

Python



Julien Despagne

Les variables sont des conteneurs pour stocker des valeurs de données.

int : stocke des nombres entiers

float : stocke les nombres à virgule

Il y a deux variables au quelle en affecter des chiffres en l'occurrence r et pi $r = 12$ et $\pi = 3,14159$.

Comment nommer une variable en Python ?

Un nom de variable en python peut être défini en lettres minuscules, en majuscules, avec les entiers de 0 à 9 et le caractère _ (underscore). Un nom de variable peut commencer par une lettre, par un underscore mais pas par un chiffre. Un nom de variable prend en compte la casse : ID \neq id.

Affectation

L'affectation est une instruction qui permet de modifier la valeur d'une variable. Plus précisément: Affecter une valeur à une variable signifie enregistrer cette valeur dans une plage mémoire allouée à cette variable.

Comment faire une affectation en python ?

Les affectations simples

Le symbole d'affectation est le signe “=”. Lors d'une affectation, Python crée à la fois la valeur et la variable qui pointe vers cette valeur. >>> a=2. Dans cet exemple, a=2 crée le nombre entier 2 et la variable a pointant sur 2.

Comment affecter sur Python ?

Pour affecter une valeur à une variable, c'est-à-dire l'initialiser ou modifier sa valeur, on utilise l'opérateur d'affectation (=). À gauche de l'opérateur on retrouve le nom de la variable et à droite la valeur qu'on souhaite lui affecter. Pour accéder au contenu d'une variable, il suffit d'utiliser son nom.

Une fonction est un type (des données ou du résultat) peut être réel, entier, chaîne de caractères, ou autre. Le commentaire précise ce que résoud la méthode, le rôle des données et en particulier s'il y a des préconditions sur les données.

La fonction input() permet de demander à l'utilisateur de saisir une chaîne de caractères au clavier. Il ne faut pas oublier d'affecter la valeur ainsi saisie à une variable. Testez le programme

```
suisvant : nom = input("Tapez votre nom : ")  
print("Bonjour ", nom, " !")
```

String est une chaîne composée de la séquence de différents caractères individuels. On retrouve de telles chaînes dans la plupart des langages de programmation Internet ; elles appartiennent aux types de données qui sont probablement les plus utilisés dans l'ensemble.

Les bases en python

Utilisation en mode interactif.

Premiers calculs.

L'opérateur /

L'opérateur %

Affectation.

Affichage - la fonction print()

La fonction range()

Accès aux éléments d'une liste.

True et false

Le booléen True est donc similaire au nombre 1 et le booléen False au nombre 0 . Cela peut sembler inutile, mais c'est très pratique si vous devez par exemple établir un compteur qui se base sur le résultat d'une expression qui résulte sur un booléen. Tous les objets de Python peuvent être évalués comme des booléens.

Comment utiliser True et False Python ?

les constantes littérales sont notées True et False. Tous les types de variables peuvent être interprétés de manière booléenne. Par exemple, pour les entiers (int), la valeur "0" correspond à "faux" et les autres valeurs à « vrai ».

C'est quoi range en Python ?

La fonction range() fournit une séquence de nombres entiers basé sur les arguments de la fonction. Des informations additionnelles peuvent être trouvées dans la documentation Python sur la fonction range() . L'argument start est la première valeur de la plage de nombres.

Comment utiliser range en Python ?

Start est le premier nombre de la séquence, stop est le dernier nombre de la séquence (non inclus), et step est l'intervalle entre les nombres de la séquence. Par défaut, start est 0, step est 1, et stop est requis. Exemple : range(5) génère la séquence [0, 1, 2, 3, 4].

Qu'est-ce que Sqrt Python ?

Cette fonction permet de calculer la racine carré d'une valeur passée en paramètre. La racine carrée d'un nombre réel positif x est le nombre positif qui, lorsqu'il est multiplié par lui-même (le carré de ce nombre), donne x.

Comment calculer la racine carré python de 1 à 13
Vous pouvez calculer la racine carrée de nombres de 1 à 13 en Python en utilisant la fonction `sqrt()` du module `math`. Voici un exemple de code qui affiche les racines carrées de 1 à 13 :

```
>>> liste = 0
>>> while liste < 12:
...     liste=liste +1
...     print(liste,liste**2,listes**3)
...
(1, 1, 1)
(2, 4, 8)
(3, 9, 27)
(4, 16, 64)
(5, 25, 125)
(6, 36, 216)
(7, 49, 343)
(8, 64, 512)
(9, 81, 729)
(10, 100, 1000)
(11, 121, 1331)
(12, 144, 1728)
>>> █
```

```
>>> import math
>>>
>>> for num in range(1, 14):
...     sqrt_num = math.sqrt(num)
...     print("La racine carree de {} est {}".format(num, sqrt_num))
...
La racine carree de 1 est 1.0
La racine carree de 2 est 1.41421356237
La racine carree de 3 est 1.73205080757
La racine carree de 4 est 2.0
La racine carree de 5 est 2.2360679775
La racine carree de 6 est 2.44948974278
La racine carree de 7 est 2.64575131106
La racine carree de 8 est 2.82842712475
La racine carree de 9 est 3.0
La racine carree de 10 est 3.16227766017
La racine carree de 11 est 3.31662479036
La racine carree de 12 est 3.46410161514
La racine carree de 13 est 3.60555127546
>>> █
```

