

Questions d'entretiens Python – Partie 1

février 20, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Qu'est-ce que Python? Quels sont les avantages d'utiliser Python?

Réponse

Python est un langage de programmation haut niveau, interprété, interactif et orienté objet. Les avantages de Python sont : simple et facile, portable, extensible, une structure de données intégrée et open source.



POURQUOI APPRENDRE PYTHON ? Python est un langage polyvalent, ce qui signifie qu'il peut être utilisé pour créer à peu près tout, et il est facile à utiliser à...**LIRE PLUS**

2. Quel type de langage est Python? Programmation ou script?

Réponse

Python est un langage de script, mais en général, il est considéré comme un langage de programmation interprété à usage général.

3. Pourquoi Python est un langage interprété?

Réponse

Un langage interprété est tout langage de programmation qui n'est pas un langage de bas niveau(code machine) avant l'exécution. Par conséquent, Python est un langage interprété.

4. Qu'est-ce que PEP 8?

Réponse

PEP 8 est une convention de codage, un ensemble de recommandations, sur la façon d'écrire votre code Python plus lisible.

5. Quels types de données sont pris en charge dans Python?

Réponse

Python a cinq types de données standard:

- **Numbers**
- **Strings**
- **Lists**
- **Tuples**
- **Dictionaries**

6. Qu'est-ce que le namespace en Python?

Réponse

Le namespace est un système de nommage utilisé pour s'assurer que les noms de classe, fonctions, etc... sont uniques, pour éviter les conflits de nommage.

7. Quelle est la différence entre list et tuple?

Réponse

La différence entre list et tuple est que list est modifiable tandis que le tuple n'est pas modifiable. Le tuple peut être haché, par exemple comme clé pour les dictionnaires. Voir la [différence entre tuple et liste en Python](#)

8. Qu'est-ce que le PYTHONPATH?

Réponse

C'est une variable d'environnement qui est utilisée lors de l'importation d'un module. Chaque fois qu'un module est importé, PYTHONPATH est également recherché pour vérifier la présence des modules importés dans divers répertoires. L'interpréteur l'utilise pour déterminer le module à charger.

9. Qu'est-ce que lambda en Python?

Réponse

Il s'agit d'une fonction anonyme à expression unique souvent utilisée comme une fonction inline. Exemple:

```
f = lambda x,y : x+y  
print(f(1, 2))
```

Sortie:

3

10. Qu'est ce qu'une variable locale et globale en Python?

Réponse

Variables globales: Les variables déclarées en dehors d'une fonction ou dans l'espace global sont appelées variables globales. Ces variables sont accessibles par n'importe quelle fonction du programme.

Variables locales: Toute variable déclarée à l'intérieur d'une fonction est appelée variable locale. Cette variable est présente dans l'espace local et non dans l'espace global.

Exemple :

```
a=1
def somme():
    b=2
    c=a+b
    print(c)
somme()
```

Sortie:

3

Lorsque vous essayez d'accéder à la variable locale « b » en dehors de la fonction somme(), elle générera une erreur.

Questions d'entretiens Python – Partie 2

février 20, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Python est-elle sensible à la casse ?

Réponse

Oui. Python est un langage sensible à la casse.

2. L'indentation est-elle nécessaire en python ?

Réponse

L'indentation est nécessaire en Python. Elle spécifie un bloc de code. Le code des boucles, classes, fonctions, etc. est spécifié dans un bloc indenté. Elle est généralement réalisée à l'aide de quatre espaces. Si votre code n'est pas indenté, il ne s'exécutera pas correctement et provoquera également des erreurs.

3. A quoi sert la conversion de type en Python?

Réponse

La conversion de type fait référence à la conversion d'un type de données vers un autre.

- **int()** : convertit n'importe quel type en type entier
- **float()** : convertit n'importe quel type en type float
- **ord()** : convertit les caractères en entier
- **str()** : convertit un entier en une chaîne.
- **hex()** : convertit les entiers en hexadécimal
- **oct()** : convertit un entier en octal
- **tuple()** : convertit n'importe quel type en tuple.
- **list()** : convertit n'importe quel type en type de liste.
- **dict()** : convertit un tuple d'ordre (clé, valeur) en dictionnaire.
- **set()** : cette fonction renvoie le type après la conversion en set.
- **complex(real, img)** : convertit les nombres réels en nombre complexe.

4. Que sont les modules en Python? Nommez certains modules intégrés couramment utilisés en Python?

Réponse

Les modules en Python sont des fichiers contenant du code Python. Ce code peut être soit des classes des fonctions soit des variables. Un module Python est un fichier .py contenant du code exécutable.

