

# Python - Opérations mathématiques de base

Python est un langage de programmation polyvalent qui peut être utilisé pour effectuer diverses opérations mathématiques. Dans ce cours, nous allons explorer les opérations mathématiques de base en Python, notamment l'addition (+), la soustraction (-), la multiplication (\*), la division (/), la division entière (//), le modulo (%), et l'exponentiation (\*\*).

3GMS

5GMS

6GMS

4GMS

4TTR

3TTR

 Découverte

Voici un article de cours sur la syntaxe de base du langage Python pour effectuer des opérations mathématiques de base.

Pour faire court: les opérations mathématiques en Python fonctionnent **comme en math**, y compris la **priorité des opérations**.

## Addition (+)

L'addition est l'opération qui permet de combiner deux nombres pour obtenir leur somme. En Python, l'opérateur d'addition est le signe "+". Voici un exemple :

```
1 | a = 5
2 | b = 3
3 | resultat = a + b
4 | print(resultat) # Affiche 8
```

## Soustraction (-)

La soustraction est l'opération qui permet de trouver la différence entre deux nombres. En Python, l'opérateur de soustraction est le signe "-". Voici un exemple :

```
1 | x = 10
2 | y = 4
3 | difference = x - y
4 | print(difference) # Affiche 6
```

## Multiplication (\*)

La multiplication est l'opération qui permet de trouver le produit de deux nombres. En Python, l'opérateur de multiplication est le signe "\*" (astérisque). Voici un exemple :

```
1 | p = 7
2 | q = 2
3 | produit = p * q
4 | print(produit) # Affiche 14
```

## Division (/)

---

La division permet de trouver le quotient de deux nombres. En Python, l'opérateur de division est le signe "/" (slash). Voici un exemple :

```
1 | dividende = 8
2 | diviseur = 2
3 | quotient = dividende / diviseur
4 | print(quotient) # Affiche 4.0
```

## Division Entière (//)

---

La division entière permet de trouver le quotient de deux nombres en ignorant la partie décimale. En Python, l'opérateur de division entière est le signe "//". Voici un exemple :

```
1 | numerator = 15
2 | denominator = 4
3 | quotient_integer = numerator // denominator
4 | print(quotient_integer) # Affiche 3
```

## Modulo (%)

---

L'opérateur modulo (%) permet de trouver le reste de la division de deux nombres. Voici un exemple :

```
1 | n = 17
2 | m = 5
3 | reste = n % m
4 | print(reste) # Affiche 2
```

## Exposants (\*\*)

---

L'exponentiation permet d'élever un nombre à une puissance donnée. En Python, l'opérateur d'exponentiation est le double astérisque "\*\*". Voici un exemple :

```
1 | base = 2
2 | exposant = 3
```

```
3 | resultat_expo = base ** exposant
4 | print(resultat_expo) # Affiche 8
```

## Séparateur de décimales

En Belgique, nous utilisons la virgule ( , ) pour séparer la partie entière de la partie décimale d'un nombre réel (ce qu'on appelle un nombre à... virgule).

En Python, on utilise le point ( . ) pour séparer la partie entière de la partie décimale d'un nombre réel.

```
1 | pi = 3.1415
2 | nombre_or = 1.61803398875
```

## Priorité des opérations

Les règles de **priorité** sont les **mêmes qu'en math**: les **\*** et **/** sont calculés avant les **+** et les **-**.

Combien donnera le calcul suivant?

```
1 | total = 5 + 2 * 3
2 |
3 | print(total)
```

21? Réessaye 😊 . La réponse est 11. Etant donné l'ordre des opérations, Python calcule d'abord  $2 * 3$  et l'additionne ensuite à 5.

Les parenthèses permettent de changer l'ordre de priorité.

Combien donnera le calcul suivant?

```
1 | total = (5 + 2) * 3
2 |
3 | print(total)
```

11? Réessaye 😊 . La réponse est 21. Etant donné l'ordre des opérations, Python calcule d'abord l'intérieur des parenthèses ( $5 + 2$ ) et multiplie le résultat par 3.

Les paranthèses à l'intérieur des parenthèses sont exécutées en premier.

**L'affectation (signe "=")** est **toujours effectuée en dernier**. Autrement dit, Python va toujours effectuer les opérations à droite du "=" d'abord.

## Conclusion

Ce cours vous a introduit aux opérations mathématiques de base en Python. Vous pouvez maintenant effectuer des opérations d'addition, de soustraction, de multiplication, de division, de division entière, de modulo, et

d'exponentiation en utilisant Python. Ces concepts sont essentiels pour résoudre des problèmes mathématiques et scientifiques à l'aide de la programmation. N'hésitez pas à pratiquer ces opérations pour les maîtriser davantage.

Vidéo à la une à regarder sur Youtube:  <http://youtu.be/9M9yz8bFzuU>