

# Introduction à la Science des données

## Travail pratique 01 – Familiarisation avec le langage Python

Professeur: Dr Stephan ROBERT

Assistant: Dr Régis HOUSOU

Emails : [prenom.nom@heig-vd.ch](mailto:prenom.nom@heig-vd.ch)

### Objectifs:

- Premiers pas avec le langage de programmation Python
- Familiarisation avec l'environnement interactif Jupyter

## 1. Langage de programmation Python

Le site [w3schools.com](https://www.w3schools.com) vous propose une série d'exercices en ligne afin de mettre en pratique le langage de programmation Python et ainsi accomplir les premiers pas avec ce langage.

Allez sur le site <https://www.w3schools.com/python/exercise.asp> et faites les exercices proposés concernant la syntaxe du langage, l'édition des commentaires, l'utilisation des variables et les différents types des données, l'utilisation des opérateurs arithmétiques et logiques, diverses structures des données, telles que les listes, tuples, sets et dictionnaires, les instructions de comparaison et des boucles, les fonctions et l'importation des modules. La programmation orientée objets ne sera pas très importante dans le cadre de ce cours (partie *classes* et *inheritance*).

## 2. Python sur des notebooks Jupyter

Créez un notebook et programmez des solutions aux problèmes ci-dessous :

- Programmez une fonction qui reçoit en entrée un entier  $n$  et qui calcule la somme :  $1 + 2^2 + 3^2 + \dots + n^2$  en utilisant une boucle `for`
- Programmez une fonction qui reçoit en entrée un entier  $n$  et qui calcule la somme :  $1 + 2^2 + 3^2 + \dots + n^2$  en utilisant la méthode `sum(liste)`
- Programmez une fonction qui reçoit en entrée une liste et rend en sortie une nouvelle liste qui contient la première séquence de valeurs et ensuite la même séquence mais à l'inverse, formant un palindrome.

Exemple : `input= [1,2,3,4,5] -> out = [1,2,3,4,5,5,4,3,2,1]`

Aide : si nécessaire, utilisez la méthode `liste.copy()` pour faire une copie de la liste en entrée

- d) Programmez un script qui génère une liste contenant un million de valeurs entières aléatoires entre 1 et 10 (y compris) et qui calcule le pourcentage de valeurs impaires dans cette liste.

Aide : vous pouvez utiliser le module *random* (`import random`) et la méthode *random.randrange* pour la génération de nombres aléatoires.

- e) Programmez un script qui utilise les compréhensions de listes Python pour imprimer les carrés des entiers impairs plus petits ou égales à 10 et les carrés des entiers pairs plus grands que 10, mais plus petits que 21.

C'est-à-dire : [1, 9, 25, 49, 81, 144, 196, 256, 324, 400]

Aide : `new_list = [function(item) for item in list if condition(item)]`

## Rapport

Il n'y a pas de rapport à rendre, mais faites SVP tous les exercices proposés pour vous préparer pour la suite. Ce cours n'est pas un cours de programmation, mais Python reste nécessaire pour la mise en pratique des expériences d'analyse et modélisation des données par la suite.