Certains des modules intégrés couramment utilisés sont:

- **os**

- **sys**
- **math**
- **random**
- **data time**
- **JSON**

5. A quoi sert une fonction en Python?

Réponse

Une fonction est un bloc de code qui n'est exécuté que lors de son appel. Pour définir une fonction Python, le mot clé « def » est utilisé. Exemple:

```
def sayWelcome():  
    print("Welcome to WayToLearnX!")  
sayWelcome();
```

Sortie:

```
Welcome to WayToLearnX!
```

6. Quelle est la différence entre Array et list en Python?

Réponse

```
import array as arr  
tab = arr.array([1,2,3,4], 'a')  
liste = [0, 'str', 5.5]  
print(tab)  
print(liste)
```

Sortie:

```
array([1, 2, 3, 4], 'a')  
  
[0, 'str', 5.5]
```

- **Array doit être déclaré. par contre les listes, car elles sont intégrées à Python. Dans l'exemple ci-dessus, vous constatez que les listes sont créées en mettant simplement une séquence d'éléments entre crochets. La création d'un array, nécessite une fonction spécifique du module Array (c'est-à-dire `Array.array()`) ou du package NumPy (c'est-à-dire `numpy.array()`). Pour cette raison, les listes sont utilisées plus souvent que les tableaux.**
- **Array peuvent stocker des données de manière très compacte et sont plus efficaces pour stocker de grandes quantités de données.**
- **Les arrays sont parfaits pour les opérations numériques; les listes ne peuvent pas gérer directement les opérations mathématiques. Par exemple, vous pouvez diviser chaque élément d'un array par le même nombre avec une seule ligne de code. Si vous essayez la même chose avec une liste, vous obtiendrez une erreur.**

7. A quoi sert le mot-clé « pass » en Python?

Réponse

Supposons que nous ayons une boucle ou une fonction qui n'est pas encore implémentée, mais que nous voulons l'implémenter après un certain temps. Ils ne peuvent pas avoir un corps vide. L'interpréteur se plaindrait. Donc, nous utilisons l'instruction « pass » pour construire un corps qui ne fait rien. Exemple:

```
liste = ['a', 'b', 'c', 'd']  
for val in liste:  
    pass
```

Nous pouvons également faire la même chose dans une fonction ou une classe vide.

```
def myFunction(args):  
    pass  
class myClass:
```

pass

8. En Python, à quoi sert les itérateurs?

Réponse

En Python, les itérateurs sont utilisés pour itérer un groupe d'éléments et des collections comme list. Exemple :

```
liste = ['a', 'b', 'c', 'd']
it = iter(liste)
print(next(it))
print(next(it))
```

Sortie:

```
a
b
```

9. Comment pouvez-vous copier un objet en Python?

Réponse

Pour copier un objet en Python, vous pouvez essayer « copy.copy() » ou « copy.deepcopy() ».

Note : Vous ne pouvez pas copier tous les objets, mais la plupart d'entre eux.

Exemple:

```
import copy
liste1 = ['a', 'b', 'c', 'd']
liste2 = copy.copy(liste1)
print(liste2)
liste3 = copy.deepcopy(liste1)
print(liste3)
```

Sortie:

```
['a', 'b', 'c', 'd']
```



```
['a', 'b', 'c', 'd']
```

10. Quelle est la différence entre Xrange et range?

Réponse

« Xrange » renvoie un objet xrange, tandis que « range » renvoie une liste.



DIFFÉRENCE ENTRE RANGE ET XRANGE EN PYTHON Les deux fonctions range() et xrange() peuvent être utilisées pour itérer un certain nombre de fois dans des boucles for en Python. Dans Python 3,...[LIRE PLUS](#)

Questions d'entretiens Python – Partie 3

février 20, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Comment convertir un nombre en une chaîne?

Réponse

Afin de convertir un nombre en une chaîne, utilisez la fonction intégrée « `str()` ». Si vous voulez une représentation octale ou hexadécimale, utilisez la fonction intégrée « `oct()` » ou « `hex()` ». Exemple:

```
s = str(5)
```

2. A quoi sert un indice négatif en Python?

Réponse

Les séquences Python peuvent être indexées en utilisant des nombres positifs et négatifs. Pour un indice positif, 0 est le premier indice, 1 est le deuxième indice et ainsi de suite. Pour un indice négatif, (-1) est le dernier indice et (-2) est l'avant-dernier indice, etc.

