

Devoir à la maison : les listes

Vous devez envoyer votre travail via la messagerie de l'ENT sous formes d'un seul fichier suivant la nomenclature `<NOM>-<Prénom>-dm-listes.py`. (Exemple : `HOLZER-Marie-dm-listes.py`)

Votre fichier doit pouvoir être exécuté **sans erreur**. Vous devez avoir **appelé les fonctions** et fait des **tests pertinents**.

Il est **interdit** d'utiliser des fonctions sur les listes de Python autres que `append`.

Vous devez faire un affichage avec des `print` pour montrer vos résultats. Vous pouvez aussi faire un `print` avec le numéro de l'exercice pour séparer les affichages (comme je fais dans les corrigés que je vous fournis).

Exercice 1

Écrire une fonction `supprime_zeros` qui prend en argument une liste de notes (entiers positifs) et qui **renvoie une nouvelle liste** sans les zéros. La liste en argument **ne doit pas être modifiée**.

Conseil : pour vérifier, afficher la liste en entrée (l'argument de la fonction) avant l'appel de la fonction et après. Les deux affichages doivent être identiques. Afficher aussi la liste renvoyée par la fonction, elle ne doit plus contenir de zéros.

Exercice 2

Écrire une fonction `remplace_zeros` qui prend en argument une liste de notes (entiers positifs) et en modifie les éléments **en place**. Tous les zéros doivent être remplacés par des vingt. Afficher la liste avant et après l'appel de la fonction. Elle doit avoir changé.

Exercice 3

Écrire une fonction `nombre_pairs` qui prend en argument une liste d'entiers positifs et qui **renvoie une nouvelle liste** ne contenant que les nombres pairs de celle en entrée.

Effectuer les affichages adaptés.

Exercice 4 Écrire une fonction `a_l_envers` qui prend en argument une liste de chaînes de caractères et qui **renvoie une nouvelle liste** contenant les éléments de la première mais à l'envers (le premier élément en dernier, le deuxième en avant-dernier, etc.).

Effectuer les affichages adaptés.

Exemple :

`["le chien", "mange", "une coquille"]` doit donner `["une coquille", "mange", "le chien"]`.

Exercice 5

Écrire une fonction `indices_des_impairs` qui prend en argument une liste d'entiers positifs et qui **renvoie une nouvelle liste** contenant les indices des éléments impairs de la liste en entrée.