Comment itérer sur toutes les combinaisons ou permutations possibles en Python

Par **Mohamed Ouamer**

Vous voulez itérer sur toutes les combinaisons ou permutations possibles d'une collection d'éléments. Le module itertools fournit trois fonctions pour cette tâche.

La première de ces fonctions, itertools.permutations(), prend une collection d'éléments et produit une séquence de tuples qui réorganise tous les éléments en toutes les permutations possibles (c'est-à-dire qu'il les mélange dans toutes les configurations possibles). Par exemple:

```
>>> items = ['a', 'b', 'c']
>>> from itertools import permutations
>>> for p in permutations(items):
...     print(p)
...
('a', 'b', 'c')
('a', 'c', 'b')
('b', 'a', 'c')
('b', 'c', 'a')
('c', 'a', 'b')
('c', 'b', 'a')
>>>
```

Si vous voulez toutes les permutations d'une longueur inférieure, vous pouvez donner un argument de longueur optionnel. Par exemple:

```
>>> for p in permutations(items, 2):
...     print(p)
...
('a', 'b')
('a', 'c')
('b', 'a')
('b', 'c')
('c', 'a')
('c', 'b')
>>>
```

Utilisez itertools.combinations() pour produire une séquence de combinaisons d'éléments provenant de l'entrée. Par exemple:

```
>>> from itertools import combinations
>>> for c in combinations(items, 3):
...     print(c)
...
('a', 'b', 'c')
>>> for c in combinations(items, 2):
...     print(c)
...
('a', 'b')
('a', 'c')
('b', 'c')
>>> for c in combinations(items, 1):
...     print(c)
...
('a',)
('b',)
('c',)
>>>
```

La méthode itertools.combinations_with_replacement() assouplit cela et permet de sélectionner plusieurs fois le même élément. Par exemple:

```
>>> for c in combinations_with_replacement(items, 3):
...     print(c)
...
('a', 'a', 'a')
('a', 'a', 'b')
('a', 'a', 'c')
('a', 'b', 'b')
('a', 'b', 'c')
('a', 'c', 'c')
('b', 'b', 'b')
('b', 'b', 'c')
('b', 'c', 'c')
('c', 'c', 'c')
>>>
```

Ce code ne démontre qu'une partie de la puissance du module itertools.

Bien que vous puissiez certainement écrire du code pour produire vous-même des permutations et des combinaisons, cela demanderait probablement plus qu'un peu de réflexion.

Lorsque vous êtes confronté à des problèmes d'itération apparemment compliqués, il est toujours préférable de voir en premier le module itertools. Si le problème est courant, il y a de fortes chances qu'une solution existe déjà.

Mohamed Ouamer

Passionné des nouvelles technologies de l'information et professeur d'informatique en classes préparatoires (Sup/Spé). J'enseigne l'informatique depuis 15 ans. De temps en temps, je publie des articles sur la programmation et les réseaux informatiques.