3. Qu'est-ce que « docstring » en Python?

Réponse

Une chaîne de documentation (doc string) Python est connue sous le nom de docstring, c'est un moyen de documenter les fonctions, modules et classes Python. Exemple:

```
def add(a, b):  
    """Cette fonction calcule la somme de deux nombres."""  
    return a + b
```

4. Qu'est-ce qu'un module et un package en Python?

Réponse

En Python, le module est un moyen pour structurer un programme. Chaque programme Python est un module, qui importe d'autres modules comme des objets et des attributs.

Le dossier où se trouve le programme Python est un package de modules. Un package peut avoir des modules ou des sous-packages.

5. Expliquez comment pouvez-vous rendre un script Python exécutable sur Unix?

Réponse

Pour rendre un script Python exécutable sur Unix, vous devez faire deux choses:

- Le mode du fichier script doit être exécutable et
- la première ligne doit commencer par # (`#!/usr/local/bin/python`)

6. Expliquez comment supprimer un fichier en Python?

Réponse

En utilisant la commande « `os.remove(myfile)` » ou « `os.unlink(myfile)` ».

Voir aussi : [Comment supprimer un fichier ou un répertoire en Python](#)

7. Expliquez comment pouvez-vous générer des nombres aléatoires en Python?

Réponse

Pour générer des nombres aléatoires en Python, vous devez importer le module `random`.

```
import random
random.random()
```

Cela renvoie un nombre aléatoire dans la plage [0,1]

8. A quoi sert l'utilisation de l'opérateur // en Python?

Réponse

Il s'agit d'un opérateur de division entière, qui est utilisé pour diviser deux opérandes affichant uniquement les chiffres avant la virgule décimale. Par exemple, $8 // 2 = 4$ et $8,0 // 4,0 = 2,0$.

9. A quoi sert la fonction split en Python?

Réponse

La fonction `split()` divise une chaîne en chaînes plus courtes en utilisant le séparateur défini. Il renvoie une liste de tous les mots présents dans la chaîne.

Exemple:

```
str = "Welcome to WayToLearnX"
liste = str.split()
print(liste)
```

Sortie:

```
['Welcome', 'to', 'WayToLearnX']
```

Voir aussi : [Comment transformer une chaîne de caractère en liste python](#)

10. Qu'est-ce que Flask?

Réponse

Flask est un framework web. Cela veut dire que flask vous fournit des outils, des bibliothèques et des technologies qui vous permettent de créer une application Web.

Questions d'entretiens Python – Partie 4

février 20, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str)
```

Réponse

Hello World!

2. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str[0])
```

Réponse

H

3. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str[1:4])
```

Réponse

ell

4. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str[1:])
```

Réponse

ello World!

5. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str * 3)
```

Réponse

Hello World!Hello World!Hello World!

6. Quelle est la sortie du code suivant?

```
str = 'Hello World!'
print(str + " bye!")
```

Réponse

Hello World! bye!

7. A quoi sert le décorateur en Python?

Réponse

Le décorateur en Python est un changement relatif que vous effectuez dans la syntaxe Python pour ajuster rapidement les fonctions.

8. A quoi sert __init__?

Réponse

`__init__` est une méthode ou un constructeur en Python. Cette méthode est automatiquement appelée pour allouer de la mémoire lorsqu'un nouvel objet/instance d'une classe est créé. Toutes les classes ont la méthode `__init__`.

Voici un exemple d'utilisation.

```
class Person:

def __init__(self, name, address):

self.name = name

self.address = address

p = Person("Alex", "Paris")

print(p.name)

print(p.address)
```

Sortie:

```
Alex

Paris
```

9. Pourquoi self en Python ?

Réponse

Self est une instance ou un objet d'une classe. En Python, cela est explicitement inclus comme premier paramètre. Cependant, ce n'est pas le cas en Java où il est facultatif. Il aide à différencier les méthodes et les attributs d'une classe avec des variables locales.

La variable `self` dans la méthode `init` fait référence à l'objet nouvellement créé tandis que dans d'autres méthodes, elle fait référence à l'objet dont la méthode a été appelée.

10. Que fait l'instruction suivant `::-1` ?

Réponse

[::-1] est utilisé pour inverser l'ordre d'un tableau ou d'une séquence. Par exemple:

```
liste = [1,2,3,4,5,6]
print(liste[::-1])
```

Sortie:

```
[6, 5, 4, 3, 2, 1]
```

Questions d'entretiens Python – Partie 5

février 20, 2020 · Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Comment pouvez-vous mélanger aléatoirement les éléments d'une liste en Python?

Réponse

