

# Langage Neon

## Présentation

**Le langage Neon est un langage interprété très simple et facile à utiliser.**

- Vous savez déjà programmer ? Alors vous pouvez apprendre en moins de 10 minutes à utiliser Neon.
- Si vous ne savez pas encore programmer, bienvenue, et vous verrez, c'est très simple.

Du côté de l'implémentation, l'interpréteur est codé en C, et est donc compilable sur une grande variété de machines. Sa syntaxe s'inspire de celle du C et du python, avec un peu d'originalité tout de même.

## Découvrons le langage Neon

### Les expressions

#### Les types

Il existe 3 types de base : - Les nombres: 1, 299.3, -34

- Les chaînes de caractères: "Hello World"
- Les booléens True, False

#### Les opérateurs

##### Opérateurs arithmétiques

- Les opérateurs classiques: +, -, \*, /
- La puissance: \*\*
- Le quotient d'une division euclidienne: %

**Opérateurs booléens** and, or, xor, not, ==, >=, <=, <, >, !=

##### Opérateurs sur les variables

- Affectation:

```
foo = 10
foo = "Hello world"
```

Notez que l'affectation est une expression comme les autres

- Opérateurs combinés

```
x = x + 12 => x += 12
x = x - 12 => x -= 12
x = x * 12 => x *= 12
```

```
x = x / 12 => x /= 12
x = x + 1 => x++
x = x - 1 => x--
```

### Opérateurs spéciaux

- `&foo`: renvoie le nom de la variable `foo` dans une chaîne de caractères.
- `:"foo"`: Renvoie la valeur de la variable `foo`
- `foo -> newVar`

### Les fonctions

La syntaxe pour appeler les fonction est la suivante:

```
fonction(arg1, arg2, ..., arg n)
```

### Les listes

Les listes littérales s'écrivent entre crochets, et les éléments sont séparés par des virgules.

```
list = [True, 12.7, "Bonjour"]
```

Une liste peut contenir tous types d'objets (chaîne de caractères, booléens, nombres, et même d'autres listes).

Les listes sont indentées de 0 à n-1, avec n la taille de la liste.

Pour accéder à l'élément en position k, la syntaxe est:

```
list[k]
```

Pour `k = 1`, `list[k]` renvoie `12.7`

### Commentaire

La syntaxe pour faire un commentaire est la suivante:

```
# Votre commentaire de folie
```

### Conditions

La syntaxe pour les blocs conditionnels est la suivante:

```
if (condition)
{
#Fait quelque chose...
}
elif (condition2)
{
#Fait autre chose...
}
```

```
else
{
#Fait encore autre chose...
}
```

Notez que ces blocs peuvent parfaitement s'écrire sur une seule ligne, en prenant soin de séparer les expressions par des points virgules.

Seul le block `if` est obligatoire, les blocks `elif` et `else` ne sont pas forcément nécessaires à la suite d'un bloc `if`.

### Les boucles

**Boucle while** Les boucles while s'écrivent comme suit:

```
while (condition)
{
#Fait quelque chose...
}
```

**Boucle for** Les boucles for s'écrivent comme suit:

```
for (variable, départ, fin)
{
#Fait quelque chose...
}
```

### Les fonctions de base

- `append(liste,valeur)`
- `remove(liste, valeur)`
- `len(listeOuChaine)`
- `sub(chaine, debut, fin)`
- `nbr(chaine)`
- `str(valeur)`
- `eval(str)`
- `reverse(str)`

Vous pouvez également récupérer les arguments du programme dans la liste `__args__`.

**Vous savez maintenant l'essentiel pour programmer en Neon !**