

❗ On n'oubliera pas et on commencera donc ce second TD, en chargeant l'environnement virtuel défini lors de la première séance à l'aide de la commande source.

3.1 Utilisation des *containers* de Python

1. Reprendre le Projet Euler n°1 dont l'objet est le calcul de la somme des entiers divisibles par 3 ou par 5 dans l'intervalle $[0, N[$ **en utilisant une liste établie "en compréhension"**. On s'aidera de la fonction intégrée `sum` pour calculer la somme.
2. Projet Euler n°6 : Calculer le résultat de la formule suivante

$$\left(\sum_{k=1}^n k \right)^2 - \sum_{k=1}^n k^2$$

où n sera saisi par l'utilisateur.

3. On s'aidera des méthodes `join` et `split` de la classe `str`. Demander à l'utilisateur de saisir une chaîne de caractères puis
 - (a) renvoyer une chaîne telle qu'un lecteur l'aurait lue de droite à gauche
 - (b) renvoyer une chaîne où le premier mot devient le dernier, le second mot devient l'avant dernier, ...
 - (c) renvoyer la chaîne de caractères saisie sans les voyelles
4. Demander à l'utilisateur de saisir son nom, prénom et âge sur une seule ligne et en utilisant le caractère `,` comme séparateur de champ.
 - (a) Présenter le résultat sous la forme "Votre nom et prénom sont XXX XXX et vous êtes agé(e) de XXX" en n'oubliant pas les majuscules aux noms propres.
 - (b) Créer ensuite un dictionnaire indexé sur le nom de la personne et conservant son âge.
5. Demander à plusieurs utilisateurs de saisir nom, prénom et âge en utilisant toujours le caractère `,` comme séparateur de champ et stocker chacune des chaînes dans une liste. On terminera la saisie en entrant le caractère `$`. Construire un dictionnaire toujours indexé sur le nom mais qui conservera le prénom et l'âge de la personne. Présenter finalement chacune des entrées du dictionnaire sous une forme intelligible.

3.2 Modularité

- Dans un fichier intitulé `euler_project.py`, reprenez les différentes solutions aux projets Euler résolus jusqu'à présent et isoler ces solutions dans des fonctions dont le nom correspondra au numéro du projet préfixé du terme `euler`. Documenter chacune de ces fonctions puis résoudre les 587 problèmes proposés^a
- Exécuter le fichier `euler_project.py` dans la console `ipython` à l'aide de la commande `%run euler_project.py`. Tester les différentes fonctions puis construire un dictionnaire qui associera le numéro du projet à la fonction idoine. Demander finalement à l'utilisateur quel numéro de projet résoudre, s'assurer que le projet a une solution et proposer cette solution.

^aje plaisante ! ne le faites pas