```
from random import shuffle
liste = ['PHP', 'Java', 'C', 'C++', 'Python']
shuffle(liste)
print(liste)
```

Sortie:

```
['Python', 'C++', 'Java', 'PHP', 'C']
```

2. Comment écrivez-vous des commentaires en python?

Réponse

Les commentaires en Python commencent par le caractère #. Exemple:

```
# Ceci est un commentaire en Python
```

3. A quoi sert les générateurs en python?

Réponse

Des fonctions qui renvoient un ensemble d'éléments itérable sont appelées générateurs.

4. Comment capitaliserez-vous la première lettre d'une chaîne?

Réponse

En Python, la méthode « `capitalize()` » met en majuscule la première lettre d'une chaîne. Si la chaîne se compose déjà d'une lettre majuscule au début, elle renvoie la chaîne d'origine. Exemple :

```
str='waytolearnx'  
print(str.capitalize()) #Waytolearnx
```

5. Comment convertir une chaîne en minuscules?

Réponse

Pour convertir une chaîne en minuscules, la fonction « `lower()` » peut être utilisée. Exemple:

```
str='WAYTOLEARNX'  
print(str.lower()) #waytolearnx
```

6. Comment commenter plusieurs lignes en python?

Réponse

Les commentaires sur plusieurs lignes apparaissent sur plusieurs lignes. Toutes les lignes à commenter doivent être préfixées par un #.

7. A quoi sert *args, **kwargs?

Réponse

Nous utilisons *args lorsque nous ne savons pas combien d'arguments vont être passés à une fonction, ou si nous voulons passer une liste ou un tuple en arguments à une fonction. **kwargs est utilisé lorsque nous ne savons pas combien d'arguments de mot-clé seront passés à une fonction, ou il peut être utilisé pour passer les valeurs d'un dictionnaire en tant qu'arguments de mot-clé. Les identifiants args et kwargs sont une convention, vous pouvez également utiliser *tata et **toto mais ce ne serait pas une bonne pratique.

Voir aussi : [*args et **kwargs en Python](#)

8. A quoi sert la fonction len()?

Réponse

Il est utilisé pour déterminer la longueur d'une chaîne, d'une liste, d'un tableau, etc.

Exemple:

```
str='waytolearnx'
```

```
print(len(str)) #11
```



FONCTION LEN() - PYTHON La fonction len() renvoie le nombre des éléments (ou la longueur) dans un objet. Si vous ne transmettez pas un argument ou vous transmettez un...**LIRE PLUS**

9. A quoi sert la fonction sub()?

Réponse

Recherche toutes les sous-chaînes où l'expression régulière correspond, puis les remplacer par une chaîne différente. Exemple:

```
import re

result = re.sub('Y', 'X', 'WayToLearnY')

print(result) #WayToLearnX
```

10. A quoi sert la fonction subn()?

Réponse

Il est similaire à « sub() » et renvoie également la nouvelle chaîne avec le nombre de remplacements. Exemple:

```
import re

result = re.subn('Y', 'X', 'WayToLearnY')

print(result) # ('WayToLearnX', 1)
```

Questions d'entretiens Python – Partie 6

février 20, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Comment ajouter des valeurs à un array en Python?

Réponse

Des éléments peuvent être ajoutés à un tableau à l'aide des fonctions suivantes `append()`, `extend()` et `insert(i, val)`. Exemple:

```
import array as arr

a = arr.array('i', [1, 2, 3])

a.append(4)

print(a)

a.extend([5, 6, 7])

print(a)

a.insert(0, 8)

print(a)
```

Sortie:

```
array('i', [1, 2, 3, 4])

array('i', [1, 2, 3, 4, 5, 6, 7])

array('i', [8, 1, 2, 3, 4, 5, 6, 7])
```

2. Comment supprimer des valeurs d'un array en Python?

Réponse

Les éléments du array peuvent être supprimés à l'aide de la méthode « `pop()` » ou « `remove()` ». La différence entre ces deux fonctions est que la première renvoie la valeur supprimée. Tandis que la deuxième renvoie le tableau avec la valeur supprimée.

Exemple:

```
import array as arr

a=arr.array('i', [10, 20, 30, 40, 50])

print(a.pop())

print(a.pop(2))
```

```
a.remove(10)

print(a)
```

Sortie:

```
50

30

array('i', [20, 40])
```

3. Quelle est la différence entre la fonction remove() et l'instruction del?

Réponse

Vous pouvez utiliser la fonction « remove() » pour supprimer un objet spécifique d'une liste.

Si vous souhaitez supprimer un objet à un emplacement spécifique (index) dans la liste, vous pouvez utiliser « del » ou « pop ».

4. A quoi sert la fonction swapcase() en Python?

Réponse

Il s'agit d'une fonction de chaîne qui convertit tous les caractères majuscules en minuscules et vice versa. Exemple:

