

Cours : Les Variables en Python

1. Définition des variables

En programmation, une **variable** est une boîte dans laquelle on peut stocker une valeur. Elle sert à nommer et manipuler ces valeurs au cours de l'exécution d'un programme.

- Exemple :

```
python
```

 Copier le code

```
age = 25 nom = "Alice"
```

Ici, `age` et `nom` sont des variables.

2. Règles pour nommer une variable

Les noms de variables doivent suivre certaines conventions et règles en Python :

1. Commencer par une lettre ou un underscore (`_`) :
 - Valide : `nom` , `_age`
 - Non valide : `1age`
 2. Ne pas utiliser d'espaces ou de caractères spéciaux :
 - Valide : `nom_utilisateur`
 - Non valide : `nom utilisateur`
 3. Éviter les mots-clés réservés par Python (comme `if` , `for` , `while` , etc.).
 4. Sensibilité à la casse :
 - `age` , `Age` , et `AGE` sont trois variables différentes.
-

3. Types de valeurs (Types de données)

Les variables en Python peuvent contenir différents types de valeurs. Python est un langage à typage dynamique, ce qui signifie que vous n'avez pas besoin de déclarer explicitement le type d'une variable.

Voici les principaux types de données en Python :

1. Booléen (`bool`) :
 - Valeurs : `True` OU `False`

- Exemple :

```
python
```

 Copier le code

```
est_actif = True
```

2. Entier (int) :

- Exemple :

```
python
```

 Copier le code

```
age = 30
```

3. Nombre à virgule flottante (float) :

- Exemple :

```
python
```

 Copier le code

```
temperature = 36.5
```

4. Chaîne de caractères (str) :

- Exemple :

```
python
```

 Copier le code

```
nom = "Alice"
```

5. Liste (list) :

- Exemple :

```
python
```

 Copier le code

```
fruits = ["pomme", "banane", "cerise"]
```

6. Dictionnaire (dict) :

- Exemple :

```
python
```

 Copier le code

```
etudiant = {"nom": "Alice", "âge": 25}
```

4. Mise à jour des variables

Les variables peuvent être mises à jour avec de nouvelles valeurs ou des résultats d'expressions.

Affectation simple

python

 Copier le code

```
x = 5 x = 10 # x contient maintenant 10
```

Mise à jour avec une expression

Les expressions permettent d'effectuer des calculs avant d'affecter une nouvelle valeur.

python

 Copier le code

```
x = 5 x = x + 3 # x vaut maintenant 8
```

Raccourcis fréquents :

python

 Copier le code

```
x += 3 # Équivaut à x = x + 3 x -= 2 # Équivaut à x = x - 2 x *= 4 # Équivaut à x = x  
* 4 x /= 2 # Équivaut à x = x / 2
```

Exercices pratiques

1. Déclarez une variable appelée `prenom` contenant votre prénom. Changez ensuite sa valeur pour un autre prénom.
2. Déclarez deux variables `a = 10` et `b = 20`. Mettez à jour `a` pour qu'elle contienne la somme de ses valeurs initiales et de `b`.
3. Créez une variable `est_vrai` contenant la valeur `False`. Changez-la ensuite en `True`.