

## Certification IT Specialist Python

RS :

Les candidats à cet examen démontreront qu'ils peuvent reconnaître, écrire et déboguer du code Python qui résoudra logiquement un problème.

 INFORMATION  
TECHNOLOGY  
SPECIALIST



## Objectifs

Effectuer des opérations à l'aide de types de données et d'opérateurs

- 1.1 Évaluer une expression pour identifier le type de données que Python affectera à chaque variable ; Identifier les types de données que Python attribue aux variables str, int, float et bool
- 1.2 Effectuer des opérations sur les données et les types de données ; Convertir d'un type de données à un autre type ; construire des structures de données ; effectuer des opérations d'indexation et de découpage
- 1.3 Déterminer la séquence d'exécution en fonction de la priorité des opérateurs : Affectation ; Comparaison ; logique ; arithmétique ; identité (est) ; confinement (dans)
- 1.4 Sélectionnez l'opérateur approprié pour obtenir le résultat souhaité : Mission ; Comparaison ; logique ; arithmétique ; identité (est) ; confinement (dans)

Contrôle de flux avec décisions et boucles

- 2.1 Construire et analyser des segments de code qui utilisent des instructions de branchement : if ; else ; expressions conditionnelles imbriquées et composées
- 2.2 Construire et analyser les segments de code qui effectuent l'itération : Tandis que ; pour ; Pause ; Continue ; passe ; boucles imbriquées et boucles qui incluent des expressions conditionnelles composées

Opérations d'entrée et de sortie

- 3.1 Construire et analyser des segments de code qui effectuent des opérations d'entrée et de sortie de fichiers ; ouvrir, fermer, lire, écrire, ajouter, vérifier l'existence, supprimer, avec instruction
- 3.2 Construire et analyser des segments de code qui effectuent des opérations d'entrée et de sortie de la console ; Lire l'entrée de la console, imprimer du texte formaté, utiliser des arguments de ligne de commande

## Documentation et structure du code

- 4.1** Segments de code de document : Utiliser l'indentation, les espaces blancs, les commentaires et les chaînes de documentation ; générer de la documentation en utilisant pydoc
- 4.2** Construire et analyser des segments de code qui incluent des définitions de fonction ; Signatures d'appel, valeurs par défaut, return, def, pass

## Dépannage et gestion des erreurs

- 5.1** Analyser, détecter et corriger les segments de code contenant des erreurs  
Erreurs de syntaxe, erreurs logiques, erreurs d'exécution
- 5.2** Analyser et construire des segments de code qui gèrent les exceptions : essayer, sauf, sinon, enfin, augmenter
- 5.3** Effectuer des tests unitaires : test unitaire, fonctions et méthodes

## Opérations à l'aide de modules et d'outils

- 6.1** Effectuer des opérations de base en utilisant des modules intégrés : math, datetime, io, sys, os, os.path, aléatoire
- 6.2** Résoudre des problèmes informatiques complexes en utilisant des modules intégrés : maths, datetime, aléatoire