```
str = "WAYTOLEARNX"

print(str.swapcase())

str = "waytolearnx"

print(str.swapcase())
```

Sortie:

```
waytolearnx
```

5. Comment supprimer les espaces d'une chaîne en Python?

Réponse

Pour supprimer les espaces d'une chaîne, Python fournit la fonction `strip(str)`. Cette fonction renvoie une copie de la chaîne après avoir supprimé les espaces, en cas de succès. Sinon, renvoie la chaîne d'origine. Exemple:

```
str = " WayToLearnX "  
  
print(str)  
  
print(str.strip())
```

Sortie:

```
WayToLearnX  
  
WayToLearnX
```

6. Pourquoi utilisons-nous la fonction « `join()` » en Python?

Réponse

La fonction « `join(str)` » est une fonction qui renvoie une chaîne dans laquelle les éléments ont été joints par le séparateur `sep`. Exemple:

```
str1 = "aaa"  
  
sep = "|||"  
  
str2 = str1.join(sep)  
  
print(str2)
```

Sortie:

7. Quelles sont les différences entre Python 2.x et Python 3.x?

Réponse

Python 2.x est une ancienne version de Python. Python 3.x est une version plus récente. Python 2.x est désormais hérité. Python 3.x est le présent et l'avenir de ce langage.

La différence clé entre Python2 et Python3 réside dans l'instruction print(). En Python 2, print est utilisé sans parenthèses, et en Python 3, print est utilisé avec des parenthèses.

La chaîne en Python2 est implicitement ASCII et en Python3, c'est Unicode.

La méthode xrange() a été supprimée de la version Python 3.

Voir aussi : [Différences entres python 2 et python 3](#)

8. Quelle est la méthode la plus courte pour ouvrir un fichier texte et afficher son contenu?

Réponse

La façon la plus courte d'ouvrir un fichier texte est d'utiliser la commande « with » de la manière suivante:

```
with open("myfile", "r") as fp:  
    data = fp.read()  
    print(data)
```

9. A quoi sert la fonction enumerate() en Python?

Réponse

La fonction enumerate() est utilisée pour parcourir une séquence et récupérer la position d'index et sa valeur correspondante en même temps.

```
for i,v in enumerate(['A','B','C']):  
    print(i,v)
```

Sortie:

```
(0, 'A')  
  
(1, 'B')  
  
(2, 'C')
```

10. Quelle est la sortie de

```
print(['Hello']*2)
```

?

Réponse

['Hello', 'Hello']

Questions d'entretiens Python – Partie 7

février 21, 2020 · Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Est-ce que Python est un langage de programmation orienté objet?

Réponse

Python est un langage de programmation orienté objet. Cependant, Python peut être traité comme un langage procédural et structurel.

2. Comment importer des modules en python?

Réponse

Les modules peuvent être importés à l'aide du mot clé « import ». Vous pouvez les importer de trois manières:

```
import array # en utilisant le nom du module
```

```
import array as arr # en utilisant le nom d'alias
```

```
from array import * # importe tout ce qui est présent dans le module array
```

3. Comment créées des classes en Python?

Réponse

Une classe en Python est créée à l'aide du mot clé 'class'. Exemple:

```
class Person:

def __init__(self, name):

self.name = name

p = Person("Alex")

print(p.name)
```

Sortie:

Alex

Voir aussi : [Les classes et l'héritage en Python](#)

4. Python prend-il en charge l'héritage multiple?

Réponse

L'héritage multiple signifie qu'une classe fille peut hériter de plusieurs classes mères. Python prend en charge l'héritage multiple, contrairement à Java.

Voir aussi : [Les classes et l'héritage en Python](#)

5. Qu'est-ce que le polymorphisme en Python?

Réponse

Le polymorphisme signifie la capacité de prendre plusieurs formes. Ainsi, par exemple, si la classe mère a une méthode nommée myMethod, la classe fille peut également avoir une méthode avec le même nom myMethod ayant ses propres paramètres et variables. Python permet le polymorphisme.

6. Définir l'encapsulation en Python?

Réponse

L'encapsulation est l'un des concepts fondamentaux de [la programmation orientée objet \(POO\)](#). Il décrit l'idée de regrouper les données et les méthodes qui fonctionnent sur ces données au sein d'une unité, par exemple, une classe en Python.

7. Comment faites-vous l'abstraction des données en Python?

Réponse

L'abstraction des données ne fournit que les détails requis et cache l'implémentation à l'extérieur. Il peut être réalisé en Python en utilisant des interfaces et des classes abstraites.

8. Comment créer une classe vide en Python?

Réponse

Une classe vide est une classe qui n'a pas de code défini dans son bloc. Il peut être créé à l'aide du mot-clé « pass ». Exemple:

