design pattern Abstract Factory
  - Le design pattern Builder
- Quelques patterns structuraux (Structural Design Pattern)
  - Le design pattern Composite
  - Le desing pattern Proxy
  - Le design pattern Decorator
  - Le design pattern Flyweight
- Quelques patterns comportementaux (Behavioral Patterns)
  - Le pattern Interpretor
  - Le pattern Iterator
  - Le pattern Chain of Responsability
  - Le pattern Observer

31, av. du Granier
38240 Meylan
Tél. 04 76 41 14 20
Fax 04 76 41 38 45

SARL au capital de 120 000€ - 390 032 050 RCS GRENOBLE
SIRET 390 032 050 00010 - APE 8559A - TVA Intracom. FR03390032050