

# Opérations arithmétiques et fonctions mathématiques en Python

## 1. Opérations arithmétiques de base en Python

Python permet d'effectuer facilement des opérations arithmétiques comme dans une calculatrice.

Opérateur	Signification	Exemple	Résultat
+	Addition	5 + 3	8
-	Soustraction	10 - 4	6
*	Multiplication	6 * 7	42
/	Division (résultat décimal)	8 / 2	4.0
//	Division entière (sans décimale)	7 // 2	3
%	Modulo (reste de la division)	7 % 2	1
**	Puissance (exponentiation)	3 ** 2	9

### Exemple simple :

```
a = 12
b = 5

print("Addition :", a + b)
print("Soustraction :", a - b)
print("Multiplication :", a * b)
print("Division :", a / b)
print("Division entière :", a // b)
```

```
python
```

```
a = 12
```

```
b = 5
```

```
print("Addition :", a + b)
```

```
print("Soustraction :", a - b)
```


```
print("Multiplication :", a * b)
```

```
print("Division :", a / b)
```

```
print("Division entière :", a // b)
```

```
print("Reste :", a % b)
```

```
print("Puissance :", a ** 2)
```



```
Programiz  
Python Online Compiler
```

```
main.py
```

```
1 a = 12  
2 b = 5  
3  
4 print("Addition :", a + b)  
5 print("Soustraction :", a - b)  
6 print("Multiplication :", a * b)  
7 print("Division :", a / b)  
8 print("Division entière :", a // b)  
9 print("Reste :", a % b)  
10 print("Puissance :", a ** 2)
```

```
Output
```

```
Addition : 17  
Soustraction : 7  
Multiplication : 60  
Division : 2.4  
Division entière : 2  
Reste : 2  
Puissance : 144  
  
=== Code Execution Successful ===
```

## 2. Bibliothèque de mathématiques et fonctions mathématiques

## **1. Exemples de Programmes Simples**

```
a = 12
b = 5
print("Addition :", a + b)
print("Soustraction :", a - b)
print("Multiplication :", a * b)
print("Division :", a / b)
print("Division entière :", a // b)
print("Reste :", a % b)
print("Puissance :", a ** 2)
```

### **1.1 Exemple 1 — Utiliser des variables et afficher leurs valeurs**

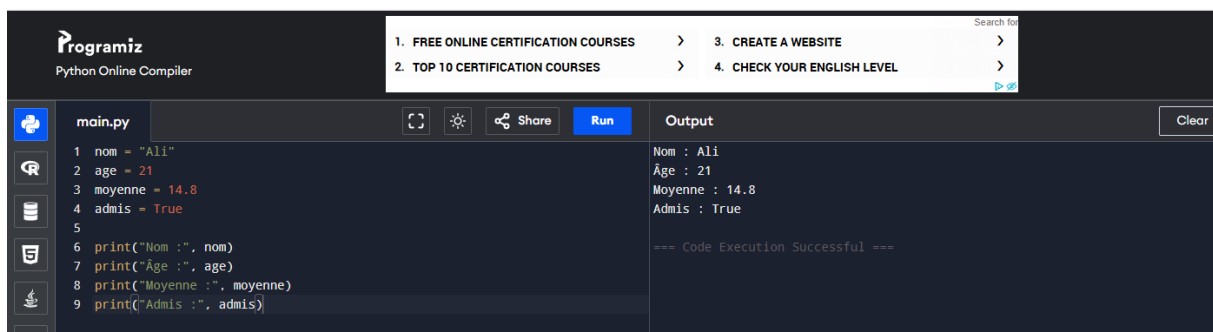
**Sous l'éditeur**

```
python

nom = "Ali"
age = 21
moyenne = 14.8
admis = True

print("Nom :", nom)
print("Âge :", age)
print("Moyenne :", moyenne)
print("Admis :", admis)
```

☞ Sous l'éditeur (partie droite programme saisi, partie gauche Sortie attendue)



## 1.2 Exemple 2 — Opérations simples avec variables numériques

```
a = 10
b = 3
somme = a + b
produit = a * b

print("La somme est :", somme)
print("Le produit est :", produit)
```

### Sous l'éditeur

```
python

a = 10
b = 3
somme = a + b
produit = a * b

print("La somme est :", somme)
print("Le produit est :", produit)
```

☞ Sous l'éditeur (partie droite programme saisi, partie gauche Sortie attendue)

Programiz  
Python Online Compiler

Understand Python better by watching your code unfold visually.

Try Now

main.py

```
1 a = 10
2 b = 3
3 somme = a + b
4 produit = a * b
5
6 print("La somme est :", somme)
7 print("Le produit est :", produit)
8
```

Output

La somme est : 13  
Le produit est : 30

=== Code Execution Successful ===

### 1.3 Exemple 3 — Combiner des types différents

```
nom = "Sara"
age = 20
print(nom + " a " + str(age) + " ans.")
```

#### Sous l'éditeur

```
python

nom = "Sara"
age = 20
print(nom + " a " + str(age) + " ans.")
```

☞ Sous l'éditeur (partie droite programme saisi, partie gauche Sortie attendue)

Programiz  
Python Online Compiler

```
main.py
```

```
1 nom = "Sara"  
2 age = 20  
3 print(nom + " a " + str(age) + " ans.")  
4
```

Output

```
Sara a 20 ans.  
=== Code Execution Successful ===
```

**(Remarque : il faut convertir l'entier en chaîne avec `str()` pour la concaténation.)**