

Exercice 1. Filtre

- (a) Copiez-collez le code de départ depuis Moodle dans un nouveau fichier pour cet exercice. Ensuite, déclarez et implémentez une fonction `filter`, qui accepte, comme premier paramètre: une liste de chaînes de caractères; et comme second paramètre: une fonction de filtre (un prédicat) qui retourne un `bool` pour un paramètre de type `str`. La fonction `filter` doit renvoyer, dans une nouvelle liste, tous les éléments de la liste initiale pour laquelle le prédicat est vérifié.

Par exemple, ceci doit afficher `['programmation', 'fonctionnelle']`:

```
1 def has_odd_length(s: str) -> bool:
2     return len(s) % 2 == 1
3
4 print(filter(["j'aime", "la", "programmation", "fonctionnelle"], has_odd_length))
```

- (b) Utilisez votre fonction `filter` pour filtrer la liste `ronsard_verses` et n'en retenir que les vers qui contiennent une virgule.
- (c) Si vous avez utilisé une déclaration de fonction traditionnelle pour (b), utilisez une fonction lambda, et vice-versa.

Exercice 2. Map

- (a) Reprenez le même code initial que pour l'exercice précédent et écrivez une fonction `map`, qui accepte, comme premier paramètre: une liste de chaînes de caractères; et comme second paramètre, une fonction qui accepte un argument de type `str` et retourne une valeur dérivée dont le type est `Any` (qui représente n'importe quel type; pour rester flexible). La fonction `map` doit retourner une liste d'éléments obtenus par l'appel de la fonction passée en second paramètre sur chacune des valeurs de la liste passée en premier paramètre.

Par exemple, ceci doit afficher `[5, 4, 5, 4, 5, 5, 5, 5, 5, 3, 5, 5, 5, 4, 4, 3, 5, 3]`, soit le nombre d'espaces dans chacun des vers:

```
1 def count_spaces(word: str) -> int:
2     return word.count(" ")
3
4 print(map(ronsard_verses, count_spaces))
```

- (b) Utilisez votre fonction `map` sur `ronsard_verses` pour obtenir tous les vers en majuscules dans une nouvelle liste.
- (c) De manière similaire, utilisez la fonction `map` pour obtenir une liste représentant, pour chaque vers de `ronsard_verses`, le nombre de lettres en majuscules dans ce vers.

Vous devriez obtenir `[1, 1, 2, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1]`.

- (d) En utilisant la fonction `sorted` vue au cours avec une clé de tri que vous définissez vous, faites en sorte de trier les vers de Ronsard avec comme premier critère de tri le nombre de lettres majuscules (du plus grand nombre au plus petit) et comme deuxième critère (lorsque le premier critère ne permet pas de différencier deux vers) la longueur du vers.

Les vers triés ainsi doivent être, dans l'ordre, doivent être (*voir la page suivante*):

Sa robe de pourpre au Soleil,
Ô vraiment marâtre Nature,
Las! las ses beautés laissé choir!
Donc, si vous me croyez, mignonne,
Cueillez, cueillez votre jeunesse:
Las! voyez comme en peu d'espace,
Comme à ceste fleur la vieillesse
Mignonne, allons voir si la rose
Mignonne, elle a dessus la place
Puis qu'une telle fleur ne dure
Tandis que votre âge fleuronne
Les plis de sa robe pourprée,
Et son teint au vôtre pareil.
Que du matin jusques au soir!
En sa plus verte nouveauté,
Qui ce matin avait déclosé
A point perdu cette vèprée
Fera ternir votre beauté.

Indice: pour spécifier plusieurs critères de tri, retournez un tuple avec plusieurs valeurs plutôt qu'une seule valeur comme fait au cours. Un tuple s'écrit comme une liste, mais avec des parenthèses au lieu des crochets, et les valeurs toujours séparées par des virgules.