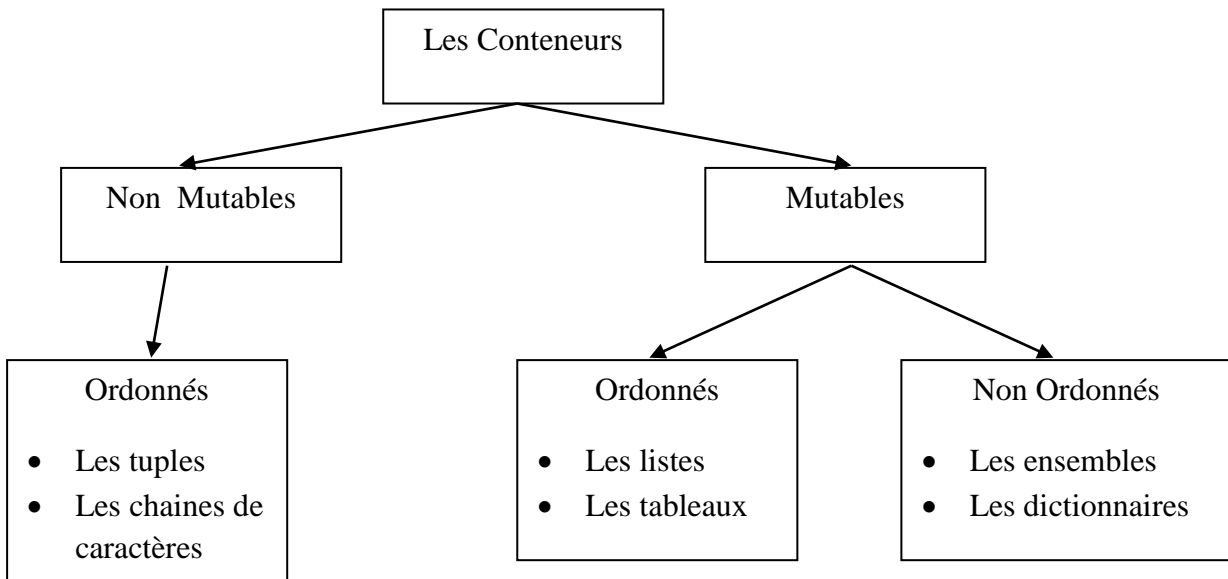


Cours Introduction à Python
Partie 3 : structures de données : Les types mutables

Introduction



Les Tableaux :

Un tableau est une liste de nombres **uniquement** (flottants, entiers, complexes voire booléens codés en binaire ou chaînes de caractères).

Les tableaux peuvent être des vecteurs ou des matrices accessibles à partir de la librairie Numpy qui doit être préalablement chargée et qui contient des fonctions permettant de réaliser des calculs matriciels utilisés en algèbre linéaire. Le type tableau en python est **array**.

A. Installation de la librairie Numpy :

1. Télécharger le package numpy à partir de <http://www.lfd.uci.edu/~gohlke/pythonlibs/#numpy> et enregistrer le dans le dossier c:\Python34

Choisissez le package qui convient à l'architecture de votre ordinateur 32 ou 64 bits :

numpy-1.11.0rc1+mkl-cp34-cp34m-win32.whl

ou

numpy-1.11.0rc1+mkl-cp34-cp34m-win_amd64.whl

2. Ajouter à la variable d'environnement système « path » le chemin de dossier d'installation python « ;C:\Python34 »
3. Mettre à jour le package « pip » : taper la commande suivante dans l'invite de commandes Windows
 > python -m pip install -U pip
4. Installer le package numpy : taper la commande suivante dans l'invite de commandes Windows
 > cd /d c:\Python34\scripts
 > pip install c:\Python34\numpy-1.11.0rc1+mkl-cp34-cp34m-win32.whl
5. Chargement de la librairie **Numpy** :

```
>>> from numpy import *
```

B. Les tableaux à une dimension :

Fonctions	Exemples	Résultats
Importer numpy	<pre>import numpy as np from numpy import * #autrement</pre>	
Créer un tableau vide	<pre>>>> t=np.empty(3) >>> type(t)</pre>	
Créer un tableau non vide	<pre>>>> tab1 = np.array([10.2,3.4,3,4,5]) >>> type(tab1)</pre>	
Taille du tableau	<pre>>>> len(tab1)</pre>	
Accès aux éléments du tableau tab [indice_deb : indice_fin]	<pre>>>> tab1[2] >>> tab1[1 :3] #éléments de 1 à 2 >>> tab1[0 :2] #éléments de 0 à 1 >>> tab1[:3]#Les 3 premiers éléments >>> tab1[3:]#Tout sauf les 3 premiers >>> tab1[-1] >>> tab1[-2] >>> arange(15)# correspond à la fonction range() pour les listes</pre>	
Tableaux particuliers 1	<pre>>>> arange(15) >>> arange(2,3,0.1)</pre>	
Tableaux particuliers 2	<pre>>>> zeros(8) # Tableau avec un nombre de 0 spécifié >>> ones(8) # Tableau avec un nombre de 1 spécifié</pre>	

C. Les tableaux à deux dimensions :

Un tableau 2D (deux dimensions) est en fait un tableau de tableaux. Ils ne sont uniquement accessibles qu'à partir de la librairie numpy. L'accès à ses éléments est identique à celui d'un tableau à une dimension excepté qu'il faut deux indices (ligne et colonne) pour repérer un élément :

Fonctions	Exemples	Résultats
Créer un tableau à 2 D	<pre>>>> tab2D1 = array([[1, 2, 3],[4, 5, 6]]) >>> tab2D1 >>> type(tab2D1)</pre>	
Dimensions du tableau	<pre>>>> shape(tab2D1)</pre>	
Nombres d'éléments	<pre>>>> size(tab2D1)</pre>	
Accès aux éléments du tableau tab [indice_deb : indice_fin]	<pre>>>> tab2D1[1][2] >>> tab2D1[0,:] #première ligne >>> tab2D1[:,0] #première colonne</pre>	
ajoute une ligne ou une colonne avec la fonction <i>append</i>	<pre>>>> t = array([[10,15,20]]) >>> #ligne : axis = 0 >>> append(t,[[1,2,3]],axis=0) >>> #colonne : axis = 1 >>> append(t,[[4],[5]],axis=1)</pre>	
Tableaux particuliers 2	<pre>>>> zeros((6,2)) # Tableau avec un nombre de 0 spécifié >>> ones((2,2)) # Tableau avec un nombre de 1 spécifié</pre>	