Dans ce code, nous avons importé le module `math` qui contient la fonction `sqrt()` pour calculer la racine carrée. Nous avons ensuite utilisé une boucle `for` pour itérer sur les nombres de 1 à 13 à l'aide de la fonction `range()`. À chaque itération de la boucle, nous avons calculé la racine carrée du nombre actuel à l'aide de la fonction `sqrt()` et avons affiché le résultat à l'aide de l'instruction `print()`.

Booléen

Il existe trois opérateurs logiques. L'opérateur && : il permet de réaliser un « ET » logique. Il faut noter que si l'opérande de gauche est évalué à false , alors le second opérande (à droite de l'opérateur) ne sera pas évalué et l'opérateur renverra false . L'opérateur || : il permet de réaliser un « OU » logique.

Quels sont les caractères booléens ?

Les principaux opérateurs booléens sont ET, OU, SAUF (AND, OR, NOT). En recherche avancée - avec certains moteurs sur le Web - l'internaute utilise des opérateurs booléens à son insu : ceux-ci sont placés automatiquement dans différentes cases du formulaire lorsqu'il tape le ou les termes à inclure et/ou à exclure. L'opérateur logique en Python correspondant à la négation se note (fort logiquement pour un anglophone) 'not'. Une particularité de ce langage et de beaucoup d'autres: Toute valeur numérique est assimilée à une valeur logique avec la convention 0=Faux et tout nombre non nul=Vrai.

Boucle

En informatique, la boucle for est une structure de contrôle de programmation qui permet de répéter l'exécution d'une séquence d'instructions. Selon les langages de programmation, différents mots-clés sont utilisés pour signaler cette structure de

contrôle : for pour les descendants d'Algol, do pour FORTRAN, PL/I, etc.

C'est quoi la boucle for en Python ?

En Python, la boucle FOR est une structure qui permet la répétition d'instructions. Imagine que tu veuilles appliquer la même instruction à 100 éléments différents : sans la boucle FOR, il faudrait écrire 100 fois cette même instruction !

Quelle est la syntaxe d'une boucle for ?

L'instruction for permet d'exécuter plusieurs fois la même instruction ou série d'instructions : c'est une boucle. La syntaxe est la suivante :

for(<Initialisation> ; <Condition> ; <Itération>)
<Instruct>.

Comment afficher les nombres de 0 à 13 sur python avec while

```
num = 0
while num <= 13:
    print(num)
    num += 1
```

0
1
2
3
4
5
6
7
8
9
10

Comment afficher les nombres de 0 à 1 000 000
0000 000000 00000 00000 000000 000000 000000
sur python avec while

[illegible]

1012870

1012871

1012872

1012873

1012874

1012875

1012876

1012877

1012878

1012879

1012880

1012881

1012882

1012883

Syntaxe

En Python, la syntaxe générale est `fonction()` . Ce qui se trouve entre les parenthèses d'une fonction est appelé argument et c'est ce que l'on « passe » à la fonction. Dans l'instruction `type(2)` , c'est l'entier 2 qui est l'argument passé à la fonction `type()` .

La syntaxe est liée à la forme d'un programme écrit en ce langage. C'est la syntaxe qui détermine si l'on utilise des accolades ou des mots-clés.

Une condition est une fonction renvoyant une propriété, alors que le Set définit la propriété. Une propriété d' un objet ne devrait jamais être accéder directement en Java.

If : Elle permet d'exécuter une série d' instructions si jamais une condition est réalisée.

else Il s'utilise en insérant un booléen entre ses parenthèses, si ce booléen vaut 1 , les instructions entre ses accolades sont exécutées. Si ce booléen vaut 0 et qu'il existe un else (qui se traduit par sinon) à la suite de ce if, les instructions entre ses accolades sont exécutées.

Elif : l'instruction Elif est une façon de dire en python "si les conditions précédentes n'étaient pas

vraies, alors essayez cette condition". Dans cet exemple a est égal à b , donc la première condition n'est pas vraie, mais la condition Elif est vraie, donc nous affichons que " a et b sont égaux".

Comment traduire Elif Python ?

