

# TD1 : Les types de base et les fonctions prédéfinies

Xavier André & Romain Tavenard

## 1 Préambule

Ce semestre, il vous sera demandé de coder dans un seul fichier par sujet de TD. Typiquement, pour la séance de cette semaine, vous devrez :

- créer un répertoire `M1_S1_Python` (dans lequel vous conserverez tous vos scripts du semestre)
- Ouvrir le répertoire `M1_S1_Python` dans VS Code (vous avez le droit d'utiliser un autre éditeur de code, mais alors vous ne pourrez pas compter sur l'assistance de vos enseignants en cas de soucis liés à votre éditeur)
- dans ce répertoire, créer un nouveau fichier `TD1_Intro.py` dont le contenu devra se présenter sous la forme suivante :

```
1 # Section 1 : Imports de module
2
3 # Section 2 : Définition de fonctions
4
5 # Section 3 : Tests de fonctions définies et manipulations en mode "
  script"
```

## 2 Types et fonctions de base

1. Essayer les instructions Python suivantes (ce code devra donc être copié-collé dans la “Section 3” de votre fichier) :

```
1 '3' * 10
2 3 * 10
3 '3' * 10.0
4 '3' + '3'
5 3 + 3
6 '3' + 3
```

Des messages d'erreurs apparaissent dans certains cas. Pour chaque erreur expliquer pourquoi il y a une erreur.

2. Un palindrome est un mot, ou une phrase, qui peut se lire indifféremment dans un sens ou dans l'autre. Dans le cas d'une phrase, on fait abstraction des espaces, majuscules, caractères accentués et caractères de ponctuation. Dans cet exercice, on ne considère que les mots. Écrire un script en Python qui part d'un mot stocké dans une chaîne de caractères puis détermine s'il s'agit d'un palindrome. Indications :
  - Les mots sont donnés en minuscules, sans accents ni cédilles ;
  - Le résultat est un test booléen à vrai ou faux ;
  - Exemples de palindromes simples : kayak, elle, ressasser.
3. Le script que vous avez codé à la question précédente a toutes les raisons d'être encapsulé dans une fonction. Quels seraient les arguments / valeurs de retour / nom que vous utiliseriez pour cette fonction ? Implémentez cette fonction dans la "Section 2" de votre fichier Python, puis testez cette fonction *via* plusieurs appels à cette fonction en "Section 3".
4. Écrire une fonction qui prend en argument un numéro de mois et une année et affiche le nombre de jours du mois (28, 29, 30 ou 31) ou 0 si le mois n'est pas compris entre 1 et 12 (vous n'utiliserez pas le module `datetime` ici).
5. Écrire une fonction qui prend en argument un nom stocké dans une variable `nom` et un entier stocké dans une variable `nombre`, affiche ce nom autant de fois qu'indiqué par le nombre (on ne se souciera pas des retours à la ligne).

### 3 Focus sur les chaînes de caractères

Pour cette partie, vous serez appelés à manipuler les principales méthodes applicables aux chaînes de caractères. Une liste non exhaustive est disponible dans le polycopié associé à ce cours ([cliquer ici](#)).

1. Soient les deux chaînes de caractères suivantes :

```
1  reponse_1_2 = "Je ne vous jette pas la pierre, Pierre,"
2  reponse_2_2 = "mais j'étais à deux doigts de m'agacer"
```

Concaténer les deux répliques ci-dessous et afficher la longueur de la chaîne créée à l'aide de la fonction `len()`.

2. Transformer la réplique concaténée en lettre minuscule puis déterminer l'indice où se trouve la sous-chaîne `"jette"`.
3. Remplacer dans la réplique le mot `"agacer"` par `"énervé"`.

4. Écrire un code Python qui permet d'afficher le nombre d'occurrence de `"pierre"` (sur la réplique en minuscule) et l'indice de toutes les occurrences dans la réplique.
5. Écrire une fonction `compte_mots` qui prend en entrée une chaîne de caractères et retourne le nombre de mots qui la composent (on suppose que les mots sont séparés par des espaces).
6. Écrire une fonction `est_un_fichier_texte` qui retourne vrai si la chaîne de caractères passée en argument est un nom de fichier qui se termine par `".txt"`, `".csv"` ou `".json"`, et faux sinon.

## 4 Modules de base

1. Écrire du code Python qui permet de tirer au hasard une valeur réelle entre 0 et 1 exclu. Le module `random` vous sera utile, consultez sa documentation.
2. Écrire du code Python qui, à partir d'un angle en degré stocké dans une variable `angle`, affiche la valeur du sinus de cet angle, en utilisant le module `math`.
3. Écrire une fonction Python qui calcule l'âge d'une personne à partir du jour, du mois et de l'année de sa naissance, à l'instant de la demande (utiliser le module `datetime`).