```
class MaClasse:  
  
    pass
```

9. Écrivez un programme en Python pour afficher un triangle d'étoiles de N lignes.

Réponse

```
def star(n):  
  
    for i in range(n):
```

```
print(' '* (n-i-1) + '*' * (2*i+1))  
  
star(5)
```

Sortie:

```
  *  
  
 ***  
  
*****  
  
*****  
  
*****
```

10. Écrivez un programme en Python pour vérifier si un nombre est premier.

Réponse

```
nbr = int(input("Entrer un nombre"))  
  
if nbr > 1:  
    for i in range(2, nbr):  
        if (nbr % i) == 0:  
            print("N'est pas un nombre premier")  
            break  
    else:  
        print("Est un nombre premier")  
else:  
    print("N'est pas un nombre premier")
```

Questions d'entretiens Python – Partie 8

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Écrivez un programme en Python pour vérifier si une séquence est un Palindrome.

Réponse

Algorithme palindrome en Python

février 21, 2020 Aucun commentaire fonction palindrome, palindrome python 3, palindrome python recursive, phrase est un palindrome python, programme palindrome, teste si une chaîne est un palindrome

Un nombre est un palindrome si il s'écrit de la même manière après l'inversion de ce dernier.

Exemples:

232, 191, 22022, 111, 666, 12012

La logique du programme

- Récupère le nombre/la chaîne à vérifier
- Garder le nombre/la chaîne dans une variable temporaire
- Inverser le nombre/la chaîne
- Comparer le nombre/la chaîne temporaire avec le nombre/la chaîne inversé

- Si les deux nombres/chaînes sont identiques, afficher « xxxx est un palindrome »
- Sinon, afficher « xxxx n'est pas un palindrome »

Programme 1: pour vérifier si le nombre/la chaîne est palindrome ou non

```
def check_palindrome(v):  
    reverse = v[::-1]  
    if (v == reverse):  
        return True  
    return False  
var = input("Entrez une valeur: ")  
if(check_palindrome(var)):  
    print("L'entrée est un palindrome")  
else:  
    print("L'entrée n'est pas un palindrome")
```

```
Entrez une valeur: 232  
L'entrée est un palindrome
```

Programme 2: pour vérifier si le nombre/la chaîne est palindrome ou non de manière récursive

```
def check_palindrome(v):  
    if len(v) < 1:  
        return True  
    else:  
        if v[0] == v[-1]:  
            return check_palindrome(v[1:-1])  
        else:  
            return False  
var = input("Entrez une valeur: ")  
if(check_palindrome(var)):  
    print("L'entrée est un palindrome")  
else:  
    print("L'entrée n'est pas un palindrome")
```

```
Entrez une valeur: 232
L'entrée est un palindrome
```

Programme 3: pour vérifier si le nombre/la chaîne est palindrome ou non

```
var = input("Entrez une valeur: ")
if(var == var[::-1]):
    print("L'entrée est un palindrome")
else:
    print("L'entrée n'est pas un palindrome")
```

```
Entrez une valeur: 232
L'entrée est un palindrome
```

2. A quoi sert la fonction map en Python?

Réponse

La fonction map exécute la fonction donnée comme premier argument sur tous les éléments de l'itérable donné comme deuxième argument. Si la fonction donnée prend plus d'un argument, alors de nombreux itérables sont donnés. Exemple:

```
def somme(v):
    return v + v

nbr = [1, 2, 3, 4, 5, 6]
s = map(somme, nbr)
print(list(s))
```

Sortie:

```
[2, 4, 6, 8, 10, 12]
```

3. Quel est la sortie du code suivant?

```
list = [ 5, 'xyz' , 1.5, 90.3 ]
```

```
print(list)
```

Réponse

[5, 'xyz', 1.5, 90.3]

4. Quel est la sortie du code suivant?

```
list = [ 5, 'xyz' , 1.5, 90.3 ]
```

```
print(list[1])
```

Réponse

xyz

5. Quel est la sortie du code suivant?

```
list = [ 5, 'xyz' , 1.5, 90.3 ]
```

```
print(list[1:3])
```

Réponse

['xyz', 1.5]

6. Quel est la sortie du code suivant?

