

## 1 Définition : p-uplet

Un **p-uplet** est une **séquence** correspondant à une collection de données (d'un type de base par exemple), souvent noté entre parenthèses.

Un p-uplet s'appelle un **tuple** en Python.

*Exemple* : Création d'un p-uplet de 4 éléments (de type de base float) en Python :

```
tup_1 = (12.3, 2.0, -56.1, 41.4)
```

## 2 Accès aux éléments d'un p-uplet

On accède à un élément d'un p-uplet par son **indice**, noté **entre crochets**, qui donne sa position dans la séquence. Le **1<sup>er</sup> élément du tuple a l'indice 0**.

valeurs	12.3	2.0	-56.1	41.4
indice (à l'endroit)	0	1	2	3
indice (à l'envers)	-4	-3	-2	-1

```
>>> elt_3 = tup_1[2]
>>> elt_3
-56.1
>>> tup_1[-1]
41.4
```

Les tuples sont des objets **non mutables** : cela signifie on ne peut pas les modifier après leur création !

```
>>> tup_1[0] = 100
Traceback (most recent call last):
  File "<input>", line 1, in <module>
TypeError: 'tuple' object does not support item assignment
```

## 3 Taille d'un p-uplet

On peut connaître le nombre d'éléments d'un tuple avec la fonction `len`.

```
>>> len(tup_1)
4
```

*Remarque* : Pour créer un tuple avec un seul élément en Python, il faut ajouter une virgule pour ne pas confondre la syntaxe avec un simple objet entre parenthèse.

```
>>> singleton = (5, ) # p-uplet à un seul élément : len(singleton) vaut 1
>>> type(singleton)
<class tuple>
>>> s = (5)           # un simple entier, noté entre parenthèses
>>> type(s)
<class int>
```