

Mettre fichier tuple.zip sur site avant le cours

1 Problématique

Dans le cadre d'une exposition sur l'évolution du climat, le professeur de SVT nous demande de réaliser un programme pour manipuler les données de température moyenne en France. Ces valeurs sont aisément accessibles sur le web.

Année	Écart par rapport à la température moyenne (en °C)
1900	0,3
1901	-0,7
1902	-0,6
1903	-0,3
1904	0,1
1905	-0,6
1906	0,0

FIGURE 1 – Anomalie de la température moyenne annuelle de l'air par rapport à la moyenne de référence. Le zéro correspond à la moyenne de l'indicateur sur la période 1961-1990, soit 11,8 °C

Il est possible de créer une variable pour chaque année, mais cela va être rapidement fastidieux.

```
1 annee1900 = 0.3
2 annee1901 = -0.7
3 ...
```

Code 1 – Une variable par année

Comment stocker les données dans le programme pour les manipuler facilement ?

2 Les tuples

2.1 Présentation

Un *tuple* est une séquence **ordonnée** de plusieurs éléments. En mathématiques on parle de *p-uplet*.

un quadruplet = 4 éléments; on peut le voir comme un tableau fixe

```
1 mon_tuple = (8, 5, 3, 9, 1, 0, 2)
```

Code 2 – Créer un tuple en Python

2.2 Accéder à un élément

On ne peut modifier un tuple : on dit qu'il est *immuable* ou *non mutable*. Par contre il est possible d'accéder aux éléments individuellement.

```
1 >>> mon_tuple[0]
2 >>> 8
```

Code 3 – Accéder à l'élément de rang 0

Le code 3 renvoie la première valeur du tuple. **L'indexation commence à 0.** Chaque élément est accessible en *temps constant* même si le tuple est très important.

								6		
h	e	y	8	5	3	9	1	0	2	T
		4		h	e	l	l	o	!	
										9

FIGURE 2 – Le tuple est enregistré dans un espace libre

2.3 Utiliser le tuple

Le tuple ci-après contient les anomalies de températures de 1900 à 2017.

```
1 temperatures = (0.3, -0.7, -0.6, -0.3, 0.1, -0.6, 0.0, -0.4, -0.4, -1.0,
2                 -0.4, 0.5, -0.4, 0.1, -0.3, -0.4, -0.2, -1.2, -0.2, -0.7,
3                 0.0, 0.6, -0.7, -0.1, -0.4, -0.6, 0.3, -0.1, 0.5, -0.2,
4                 0.4, -0.6, -0.3, -0.2, 0.4, -0.1, 0.0, 0.5, -0.1, -0.3,
5                 -0.9, -0.9, -0.4, 0.7, -0.4, 0.6, -0.3, 0.8, 0.3, 0.8,
6                 0.2, -0.1, 0.1, 0.1, -0.5, 0.0, -1.2, 0.0, 0.0, 0.8,
7                 0.0, 0.8, -0.8, -1.1, -0.1, -0.5, 0.2, 0.1, -0.2, -0.2,
8                 -0.1, -0.2, -0.5, -0.2, 0.1, 0.0, 0.3, 0.1, -0.4, -0.2,
9                 -0.6, 0.1, 0.8, 0.6, -0.2, -0.5, -0.2, -0.1, 0.7, 1.2,
10                1.2, 0.2, 0.5, 0.2, 1.5, 1.0, 0.1, 1.3, 0.7, 1.2,
11                1.3, 1.0, 1.3, 1.7, 0.8, 0.8, 1.4, 1.1, 0.7, 1.2,
12                0.1, 1.8, 1.0, 0.6, 1.9, 1.7, 1.2, 1.6)
```

Activité 1 :

1. Récupérer le code *temperature.py* sur le site <https://cviroulaud.github.io>
2. Afficher l'anomalie de température pour l'année 1910.
3. Écrire un programme qui demande une année à l'utilisateur et affiche l'anomalie correspondante.
4. Tester le code 4 qui utilise la bibliothèque *pyplot*. Prendre le temps de comprendre la ligne 3.

```
1 import matplotlib.pyplot as plt
2
3 plt.bar(range(1900, 2018), temperatures)
```

```
4 plt.title("Anomalie des températures 1900-2017")
5 plt.show()
```

Code 4 – Première utilisation de la bibliothèque matplotlib.pyplot

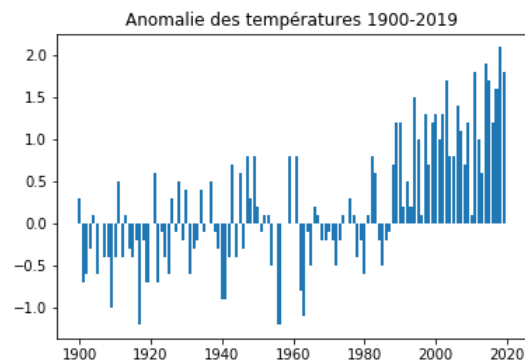


FIGURE 3 – Anomalie des températures 1900 - 2017

3 Les tableaux

3.1 Présentation

Les tuples sont très utiles mais restent limités : on ne peut pas les modifier. Or si nous voulons rajouter les années 2018 et 2019 dans nos données, nous sommes obligés de créer un nouveau tuple. Ce n'est guère satisfaisant.

Un deuxième type de structure semble alors plus adapté : *les tableaux*.

```
1 mon_tableau = [8, 5, 3, 9, 1, 0, 2]
```

Code 5 – Créer un tableau

À savoir : En Python, ces structures portent le nom *list*. Cependant cela peut prêter confusion avec un autre type de construction que nous verrons l'année prochaine.

3.2 Accéder à un élément

La syntaxe est la même que pour les tuples.

```
1 >>> mon_tableau[0]
2 >>> 8
```

Code 6 – Accéder au premier élément

Il est également possible de modifier un élément.

```
1 mon_tableau[2] = 19
```

Code 7 – Modification du troisième élément

3.3 Ajouter un élément

Les tableaux sont *mutables*. Il est possible de modifier leur contenu, ajouter voire supprimer un élément.

```
1 mon_tableau.append(12)
```

Code 8 – Ajouter un élément en fin de liste

3.4 Pour aller plus loin

Il existe de nombreuses autres méthodes pour manipuler les tableaux. La documentation Python présente ces outils.

<https://docs.python.org/fr/3/tutorial/datastructures.html>

3.5 Utiliser le tableau

En 2018 l'anomalie de température était de +2.1. En 2019, elle s'élevait à +1.8.

Activité 2 :

1. Modifier le programme précédent pour qu'il utilise un tableau au lieu d'un tuple.
2. Ajouter les anomalies de 2018 et 2019 à l'aide de la méthode *append*.
3. Réaliser la représentation graphique des anomalies.