```
list = [ 5, 'xyz' , 1.5, 90.3 ]
```

```
print(list[2:])
```

Réponse

[1.5, 90.3]

7. Quel est la sortie du code suivant?

```
list = [ 5, 'xyz' ]
```

```
print(list * 2)
```

Réponse

[5, 'xyz', 5, 'xyz']

8. Quel est la sortie du code suivant?

```
list1 = [ 1, 2, 3, 4 ]  
list2 = [ 5, 6 ]  
print(list1 + list2)
```

Réponse

[1, 2, 3, 4, 5, 6]

9. Quel est la sortie du code suivant?

```
tuple = ( 5, 'xyz' , 1.5, 90.3 )  
print(tuple)
```

Réponse

(5, 'xyz', 1.5, 90.3)

10. Quel est la sortie du code suivant?

```
tuple = ( 5, 'xyz' , 1.5, 90.3 )  
print(tuple[0])
```

Réponse

5

Questions d'entretiens Python – Partie 9

février 21, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test
recrutement, Test technique

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python »

afin de pouvoir y répondre en toute confiance.

1. Quel est la sortie du code suivant?

```
tuple = ( 5, 'xyz' , 1.5, 90.3 )  
  
print(tuple[1:3])
```

Réponse

('xyz', 1.5)

2. Quel est la sortie du code suivant?

```
tuple = ( 5, 'xyz' , 1.5, 90.3 )  
  
print(tuple[2:])
```

Réponse

(1.5, 90.3)

3. Quel est la sortie du code suivant?

```
tuple = ( 5, 'xyz' )  
  
print(tuple * 2)
```

Réponse

(5, 'xyz', 5, 'xyz')

4. Quel est la sortie du code suivant?

```
tuple1 = ( 1, 2, 3, 4 )  
  
tuple2 = ( 5, 6 )  
  
print(tuple1 + tuple2)
```

Réponse

(1, 2, 3, 4, 5, 6)

5. A quoi servent les dictionnaires en Python?

Réponse

Les dictionnaires en Python sont une sorte de table de hachage. Ils fonctionnent comme des tableaux associatifs et se composent de paires clé-valeur. La clé peut être n'importe quel type Python, mais ce sont généralement des nombres ou des chaînes. Les valeurs peuvent être n'importe quel objet Python.

6. Comment créer un dictionnaire en Python?

Réponse

Les dictionnaires sont entourés par des accolades ({}), et les valeurs sont accessibles à l'aide des crochets carrés ([]). Exemple:

```
dict = {'alex': 25, 'bob': 22}
dict['bob'] = 15
print(dict)
```

Sortie:

```
{'bob': 15, 'alex': 25}
```

7. Comment récupérer toutes les clés d'un dictionnaire?

Réponse

En utilisant la fonction « dictionary.keys() », nous pouvons obtenir toutes les clés d'un dictionnaire. Exemple:

```
dict = {'alex': 25, 'bob': 22}
print(dict.keys())
```

Sortie:

```
dict_keys(['alex', 'bob'])
```

8. Comment récupérer toutes les valeurs d'un dictionnaire?

Réponse

En utilisant la fonction « `dictionary.values()` », nous pouvons obtenir toutes les valeurs d'un dictionnaire. Exemple:

```
dict = {'alex': 25, 'bob': 22}
print(dict.values())
```

Sortie:

```
dict_values([25, 22])
```

9. Comment convertir une chaîne en entier en Python?

Réponse

`int(n [, base])` – Convertit `n` en entier. `base` spécifie la base si `n` est une chaîne.

Voir aussi : [Convertir une chaîne de caractères en entier python](#)

10. Comment convertir une chaîne en un long en Python?

Réponse

`long(n [, base])` : Convertit `n` en un entier long. `base` spécifie la base si `n` est une chaîne.

Questions d'entretiens Python – Partie 10

février 21, 2020 Aucun commentaire correction, corrigé, entretien, pdf, QCM, question, réponse, test recrutement, Test technique

Q

uestions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage

systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Comment créer un dictionnaire en utilisant des tuples en Python?

Réponse

dict(d) – Crée un dictionnaire. d doit être une séquence de tuples (clé, valeur). Exemple:

```
tuple = (('alex', 20), ('bob', 15))  
dict = dict((a, b) for a, b in tuple)  
print(dict)
```

Sortie:

```
{'bob': 15, 'alex': 20}
```

2. À quoi sert l'opérateur **?

Réponse

Effectue un calcul exponentiel sur les opérateurs. 42 => 4 à la puissance de 2. Exemple:**

