

Les listes

Exercice 1 :

Soient les listes suivantes :

$t1 = [31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31]$

$t2 = ['Janvier', 'Février', 'Mars', 'Avril', 'Mai', 'Juin', 'Juillet', 'Août', 'Septembre', 'Octobre', 'Novembre', 'Décembre']$

Ecrivez un petit programme qui crée une nouvelle liste $t3$. Celle-ci devra contenir tous les éléments des deux listes en les alternant, de telle manière que chaque nom de mois soit suivi du nombre de jours correspondant : $['Janvier', 31, 'Février', 28, 'Mars', 31, \text{etc...}]$.

Exercice 2 :

Ecrire une fonction qui échange le premier et dernier éléments d'une liste passée en paramètre. Documentez la fonction.

Exercice 3 :

Ecrire une fonction qui échange les éléments d'indices i et j d'une liste passée en paramètre avec i et j .

Exercice 4 :

Ecrivez une fonction qui affiche « proprement » tous les éléments d'une liste. Si on l'appliquait par exemple à la liste $t2$ de l'exercice ci-dessus, on devrait obtenir :

Janvier Février Mars Avril Mai Juin Juillet Août Septembre Octobre Novembre Décembre

Ecrivez la documentation de cette fonction.

Exercice 5

Ecrivez un programme qui recherche le plus grand élément présent dans une liste donnée.

Par exemple, si on l'appliquait à la liste $[32, 5, 12, 8, 3, 75, 2, 15]$, ce programme devrait afficher :

le plus grand élément de cette liste a la valeur 75.

Ecrivez ensuite une fonction `indexMax(liste)` qui renvoie l'index de l'élément ayant la valeur la plus élevée dans la liste.

Exercice 6 :

Ecrivez un programme qui analyse un par un tous les éléments d'une liste de nombres entiers pour générer deux nouvelles listes. L'une contiendra seulement les nombres pairs de la liste initiale, et l'autre les nombres impairs.

Exercice 7 :

Ecrivez un programme qui analyse un par un tous les éléments d'une liste de mots (par exemple : `['utf', 'unicode', 'soir', 'journée', 'compas', 'ascii']`) pour générer deux nouvelles listes.

L'une contiendra les mots comportant moins de 6 caractères, l'autre les mots comportant 6 caractères ou davantage.

Exercice 8 :

Ecrire le programme précédent dans une fonction de paramètre liste dont on écrira la documentation.

Exercice 9 :

Trouvez sur internet la méthode `split()`.

Ecrivez alors une fonction `chaineListe()` qui convertisse une phrase en une liste de mots. Documentez.