

Variables et Types de Données en Python

1. Les Types de Données de Base

Type de donnée	Nom	Exemple	Description
int	Entier	x = 10	Représente des nombres entiers (positifs, négatifs ou zéro).
float	Nombre décimal (flottant)	y = 3.14	Représente des nombres réels avec une partie décimale.
str	Chaîne de caractères	nom = "Alice"	Représente du texte ou une suite de caractères.
bool	Booléen	vrai = True ou faux = False	Représente une valeur logique : vrai ou faux .

2. Définition d'une Variable

Une **variable** est un espace mémoire nommé qui stocke une valeur.

```
nom = "Alice"
age = 25
pi = 3.14159
est_etudiant = True
```

👉 En Python, vous **n'avez pas besoin de déclarer le type** : il est **déterminé automatiquement**.

3. Règles de Nommage des Variables

Règle	Exemple correct	Exemple incorrect
Le nom doit commencer par une lettre ou un underscore (_)	nom, _age	1nom, @data
Il peut contenir des lettres, chiffres et underscores	score_total	score-total
Python est sensible à la casse (maj/min)	Age ≠ age	—
Évitez les mots réservés du langage	compteur	print, class, if

4. Exemples de Programmes Simples

4.1 Exemple 1 — Utiliser des variables et afficher leurs valeurs

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

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

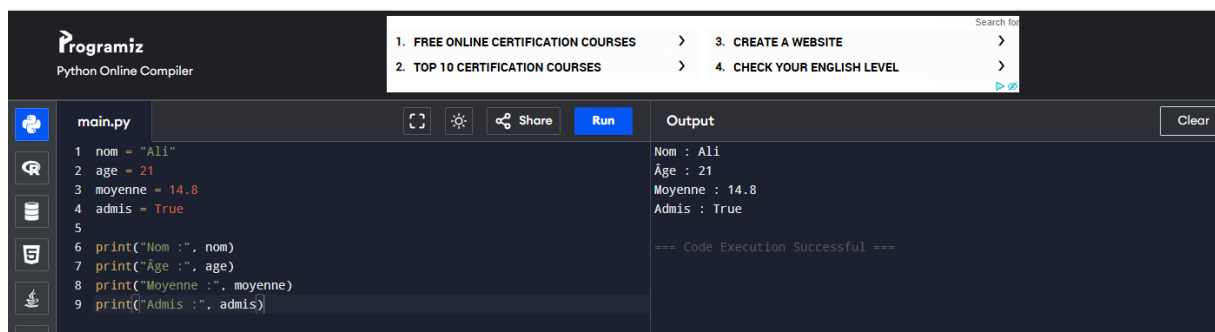
Sous l'éditeur

```
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)



4.2 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)
```

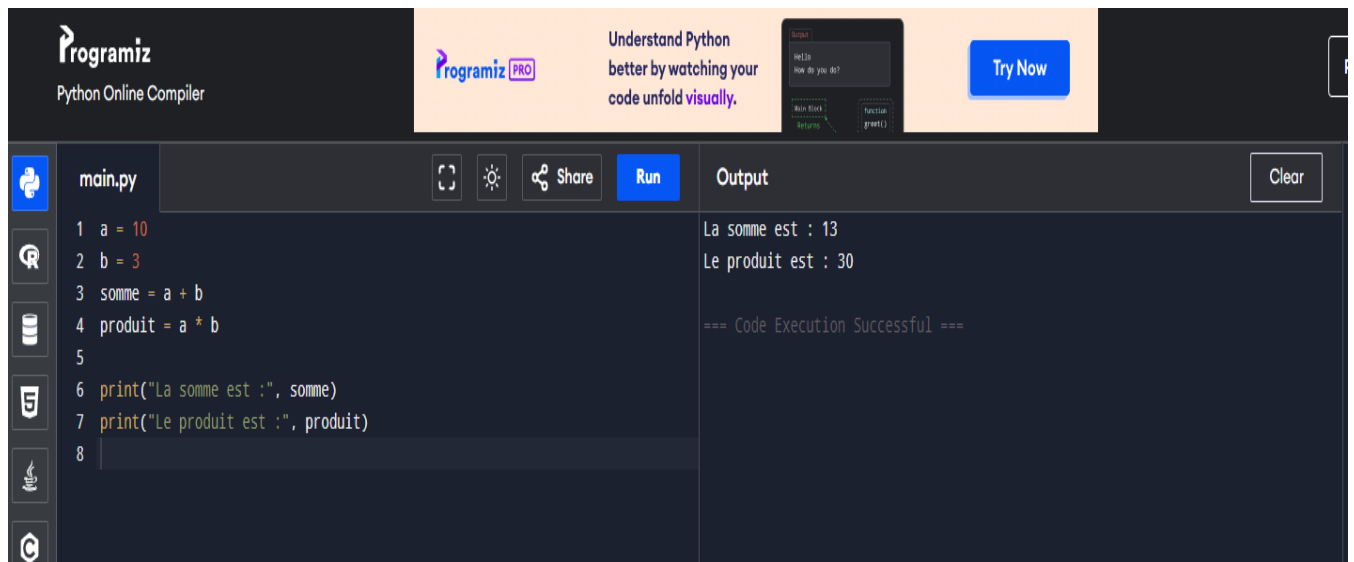
Sous l'éditeur

```
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)



4.3 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

Share

Run

Clear

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

Sara a 20 ans.

=== Code Execution Successful ===

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