



Les bases

Modules standards

Modules non standards



Types des variables

Opérateurs de comparaisons

Opérations

Instructions

Chaînes de caractères

Listes

Tuples

Séquences en général

Dictionnaires

Sets

Arrays

Fonctions

Fichiers

Exceptions

Générateurs

Décorateurs

Evaluation dynamique de code

Inspection

Unicode

différences python2 / python3

Type Checking

Online-Druck weiter gedacht

E
L
F> [Les bases](#) > [Le langage](#) > [Types des variables](#)

Types des variables

* Une variable sans valeur est définie par : `myVar = None` (None est l'équivalent de null dans d'autres langages).

* Valeurs qui sont considérées comme fausses : `False`, `None`, `0`, `"`, `[]`, `()`, `{}`

* Types primitifs :

- `bool` : booléen (`True` ou `False`)
- `int` : entier.
- `float` : nombre flottant qui a la précision d'un double.
- `str` : chaîne de caractère (`string`).

* Pour avoir l'entier de taille maximum : `sys.maxint`

* conversion de n'importe quoi en string : avec ```` ou `repr()` :

```
x = 10
s = 'valeur = ' + `x`
s = 'valeur = ' + repr(x)
```

* Conversion d'une string en int ou float :

- `x = int('2')`
- déclenche une `ValueError` si ce n'est pas le bon type.

* Conversions :

- `float(3)` : convertit l'int en float.
- `int(3.0)` : convertit le float en int.
- `float(3.6)` : convertit le float en int, en donnant ici 3.

* Examen du type d'une variable :

- `type(var)` : renvoie le type de la variable, par exemple `<type 'int'>` ou `<type 'list'>`
- pour tester le type d'une variable, on peut faire : `type(var) == list` (ou `str` ou `int` ou `float`)
- mais pour tester le type d'une variable, le mieux est `isinstance(var, list)`.
- `isinstance` donne `True` si on teste si un objet contre sa classe, mais aussi contre ses classes de base.

* Pour détruire une variable : `del v`

Copyright python-simple.com

contact@python-simple.com

Mis à jour le 2019-08-19, 22:43