

# Exercice 1

Décrire l'algorithme qui permettra de trier une liste de mots simple, sans doublons :

on crée un premier tableau avec les doublons et un deuxième qui est vide, avec deux boucle for on parcourt le tableau avec les doublons et on ajoute dans le tableau vide chaque mot dans ce tableau s'il n'y est pas déjà.

# Exercice 2

Ecrire le pseudo code correspondant à cet algorithme :

## Début

words est une liste [

“coco”,

“pedro”,

“pedro”,

“pedro”,

“pedro”,

“jaco”,

“coco”,

“jaco”

]

newList est une liste

i : entier

**pour** i de 0 à longueur(words)

indexAtWord : entier

word : chaîne de caractères.

word ← words[i] // je suis pas très sur

j : entier

**pour** j de 0 à longueur(newList)

**si** newList[j] = null ou newList[j] = word // pas sûr

indexAtword ← j

**finsi**

**finpour**

**si** (indexAtWord >= 0)

newList[indexAtWord] ← word // pas sûr

**finsi**

**finpour**

afficher newList

**fin**

## Exercice 3

Implémenter en Java

[https://github.com/idembele70/Cornillon\\_Devoir\\_1-2/blob/master/Devoir\\_1/Doublon/doublon.java](https://github.com/idembele70/Cornillon_Devoir_1-2/blob/master/Devoir_1/Doublon/doublon.java)