Il est possible d'ajouter autant de conditions précises que l'on souhaite en ajoutant le mot clé elif , contraction de "else" et "if", qu'on pourrait traduire par "sinon".

Pourquoi Python Est-il extensif ?

Python a été conçu pour être un langage lisible. Il vise à être visuellement épuré. Par exemple, il possède moins de constructions syntaxiques que de nombreux langages structurés tels que C, Perl, ou Pascal.

Le type ça permet de connaître les types de variable.

PEMDAS

P = pour les parenthèses

E = exposants

Un exposant consiste à multiplier un élément par lui même plusieurs fois à la suite.

M et D = pour multiplication et division qui sont la même priorité

Elles sont évaluées avant l'addition A et la soustraction S, lesquelles sont effectuées en dernier lieu.

Un code binaire représente un texte, des instructions de processeur ou toute autre donnée utilisant un système à deux symboles. Le système à deux symboles utilise souvent des "0" et "1" dans le système de numération binaire.

Comment écrire un code binaire ?

En base ou binaire, on n'utilise que deux chiffres le 0 et le 1. Arrivé à 1, le 2 n'existant pas, on passe à 10, 11, 100 ... En base 12 (base duodécimale), nous utilisons les douze "chiffres" suivants: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B. De sorte que, arrivé à B, nous passons à 10, 11, 12 ...

Pour convertir en binaire, il suffit d'utiliser la fonction native bin. Remarque : La réponse correspond bien au code binaire précédé du préfixe 0b . Par contre, la réponse n'est pas réellement un nombre mais un string comme l'indique la présence des guillemets simples.

Comment écrire en binaire sur Python ?

Ecrire dans un fichier binaire

Pour écrire du texte dans un fichier binaire, vous devez préfixer la chaîne avec le caractère " b "

pour indiquer à Python qu'il s'agit d'une chaîne binaire, alors convertissez-la vous-même en une séquence d'octets.

Le code suivant stocke une liste de nombres dans un fichier binaire.

Comment trouver le code binaire ?

Dans le système binaire, les calculs s'effectuent comme dans le système décimal. Ainsi, l'addition $1100 + 1010$ donne 10110 .

En posant le calcul comme on le fait à l'école et en additionnant de droite à gauche, on a :

$$0 + 0 = 0.$$

$$0 + 1 = 1.$$

$$1 + 0 = 1.$$

$$1 + 1 = 0 \text{ avec la retenue}$$

Instruction est une opération de base du processeur ou d'un langage de programmation, une opération que le programmeur demande à la machine d'exécuter.

C'est quoi une instruction en programmation ?

Une instruction informatique désigne une étape dans un programme informatique. Une instruction dicte à l'ordinateur l'action nécessaire qu'il doit effectuer avant de passer à l'instruction suivante.

```
>>> a = 5.5678
>>> b = 8.97531
>>> c = 709721
>>> e = 3.764321
>>> a * b /c -e + a
1.8035494117970559
>>> a * b /c -e + b
1.211059411797056
>>> a * b /c -e + c
709717.2357494117
>>> a * b
49.9727310180000005
>>> a + b
14.54311
>>> a- b
-3.40751000000000003
>>> 
```



```
>>> c = 550
```

```
>>> a * c / x
```

```
3666
```

```
>>> a * c
```

```
55000
```

```
>>> a + c
```

```
650
```

```
>>> w = 450
```

```
>>> w * c / a
```

```
2475
```

```
>>> 
```

```
>>> result = a, c, w
>>> print(result)
(100, 550, 450)
>>> resultat = a + c / w
>>> d = 670
>>> resultat = a + c * d / w
>>> print(resultat)
918
>>> e = 560
>>> resultat = a + e - c * d / w
>>> print(resultat)
-158
>>> █
```

```
>>> resultat = a * b + c - e / b
>>> print(resultat)
709770.553323
>>> resultat = a + b - c * e / b
>>> print(resultat)
-297648.452869
>>> █
```

Un programme informatique est constitué d'une suite d'instructions.

Comment Ecrire une chaîne de caractère sur Python ?

On représente une chaîne de caractères en plaçant les caractères la consti- tuant entre une paire de guillemets ou une paire d'apostrophes : In [1]: ch1 = "Hello World!" 1.

```
>>> ma_liste = 4,1,3
>>> print(ma_liste)
(4, 1, 3)
>>> █
```

```
>>> print(prenom , nom ,taille)
('Julien', 'Despagne', 35)
>>> ville = "Thorigne Fouillard"
>>> print(prenom , nom ,taille, ville)
('Julien', 'Despagne', 35, 'Thorigne Fouillard')
>>> poids = 45
>>> print(prenom , nom ,taille,poids, ville)
('Julien', 'Despagne', 35, 45, 'Thorigne Fouillard')
>>> █
```



Julien Despagne