

Objectifs :

Domaine 1 : Langage et syntaxe

Les compétences évaluées :

1.1 Maîtriser la syntaxe de base et les structures de contrôle : maîtriser la syntaxe de base de Python, incluant les boucles, les conditionnels et la gestion des erreurs, pour développer des programmes robustes et efficaces.

1.2 Définir et utiliser des fonctions : créer des fonctions en Python, comprendre la portée des variables, gérer les arguments et retours, et utiliser les fonctions lambda pour simplifier le code et améliorer sa lisibilité.

1.3 Appliquer des concepts avancés de programmation : utiliser les décorateurs, générateurs, compréhensions de listes et techniques asynchrones pour écrire un code Python plus performant et concis.

Domaine 2 : Structures de données et objets

Les compétences évaluées :

2.1 Manipuler des types de données primitifs et composés : utiliser les listes, tuples, dictionnaires et ensembles, ainsi que les opérations courantes sur ces structures pour organiser et manipuler efficacement les données en Python.

2.2 Utiliser la programmation orientée objet : créer des classes, utiliser l'héritage et le polymorphisme, et appliquer des principes de conception avancée tels que les classes abstraites et les interfaces pour structurer des programmes orientés objet, facilitant la réutilisation et la maintenance du code.

2.3 Gérer et optimiser les données : manipuler les données de manière avancée avec des bibliothèques comme pandas et optimiser les structures de données pour améliorer les performances des programmes Python.

Domaine 3 : Modules et packages

Les compétences évaluées :

3.1 Utilisation et création de modules : importer des modules existants, créer de nouveaux modules et structurer le code en modules réutilisables pour favoriser la modularité et la maintenabilité en Python.

3.2 Développer et distribuer des packages : créer et configurer des packages avec setuptools pour assurer leur distribution efficace.

3.3 Gérer les environnements et les dépendances : utiliser des environnements virtuels, gérer les dépendances avec pip et automatiser les configurations via Docker pour garantir des environnements de développement cohérents et reproductibles.

Domaine 4 : Optimisation de code

Les compétences évaluées :

4.1 Analyser et profiler le code : utiliser des outils de mesure pour analyser la performance du code, identifier les goulots d'étranglement, et appliquer des méthodologies afin d'optimiser l'efficacité du programme.

4.2 Améliorer l'efficacité du code : appliquer des techniques pour réduire la complexité du code, optimiser les boucles et utiliser efficacement les ressources afin d'améliorer la performance globale des programmes.

4.3 Tester et déboguer : écrire des tests unitaires et d'intégration, utiliser des frameworks comme 'pytest' et appliquer des stratégies avancées de débogage pour garantir la qualité et la fiabilité du code.

Prérequis

Être professionnel ou futur professionnel devant utiliser Python dans le cadre de son activité

Programme

MODULE 1 : Syntaxe, fonctions et programmation avancé

MODULE 2 : Structures de données et POO

MODULE 3 : Modules, packages et gestion d'environnement

MODULE 4 : Manipulation de données avec Pandas / NumPy

MODULE 5 : Optimisation, performances et qualité du code

MODULE 6 : Projets et automatisations

MODULE 7 : Préparation à la certification TOSA Python

L'accompagnement pédagogique

L'accompagnement pédagogique est assuré avec un formateur qui forme l'apprenant dans un parcours individuel, précisément adapté au besoin de l'apprenant dans le but d'obtenir la certification visée. En dehors des heures de formations prévues, l'organisme prévoit un accompagnement pour s'assurer de la satisfaction de l'apprenant, de la conformité administrative de son dossier ainsi qu'un temps particulier pour la préparation à la certification.

L'accompagnement technique

En début de parcours, un accès e-learning est envoyé à l'apprenant. Son utilisation est facultative mais permet à l'apprenant d'asseoir ses compétences théoriques. Les heures de connexion ne sont jamais décomptées des heures de formation prévues. L'apprenant a un accès à une plateforme développée en interne par le centre de formation pour gérer les documents pédagogiques (Attestations, émargement, etc..).

INFORMATIONS

- **Certification** : RS6962
- **Public** : Professionnels ou futurs professionnels devant utiliser Python dans le cadre de son activité
- **Durée** : A définir lors de votre entretien

FINANCEMENT

Une action de formation peut faire l'objet d'une prise en charge OPCO et CPF. Le centre de formation étudie les dossiers au par cas et vous accompagne dans les démarches de financement.

MOYENS

Pédagogiques :

- Supports de cours théoriques
- Études de cas
- Questions/Réponses

Matériels :

- Ordinateurs / Tablettes / Smartphones
- Paperboard

Méthodes d'évaluation :

Algorithme adaptatif : le niveau des questions s'adapte au niveau du candidat tout au long du déroulement du test

Scoring mathématique IRT (Item Response Theory) : score sur 1000

Typologies de questions : activités interactives

Format du test : 35 questions – 90 minutes

CONTACT

Tél : 01 87 66 02 19

Mail : contact@competencesfirst.fr

Nom de l'accompagnateur : Anthony PITKANITSOS