

```

# -*- coding: utf-8 -*-
"""
@auteur: David
"""

#####
print('\n'+ 'Réponse 1 : calculIndiceMaximum(tab)')
#####

# David
def calculIndiceMaximum(tab):
    for indice, valeur in enumerate(tab):
        if valeur == max(tab):
            return indice

exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(exemple)

resultat = calculIndiceMaximum(exemple)
print(resultat)
# doit retourner 5

#####
print('\n'+ 'Réponse 2 : nombrePlusPetit(tab,val)')
#####

def nombrePlusPetit(tab,val):
    compteur = 0
    for valeur in tab:
        if valeur < val:
            compteur += 1
    return compteur

exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(exemple)

resultat = nombrePlusPetit(exemple,8)
print(str(resultat))
# doit retourner 5

#####
print('\n'+ 'Réponse 3 : partition(tab,a,b,indicePivot)')
#####

def partition(tab,indicePivot):
    tab1 = []; tab2 = []; tab3 = []
    pivot = tab[indicePivot]
    for valeur in tab:
        if valeur < pivot: tab1.append(valeur)
        elif valeur == pivot: tab2.append(valeur)
        else: tab3.append(valeur)
    del(tab[:])
    for valeur in tab1 + tab2 + tab3:
        tab.append(valeur)
    return len(tab1)

exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(exemple)

print(partition(exemple,3))

```

```

print(exemple)
# doit retourner 5 car [3, 2, 5, 1, 6, 8, 8, 34, 21, 9, 14]

#####
print('\n'+ 'Réponse 4 : elementK(tab,k)')
#####

def elementK(tab,k):
    if k==0 and len(tab)==1: return tab[0]
    else:
        i = partition(tab,0)
        if i > k: return elementK(tab[:i],k)
        if i == k: return tab[i]
        if i < k: return elementK(tab[i+1:],k-i-1)

exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(exemple)
#exemple.sort()
#print(exemple)
#exemple = [3,2,5,8,1,34,21,6,9,14,8]

print(elementK(exemple,4))
print(exemple)
# doit retourner 6 car [1,2,3,5,6,8,8,9,14,21,34]

#####
print('\n'+ 'Réponse 6 : ChoixPivot(tab)')
#####

import numpy

def petiteMediane(tab):
    assert len(tab) <= 5
    n = len(tab)
    return numpy.sort(tab)[n//2]

def ChoixPivot(tab):
    if len(tab) <= 5:
        return petiteMediane(tab)
    else:
        t = []
        while len(tab) > 5:
            t.append(petiteMediane(tab[:5]))
            for i in range(5): tab.pop(0)
        if len(tab)>0: t.append(petiteMediane(tab[:5]))
        return ChoixPivot(t)

exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(ChoixPivot(exemple.copy()))
# doit retourner 8 car [3,2,5,8,1,34,21,6,9,14,8] --> [3,14,8] --> [8]

def elementK_opt(tab,k):
    if k==0 and len(tab)==1: return tab[0]
    else:
        i = partition(tab,ChoixPivot(tab.copy()))
        if i > k: return elementK(tab[:i],k)
        if i == k: return tab[i]
        if i < k: return elementK(tab[i+1:],k-i-1)

```

```
exemple = [3,2,5,8,1,34,21,6,9,14,8]
print(exemple)
print(elementK_opt(exemple,4))
print(exemple)
# doit retourner 6 car [1,2,3,5,6,8,8,9,14,21,34]
```