



Forum: Python & Plugins

Topic: Comparaisons d'un ensemble de valeurs

Subject: Re: Comparaisons d'un ensemble de valeurs

Posté par: DaWaaaaghBabal

Contribution le : 10/10/2019 20:46:51

Pour avoir quelque chose qui tient la charge quand on augmente le nombre de variables, il faut éviter de comparer toutes les paires.

Première étape, ne comparer chaque valeur qu'aux précédentes. Imaginons un instant que dans (a, b, c, d, e, f, g) tu aies un doublon, a == d. Tu ne veux pas un code qui commence par "if a == d: ..." et continue un peu plus loin par "if d == a: ...".

Ensuite, idéalement, on ne veut pas comparer les paires du tout en fait. C'est quadratique : chacune des N variables doit être comparée aux N-1 précédentes, et le nombre total de comparaisons est de l'ordre de N^2 (plus précisément, $(N^2 + N)/2$). N^2 , c'est trop, alors qu'on peut faire ça en temps linéaire : plutôt que de comparer chaque variables à toutes les précédentes une par une, on va regarder si chaque variable est dans l'ensemble des précédentes. Le temps de vérification ne dépend pas du nombre d'éléments dans l'ensemble (parce que les HashSets, c'est de la sorcellerie).

Ce qui nous donne :

```
variables = [g, h, i, j, k, l, m] précédentes = set()
for variable in variables:
    if variable in précédentes:
        print("On a un doublon, faire un truc !")
    précédentes.add(variable)
```

Bonus : un set est une collection non ordonnée mais sans doublon. Si ton but est seulement d'éliminer les doublons, on doit assurer qu'il n'y en a pas, tu peux faire encore plus simple : variables = {g, h, i, j, k, l, m, n} # tu peux itérer sur variables, il n'y aura pas de doublon