

Module 2 :: n-uplets

version 1.0 - dimanche 6 décembre 2020 - 17:12:19

Objectifs☒ **Connaissances visées**☐ *tuples, p-uplets, p-uplets nommés*☒ **Compétences à développer**☐ *Écrire une fonction renvoyant un p-uplet de valeurs.***Ce Qu'il Faut Retenir...**

Un **n-uplet** – nommé **tuple** dans la terminologie de Python – est une **séquence ordonnée** de données.

Déclaration d'un tuple

Un jeu de n valeurs (val) peut être rangé dans un tuple de la manière suivante :

```
>>> t = val1, val2, ... ,valn
```

Pour une meilleure visibilité de l'écriture, une habitude veut que l'on encadre ces valeurs par des parenthèses.

```
>>> t = (val1, val2, ... ,valn)
```

Les valeurs rangées dans un tuple peuvent être de type simple : entier naturel, entier relatif, flottant, caractère, booléens ; mais aussi de type construit : chaîne de caractères, tuple, liste ou dictionnaire ;

Un tuple peut contenir des données de n'importe quel type et de types différents.

```
>>> t = (int1, char1, ... ,str1)
```

```
>>> t = (int1, (char1, char2, ... charn), ... ,str1)
```

Pour placer un tuple parmi les valeurs d'un tuple, il faut obligatoirement encadrer de parenthèse le tuple qui est inséré dans le jeu de données.

Pour créer un tuple vide il faut mettre à la suite une parenthèse ouvrante et une parenthèse fermante :

```
>>> t = ()
```

Pour créer un tuple avec une seule valeur, il faut ajouter une virgule après la valeur

```
>>> t = val,
```

```
>>> t = (val,)
```

La fonction 'type(t)' renvoie <class 'tuple'> quand une variable est de type 'tuple'

Accès à une valeur d'un tuple

Les différentes valeurs d'un tuple non vide sont accessibles par un index.

Pour accéder à une valeur, on écrit le nom du tuple suivi de son index écrit entre crochets.

```
>>> t[i]
```

Quand on lit les valeurs d'un tuple de gauche à droite, l'index de la première valeur est 'zéro'.

Quand on lit les valeurs d'un tuple de droite à gauche, l'index de la première valeur est '-1'.

Pour accéder à la valeur d'index j d'un tuple qui se trouve dans un autre tuple à l'index i, on écrit :

```
>>> t[i][j]
```

Si l'on demande à accéder à une valeur correspondant à un index qui n'existe pas – c'est à dire supérieur ou égal à 'len(t)', une exception est levée : **IndexError: tuple index out of range**

Module 2 :: n-uplets

version 1.0 - dimanche 6 décembre 2020 - 17:12:19

Longueur d'un tuple

La longueur d'un tuple correspond au nombre de valeur qu'il contient.

Cette longueur est obtenue en utilisant la fonction 'len()'

```
>>> len(t)
```

Les tuples sont concaténables

```
>>> t1 = (val1, val2, ... , valn)
```

```
>>> t2 = (data1, data2, ... , data_n)
```

```
>>> t1 + t2
```

```
(val1, val2, ... ,valn, data1, data2, ... , data_n)
```

Les tuples sont duplicables

```
>>> t = (val1, val2, ... , valn)
```

```
>>> t * 2
```

```
(val1, val2, ... ,valn, val1, val2, ... , valn)
```

Tester la présence d'une valeur dans un tuple

Ce test s'effectue avec l'opérateur 'in' : il renvoie une valeur booléenne

```
>>> t = (val1, val2, ... , valn)
```

```
>>> val_x in t
```

```
'True' ou 'False'
```

Les tuples sont itérables

```
>>> t = (val1, val2, ... , valn)
```

```
>>> for i in range(0, len(t)) :
```

```
... print(t[i])
```

ou

```
>>> for val in t :
```

```
... print(val)
```

Les tuples sont non mutables

Il n'est pas possible d'assigner une valeur à un nouvel index : $i = \text{len}(t)$

```
>>> t = (val1, val2, ... , valn)
```

```
>>> t[i] = val_x
```

TypeError: 'tuple' object does not support item assignment

Il n'est pas possible de modifier la valeur correspondant à un index existant : $0 \leq i < \text{len}(t)$

```
>>> t[i] = val_x
```

TypeError: 'tuple' object does not support item assignment

Il n'est pas possible de supprimer une valeur correspondant à un index existant : $0 \leq i < \text{len}(t)$, en utilisant la fonction 'del()'

```
>>> del(t[i])
```

TypeError: 'tuple' object doesn't support item deletion