```
print(4**2) #16
```

3. Comment vérifier que tous les caractères dans une chaîne sont en minuscules ?

Réponse

islower() – Retourne « True » si tous les caractères dans la chaîne sont en minuscules, sinon, elle renvoie « False ».

4. Comment vérifier que tous les caractères dans une chaîne sont des chiffres ?

Réponse

isnumeric() – Renvoie « True » si une chaîne unicode ne contient que des caractères numériques, sinon, elle renvoie « False ».

5. Comment vérifier que tous les caractères dans une chaîne sont des espaces ?

Réponse

isspace() – Renvoie « True » si la chaîne ne contient que des des espaces, sinon, elle renvoie « False ».

6. Comment vérifier que tous les caractères dans une chaîne sont en majuscules ?

Réponse

isupper() – Retourne « True » si tous les caractères de la chaîne sont en majuscules, sinon, elle renvoie « False ».

7. Comment remplacer toutes les occurrences d'une chaîne par une nouvelle chaîne?

Réponse

replace(old, new) – Remplace toutes les occurrences de old dans la chaîne par de nouvelles occurrences. Exemple:

```
str = "Hello World!"  
print(str.replace("World", "WayToLearnX"))
```

Sortie:

```
Hello WayToLearnX!
```

8. Quelle est la sortie de

```
len(['a', 'b', 'c', 'd'])
```

?

Réponse

4

9. Quelle est la sortie de

```
[1, 2] + [3, 4]
```

?

Réponse

[1, 2, 3, 4]

10. Quelle est la sortie de

```
print(2 in [1, 2, 3])
```

?

Réponse

TRUE

Questions d'entretiens Python – Partie

11

février 21, 2020 [Aucun commentaire](#) [correction](#), [corrigé](#), [entretien](#), [pdf](#), [QCM](#), [question](#), [réponse](#), [test](#)
[recrutement](#), [Test technique](#)

Questions sur Python avec des réponses pour la préparation des entretiens d'embauche, des tests en ligne, aux examens et aux certifications. Ces questions et réponses sur Python comprennent divers sujets. Et sont tirés d'un vrai entretien écrit et certaines parties sont en direct. Cette méthode d'apprentissage systématique préparera facilement toute personne pour réussir son test sur Python.

Préparez-vous avec cette liste de questions fréquemment posées sur « Python » afin de pouvoir y répondre en toute confiance.

1. Quel est la sortie du code suivant?

```
for n in [1, 2, 3]: print(n)
```

Réponse

1
2
3

2. Comment comparer deux listes?

Réponse

cmp(list1, list2) – Compare les éléments des deux listes.

3. Comment récupérer la valeur maximale d'une liste?

Réponse

max(liste) – Renvoie la valeur maximale d'une liste.



FONCTION MAX() - PYTHON La fonction max() renvoie la plus grande valeur d'une série de données. Si max() est appelé sur un itérable, il renvoie l'élément le plus grand....**LIRE PLUS**

4. Comment récupérer la valeur minimale d'une liste?

Réponse

min(liste) – Renvoie la valeur minimale d'une liste.



FONCTION MIN() - PYTHON La fonction min() renvoie la plus petite valeur d'une série de données. Si min() est appelé sur un itérable, il renvoie l'élément le plus petit....**LIRE PLUS**

5. Python est-elle indépendante de la plate-forme?

Réponse

Non, il existe certains modules et fonctions en python qui ne peuvent s'exécuter que sur certaines plates-formes.

6. Pensez-vous que Python a un compilateur?

Réponse

Oui, il a un compilateur qui fonctionne automatiquement, et nous ne remarquons pas qu'il existe un compilateur sous Python.

7. Existe-t-il le type « double » en Python?

Réponse

Non.

8. Une chaîne en Python est-elle immuable?

Réponse

Oui. Une chaîne est immuable en Python. Une chaîne représentera toujours la même valeur, il ne peut pas être modifié. Exemple:

```
str = 'WayToLearnX'  
  
str[0] = 'S' #Cela ne changera pas la chaîne d'origine en SayToLearnX.  
  
#nous devons créer une nouvelle chaîne.
```

9. Quel module de Python est utilisé pour appliquer les méthodes liées au système d'exploitation?

Réponse

os

10. Quel bibliothèque Python est utilisé pour Machine learning?

Réponse

La bibliothèque Python « Scikit-learn » est utilisée pour le Machine learning.

11. Quels sont les outils que python utilise pour trouver les bugs.

Réponse

Pylint et Pychec