

Cours : Les Boucles en Python

1. Définition des boucles

Une boucle est une structure qui permet de répéter un bloc de code plusieurs fois, tant qu'une condition est vraie ou pour un ensemble d'éléments.

- **Pourquoi utiliser des boucles ?** Elles évitent de réécrire plusieurs fois le même code, en automatisant les tâches répétitives.
-

2. Les opérateurs d'affection

Avant d'entrer dans les détails des boucles, rappelons les opérateurs d'affection qui permettent de mettre à jour les variables souvent utilisées dans les boucles.

Opérateurs classiques

1. = : Affectation simple

```
python
```

 Copier le code

```
compteur = 0
```

2. Opérateurs combinés (opérateurs d'affection raccourcis) :

- += :Ajouter une valeur

```
python
```

 Copier le code

```
compteur += 1 # Équivaut à compteur = compteur + 1
```

- -= : Soustrait une valeur

```
python
```

 Copier le code

```
compteur -= 1 # Équivaut à compteur = compteur - 1
```

- *= : Multiplier par une valeur
 - /= :Diviser par une valeur
-

3. La boucle while

La boucle `while` continue d'exécuter un bloc de code tant qu'une condition reste vraie. Elle est utile lorsqu'on ne sait pas à l'avance combien de fois la boucle doit s'exécuter.

Syntaxe :

python

 Copier le code

```
while condition: # Bloc de code à répéter
```

Exemple :

Afficher les nombres de 1 à 5 :

python

 Copier le code

```
compteur = 1 while compteur <= 5: print(compteur) compteur += 1 # Incrémenter la variable
```

Sortie :

1 2 3 4 5

 Copier le code

Précautions avec `while` :

- Assurez-vous que la condition devienne **fausse** à un moment donné pour éviter des boucles infinies.

4. La boucle `for`

La boucle `for` est utilisée pour parcourir un ensemble de valeurs (comme une liste ou une plage de nombres). Elle est utile lorsqu'on connaît à l'avance le nombre d'itérations.

Syntaxe :

python

 Copier le code

```
for variable in sequence: # Bloc de code à répéter
```

Exemple 1 : Parcourir une plage de nombres

Afficher les nombres de 1 à 5 :

python

 Copier le code

```
for i in range(1, 6): print(i)
```

Sortie :

 Copier le code

```
1 2 3 4 5
```

Exemple 2 : Parcourir une liste

python

 Copier le code

```
fruits = ["pomme", "banane", "cerise"] for fruit in fruits: print(fruit)
```

Sortie :

 Copier le code

```
pomme banane cerise
```

Exemple 3 : Utiliser break et continue

- break : Arrêtez immédiatement la boucle.
- continue : Sauter à l'itération suivante.

python

 Copier le code

```
for i in range(1, 10): if i == 5: break # Arrête la boucle quand i vaut 5 if i % 2 == 0: continue # Ignore les nombres pairs print(i)
```

Sortie :

 Copier le code

```
1 3 5
```

Exercices pratiques

1. Utilisez une boucle while pour afficher les nombres paires de 2 à 10 inclus.
2. Utilisez une boucle for pour parcourir une chaîne de caractères, et afficher chaque lettre sur une nouvelle ligne.

3. Utilisez `for` et `break` pour parcourir une liste et arrêter la boucle dès que vous rencontrez la valeur `"stop"` .