

# Memento Python

## I Fondamentaux

- Affichage – La fonction `print()` : `print("bonjour")`
- Déclaration de variable :
  - `var = 1` sera un entier
  - `var = 1.0` sera un réel
  - `var = "1"` sera une chaîne de caractère
  - `var = [var1, var2, var3]` sera une liste
- Changer le type d'un entier : `var = int("1")` sera un entier, `float` un réel et `str` une chaîne de caractère
- Opérateurs : `+`, `-`, `*`, `/`, `%` modulo, `//` quotient de la division euclidienne, `**` puissance, `=` affectation

## II Conditions et boucles

Un programme doit être structuré par des décalages de texte lors de l'utilisation de certaines instructions (`if`, `while`, `for`, etc.). On appelle cela l'indentation du code. Ces décalages s'effectuent avec la touche tabulation du clavier.

- Condition « si » :

```
if condition 1 :
    instruction 1
elif condition 2 :
    instruction 2
else:
    instruction finale
```
- Boucle « pour » ( boucle bornée, quand on sait combien de fois doit avoir lieu la répétition ) :

```
for i in range (10) :
    instruction à répéter pour i allant de 0 à 9
```
- Boucle « tant que » ( boucle non bornée, si on ne connaît pas à l'avance le nombre de répétitions ) :

```
while condition :
    instruction tant que condition est vraie
```

## III Listes

`L = [elem0, elem1, elem2]` crée une liste, une liste est modifiable.  
`L[1] = "toto"` affecte la chaîne de caractère "toto" à la place de `elem1`  
`L.append(élément à ajouter)` ajoute cet élément à la fin de la liste  
`var = len(L)` donne la longueur de la liste

## IV Fonctions

- Importer un module
  - `from math import *` importe toutes les fonctions du module  
`from math import sqrt`, on importe seulement la fonction `sqrt` du module
  - `from random import *` importe toutes les fonctions pour générer des nombres aléatoires  
`randint(0, 10)` renvoie un nombre entier aléatoire entre 0 et 10 inclus.
  - `import numpy as np` importe la bibliothèque Numpy pour effectuer de calculs numériques et gérer des tableaux de nombres.  
`np.pi` permet d'obtenir la valeur de pi

- Définir une fonction

```
def nom_fonction (liste de paramètres) :
    """ commentaire décrivant la fonction """
    instructions
    return resultat
```

exemple :

```
def factorielle (n) :
    """ la fonction factorielle passe
    l'entier n en argument et renvoie le
    résultat n! """
    facto = 1
    for i in range (1, n+1) :
        facto = facto*i
    return facto
```